TU Delft

Delft University of Technology

Document Version
Final published version

Licence
CC BY

Citation (APA)
Schumann, J. F. (2026). *Modeling Human Driving: Evaluating Predictions and Explaining Behavior*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:a5b7e70f-9681-4148-a85c-adab5c8dfc2c

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
In case the licence states “Dutch Copyright Act (Article 25fa)”, this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

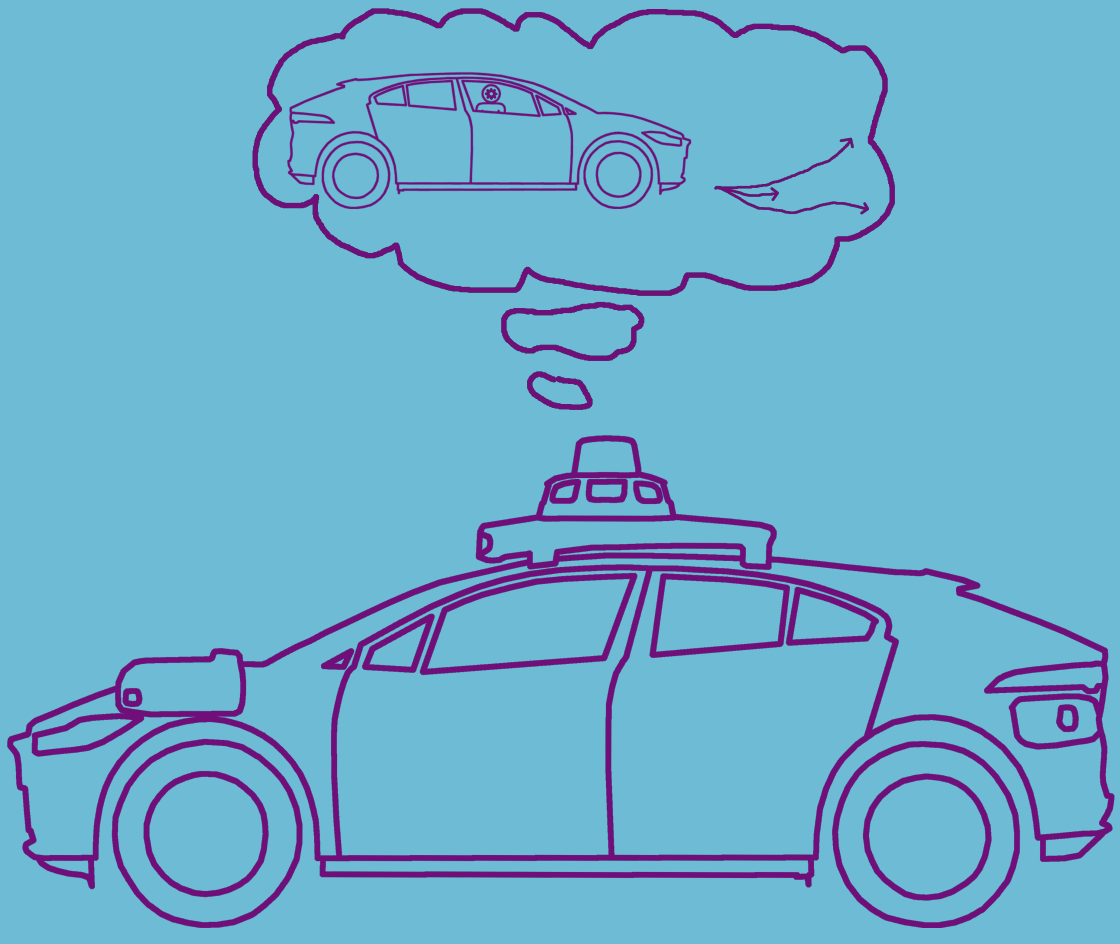
Sharing and reuse
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.

Modeling Human Driving

Evaluating Predictions and Explaining Behavior



Julian Frederik Schumann

Modeling Human Driving: Evaluating Predictions and Explaining Behavior

Modeling Human Driving: Evaluating Predictions and Explaining Behavior

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology,
by the authority of the Rector Magnificus
Prof. dr. ir. H. Bijl,
chair of the Board for Doctorates
to be defended publicly on
Friday, October 9, 2026, at 10:00

by

Julian Frederik SCHUMANN

This dissertation has been approved by the (co)promotors and the external advisor.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Prof. dr.-ing. J. Kober,	Delft University of Technology & University of Stuttgart, *promotor*
Dr. A. Zgonnikov,	Delft University of Technology, *copromotor*
Dr. J. Engström,	Waymo LLC, United States of America, *external advisor*

Independent members:

Prof. dr. ir. M. Wisse,	Delft University of Technology
Prof. dr. ir. J. C. F. de Winter,	Delft University of Technology
Prof. dr. K. J. Friston	University College London, United Kingdom
Dr. K. Messaoud	Télécom Paris, France
Prof. dr.-ir. R. Happee,	Delft University of Technology, *reserve member*

This research was supported by *Waymo LLC*. All content represents the opinion of the author, which is not necessarily shared or endorsed by the sponsor.





Keywords: autonomous vehicles, behavior prediction, evaluation, active inference.

Printed by: Ridderprint

Cover by: Julian Schumann and Yijia Chen

Copyright © 2026 by Julian Frederick Schumann
ISBN 978-94-6518-443-2

An electronic version of this dissertation is available at
`http://repository.tudelft.nl/`.

The man who moves a mountain begins by carrying away small stones.

Confucius

Contents

Summary **xi**

Samenvatting **xv**

Zusammenfassung **xix**

1 Introduction **1**

I Evaluating Prediction Models **7**

2 Benchmarking Behavior Prediction Models in Gap Acceptance Scenarios **13**
2.1 Introduction . . . 14
2.2 Defining gap acceptance . . . 17
2.3 Benchmarking gap acceptance models . . . 18
2.4 Benchmark implementation . . . 25
2.5 Results . . . 28
2.6 Conclusion . . . 31

3 Structured Training and Evaluation Platform for Benchmarking Trajectory Prediction Models **33**
3.1 Introduction . . . 34
3.2 Related work . . . 35
3.3 STEP . . . 37
3.4 Experiments . . . 41
3.5 Results . . . 43
3.6 Conclusion . . . 48

4 Robust Multi-Modal Density Estimator **49**
4.1 Introduction . . . 50
4.2 Related work . . . 51
4.3 ROME . . . 52
4.4 Experiments . . . 57
4.5 Results . . . 60
4.6 Conclusion . . . 64

5 Realistic Adversarial Attacks for Robustness Evaluation of Trajectory Prediction Models **67**
5.1 Introduction . . . 68
5.2 Related work . . . 70
5.3 Background . . . 71
5.4 Method . . . 73

5.5 Experimental setting . . . 78
5.6 Results . . . 82
5.7 Discussion . . . 91
5.8 Conclusion . . . 93

II Explaining Human Behavior 95

6 Cognitively Inspired Modeling of Human Gap Acceptance Behavior in Traffic 99
6.1 Introduction . . . 100
6.2 Background . . . 101
6.3 Commotions as general model of gap acceptance behavior . . . 103
6.4 Evaluating Commotions' configurations . . . 104
6.5 Comparing Commotions to established models . . . 107
6.6 Conclusion . . . 109

7 Active Inference as a Model of Human Collision Avoidance Behavior 111
7.1 Introduction . . . 112
7.2 Methods . . . 114
7.3 Results . . . 123
7.4 Evaluating individual model mechanisms . . . 132
7.5 Parameter sensitivity analysis . . . 138
7.6 Discussion . . . 141
7.7 Conclusion . . . 148

8 Active Inference as a Model of Human Resolution of Space-Sharing Conflicts 149
8.1 Introduction . . . 150
8.2 Method . . . 152
8.3 Results . . . 159
8.4 Discussion . . . 171
8.5 Conclusion . . . 175

III Conclusion and future work 177

9 Conclusion 179
9.1 Evaluating Prediction Models . . . 179
9.2 Explaining Human Behavior . . . 182

10 Extensions and Impact 185
10.1 Limitations . . . 185
10.2 Future work . . . 186
10.3 Impact . . . 189

IV Appendices 193

A Details of the Gap Acceptance Benchmark Framework 195
A.1 Extracting 1D data from 2D data . . . 195

A.2 Datasets . . . 196
A.3 Models . . . 200
A.4 Metrics . . . 205

B An Active Inference Model of Human Behavior 207
B.1 The general active inference model . . . 207
B.2 Model implementation details . . . 213
B.3 Implementations of simulation scenarios . . . 222

C Additional Experiments for the Active Inference Model 227
C.1 Norm conditioning in a non-incursion scenario . . . 227
C.2 Epistemic value in a limited visibility scenario . . . 228
C.3 Evaluating deep learning models in collision avoidance . . . 229

D The Epistemic Value of Prompting in the Intersection Interaction 231

Bibliography 233

Acknowledgments 265

List of Publications 269

Curriculum Vitæ 271

Summary

Autonomous driving has emerged as one of the most prominent research fields of the past decades, driven by both substantial economic prospects and the promise of significantly improving road safety by reducing avoidable and potentially deadly collisions caused by human error. Despite this potential, progress toward fully autonomous systems has consistently lagged behind early optimistic projections. A central reason for this gap lies in the difficulty of handling interactions with human road users. In such mixed-traffic environments, AVs must continuously interpret and anticipate the behavior of others. Early systems have often struggled in such settings, frequently adopting overly conservative or timid strategies that prioritize safety but degrade efficiency and usability.

Improving models of human driving behavior is therefore a critical step toward advancing autonomous driving along several dimensions. First, accurate and precise prediction of human actions allows AVs to reason about a broader set of safe future trajectories, enabling more assertive and efficient behavior without compromising safety. Second, realistic behavior models are essential for simulation, which remains one of the primary tools for developing and validating AV systems, especially in rare and safety-critical scenarios that are difficult or dangerous to reproduce in the real world. Third, understanding the mechanisms underlying human behavior enables AVs to act in ways that align with human expectations, thereby facilitating smoother interactions and increasing user trust and societal acceptance. Taken together, these aspects highlight that progress in autonomous driving is tightly coupled to advances in modeling human behavior.

This thesis addresses this challenge along two complementary directions. The first direction focuses on prediction models, where recent developments have been dominated by deep learning approaches. In particular, we try to improve the evaluation of such models to test whether they are truly reliable and generalizable enough for real-world deployment. Problematically, current prediction models are typically assessed on standardized benchmarks, which are dominated by benign interactions and therefore provide limited insight into performance in safety-critical situations. Moreover, commonly used metrics treat all prediction errors equally, ignoring the fact that some mistakes – such as failing to predict a collision – are far more consequential than others. Finally, models are often trained and evaluated only once, neglecting the inherent stochasticity in training procedures and leading to comparisons between individual model instances rather than underlying architectures.

Therefore, Chapters 2 and 3 introduce a novel framework for evaluating human behavior prediction models, aiming to overcome the key limitations of current practice. Specifically, the framework enables model evaluation focused on safety-critical scenarios, while incorporating outcome-based evaluation metrics that reflect the severity of prediction errors. It also allows the systematic evaluation of multiple instances of each model architecture. Together, this enables a more robust and meaningful comparison of different approaches.

Chapter 4 addresses the problem that many models can generate multiple predictions but do not have an accessible representation of the underlying probability distribution. With many important metrics relying on such a representation, we need methods to approximate this distribution from the generated samples. Therefore, we introduce a novel method for approximating the underlying probability distribution from generated samples. In particular, this method is based on clustering the samples into approximately normal distributions. This approach provides a more principled assessment of uncertainty and better captures the strengths and weaknesses of probabilistic models. Meanwhile, Chapter 5 presents a novel method for generating adversarial attacks on human behavior models. Here, such adversarial attacks involve making small, carefully designed perturbations to the observed past behavior of an agent in order to induce large errors in the model' s predictions. Our new method ensures that these perturbations remain dynamically feasible and that the resulting scenarios remain comparable to the original ground truth. Thereby, we enable the systematic generation of challenging but meaningful test cases, which can provide valuable insights into model robustness and can expose model vulnerabilities. Consequently, this will be helpful for the development of more reliable prediction systems.

Applying these evaluation methods to a range of state-of-the-art models, we find that models that perform well on standard benchmarks still exhibit significant weaknesses. Especially in safety-critical scenarios, many models show substantial brittleness, with both adversarial attacks and domain shifts causing significant deterioration in performance. Furthermore, our results indicate that the improvements claimed by many models on these standard benchmarks are often not statistically significant when considering multiple model instances, reducing our ability to identify the most promising model architectures in great detail.

The second direction of this thesis tries to mitigate some of these limitations by investigating the promise of cognitively inspired models of human behavior. While there has been previous work on such mechanistic modeling, the scope of those examples was often too narrow. For example, many such models focus exclusively on certain agent types in limited scenarios, such as pedestrian street crossing. This limits the generalizability of the resulting models and restricts their applicability in the diverse and dynamic environments encountered in real-world traffic.

Chapter 6 shows that those limitations can be addressed in principle by expanding the focus of a previously limited model. Specifically, a model originally developed for pedestrian street crossing is extended to a more general framework for human gap acceptance – a fundamental decision-making process that arises in a variety of contexts, including intersections, merging maneuvers, and roundabouts. The chapter demonstrates that, by relying on the baseline model's underlying principles such as evidence accumulation and noisy perception, it can be generalized across different scenarios. This highlights the potential of mechanistic approaches to capture common structures in human behavior rather than focusing on isolated cases.

Chapter 7 introduces a novel model of collision avoidance in human drivers using the active inference framework. Active inference is a principled framework rooted in the idea that agents act to minimize a quantity known as free energy. Here, free energy minimization effectively balances exploration and exploitation. Our active inference model

can both reproduce many aspects of human behavior such as response times and chosen avoidance maneuvers. Additionally, we show strong generalization across scenarios, with model parameters shown to transfer successfully to a previously unseen cases. This contrasts with many existing models, which require scenario-specific tuning, and suggests that active inference provides a promising foundation for capturing general aspects of human behavior.

Chapter 8 further strengthens the promise of the active inference model's general applicability by incorporating additional mechanisms that are critical for real-world interactions. In particular, the model is augmented to account for the influence of interactive traffic norms (*e.g.*, priority rules) as well as explicit communication between agents. These extensions are shown to play an important role in resolving interactions at intersections. The ease with which these mechanisms can be integrated further supports the flexibility and generality of the active inference approach.

In conclusion, this thesis both improves the evaluation of existing behavior models and proposes a new general active inference model of human driving. On the evaluation side, it introduces a comprehensive framework that enables a standardized and efficient validation of prediction models, with a particular emphasis on safety-critical performance. This has the potential to accelerate the development of improved models by providing clearer guidance on their strengths and weaknesses. On the modeling side, the thesis demonstrates that cognitively inspired approaches such as active inference offer a viable alternative to purely data-driven methods. Specifically, such models can improve upon key limitations of deep learning approaches, including their lack of explainability and potential brittleness in critical scenarios. At the same time, they can also provide deeper insights into the principles governing human behavior. This suggests a path toward unified models of human behavior that can support safer and more effective interaction between autonomous vehicles and human road users.

Samenvatting

Autonoom rijden is in de afgelopen decennia uitgegroeid tot een van de meest prominente onderzoeksgebieden. Deze ontwikkeling wordt gedreven door zowel aanzienlijke economische vooruitzichten, alsmede de belofte de verkeersveiligheid aanzienlijk te verbeteren door vermijdbare en potentieel dodelijke ongevallen als gevolg van menselijke fouten te verminderen. Ondanks dit potentieel is de vooruitgang richting volledig autonome systemen consequent achtergebleven bij de vroege optimistische verwachtingen. Een belangrijke oorzaak van deze kloof is dat interacties met menselijke verkeersdeelnemers moeilijk te verwerken zijn. In dergelijke gemengde verkeersomgevingen moeten autonome voertuigen (AV's) voortdurend het gedrag van anderen interpreteren en anticiperen. Vroege systemen hadden vaak moeite met dergelijke situaties en kozen regelmatig voor overdreven voorzichtige strategieën die veiligheid prioriteren, maar ten koste gaan van efficiëntie en bruikbaarheid.

Het verbeteren van modellen van menselijk rijgedrag is daarom een cruciale stap om de ontwikkeling van autonoom rijden op meerdere fronten te bevorderen. Ten eerste maakt een nauwkeurige en precieze voorspelling van menselijke handelingen het voor AV's mogelijk om over een bredere scala aan veilige toekomstige trajecten te redeneren, waardoor zij besluitvaardiger en efficiënter kunnen handelen zonder de veiligheid in gevaar te brengen. Ten tweede zijn realistische gedragsmodellen essentieel voor simulatieomgevingen, die nog steeds een van de belangrijkste instrumenten vormt voor de ontwikkeling en validatie van AV-systemen. Dat is bijzonders belangrijk in zeldzame en veiligheidskritieke scenario's die moeilijk of gevaarlijk zijn om in de praktijk te reproduceren. Ten derde stelt inzicht in de mechanismen die ten grondslag liggen aan menselijk gedrag AV's in staat zich te gedragen op een manier die aansluit bij menselijke verwachtingen, waardoor soepelere interacties ontstaan en het vertrouwen van gebruikers en de maatschappelijke acceptatie toenemen. Gezamenlijk onderstrepen deze aspecten dat vooruitgang in autonoom rijden nauw verbonden is met vooruitgang in het modelleren van menselijk gedrag.

Dit proefschrift behandelt deze uitdaging langs twee complementaire richtingen. De eerste richting focust op voorspellingsmodellen, waarin recente ontwikkelingen worden gedomineerd door deep-learningmethoden. In het bijzonder proberen wij de evaluatie van dergelijke modellen te verbeteren door te bestuderen of zij daadwerkelijk voldoende betrouwbaar en generaliseerbaar zijn voor toepassing in de praktijk. Problematisch is dat huidige voorspellingsmodellen doorgaans worden beoordeeld aan de hand van gestandaardiseerde vergelijkingsmethodes, die worden gedomineerd door onschuldige interacties en daardoor slechts beperkt inzicht bieden in de prestaties in veiligheidskritieke situaties. Bovendien behandelen veelgebruikte evaluatiematen alle voorspellingsfouten als gelijkwaardig, terwijl sommige fouten – zoals het niet voorspellen van een botsing – veel ernstigere gevolgen hebben dan andere. Ten slotte worden modellen vaak slechts één keer getraind en geëvalueerd, waardoor de inherente stochastiek van trainingsprocedu-

res wordt genegeerd en vergelijkingen plaatsvinden tussen individuele modelinstanties in plaats van tussen de onderliggende modelarchitecturen.

Daarom introduceren Hoofdstukken 2 en 3 een nieuw kader voor het evalueren van voorspellingsmodellen voor menselijk gedrag, met als doel de belangrijkste beperkingen van de huidige praktijk te overwinnen. Concreet maakt idem een modelevaluatie mogelijk die is gericht op veiligheidskritieke scenario's, terwijl tegelijkertijd uitkomstgebaseerde evaluatiematen worden toegepast die rekening houden met de ernst van voorspellingsfouten. Daarnaast maakt het een systematische evaluatie van meerdere instanties van elke modelarchitectuur mogelijk. Hierdoor ontstaat een robuustere en betekenisvollere vergelijking van verschillende methodes.

Hoofdstuk 4 behandelt het probleem dat veel modellen meerdere voorspellingen kunnen genereren, maar geen toegankelijke representatie hebben van de onderliggende kansverdeling. Omdat veel belangrijke evaluatiematen afhankelijk zijn van een dergelijke representaties, zijn methoden nodig om deze kansverdeling te benaderen op basis van de gegenereerde steekproeven. Daarom introduceren wij een nieuwe methode om de onderliggende kansverdeling uit gegenereerde steekproeven te benaderen. Deze methode is gebaseerd op het clusteren van de steekproeven in bij benadering normale verdelingen. Deze aanpak biedt een meer principiële beoordeling van onzekerheid en legt de sterke en zwakke punten van probabilistische modellen beter in kaart. Hoofdstuk 5 presenteert daarnaast een nieuwe methode voor het genereren van adversariële aanvallen op modellen voor menselijk gedrag. Dergelijke adversariële aanvallen bestaan uit het aanbrengen van kleine, zorgvuldig ontworpen verstoringen in het geobserveerde gedrag uit het verleden van een agent, met als doel grote fouten in de voorspellingen van het model te veroorzaken. Onze nieuwe methode garandeert dat deze verstoringen dynamisch uitvoerbaar blijven en dat de resulterende scenario's vergelijkbaar blijven met de oorspronkelijke referentiewaardes. Daarmee maken wij de systematische constructie mogelijk van uitdagende maar betekenisvolle testscenarios, die waardevolle inzichten kunnen bieden in de robuustheid van modellen en kwetsbaarheden van modellen aan het licht kunnen brengen. Dit zal uiteindelijk bijdragen aan de ontwikkeling van betrouwbaardere voorspellingssystemen.

Door deze evaluatiemethoden toe te passen op een reeks state-of-the-artmodellen constateren wij dat modellen die goed presteren op standaardidem, nog steeds aanzienlijke zwakke punten vertonen. Vooral in veiligheidskritieke scenario's blijken veel modellen aanzienlijk kwetsbaar te zijn, waarbij zowel adversariële aanvallen als domeinverschuivingen leiden tot een duidelijke verslechtering van de prestaties. Bovendien wijzen onze resultaten erop dat de verbeteringen die door veel modellen op deze standaardbenchmarks worden geclaimd vaak niet statistisch significant zijn wanneer rekening wordt gehouden met meerdere modelinstanties. Hierdoor wordt het mogelijk de meest veelbelovende modelarchitecturen in detail te identificeren.

De tweede richting van dit proefschrift probeert enkele van deze beperkingen te ondervangen door het potentieel van cognitief geïnspireerde modellen van menselijk gedrag te onderzoeken. Hoewel eerder onderzoek naar dergelijke mechanistische modellen is verricht, was de reikwijdte van deze voorbeelden vaak te beperkt. Zo richten veel van deze modellen zich uitsluitend op specifieke typen agenten in beperkte scenario's, zoals overstekende voetgangers. Dit limiteert de generaliseerbaarheid van de resulterende modellen en verkleint hun toepasbaarheid in de diverse en dynamische omgevingen die kenmerkend

zijn voor het verkeer in de praktijk.

Hoofdstuk 6 laat zien dat deze beperkingen kunnen worden aangepakt door de reikwijdte van een eerder beperkt model uit te breiden. Concreet wordt een model dat oorspronkelijk is ontwikkeld voor overstekende voetgangers uitgebreid tot een algemener idem voor menselijke *gap acceptance* – een fundamenteel besluitvormingsproces dat voorkomt in uiteenlopende contexten, waaronder kruispunten, invoegmanoeuvres en rotondes. Het hoofdstuk laat zien dat het model, door voort te bouwen op de onderliggende principes van het basismodel, zoals evidentieaccumulatie en ruis in de waarneming, kan worden gegeneraliseerd naar verschillende scenario's. Dit onderstreept het potentieel van mechanistische benaderingen om gemeenschappelijke structuren in menselijk gedrag te beschrijven, in plaats van zich te richten op geïsoleerde gevallen.

Hoofdstuk 7 introduceert een nieuw model voor botsingsvermijding door menselijke bestuurders op basis van het active-inference-methode. Active inference is een principieel methode dat is gebaseerd op het idee dat agenten handelen om een grootheid, bekend als vrije energie, te minimaliseren. In deze context zorgt de minimalisatie van vrije energie voor een effectieve balans tussen exploratie en exploitatie. Ons active-inference-model kan vele aspecten van menselijk gedrag reproduceren, zoals reactietijden en gekozen uitwijkmanoeuvres. Daarnaast laten wij een sterke generalisatie over verschillende scenario's zien, waarbij modelparameters met succes kunnen worden overgedragen naar eerder ongeziene situaties. Dit staat in contrast met veel bestaande modellen, waarvoor afstemming per scenario noodzakelijk is, en suggereert dat active inference een veelbelovende basis vormt voor het modelleren van algemene aspecten van menselijk gedrag.

Hoofdstuk 8 versterkt de belofte van de brede toepasbaarheid van het active-inference-model verder door aanvullende mechanismen te integreren die essentieel zijn voor interacties in de praktijk. In het bijzonder wordt het model uitgebreid om rekening te houden met de invloed van interactieve verkeersnormen (bijvoorbeeld voorrangsregels) en expliciete communicatie tussen agenten. Deze uitbreidingen blijken een belangrijke rol te spelen bij het oplossen van interacties op kruispunten. Het gemak waarmee deze mechanismen kunnen worden geïntegreerd, ondersteunt bovendien de flexibiliteit en algemene toepasbaarheid van de active-inference-methode.

Concluderend verbetert dit proefschrift zowel de evaluatie van bestaande gedragsmodellen als introduceert het een nieuw algemeen active-inference-model voor menselijk rijgedrag. Betreffende de evaluatie wordt een uitgebreid idem geïntroduceerd dat een gestandaardiseerde en efficiënte validatie van voorspellingsmodellen mogelijk maakt, met bijzondere nadruk op prestaties in veiligheidskritieke situaties. Dit heeft het potentieel de ontwikkeling van verbeterde modellen te versnellen door duidelijker inzicht te bieden in hun sterke en zwakke punten. Aan de modelleerzijde laat het proefschrift zien dat cognitief geïnspireerde benaderingen, zoals active inference, een levensvatbaar alternatief vormen voor puur datagedreven methoden. Dergelijke modellen kunnen met name belangrijke tekortkomingen van deep-learningbenaderingen verminderen, waaronder hun gebrek aan verklaarbaarheid en hun potentiële kwetsbaarheid in kritieke scenario's. Tegelijkertijd kunnen zij diepere inzichten verschaffen in de principes die ten grondslag liggen aan menselijk gedrag. Dit biedt perspectief op de ontwikkeling van uniforme modellen voor menselijk gedrag die veiligere en effectievere interacties tussen autonome voertuigen en menselijke verkeersdeelnemers kunnen ondersteunen.

Zusammenfassung

Autonomes Fahren hat sich in den vergangenen Jahrzehnten zu einem der bedeutendsten Forschungsgebiete entwickelt, angetrieben sowohl durch wirtschaftliche Vorteile als auch durch das Versprechen einer deutlich verbesserten Verkehrssicherheit, indem vermeidbare und potenziell tödliche Unfälle aufgrund menschlicher Fehler reduziert werden. Trotz dieses Potenzials ist der Fortschritt hin zu vollständig autonomen Fahrzeugen kontinuierlich hinter optimistischeren Prognosen zurückgeblieben. Ein wesentlicher Grund für diese Diskrepanz ist in der Schwierigkeit, Interaktionen mit menschlichen Verkehrsteilnehmern zu bewältigen, zu finden. In solchen gemischten Verkehrsumgebungen müssen autonome Fahrzeuge das Verhalten anderer Verkehrsteilnehmer fortlaufend interpretieren und antizipieren. Die ersten erprobten Systeme hatten in solchen Situationen häufig Schwierigkeiten und verfolgten oft übermäßig vorsichtige und zurückhaltende Strategien. Diese konnten zwar die Sicherheitsanforderungen erfüllen, hatten jedoch eine geringe Effizienz und Nutzbarkeit.

Hier ist die Verbesserung von Modellen des menschlichen Fahrverhaltens ein entscheidender Schritt, um die Entwicklung des autonomen Fahrens in mehrfacher Hinsicht voranzubringen. Erstens ermöglicht die genaue und präzise Vorhersage menschlicher Handlungen es autonomen Fahrzeugen, entschlossener und effizienter zu handeln ohne dabei die Sicherheit zu beeinträchtigen. Zweitens sind solche realistischen Verhaltensmodelle unerlässlich für Simulationen, eines der wichtigsten Werkzeuge für die Entwicklung und Validierung autonomer Fahrzeugsysteme insbesondere in seltenen und sicherheitskritischen Szenarien, welche in der realen Welt nur schwer sicher nachzustellen sind. Drittens ermöglicht ein besseres Verständnis der Mechanismen, die menschlichem Verhalten zugrunde liegen, autonomen Fahrzeugen, sich in einer Weise zu verhalten, die den Erwartungen menschlicher Verkehrsteilnehmer entspricht. Dadurch können reibungslosere Interaktionen sowie ein größeres Vertrauen der Nutzer und eine höhere gesellschaftliche Akzeptanz gefördert werden. Zusammengenommen verdeutlichen diese Aspekte, dass Fortschritte im autonomen Fahren eng mit denen bei der Modellierung menschlichen Verhaltens verknüpft sind.

Diese Dissertation begegnet dieser Herausforderung entlang zweier komplementärer Forschungsrichtungen. Die erste Richtung konzentriert sich auf Vorhersagemodelle, deren jüngste Entwicklungen von Deep-Learning-Ansätzen dominiert werden. Insbesondere versuchen wir, die Evaluierung solcher Modelle zu verbessern, um zu überprüfen, ob sie tatsächlich zuverlässig und generalisierbar genug für den Einsatz in der Praxis sind. Problematisch ist, dass aktuelle Vorhersagemodelle typischerweise anhand standardisierter Benchmarks bewertet werden, die überwiegend harmlose Interaktionen enthalten und daher nur begrenzte Einblicke in die Leistungsfähigkeit in sicherheitskritischen Situationen bieten. Darüber hinaus behandeln häufig verwendete Metriken alle Vorhersagefehler gleich, obwohl manche Fehler – beispielsweise das Nichterkennen einer bevorstehenden Kollision – deutlich schwerwiegendere Folgen haben als andere. Schließlich wer-

den Modelle häufig nur einmal trainiert und evaluiert, wodurch die inhärente Stochastizität von Trainingsverfahren vernachlässigt wird und man damit statt der zugrunde liegenden Modellarchitekturen nur einzelne Modellinstanzen vergleicht.

Daher führen die Kapitel 2 und 3 einen neuartiges System zur Evaluierung von Vorhersagemodellen menschlichen Verhaltens ein. Dieses zielt vor allem darauf ab, die wesentlichen Einschränkungen der bisherigen Praxis zu überwinden. Konkret ermöglicht dieses System eine auf sicherheitskritische Szenarien fokussierte Modellevaluierung unter Verwendung ergebnisbasierter Bewertungsmetriken, welche die Schwere von Vorhersagefehlern berücksichtigen. Darüber hinaus erlaubt es die systematische Evaluierung vieler Instanzen jeder Modellarchitektur. Dadurch wird ein robusterer und aussagekräftigerer Vergleich verschiedener Ansätze ermöglicht.

Kapitel 4 behandelt das Problem, dass viele Modelle zwar mehrere variierende Vorhersagen erzeugen können, jedoch keine zugängliche Repräsentation der zugrunde liegenden Wahrscheinlichkeitsverteilung besitzen. Da viele wichtige Metriken auf einer solchen Repräsentation beruhen, werden Methoden benötigt, um diese Verteilung aus den erzeugten Stichproben zu approximieren. Daher stellen wir eine neuartige Methode zur Approximation der zugrunde liegenden Wahrscheinlichkeitsverteilung aus erzeugten Stichproben vor. Insbesondere basiert diese Methode auf dem Clustern der Stichproben zu annähernd normalverteilten Komponenten. Dieser Ansatz ermöglicht eine fundiertere Bewertung von Unsicherheit und bildet die Stärken und Schwächen probabilistischer Modelle besser ab. Kapitel 5 präsentiert darüber hinaus eine neuartige Methode zur Erzeugung adversarialer Angriffe auf Modelle menschlichen Verhaltens. Solche adversarialen Angriffe bestehen darin, kleine, sorgfältig entworfene Störungen des beobachteten vergangenen Verhaltens eines Agenten einzuführen, um große Fehler in den Modellvorhersagen hervorzurufen. Unsere neue Methode stellt sicher, dass diese Störungen dynamisch realisierbar bleiben und die resultierenden Szenarien mit dem ursprünglich aufgenommenen Verhalten vergleichbar sind. Dadurch ermöglichen wir die systematische Erzeugung schwierig zu verhersagenden, aber aussagekräftigen Testfällen, die wertvolle Einblicke in die Robustheit von Modellen liefern und deren Schwachstellen aufdecken können.

Durch die Anwendung dieser Evaluierungsmethoden auf eine Reihe von Modellen auf dem Stand der aktuellen Technik stellen wir fest, dass diese Modelle, die auf Standardbenchmarks gut abschneiden, dennoch erhebliche Schwächen aufweisen. Insbesondere in sicherheitskritischen Szenarien zeigen viele Modelle eine ausgeprägte Fragilität, wobei sowohl adversariale Angriffe als auch Domänenverschiebungen zu einer deutlichen Verschlechterung der Leistung führen. Darüber hinaus zeigen unsere Ergebnisse, dass die von vielen Modellen auf diesen Standardbenchmarks berichteten Verbesserungen häufig nicht statistisch signifikant sind, wenn mehrere Modellinstanzen berücksichtigt werden. Dadurch wird unsere Fähigkeit eingeschränkt, die vielversprechendsten Modellarchitekturen im Detail zu identifizieren.

Die zweite Forschungsrichtung dieser Dissertation versucht, einige dieser hier gefundenen Einschränkungen durch die Untersuchung des Potenzials kognitiv inspirierter Modelle menschlichen Verhaltens zu überwinden. Obwohl bereits frühere Arbeiten zu solchen mechanistischen Modellen existieren, war deren Anwendungen häufig zu eng gefasst. So konzentrieren sich viele dieser Modelle ausschließlich auf einzelne Verkeihrsteilnehmer in begrenzten Szenarien, etwa auf Fußgänger beim Überqueren einer Straße. Dies schränkt

die Generalisierbarkeit der resultierenden Modelle ein und begrenzt ihre Anwendbarkeit in den vielfältigen und dynamischen Verkehrsumgebungen der realen Welt.

Kapitel 6 zeigt, dass diese Einschränkungen durch eine Erweiterung des Anwendungsbereichs eines zuvor eingeschränkten Modells überwunden werden können. Konkret wird ein ursprünglich für das Überqueren von Straßen durch Fußgänger entwickeltes Modell zu einem allgemeineren Rahmen für die menschliche *Gap Acceptance* (zu Deutsch wortwörtlich *Lückenakzeptanz*) erweitert – einem grundlegenden Entscheidungsprozess, der in einer Vielzahl von Situationen auftritt, darunter Kreuzungen, Einfädelvorgänge und Kreisverkehre. Das Kapitel zeigt, dass sich das Modell auf Grundlage seiner zugrunde liegenden Prinzipien wie Evidenzakkumulation und verrauschter Wahrnehmung auf unterschiedliche Szenarien verallgemeinern lässt. Dies unterstreicht das Potenzial mechanistischer Ansätze, gemeinsame und generelle Strukturen menschlichen Verhaltens zu erfassen, anstatt sich auf isolierte Einzelfälle zu konzentrieren.

In einem noch größeren Scritt stellt Kapitel 7 ein neuartiges Modell der Kollisionsvermeidung menschlicher Fahrer auf Grundlage *aktiver Inferenz* vor. Aktive Inferenz ist ein Prinzip, das auf der Idee beruht, dass Agenten handeln, um eine Größe zu minimieren, die als freie Energie bezeichnet wird. Die Minimierung freier Energie schafft hierbei einen effektiven Ausgleich zwischen Erkundung und Ausnutzung. Unser Modell basierend auf aktiver Inferenz kann zahlreiche Aspekte menschlichen Verhaltens reproduzieren, darunter Reaktionszeiten und die gewählte Ausweichmanöver. Darüber hinaus zeigen wir eine starke Generalisierungsfähigkeit über verschiedene Szenarien hinweg, wobei die Modellparameter erfolgreich auf zuvor unbekannte Fälle übertragen werden können. Dies steht im Gegensatz zu vielen bestehenden Modellen, die eine szenariospezifische Parametrisierung erfordern, und deutet darauf hin, dass aktive Inferenz eine vielversprechende Grundlage für die Modellierung allgemeiner Aspekte menschlichen Verhaltens bietet.

Kapitel 8 untermauert das Potenzial der allgemeinen Anwendbarkeit dieses Ansatzes, indem zusätzliche Mechanismen integriert werden, die für Interaktionen in realen Verkehrssituationen entscheidend sind. Insbesondere wird das Modell erweitert, um den Einfluss interaktiver Verkehrsregeln (beispielsweise Vorfahrtsregeln) sowie expliziter Kommunikation zwischen Agenten zu berücksichtigen. Es wird gezeigt, dass diese Erweiterungen eine wichtige Rolle bei der Auflösung von Interaktionen an Kreuzungen spielen. Die problemlose Integration dieser Mechanismen unterstreicht zusätzlich die Flexibilität und Allgemeingültigkeit aktiver Inferenz.

Zusammenfassend verbessert diese Dissertation sowohl die Evaluierung bestehender Verhaltensmodelle als auch die Entwicklung eines neuen allgemeinen Modells des menschlichen Fahrverhaltens basierend auf aktiver Inferenz. Auf der Evaluierungsseite wird ein umfassender Rahmen eingeführt, der eine standardisierte und effiziente Validierung von Vorhersagemodellen mit besonderem Schwerpunkt auf der Leistung in sicherheitskritischen Situationen ermöglicht. Dies besitzt das Potenzial, die Entwicklung verbesserter Modelle zu beschleunigen, indem ihre Stärken und Schwächen klarer aufgezeigt werden. Auf der Modellierungsseite zeigt die Dissertation, dass kognitiv inspirierte Ansätze wie aktive Inferenz eine vielversprechende Alternative zu rein datenbasierten Methoden darstellen. Wesentliche Einschränkungen dieser Methoden, wie beispielsweise eine mangelnde Erklärbarkeit und ihre potenzielle Fragilität in kritischen Szenarien, können dadurch überwunden werden. Gleichzeitig ermöglichen sie tiefere Einblicke in die Prinzip-

ien, die menschlichem Verhalten zugrunde liegen. Dies weist den Weg zu vereinheitlichten Modellen menschlichen Verhaltens, die sicherere und effektivere Interaktionen zwischen autonomen Fahrzeugen und menschlichen Verkehrsteilnehmern unterstützen können.

1

Introduction

Over recent years, autonomous driving – alongside electric vehicles – has emerged as one of the most significant developments in the automotive industry, attracting billions of dollars in annual investment [2]. On one hand, this investment is driven by financial incentives (such as eliminating the cost of human drivers in taxis or logistics [3, 4]). On the other hand, autonomous driving also offers considerable potential for societal benefit, foremost being the potential reduction in traffic accidents. In particular, of the more than one million road deaths worldwide each year [5], over 90% are commonly attributed to some form of human error [6]. Admittedly, the exact proportion of these fatalities that could be prevented through autonomous driving remains debated [7]. However, a reasonably well-designed autonomous system should in principle be able to avoid the large share of accidents resulting from causes such as drunk driving, distraction, or speeding. As road traffic accidents are the leading cause of death among children and young adults [5] in many countries, the potential life-saving impact of reliable autonomous driving systems – even under conservative assumptions – is therefore substantial. In addition to preventing fatalities, widespread adoption could similarly reduce serious injuries and significantly limit property damage. Beyond improvements in safety, autonomous driving also promises a number of further advantages. These range from easier access to mobility for people currently unable to drive (whether because of young age or physical impairments), to reductions in traffic congestion and emissions through more efficient driving behavior, and increased productivity during travel time, as occupants can focus on other tasks instead of driving [8–10].

However, for autonomous vehicles to become a viable and widely accepted alternative to human drivers, achieving a lower accident rate alone will not be sufficient – and even this is far from being realized, for instance, in scenarios with difficult visibility [11]. They must also be capable of navigating complex interactions with other road users (such as in Figure 1.1) at a level comparable to competent human drivers. This includes the ability to coordinate and communicate implicitly in everyday traffic situations. To date, this remains a significant challenge. Early autonomous systems, in particular, were often characterized by an overly timid and conservative driving style [12, 13]. While such behavior may appear cautious, it can itself introduce risks: Human road users do not always antic-

1

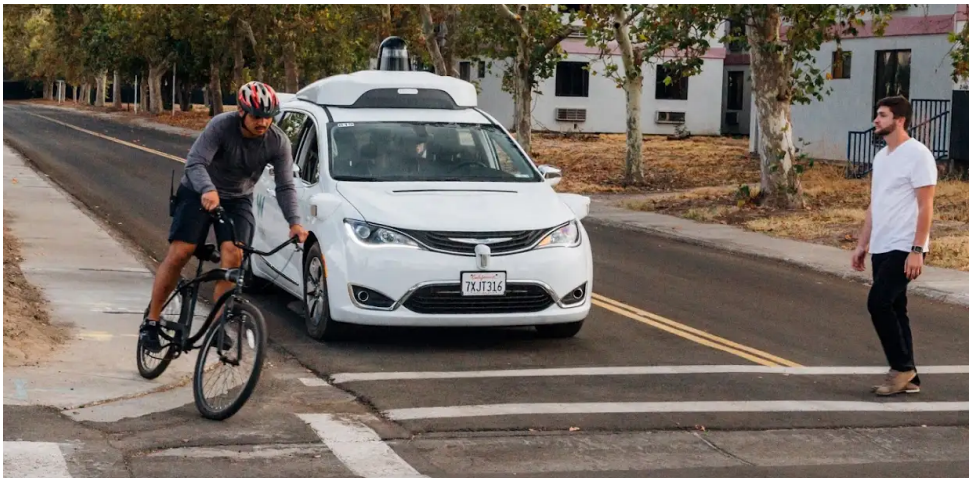


Figure 1.1: Illustration of an autonomous vehicle (AV) operating in a traffic environment with other human road users. Adjusted from Source: Waymo LLC [1]

ipate unusually hesitant maneuvers, which can lead to conflicts – for example, rear-end collisions involving autonomous vehicles [12, 14]. This highlights that caution alone does not guarantee the safe deployment of autonomous vehicles, even if efficiency is seen as secondary. Instead, a deeper understanding of human traffic participants is necessary to successfully integrate autonomous vehicles into mixed traffic environments.

A promising approach to this challenge can be found in human behavior models. On the one hand, such models can be used to predict the actions of other road users. By reducing the uncertainty about their future actions, an autonomous vehicle could avoid overly cautious autonomous driving (as shown in Figure 1.2a). In principle, this enables more confident and efficient decision-making – particularly in cases where other vehicles are likely to yield [15, 16]. Specifically, precise predictions about other agents' future trajectories expand the set of safe states the autonomous vehicle can exploit and, thereby, facilitate policies that are both safe and efficient. Crucially, this benefit depends on behavioral models also being sufficiently accurate and reliable, so as not to give the autonomous system a false sense of security (Figure 1.2b and c). On the other hand, behavior models can serve as stand-ins for human road users (*i.e.*, behavior reference models) in simulated environments, where they can provide two key functions. First, as reactive surrounding agents, they enable closed-loop testing and validation of autonomous vehicle control policies. This allows for both larger scale than on-road testing and more targeted evaluation of safety-critical crash or near-crash scenarios that are difficult or unsafe to reproduce in the real world [17, 18]. Second, they can provide a human baseline against which the performance of the autonomous vehicle can be compared – a crucial step for building consumer and regulatory trust in the safety of autonomous systems [19, 20]. Especially when used as a behavior reference, models capable of accurately reproducing human behavior would further benefit from the ability to explain that behavior as well. Specifically, explainable models can reveal the cognitive mechanisms underlying human decision making. This

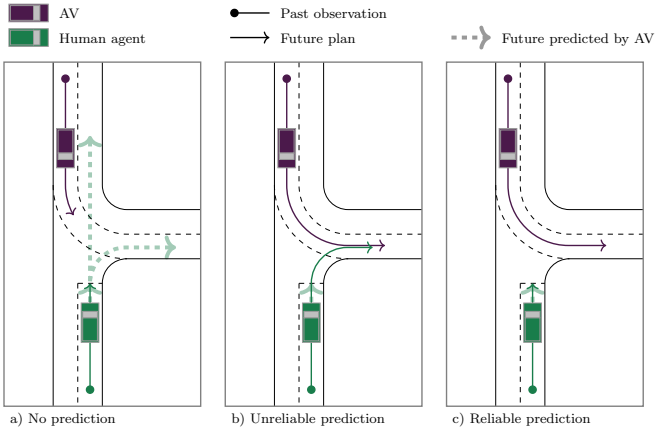


Figure 1.2: An example traffic scenario highlighting the need for accurate, precise, and reliable human behavior prediction models in autonomous driving. In this scenario, the purple autonomous vehicle (AV) wants to make a left turn at the shown intersection with priority. There, it has to interact with the green human-driven vehicle, which could either yield or proceed first (going straight or turning left). All of these human behaviors are clearly kinematically possible. a) In the case where the AV has no prediction model, it has to predict the human agent's future behavior (thick dashed lines) solely based on kinematic feasibility. As many of these predictions would result in a collision if the AV were to move ahead as intended, braking is the only option that guarantees safety. This approach is technically accurate, as it is highly unlikely for the human vehicle not to conform to these predictions. However, it is too imprecise to be of practical use. b) In the case where the AV is equipped with a precise but inaccurate model, it might now appear as if the AV can find an efficient resolution to this interaction. Specifically, given the high certainty that the human vehicle will yield, proceeding is considered safe. However, if this prediction is wrong (as the human agent intends to actually turn right), an unsafe scenario will occur, potentially resulting in a collision. c) Only with an accurate and precise model can the AV find a good solution to this interaction that is both safe and efficient. In this case, the AV can anticipate with high certainty that the human vehicle will will yield. Consequently, the AV can safely proceed with its plan of turning left.

would enable the targeted design of autonomous vehicle policies that are interpretable to humans and capable of influencing behavior in desired ways, facilitating smooth interactions with other road users [21, 22]. Moreover, explainability allows for easier identification of model failure modes [23] and assessment of the reasons behind specific behaviors. This can increase trust in the model and improve its generalizability to unseen scenarios [24]. Combined, sufficiently developed behavior models allow us to both understand and predict the actions of other road users. Specifically, providing insight into their intentions and the interactive dynamics that shape human driving can support safer interactions in mixed traffic.[1]

In the literature, a major focus for behavior modeling has been on the development of data-driven deep-learning methods used for the prediction of other road users' trajectories [15]. Initially, these were predominantly based on recurrent neural networks [28] and conditional variational autoencoders [29] architectures [30, 31]. Later, these recurrent neural networks were increasingly superseded by transformer-based [32] encodings [33–36]. Furthermore, alternatives to conditional variational autoencoders have become increas-

[1]While interactions between autonomous vehicles are also of great concern [25–27], they are out of scope for this thesis, which focuses on interactions between autonomous vehicles and human road users.

ingly common. Examples include normalizing flow-based models [37, 38], but the most prevalent approaches today are based on denoising diffusion models [39–42]. Nevertheless, there has also been research into the development of models based on behavioral theory that are more explainable than the "black-box" deep learning approach, such as game-theoretic models [43]. There has also been growing interest in the application of cognitive principles. For example, models using mechanisms such as evidence accumulation [44, 45] are used to explain human decision making in scenarios such as left turns at intersections [46] or pedestrian street crossing [47, 48]. However, many of those approaches – even when successful as accurate models – are relatively narrow in scope. In particular, they are often limited to specific agents in well-defined scenarios, with typical examples being pedestrian street crossing [49, 50] or highway lane changes [51, 52]. Other models are only able to reproduce certain aspects of human behavior, such as response times for braking [53] or takeovers from automated vehicles [54, 55].

While especially the deep learning approach has led to significant progress in prediction performance, even current models still struggle with generalizability and reliability in safety-critical scenarios [56]. Additionally, current human behavior prediction models are often evaluated primarily through simple metrics on standardized datasets, which may obscure failure modes in rare but safety-critical scenarios and provide limited insight into closed-loop performance. Part I of this thesis therefore focuses on the evaluation of human behavior prediction models. Concretely, we want to move towards holistic assessment methodologies that better reflect safety, robustness, and reliability in highly interactive and uncertain traffic situations. As seen in Figure 1.3, this part is split into four chapters. Chapter 2 introduces a simplified framework for benchmarking human behavior prediction models in gap acceptance scenarios. These scenarios are characterized by an agent without priority deciding whether to go in front of an oncoming agent with priority (accept the gap) or yield to them (reject the gap). The focus on such a well-defined scenario enables controlled and safety-critical comparisons of both trajectory prediction models and those predicting categorical behavior. Chapter 3 then expands this idea by introducing *STEP*, a comprehensive and scalable framework for evaluating human behavior prediction models. It covers a wide range of behaviors, datasets, metrics, and experimental protocols. Chapter 4 presents *ROME*, a sample-based density estimation method used in *STEP*. It enables the use of probability density-based evaluation metrics for models without explicit density representations. Lastly, Chapter 5 introduces novel adversarial attacks against trajectory prediction models that are both meaningful and kinematically feasible. When used in *STEP*, this allows for the evaluation of the robustness of human behavior prediction models against such attacks.

Meanwhile, while cognitively inspired models of human behavior in traffic have been proposed, they often lack a general underlying theory of human behavior. This limits their ability to achieve broader scope and general applicability. Consequently, to properly explore the potential of this approach, the thesis also aims to develop general human behavior models incorporating principled inductive biases (Part II). By doing so, we validate the underlying principles, improve generalization, and provide greater interpretability compared to both purely data-driven methods and prior cognitive science-based models. This part introduces three major contributions (Figure 1.3). First, Chapter 6 extends a cognitively plausible model of vehicle-pedestrian interactions to a more general model of

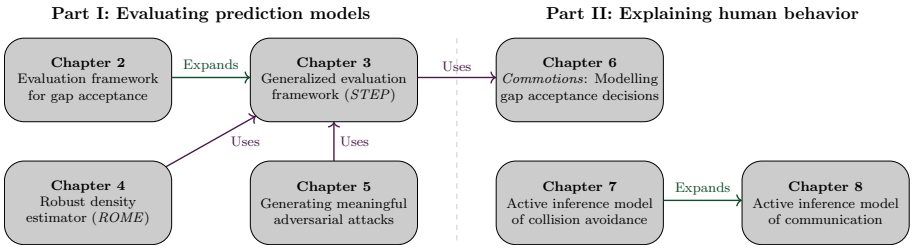


Figure 1.3: An overview of the chapters in this thesis and the connection between them.

human gap acceptance behavior. This model is evaluated across benign and safety-critical scenarios against a state-of-the-art trajectory prediction model, using *STEP*. While this is not a fully general model of human behavior, it shows that the underlying cognitive principles allow the expansion of originally narrow models to a wider domain. In a next step, we further increase the scope by utilizing the more general concept of active inference [57]. In particular, Chapter 7 presents an active inference model of human driving behavior in collision avoidance scenarios. There, we introduce several novel mechanisms such as evidence-accumulation-based delays and norm-conditioned behavior prediction. The model and these novel contributions are validated through comparisons to human data in multiple scenarios. Finally, Chapter 8 extends this active inference framework to interactive conflict and space-sharing scenarios. Concretely, we implement and investigate the influence of explicit communication and deontic cues such as traffic signs, as well as the risk introduced by malign agents using deceptive communication.

Together, these contributions seek to advance both the assessment and the design of human behavior models by addressing complementary challenges in evaluation and cognitive modeling. On the one hand, the introduction of more expressive and robust model evaluation methodologies can expose failure modes in rare and safety-critical situations. On the other hand, the use of cognitively motivated modeling principles can be used to avoid those failure modes. This promises to improve generalization, interpretability, and robustness of human behavior models. Considered jointly, this work supports the principled development and deployment of human behavior prediction models that are reliable in safety-critical scenarios, providing a crucial building block for safe and efficient autonomous driving systems.

I

Evaluating Prediction Models

As pointed out in the Introduction (Chapter 1), accurate and reliable prediction of human behavior is an important aspect of the safe and efficient operation of autonomous vehicles. While the design of human behavior prediction models has seen significant development [15, 58–60], their evaluation is often still lagging behind. Most commonly, models are evaluated based on their performance on a randomly selected subset of a larger dataset, often using only a handful of metrics such as displacement error or negative log-likelihood. However, one issue with these approaches is the symmetry with which those metrics consider different types of prediction errors [61, 62]. Specifically, these metrics ignore the varying consequences of different false predictions [63]. For example, a deviation that leads to the prediction of a collision where none would occur would very likely not result in anything more than overly conservative behavior. Meanwhile, a similar prediction error that misses a collision that would occur is assigned the same metric value but could instead lead to a potentially dangerous situation. In another case, incorrect longitudinal predictions are far less likely to have significant consequences than incorrect lateral predictions in highway driving. Furthermore, standard metrics put insufficient weight on the interactions between vehicles, where trajectroy predictions are most needed [64]. Overall, such metrics do not test the ability of models to differentiate between distinct human behaviors [65, 66].

Additionally, the random selection of test sets poses another problem. As most scenarios included in large datasets are benign and of low criticality [67, 68] (*i.e.*, few, if any, meaningful interactions between road users occur), random sampling leads to evaluations being dominated by such scenarios [69]. Consequently, a model that is highly capable of predicting benign scenarios but unreliable in rare edge cases – where unusual behavior, such as human error or high aggression, can pose a real threat of causing accidents [70, 71] – could still achieve a better metric score than a model that is far more reliable in such critical scenarios but slightly worse in common benign cases. As a result, without further insight, the conclusions that can be drawn from such evaluations are limited.

Furthermore, on most common benchmarks used for evaluating prediction models (such as Argoverse [72] or the Waymo Open Motion Dataset [73]), models are typically trained and evaluated only once. Consequently, comparisons between different models often ignore potential noise introduced during training, such as random parameter initialization or the order in which training samples are presented. As a result, the same model architecture can produce different model instances with varying performance characteristics. Therefore, performance differences observed between single instances of different model architectures may not be statistically significant or may even be statistical outliers [38]. Nevertheless, most papers proposing new model architectures still rely on such single-instance comparisons to claim the superiority of their novel approaches [30, 40, 74–83].

Finally, by training and testing models on randomly selected subsets of the same dataset [67], their ability to generalize to unseen scenarios is not sufficiently evaluated. This is particularly problematic, as robustness in adverse environments is a critical property for real-world applications [84, 85]. While some recent work has addressed the robustness of prediction models under outlier conditions, these efforts are mostly focused on naive adversarial attacks [86–88]. In such an attack, the general goal is to add small perturbations to the observed past behavior of an adversarial agent, which, when given

as input to the behavior prediction model, result in predictions that diverge from the observed future behavior. However, in those particular examples, the constraints imposed on these attacks are often too loose to preserve meaningfulness (*e.g.*, comparing observed ground-truth behavior to predictions based on perturbed past observations that make it kinematically impossible for the vehicle to come close to that ground-truth behavior lacks significance). Consequently, more meaningful robustness evaluations are required to properly assess the capabilities of human behavior prediction models.

Thesis contribution

Consequently, the first part of this thesis is dedicated to addressing these evaluation shortcomings. Specifically, in Chapter 2, we propose a novel benchmark for evaluating human behavior prediction, focused on gap acceptance scenarios. These scenarios, in which a human road user without priority must decide whether to cross the path of an oncoming road user (accept the gap) or wait and let them pass (reject the gap), represent a major class of human traffic interactions [89]. By focusing on such scenarios – for which we also propose a unified method for estimating criticality – our benchmark enables comparisons between both data-driven trajectory prediction models and simpler knowledge-based models that predict the binary decision of accepting or rejecting the gap. Additionally, by restricting the evaluation to a well-defined scenario, we can exert greater control over the selection of the test set, allowing us to specifically focus the evaluation on the most safety-critical situations. Furthermore, we also introduce a novel metric that takes into account the asymmetrical consequences of different types of prediction errors in such scenarios.

In an effort to generalize this idea, in Chapter 3 we introduce *STEP*, a comprehensive and general framework for the evaluation of human behavior prediction models. Compared to the earlier work in Chapter 2, we allow for the inclusion of a wider range of classifiable behaviors (*e.g.*, instead of only gap acceptance decisions, we also evaluate predictions based on the direction taken at intersections). Furthermore, we enable the inclusion of larger datasets that the previous framework was unable to handle, relying on disk storage when RAM is insufficient. Lastly, we further expand control over dataset splitting by implementing methods such as cross-validation, allowing for straightforward assessment of the statistical significance of performance differences between model architectures. We then demonstrate the capability of this framework in a number of experiments, where we, among others, test a model's generalizability across different datasets or the variability of model performance introduced by randomness in the training process.

Another novelty of STEP is the comprehensive inclusion of probability density-based metrics (such as negative log-likelihood), which we make applicable to models without an explicit density representation. This is achieved by the inclusion of *ROME* (Chapter 4), a robust method for estimating probability densities based on samples generated by an arbitrary distribution. While density estimators such as *ROME* have many general applications, in the context of human behavior prediction they enable a more comprehensive evaluation of such models, using metrics like expected calibration error [90] or Brier score / displacement-based metrics [72]. In this work, we also introduce a methodology for evaluating such density estimators that does not require a queryable representation of the estimated distribution and operates solely on samples drawn from the ground-truth distribution. Using these methods, we compare *ROME* against a number of state-of-the-art

methods on multiple different problems, including sets of human trajectories.

Finally, *STEP* also has the capability to perform robustness evaluations against adversarial attacks (Chapter 5) to further enhance the evaluation capabilities of the framework. While different methods for adversarial attacks are included, we want to specifically highlight a novel method for generating meaningful and kinematically feasible attacks introduced in more detail in Chapter 5. Specifically, we ensure the meaningfulness of these robustness evaluations by introducing constraints during the adversarial attack that ensure the resulting metrics evaluating the model performance on those perturbed inputs are well-defined and meaningful. In particular, as these metrics are typically based on comparisons with unperturbed ground-truth behavior, we constrain the adversarial perturbations such that the ground-truth behavior remains kinematically reachable by the adversarial agent. Furthermore, by perturbing the underlying control states such as acceleration instead of positions directly, we ensure these adversarial trajectories are physically realizable in the real world. We test this attack on multiple datasets against state-of-the-art prediction models to assess whether the focus on feasibility and meaningfulness can degrade model performance. Additionally, we investigate whether it is necessary for such adversarial attacks to be performed directly against the evaluated model, or whether these attacks transfer sufficiently well between different models.

2

Benchmarking Behavior Prediction Models in Gap Acceptance Scenarios

As discussed in the previous introduction, autonomous vehicles require reliable behavior prediction models to be both safe and efficient. However, the evaluation of such models is commonly too simplistic, ignoring the asymmetric importance of prediction errors and the heterogeneity of the datasets used for testing. In this chapter, we hence examine the potential of recasting interactions between traffic participants as gap acceptance scenarios and evaluating models in this structured environment. By developing a framework aiming to facilitate the evaluation of various prediction models, using for instance metrics designed specifically for gap acceptance scenarios, we show that many state-of-the-art models are unreliable in the most safety-critical situations.

This chapter is based on *J. F. Schumann, J. Kober, A. Zgonnikov. "Benchmarking behavior prediction models in gap acceptance scenarios." IEEE Transactions on Intelligent Vehicles 8 (2023).* [16]

2.1 Introduction

Successfully implementing autonomous driving is one of the key technical challenges faced by the automotive industry as well as large parts of the research community, with tens of billions of dollars invested in recent years towards this goal [2]. The provision of those funds is motivated by several benefits promised by this technology. The foremost of these is safer driving, expressed by a significant decrease in accidents and, correspondingly, a reduction of bodily harm and financial losses. Additional advantages are also expected, such as more accessible mobility for people unable to drive or an easing of road congestion and traffic [8–10].

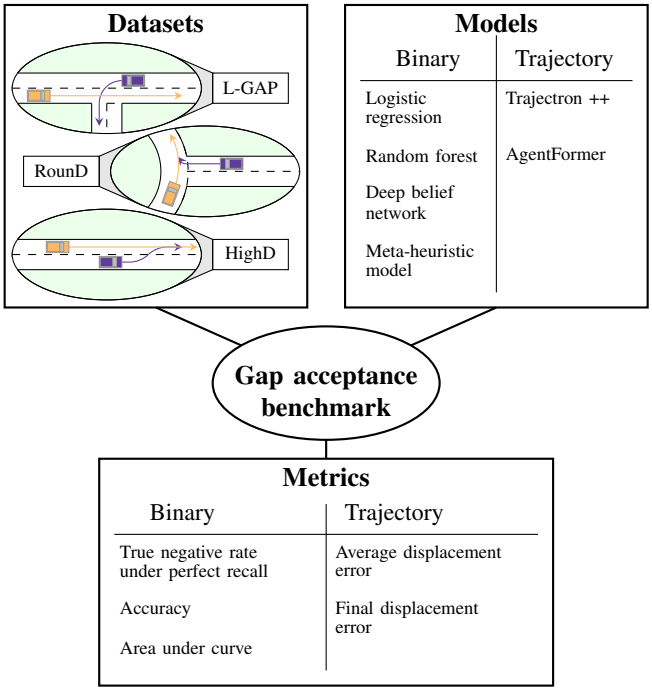


Figure 2.1: The proposed framework allows researchers to evaluate the performance of various prediction models for human behavior according to several metrics on different datasets that include gap acceptance scenarios. In this work, three datasets, six models, and four metrics (dealing both with binary and trajectory predictions) are implemented.

But despite all these investments, autonomous vehicles still suffer from many problems preventing widespread use [12, 13]. One such problem is their timidity in interactions with human traffic participants, caused by the uncertainty about the future behavior of those human agents. This uncertainty can prevent the autonomous vehicle from taking the most time-efficient actions if the resulting probability of a crash or near-crash is too high, resulting in the cautious driving style observed. Paradoxically, this can also be a safety risk, as such caution by an autonomous vehicle is often not expected by the surrounding humans, which can result in accidents such as being rear-ended [12, 14].

To reduce this uncertainty and to allow for a more efficient driving style without com-

promising on safety requirements, behavior prediction models can be used [15, 43], which project the future position of traffic participants. Those can range from models able to deal with any kind of traffic participant [30, 33, 34] to others focused on predicting the behavior of a specific kind of participant, such as cars [91–97] or pedestrians [98–102].

However, the utility of those models – primarily designed to minimize the necessary trade-off between safety and efficiency in trajectory planning – is questionable, as the common methods for their evaluation diverge from the models' purpose. First, most common metrics for evaluating prediction models, such as the final or average displacement error, ignore that the consequences of a false prediction are inherently asymmetric [61, 62]. For example, on a highway, wrong longitudinal predictions are far less dangerous than wrong lateral predictions, which might result in an autonomous vehicle reacting to a lane change too late. Similarly, such metrics also lack the ability to evaluate how good models are at capturing distinct human behaviors [65, 66]. Second, the common approach of randomly selecting test cases from datasets [30, 34] is problematic due to the heterogeneity of those datasets, which typically include samples that can vary widely in their importance and difficulty. Such samples can range from a single vehicle following a lane to complex space-sharing conflicts with multiple agents at unsignalized intersections, where the behavior of human agents is often multi-modal and can change rapidly. Rare edge cases, where some traffic participants are very aggressive or even violate traffic rules and accidents are far more likely [70, 71], are also possible. But with randomly selected test cases, potentially poor performance in the most important situations can be compensated by good performance in less important but more numerous ones. For these reasons, a model that is unreliable in actual safety-critical situation might still appear promising, making the whole evaluation meaningless and hampering further progress.

One possible approach to overcome these issues is including a path planning algorithm in evaluations, as suggested by Ivanovic and Pavone [61]. However, this adds further computational loads to an evaluation and only addresses the symmetry of common metrics, neglecting the varying difficulty and importance between testing samples. To cover both these problems, we suggest instead narrowing the evaluation to the most critical situations. In particular, we focus the evaluation of behavior prediction models on *gap acceptance* scenarios, a concept that encompasses most of the safety critical interactions between autonomous vehicles and humans [89]. In a gap acceptance scenario, an autonomous vehicle follows a particular trajectory over which a second traffic participant (*e.g.*, a pedestrian or another vehicle) can move either in front of or behind the autonomous vehicle. Here, the first option (*i.e.*, the human accepting the gap) would require the autonomous vehicle to potentially alter its trajectory planning, while the latter one of rejecting the gap would not. Due to the narrow focus, estimating the importance and difficulty of a particular situation can become much more straightforward. Additionally, as the human has only two options to decide between, such gap acceptance scenarios allow the usage of simple binary prediction models to estimate if the human behavior requires an adjustment of trajectory planning.

Many binary prediction models have been developed for gap acceptance scenarios. However, those mostly focus and are trained on a specific scenario, such as the street crossing behavior of pedestrians [47, 49, 50, 103], the crossing behavior of cars at intersections [46, 104–107], or lane change decisions on high ways [51, 52, 108–110]. Addition-

ally, the development of those models still suffers from similar problems as the trajectory prediction models, such as the anisotropy of common metrics like accuracy. Likewise, a random selection of test cases [46, 47] and neglect of the varying importance of different samples are also common. Additionally, in contrast to trajectory prediction models, which are commonly compared to each other on accepted benchmarks (such as on the ETH dataset [30, 33, 34] when predicting pedestrian crowds), an equivalent benchmark does not exist for binary prediction models [58]. Instead, those models are mostly trained and tested on datasets exclusive to the respective work and are – if at all – only compared against a small number of other selected models [46, 47, 103, 106, 107, 111–115].

Our goal in this work is to overcome these limitations of the current literature on both binary and trajectory prediction models and enable a meaningful evaluation of these models in gap acceptance scenarios. Such an evaluation cannot only make the development of trajectory prediction models more goal-oriented but also help determine to what extent the inclusion of specialized binary prediction models can improve the performance and reliability of general trajectory prediction models. To that end, this paper makes three main contributions:

- We develop a formal description of the gap acceptance process that applies to all possible gap acceptance scenarios. This description includes a detailed timeline of gap acceptance (Section 2.2), which serves as a foundation for methods to estimate the criticality concerning the safety of each sample, which is a fundamental requirement for selecting meaningful test cases.

- We devise a framework for evaluating behavior prediction models in gap acceptance scenarios. This novel framework allows the integration of varied gap acceptance datasets, models, and evaluation metrics (Section 2.3). It is inspired by similar works by Müller *et al.* for computer-based image retrieval algorithms [116], by Zaffar *et al.* in the field of visual place recognition [117], or Cao *et al.* for evaluating robustness to adversarial attacks of trajectory prediction models [86]. This approach would allow one to test and compare models in a chosen environment more easily, no matter if they make binary or trajectory predictions. For example, the proposed framework enables evaluating trajectory prediction models using binary metrics in gap acceptance scenarios, something that has not been previously investigated. Simultaneously, the framework allows precise control over the splitting of the data into training and testing samples to comprehensively evaluate the models' reliability in the most difficult gap acceptance situations.

- Using the proposed framework, we compare six prediction models in their performance on three gap acceptance datasets (Section 2.4 and Figure 2.1). First, this demonstrates the general viability of our proposed method, by including models making and metrics evaluating both binary and trajectory predictions. Second, it allows us to test the reliability of those models specifically in safety-critical edge cases, which is currently missing from the literature. Third, it allows the testing of the hypothesis that including dedicated binary models for gap acceptance problems could improve the performance of state-of-the-art trajectory prediction models. Lastly, this gives researchers easy access to an already implemented baseline to compare their own behavior prediction models to.

2.2 Defining gap acceptance

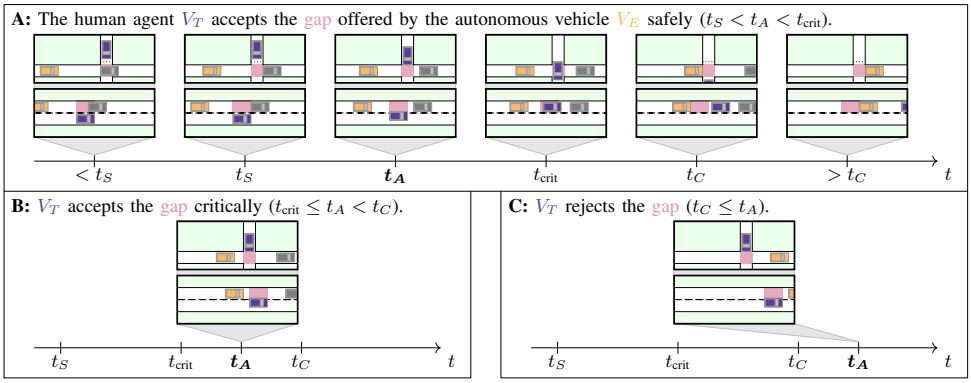


Figure 2.2: The characteristic time-points of the gap acceptance process – defined by the relation of the agents to the contested space (in purple) – of two different examples of gap acceptance, intersection crossing (upper panels) and lane changing (lower panels). In both examples, the autonomous vehicle V_E (in red) offers a gap to the human-driven vehicle V_T (in blue). In total, three cases are possible (**A**–**C**), depending on t_A, *i.e.*, the time the target vehicle enters the contested space. In **B**, the accepted gap decision by the human is considered to be unsafe, as V_E cannot guarantee the avoidance of a crash, having potentially not enough time for braking. Meanwhile, in **C**, it might be possible that V_T crashes into V_E.

To estimate the difficulty and control the importance of prediction tasks over disparate datasets, a coherent formal definition of gap acceptance scenarios is needed. Here we propose such a definition.

In a gap acceptance scenario, an autonomous vehicle V_E – also referred to as the ego-vehicle – plans to follow along a certain trajectory P_E along which it has the right of way. This trajectory overlaps with the trajectory P_T of another, human-controlled vehicle V_T (also named target vehicle). Such an overlap might, for example, happen at unsignalized intersections, where the agents move along crossing streets or on highways, where V_T wants to merge into the faster lane along which V_E is driving. In such situations, V_T can decide to move onto P_E either in front of or behind V_E, *i.e.*, to accept or reject the gap offered by V_E. We assume that V_E has the right of way along P_E, as otherwise, traffic rules would obligate it to preemptively yield.

Under these conditions, a gap acceptance scenario is characterized by the spatiotemporal relation between the agents towards the so-called contested space [89]. There, the trajectories P_E and P_T would start to overlap, making this the location of a potential collision. An example is the overlap of two crossing lanes at an intersection. However, in specific scenarios (such as changing lanes on highways), the exact location of the meeting point of P_E and P_T can be at the discretion of the human agent V_T and therefore be unknown before the actual decision. In such cases, we then place the contested space under the assumption that V_T would decide to accept the gap immediately. For example, in the scenario of highway lane changes, the contested space would therefore move in parallel to V_T, only stopping to move once V_T starts to enter the lane of V_E.

The following time points then characterize the gap acceptance process (illustrated together with the contested space in Figure 2.2):

t_S: At the **s**tarting time t_S, there is no longer any other vehicle along P_E in between V_E and the contested space. This is primarily the case when the vehicle preceding V_E leaves the contested space, but other options are imaginable, like the vehicle in front of V_E leaving P_E.

t_C: At t_C, V_E starts to enter the contested space, **c**losing the gap.

$t_{\underline{C}}(t)$: A prediction of t_C by the ego vehicle, made at t, needed to allow gap size estimations during online applications. While this is scenario-dependent, the following condition has to be satisfied so that an open gap can still be characterized as such, even if V_E is moving away from the contested space:

$$\operatorname{sgn}\left(t_{\underline{C}}(t) - t\right) = \operatorname{sgn}(t_C - t)\,.$$

t_{crit}: The last time V_E can safely prevent a collision even in the case of malicious behavior by V_T; *e.g.*, at this point, a safe braking process could bring V_E to a stop before the intersection. t_{crit} can be formalized in the following condition:

$$\Delta t_D(t) = t_{\underline{C}}(t) - t - t_{\text{brake}}(t) = 0 \tag{2.1}$$

Here, the required braking time t_{brake} is not based on the maximum deceleration V_E is technically capable of, but instead, one that is considered safe. The time point t_{crit} is also the last time a prediction can be considered useful for further trajectory planning.

t_A: At t_A, V_T enters the contested space, potentially **a**ccepting the gap.

We count V_T as rejecting the gap if V_E is allowed to move first onto the contested space, *i.e.*, if $t_C \leq t_A$. If this is not the case and the human moves first ($t_A < t_C$), the gap is considered accepted.

2.3 Benchmarking gap acceptance models

After defining the fundamental characteristics of a gap acceptance scenario, we will use this groundwork to build a framework for benchmarking gap acceptance models. This framework should allow for the performance assessment of a prediction model M on a dataset D according to evaluation metric E. The following requirements need to be met for such an assessment to be both meaningful and possible for as many of the aforementioned modules as possible:

R 1 The time point t_0 of a prediction must be controllable, as it influences not only the difficulty of the prediction but also its importance due to changing consequences of a false prediction.

R 2 To evaluate models in critical situations, the framework should allow control over splitting all available samples into training and testing sets.

R 3 Models producing (as well as metrics evaluating) for example binary or trajectory predictions should fit into the framework. Therefore, the framework should allow transformations between those forms of model output.

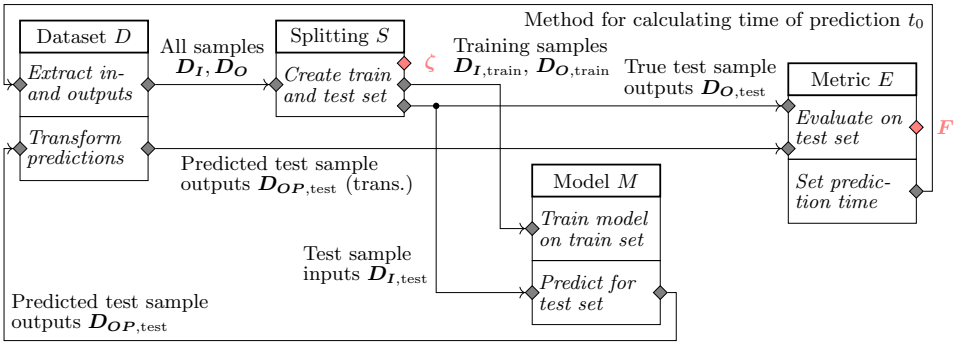


Figure 2.3: To evaluate a model M on a dataset D with the metric E and splitting method S, a method for determining the prediction time t_0 has to be chosen first (a detailed description can be found in Section 2.3.1). This method is used to extract input and output trajectories ($\boldsymbol{D}_I$ and $\boldsymbol{D}_O$ respectively) from the dataset D (2.3.2). Those samples are split in training and testing set using the splitting method S (2.3.3), with the training one ($\boldsymbol{D}_{I,\text{train}}$, $\boldsymbol{D}_{O,\text{train}}$) used to train a model M (2.3.4). Subsequently, predictions $\boldsymbol{D}_{OP,\text{test}}$ are made for the test samples $\boldsymbol{D}_{I,\text{test}}$ with the trained model (2.3.5). It might be necessary to transform these predictions into another form (2.3.6), before the metric E compares them to the true outputs $\boldsymbol{D}_{O,\text{test}}$ (2.3.7). These steps produce two outcomes (red diamonds): the similarity $\boldsymbol{\zeta}$ between training and test data, provided by S, and the model performance $\boldsymbol{F}$ according to metric E.

Considering these requirements, seven functionalities will constitute the proposed framework. Figure 2.3 illustrates these functionalities and their mutual dependencies. These functionalities are grouped in four modules (dataset D, splitting method S, model M, and Metric E); in this section we describe them in the order in which they are employed in the process of a single evaluation.

2.3.1 Setting the prediction time – Metric E

To satisfy requirement **R 1**, this functionality enables the selection of the time-point t_0 at which the prediction has to be made. As the prediction time influences both the importance of such predictions and the meaningfulness of different metrics, this functionality is attached to the metric module.

Specifically, for $t_0 \ll t_{\text{crit}}$, the consequences of a wrong prediction are generally minor, as time is left to wait for future information before more significant changes to trajectory planning are necessary. Furthermore, even if the target vehicle would immediately accept the gap after t_0, the necessary response is likely neither uncomfortable nor risky. Consequently, symmetric metrics can be used here.

For $t_0 \approx t_{\text{crit}}$ however, no more time for further observations is left, resulting in far more severe consequences for both false positive and false negative predictions. A false positive prediction would unnecessarily result in a harsh and uncomfortable braking maneuver. Meanwhile, a wrong negative prediction leads to an unsafe gap acceptance maneuver, with the safety of the interaction between V_E and V_T no longer in the control of the autonomous vehicle V_E. Accidents or the need for dangerous emergency maneuvers, which could result in material damage or even bodily harm, are then possible. As the latter should be avoided at all costs, a false negative prediction at this time is far more consequential, which should be reflected in the evaluation metric.

Currently, three methods are implemented into the framework to determine t_0:

- *Prediction at the initial opening of the gap*: $t_0 = t_S$. The prediction is made when the gap first appears, and this is the baseline most commonly used in the literature [46, 51, 111].

- *Prediction at gaps with fixed size*: $t_0 = \min\{t \mid t_{\underline{C}}(t) - t = \Delta t\}$. The prediction is made when the gap offered has a uniform duration Δt, which should make every prediction equally difficult due to a similar prediction horizon.

- *Last useful prediction for critical gaps*: $t_0 = t_{\text{crit}} - t_\epsilon$. The prediction is made at the last point in time when it would still be useful, with t_ϵ being used to allow time for calculations.

Here, t_0 has to be calculated without hindsight knowledge for online predictions at a time when t_C or t_A are not known. Importantly, the method for determining the time t_0 can impact the size of the resulting dataset (Table 2.2) due to the condition from Equation (2.3). Namely, for constant gap size ($t_0 = \min\{t \mid t_{\underline{C}}(t) - t = \Delta t\}$), the number of available samples will be reduced, as all gaps with an initial smaller gap size ($t_{\underline{C}}(t_S) - t_S < \Delta t$) will be excluded. The same is the case for gaps already accepted before t_0. For critical gap sizes instead (*i.e.*, $t_0 = t_{\text{crit}} - t_\epsilon$), all gaps accepted before t_{crit} will be excluded, leading to extremely biased datasets, sometimes even removing all accepted gaps.

2.3.2 Extracting input and output – Dataset D

Next, the input and output data for each sample are extracted from a given trajectory $\boldsymbol{X_T}$, which includes positions at different time points $\boldsymbol{T}$ from different actors $\boldsymbol{V} = \{V_E, V_T, V_1, \ldots\}$:

$$\begin{aligned} \boldsymbol{X_T} &= \left\{\boldsymbol{x}(t) \mid t \in \boldsymbol{T}\right\}, \text{ where } I_T = [\max \boldsymbol{T}, \min \boldsymbol{T}] \\ \boldsymbol{x}(t) &= \left\{\boldsymbol{x}_i(t) \mid V_i \in \boldsymbol{V}\right\} \\ \boldsymbol{x}_i(t) &= \left(x_i(t), y_i(t)\right) \in \mathbb{R}^2 \end{aligned}$$

This functionality, requiring access to the raw data from the scenario and thus being part of the dataset module, consists of ten consecutive steps:

1. $t_{\underline{C}}$ and t_{brake} are estimated at every time point in $\boldsymbol{T}$, which is scenario-dependent.

2. The characteristic time points t_S, t_A and t_C are determined. Using the scenario-specific conditions C_S, C_C, and C_A respectively (C_S is true if the gap is offered, C_C and C_A are true if respectively the ego vehicle V_E and target vehicle V_T are inside the contested space), one then has to to find the specific times $\boldsymbol{T}_{C_i} = \left\{t \mid C_i(t) \forall\, t \in I_{\boldsymbol{T}}\right\}$ at which those conditions are true (*e.g.*, $\boldsymbol{T}_{C_A}$ is the time during which V_T is inside

the contested space); from this, the characteristic time points are extracted:

$$t_S = T_S(\boldsymbol{X_T}) = \begin{cases} \min \boldsymbol{T} & \boldsymbol{T}_{C_S} = \emptyset \\ \max \boldsymbol{T}_{C_S} & \text{else} \end{cases}$$
$$t_C = T_C(\boldsymbol{X_T}) = \begin{cases} t_{\underline{C}}(\max \boldsymbol{T}) & \boldsymbol{T}_{C_C} = \emptyset \\ \min \boldsymbol{T}_{C_C} & \text{else} \end{cases} \tag{2.2}$$
$$t_A = T_A(\boldsymbol{X_T}) = \begin{cases} \max \boldsymbol{T} + t_\epsilon & \boldsymbol{T}_{C_A} = \emptyset \\ \min \boldsymbol{T}_{C_A} & \text{else} \end{cases}$$

Here, a sample is excluded from the dataset, if no decision can be observed (*i.e.*, if $\boldsymbol{T}_{C_C} = \emptyset \wedge \boldsymbol{T}_{C_A} = \emptyset$).

3. The binary decision a, with $a = 1$ for accepted gaps ($t_A < t_C$) and $a = 0$ for rejected gaps, is extracted.

4. t_{crit} is extracted next, with

$$t_{\text{crit}} = \begin{cases} t_S & \Delta t_D(t_S) \leq 0 \\ t_A + t_\epsilon & \min\{\Delta t_D(t) \,|\, t_S \leq t < t_A\} > 0 \\ t_D & \text{else} \end{cases},$$

where

$$t_D = \min\left\{t \,|\, t > t_S \wedge \Delta t_D(t) = 0\right\}$$

satisfies both requirements in Equation (2.1).

5. The time of prediction t_0 is calculated accordingly to the method chosen previously (2.3.1). Only samples that meet the condition

$$t_S \leq t_0 < \min\{t_A, t_{\text{crit}}\} \tag{2.3}$$

are included in the final dataset, to ensure that gaps are already offered, V_T has not made a decision yet, and that the prediction is still useful.

6. The number of input time-steps n_I and the time-step size δt are chosen.

7. One also has to determine the number of output time steps n_O:

$$n_O = \left\lceil \frac{t_C - t_0}{\delta t} \right\rceil \tag{2.4}$$

The resulting prediction horizon $n_O \delta t$ is therefore large enough to see the outcome of the gap acceptance scenario, *i.e.* the acceptance or closing of the gap.

8. Based on t_0, n_I, n_O, and δt, the time-steps for input and output data are selected, named $\boldsymbol{T}_I$ and $\boldsymbol{T}_O$ respectively:

$$\begin{aligned} \boldsymbol{T}_I &= \{t_0 + i\delta t \,|\, i \in \{-n_I + 1, \dots, 0\} \\ \boldsymbol{T}_O &= \{t_0 + i\delta t \,|\, i \in \{1, \dots, n_O\}\} \end{aligned} \tag{2.5}$$

9. For those time-steps, the input trajectories $\boldsymbol{X}_{\boldsymbol{T}_I}$ and output trajectories $\boldsymbol{X}_{\boldsymbol{T}_O}$ are extracted from $\boldsymbol{X}_{\boldsymbol{T}}$, using interpolation if necessary.

10. Certain domain information k is collected, for instance, the location at which the trajectories $\boldsymbol{X}_{\boldsymbol{T}}$ were collected or the test subjects involved in gathering the data.

The input data $\boldsymbol{D}_{\boldsymbol{I}}$ then includes from each sample the input trajectory $\boldsymbol{X}_{\boldsymbol{T}_I}$ and the corresponding time-steps $\boldsymbol{T}_I$. Meanwhile, the output data $\boldsymbol{D}_{\boldsymbol{O}}$ takes the output trajectory $\boldsymbol{X}_{\boldsymbol{T}_O}$ and the corresponding time-steps $\boldsymbol{T}_O$, as well as the binary decision a, the time of accepting the gap t_A, and the domain information k from each sample.

2.3.3 Creating training and testing set – Splitting method S

To fulfill requirement **R 2**, the splitting method S, separating the given samples created in the previous step (2.3.2) into training and testing sets, is a crucial part of the framework. As this functionality should be independent of the scenario, it is part of the separate splitting module. Examples for this range from random splitting to methods taking into account all the information in $\boldsymbol{D}_{\boldsymbol{I}}$ and $\boldsymbol{D}_{\boldsymbol{O}}$.

Besides the potential similarity measure ζ of training samples to the training set, this functionality creates the training data $\boldsymbol{D}_{\boldsymbol{I},\text{train}}$ and $\boldsymbol{D}_{\boldsymbol{O},\text{train}}$ as well as the test data $\boldsymbol{D}_{\boldsymbol{I},\text{test}}$ and $\boldsymbol{D}_{\boldsymbol{O},\text{test}}$.

2.3.4 Training the model on the training set – Model M

After splitting the samples into training and testing sets (2.3.3), the model has to be trained on the training set ($\boldsymbol{D}_{I,\text{train}}$, $\boldsymbol{D}_{O,\text{train}}$), which is one of the functionalities of the model module that has to be individually implemented for each model. If the model, for instance, requires input velocities, extracting those from the given position data $\boldsymbol{X}_{\boldsymbol{T}_I}$ and $\boldsymbol{X}_{\boldsymbol{T}_O}$ is done here.

2.3.5 Making the predictions for the testing set – Model M

For every sample from the input testing set $\boldsymbol{D}_{\boldsymbol{I},\text{test}}$, a prediction $\boldsymbol{d}_{\text{pred}}$ is made by the model trained previously (2.3.4), with all $\boldsymbol{d}_{\text{pred}}$ constituting the set of predictions $\boldsymbol{D}_{\boldsymbol{OP},\text{test}}$. As such predictions rely on a trained model, this functionality is also part of the model module. Depending on this model, each prediction $\boldsymbol{d}_{\text{pred}}$ might take different forms. Three different forms of stochastic predictions are implemented into the framework.

- Binary prediction: $\boldsymbol{d}_{\text{pred}} = a_{\text{pred}}$, *i.e.*, only the probability $a_{\text{pred}} \in [0, 1]$ of V_T accepting the offered gap is predicted.

- Timing prediction: $\boldsymbol{d}_{\text{pred}} = \{a_{\text{pred}}, \boldsymbol{t}_{A,\text{pred}}\}$, *i.e.*, not only a_{pred} is predicted but also the time $\boldsymbol{t}_{A,\text{pred}}$ at which the gap acceptance might take place. Specifically, $\boldsymbol{t}_{A,\text{pred}}$

is expressed using decile values:

$$\boldsymbol{t}_{A,\text{pred}} = \boldsymbol{Q}_9(\mathcal{T}_A) = \left\{Q_{\mathcal{T}_A}(p) \,|\, p \in \{0.1, 0.2, \ldots, 0.9\}\right\} \in \mathbb{R}^9 \tag{2.6}$$

Here, $Q_{\mathcal{T}_A}$ is the quantile function associated with this underlying distribution of $\mathcal{T}_A$, which might for example be expressed as a set of individual time points t_A. This means that with a likelihood of p, $t_A < Q_{\mathcal{T}_A}(p)$ will be the case, if the human decides to accept the gap ($t_A < t_C$):

$$P\left(t_A < Q_{\mathcal{T}_A}(p) \,|\, t_A < t_C\right) = p$$

As the likelihood of accepting the gap is given by a_{pred}, one can get:

$$\begin{aligned} P\left(t_A < Q_{\mathcal{T}_A}(p)\right) &= P\left(t_A < Q_{\mathcal{T}_A}(p) \,|\, t_A < t_C\right) P(t_A < t_C) \\ &= p\, a_{\text{pred}} \end{aligned}$$

As t_A is predicted under the assumption that the gap is accepted, every decile value in $\boldsymbol{t}_{A,\text{pred}}$ should be smaller than t_C and larger than t_S.

- Trajectory prediction: $\boldsymbol{d}_{\text{pred}} = \boldsymbol{X}_{\boldsymbol{T}_O,\text{pred}}$, *i.e.*, the full trajectory of V_T is predicted. This prediction consists of n_p trajectories $\boldsymbol{X}_{\boldsymbol{T}_O,p}$ – all equally likely – to represent probabilistic outputs:

$$\boldsymbol{X}_{\boldsymbol{T}_O,\text{pred}} = \left\{\boldsymbol{X}_{\boldsymbol{T}_O,1}, \ldots, \boldsymbol{X}_{\boldsymbol{T}_O,n_p}\right\}$$

2.3.6 Transforming the predictions – Dataset D

Table 2.1: Transformation between the three possible prediction types $\boldsymbol{d}_{\text{pred}}$ implemented into the proposed framework, using the three implemented functions T_i.

Input ⟹ Output	Binary prediction a_{pred}	Timing prediction $\{a_{\text{pred}}, \boldsymbol{t}_{A,\text{pred}}\}$	Trajectory prediction $\boldsymbol{X}_{\boldsymbol{T}_O,\text{pred}}$
a_{pred}	–	$\left\{a_{\text{pred}}, T_3\left(a_{\text{pred}}\right)\right\}$	$T_2\left(\left\{a_{\text{pred}}, T_3\left(a_{\text{pred}}\right)\right\}\right)$
$\{a_{\text{pred}}, \boldsymbol{t}_{A,\text{pred}}\}$	–	–	$T_2\left(\left\{a_{\text{pred}}, \boldsymbol{t}_{A,\text{pred}}\right\}\right)$
$\boldsymbol{X}_{\boldsymbol{T}_O,\text{pred}}$	in $T_1\left(\boldsymbol{X}_{\boldsymbol{T}_O,\text{pred}}\right)$	$T_1\left(\boldsymbol{X}_{\boldsymbol{T}_O,\text{pred}}\right)$	–

To fulfill requirement **R 3**, we then must be able to transform a prediction $\boldsymbol{d}_{\text{pred}}$ from the previous step (2.3.5) to another prediction form if necessary. This functionality is a part of a specific dataset, as this requires the context information to, for example, classify different trajectories as accepted or rejected gap.

To facilitate those transformations, three different functions T_i are needed, as can be seen in Table 2.1. For the implementation of this, two instances of the trajectory prediction model *Trajectron++* [30] are used, namely M_A, trained on all samples from the specific dataset D ($\boldsymbol{D_I}$ and $\boldsymbol{D_O}$) where $a = 1$, and $M_{\neg A}$, trained on the remaining samples where $a = 0$:

T_1: takes the trajectory prediction $\boldsymbol{X}_{T_O,\text{pred}}$ and then provides $\{a_\text{pred}, \boldsymbol{t}_{A,\text{pred}}\}$. This is similar to the extraction of the time points in Section 2.3.2, using the function f_a, which extracts the gap acceptance decision from a single predicted trajectory, using T_A from Equation (2.2) (assuming $t_C \approx \max \boldsymbol{T}_O$ based on Equations (2.4) and (2.5)):

$$f_a\left(\boldsymbol{X}_{T_O,p}\right) = \begin{cases} 1 & T_A\left(\boldsymbol{X}_{T_O,p}\right) < \max \boldsymbol{T}_O \\ 0 & \text{else} \end{cases} . \tag{2.7}$$

Together with Equation (2.6), this results in

$$\begin{aligned} a_\text{pred} &= \frac{1}{n_p} \sum_p f_a\left(\boldsymbol{X}_{T_O,p}\right) \\ \boldsymbol{t}_{A,\text{pred}} &= Q_9\left(\left\{T_A\left(\boldsymbol{X}_{T_O,p}\right) \mid f_a\left(\boldsymbol{X}_{T_O,p}\right) = 1\right\}\right) . \end{aligned} \tag{2.8}$$

T_2: takes the prediction $\{a_\text{pred}, \boldsymbol{t}_{A,\text{pred}}\}$ and then based thereon provides the trajectory prediction $\boldsymbol{X}_{T_O,\text{pred}}$. Specifically, M_A and $M_{\neg A}$ are used to respectively predict two sets of trajectories, namely $\boldsymbol{X}_{T_O,\text{pred},A}$ and $\boldsymbol{X}_{T_O,\text{pred},\neg A}$. The final set of trajectories $\boldsymbol{X}_{T_O,\text{pred}}$ is assembled from select trajectories out of these two sets:

$$\boldsymbol{X}_{T_O,\text{pred}} = \{\boldsymbol{X}_{T_O,p,\neg A} | p \in \boldsymbol{R}_{\neg A}\} \cap \{\boldsymbol{X}_{T_O,p,A} | p \in \boldsymbol{R}_A\}.$$

This selection is based on the random selection function $R(m, \boldsymbol{M}, \boldsymbol{W})$, which randomly selects m samples from a set $\boldsymbol{M}$, with the probabilities of selection being proportional to the the weights $\boldsymbol{W}$. Only samples that actually represent the desired decision are viable (*e.g.*, in samples from $\boldsymbol{X}_{T_O,p,A}$, the gap must be accepted):

$$\begin{aligned} \boldsymbol{R}_{\neg A} &= R(n_p(1 - a_\text{pred}), \{p \mid f_a(\boldsymbol{X}_{T_O,p,\neg A}) = 0\}, 1) \\ \boldsymbol{R}_A &= R(n_p a_\text{pred}, \{p \mid f_a(\boldsymbol{X}_{T_O,p,A}) = 1\}, \boldsymbol{W}_A) . \end{aligned}$$

Here, $\boldsymbol{W}_A$ is chosen so that the distribution of t_A described by the decile values $t_{A,\text{pred},i}$ of $\boldsymbol{t}_{A,\text{pred}}$ (Equation (2.6)) is maintained:

$$\sum_{\left\{p | t_{A,\text{pred},i} < T_A\left(\boldsymbol{X}_{T_O,p}\right) < t_{A,\text{pred},i+1}\right\}} w_{A,p} = 1 \quad \forall i$$

This approach of using conditional trajectory prediction models is inspired by Xie *et al.* [51] and Hu *et al.* [107]. Importantly, this selection ensures that T_2 is the inverse of T_1

T_3: takes a binary prediction a_pred and provides the predicted time of accepting the gap $\boldsymbol{t}_{A,\text{pred}}$, by using M_A to generate $\boldsymbol{X}_{T_O\text{pred},A}$. We extract the desired output from this set of trajectories, similar to T_1:

$$\boldsymbol{t}_{A,\text{pred}} = Q_9\left(\left\{T_A\left(\boldsymbol{X}_{T_O,p,A}\right) \mid f_a\left(\boldsymbol{X}_{T_O,p,A}\right) = 1\right\}\right) .$$

2.3.7 Evaluating the predictions – Metric E

This functionality – the main part of the metric module – implements the performance evaluation, comparing the actual outputs $\boldsymbol{D}_{\boldsymbol{O},\text{test}}$ (from 2.3.3) with the predicted ones (*i.e.*, $\boldsymbol{D}_{\boldsymbol{OP},\text{test}}$, see 2.3.6). It returns either a combined value F or instead a separate value for each sample $\boldsymbol{d}_{\text{pred}} \in \boldsymbol{D}_{\boldsymbol{OP},\text{test}}$, resulting in the output $\boldsymbol{F}$.

2.4 Benchmark implementation

We implemented the framework described above by linking together several datasets, models, splitting methods, and metrics. These were chosen not to comprehensively cover all possible gap acceptance scenarios and prediction models but to demonstrate the flexibility and utility of the proposed framework. Still, our implementation can already serve as a benchmark for new prediction models. This section only presents an overview of the implementation, with more details available in Appendix A.

2.4.1 Datasets

Table 2.2: The number of accepted gaps N_A and rejected gaps $N_{\neg A}$ in the implemented datasets (*highD* with lane changes, *rounD* with entering roundabouts, and *L-GAP* with left turns), in the form: $N_A/N_{\neg A}$ (Median $t_{\underline{C}}(t_0)$). The numbers depend on the method for choosing the time of prediction t_0

Dataset	Gaps at opening	Fixed size gaps	Critical gaps
highD	1406/7026 (8.9 s)	461/1568 (11.8 s)	0/7025 (0.8 s)
highD – restricted	1406/1001 (12.9 s)	392/241 (8.7 s)	0/1000 (0.7 s)
rounD	662/917 (2.5 s)	168/168 (2.9 s)	33/913 (1.0 s)
L-GAP	703/724 (4.6 s)	496/572 (3.5 s)	369/723 (2.3 s)

Different datasets are implemented into the framework (Table 2.2), including data recorded on real roads as well as data from a driving simulator study. The naturalistic datasets used here are captured by drones and distinguished by accurate position labeling. They cover lane changes on German highways (the *highD* dataset [118]) and roundabouts (the *rounD* dataset [119]). The L-GAP dataset covers left turns at unsignalized intersections through oncoming traffic recorded in a driving simulator [46]. It has been chosen due to the simplicity of its environment, contrasting the more complex scenarios in the naturalistic datasets.

Lane changes

Here we focus on lane changes of the target vehicle V_T toward a faster lane to the left, along which the ego vehicle V_E driving there has the right of way. While it could be argued that predictions in such situations could be simply based on turn signals, one cannot rely on human drivers to correctly use these [114]. As a source of lane change data, we used two versions of the *highD* dataset, *full* and *restricted*.

The full *highD* dataset, not employing any filters, is heavily biased toward trajectories without a lane change. This is not a problem per se, but in such trajectories it is not known

whether the target vehicle V_T even had an intention to change lanes (*i.e.*, if there was a gap acceptance situation in the first place). For this reason, in addition to the full *highD* dataset, we added a restricted version of it which only included samples for which it can be inferred that the target vehicle V_T indeed considered a lane change. Criteria are either a lane change of V_T after V_E has passed or V_T braking to not collide with the preceding vehicle instead of changing lanes. Still, in both versions of *highD*, the gaps are always accepted with large safety margins (Table 2.2).

Roundabout
In the *rounD* dataset, the target vehicle V_T has to enter a roundabout, which it can do in front of or behind the ego vehicle V_E already in the roundabout. As the trajectories are recorded in Germany, the ego vehicle inside the roundabout has the right of way. Compared to *highD*, this dataset is far more balanced between accepted and rejected gaps, but still only includes few critically accepted gaps.

Left turns
In the *L-GAP* dataset [46], the driver of the target vehicle V_T intends to turn left at an intersection. The driver had to decide whether to do this in front of or behind the ego vehicle V_E approaching the intersection from the opposite direction with the right of way. While the number of samples in this dataset is comparatively small, they are relatively balanced between accepted and rejected gaps. Also, they include many gaps accepted after t_{crit} (Table 2.2). Nonetheless, as V_T starts in an idling position at some distance to the contested area, this might not be the most challenging dataset, as an onset of movement before t_0 in most cases is an apparent indicator of V_T intending to accept the gap.

2.4.2 Splitting the dataset

Two splitting methods are implemented, without a method for calculating the similarity measure ζ. Nonetheless, to enable at least a qualitative approximation of a model's robustness, the methods are designed to produce testing sets of varying difficulty for the prediction models.

The easier variant performs a stratified random splitting, while the second, more extreme method sorts the most unintuitive behavior of the target vehicle into the testing set (*e.g.*, accepting a very small gap or rejecting a very large gap).

In both cases, the testing set includes 20% of the samples and the training set the remaining 80%.

2.4.3 Models

The benchmark includes two state-of-the-art trajectory prediction models

- *Trajectron++* (also referred to as *T+*), a deep-learning model mainly based on long-short-term memory cells [30].

- *AgentFormer* (*AF*), a deep-learning model based on transformers [34]. Compared to *T+*, it has ten times more trainable parameters.

For the binary prediction models for gap acceptance, there is, as mentioned above, a lack of a common benchmark, making the models' selection more contentious. Four models have been selected nonetheless:

- Logistic regression (*LR*) is commonly used for predicting human gap acceptance decisions [49] and is therefore included as a simple baseline.
- Random forests (*RF*) have been shown to outperform other approaches such as logistic regression and standard decision trees in gap acceptance prediction [104].
- Deep belief networks (*DB*), also used previously to predict human gap acceptance decisions [51].
- A metaheuristic model based on combining all other five models above (*MH*); previously a similar approach for lane changes has been shown to outperform each of the models included in it [110].

The benchmark does not include any dedicated timing prediction models yet, as their primary representative, the drift-diffusion model [46, 47], can currently not be trained on datasets with a large number of unique samples in a reasonable amount of time. Nonetheless, to allow for future expansion of the benchmark, the framework has been designed with such models in mind.

2.4.4 Evaluation metrics

We have included several metrics that characterize models in terms of the quality of binary predictions (accept/reject gap) as well as full trajectory predictions. The following metrics are commonly used in the literature:

- Accuracy: This metric is a widespread method to evaluate the performance for binary prediction models [51, 104, 110]. However, accuracy is a symmetric metric (*i.e.*, it is unable to differentiate between false negative and false positive predictions). Consequently, it is best used in cases where $t_0 \ll t_{\text{crit}}$, as the consequences of false predictions are not too different there (Section 2.3.1).
- *AUC*: This metric for binary prediction models, the Area Under Curve of a receiver operating characteristics curve, addresses one central point of criticism of the accuracy metric, namely its sensitivity to biases in the testing set. Nonetheless, like the accuracy metric, it does not consider the potentially differing severity of false predictions. Hence, we only apply it to rate a prediction model's performance when $t_0 \ll t_{\text{crit}}$.
- ADE_β and FDE_β: These metrics, the Average or Final Displacement Error of the $n_p\beta$ least erroneous predicted trajectories, are commonly applied to trajectory predictions [30, 33, 34], with $\beta = 1$ (*i.e.*, average for all predictions) or $\beta = 0.05$ (mimicking the "best-of-20" metric used in [30]) being used in this work. As these metrics also does not take into account the severity of different false predictions [61, 62] and requires equally long prediction horizons as well, they are only applied for constant gap sizes ($t_0 = \min\{t \mid t_{\underline{C}}(t) - t = \Delta t\} \ll t_{\text{crit}}$). However, due to their similarity, only the *ADE* is discussed further in this work.

Furthermore, we propose a novel metric that considers the potential consequences of a wrong prediction.

- *TNR-PR*: The True Negative Rate under Perfect Recall is a metric applied to binary predictions, with the explicit goal to consider the vastly different consequences of false negative and positive predictions at $t_0 \approx t_{\text{crit}}$ (*i.e.*, a potential accident vs unnecessary braking, refer to Section 2.3.1). To that end, the threshold for classifying a prediction a_{pred} as positive is set as low as necessary to achieve perfect recall on the test set (*i.e.*, there are no false negative predictions). It then estimates the usefulness of prediction models in improving the efficiency of path planning by evaluating their true negative rate (TNR) under this decision threshold. This metric is therefore equivalent to the likelihood that a prediction model can prevent needless braking while guaranteeing safe interactions. As earlier predictions do not necessitate a model with perfect recall, due to available time to wait for further data, this metric is only applied to predictions made at the last possible time (*i.e.*, $t_0 = t_{\text{crit}} - t_\epsilon$).

2.5 Results

Our benchmark[1] provides insights into performance of tested models under different conditions (Figures 2.4, 2.5). In this section, we first discuss our findings regarding the prediction performance across models depending on specific prediction problems, which are defined by number of input time steps n_I, the splitting method, the scenarios, and the prediction time t_0. We will then discuss the differences between individual models.

2.5.1 Prediction problem's influence on model performance

As one may expect, when provided with more input time steps ($n_I = 10$ vs. $n_I = 2$), *i.e.*, more information is provided to extract signs of future behavior from, the models' predictions were generally better. Second, performance of the models tested on the most unintuitive samples (the extreme splitting case) was worse than in the random splitting case. There, the models have to extrapolate to situations outside the training domain, a typically far more difficult task than the interpolation inside the training domain needed for prediction on random test samples [120]. The poor performance on unintuitive samples is especially pronounced when looking at the *TNR-PR* at critical gaps (Figure 2.5), where no model could substantially outperform a random predictor on both datasets.

When comparing the predictions of models on the two lane-change datasets (the two left columns in Figure 2.4), it can be expected that the models' performance should be better on the restricted one, due to the removal of many large gaps that were unintuitively rejected. However, this was only observed in the *AUC* metric, while the opposite could be observed for Accuracy (caused by the exploitation of the larger bias towards rejected gaps in the unrestricted lane change dataset) and *ADE*. Therefore, it can be assumed that *AUC* is the most reliable metric here, a result which is supported for example by Huang and Ling [121]; this metric will be our main focus from here on.

When comparing the difficulty of different scenarios, it can be seen that, generally, the prediction of human behavior at roundabouts seems to be the easiest, having the best

[1]The data underlying these results can be found at `https://github.com/DAI-Lab-HERALD/Framework-for-benchmarking-gap-acceptance`.

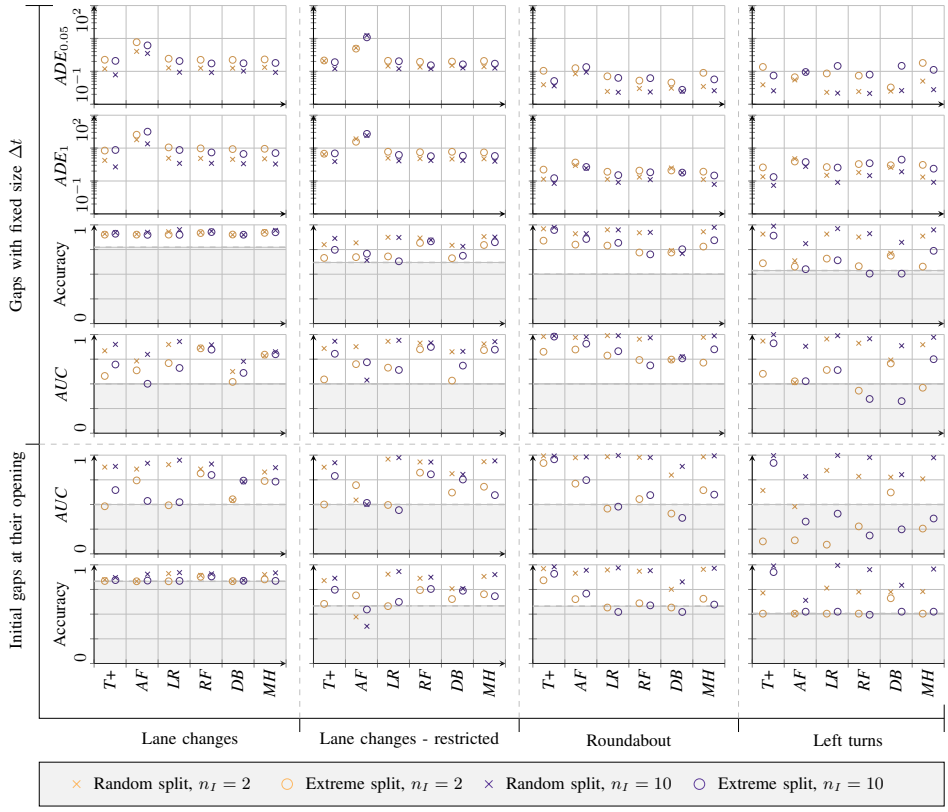


Figure 2.4: The results of evaluating behavior prediction models in gap acceptance scenarios on different datasets, with prediction being made either at the initial opening of the gap ($t_0 = t_S$) or with fixed gap sizes ($t_0 = \min\{t \mid t_C(t) - t = \Delta t\}$). The color indicates the number of input time-steps n_I given to the models, and the marker type denotes the splitting method, which can be *random* or *extreme*. The dashed gray lines indicate the performance F_r of a uniformly random binary predictor.

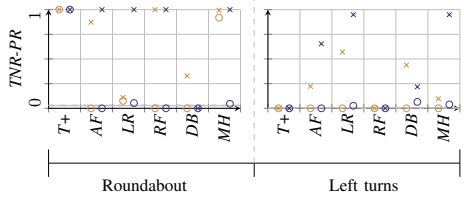


Figure 2.5: The true negative rate under perfect recall (*TNR-PR*) of different prediction models, using the same visualizations as in Figure 2.4, tested on the last useful predictions ($t_0 = t_{\text{crit}} - t_\epsilon$).

AUC values in 30 of 48 cases (each two prediction times, input steps, and splitting methods on six models). A possible explanation here might be the short prediction horizon of less than three seconds (Table 2.2), which leaves less room for the target vehicle to behave unexpectedly, although scenario-specific reasons cannot be excluded. This explanation

is also supported by the finding that – at least for $n_I = 10$ – the scenario with the next shortest gap, the left turns, gets the best results out of the remaining three datasets in 12 of 24 cases. This is not the case for $n_I = 2$, where the left-turn dataset is the worst of all four datasets in 15 of 24 cases. This can be explained by the fact that in this scenario, the target vehicle starts in an idling position, from which not much information can be gained. Technically, predictions from both $n_I = 2$ and $n_I = 10$ should be made at the same time, but due to the lack of trajectory data before t_S in this dataset, the predictions are made in fact at $t_0 > t_S + (n_I - 1) * \delta t$. Therefore, prediction for $n_I = 10$ are generally made later, which leaves more room for the onset of motion, making predictions easier.

When comparing the *AUC* at different prediction times t_0, it can be observed that on those datasets with a lower median $t_{\underline{C}}(t_0)$ (Table 2.2) the prediction were better more often than not, in 59 of 96 cases. This again lends credence to the hypothesis that earlier predictions with longer prediction horizons are more challenging. In the case of trajectory prediction models, this seems logical, as a longer prediction horizon leaves more time for prediction errors to propagate and compound on each other. There is also the higher probability that the duration between t_0 and the actual human decision is larger, which leads to a higher probability that there are no indications of that decision in the human behavior yet.

2.5.2 Differences between models

Comparing the trajectory prediction models, we found that the Trajectron++ model (*T+*) consistently outperforms the AgentFormer model (*AF*). This is surprising, as *AF* previously outperformed *T+* on pedestrian trajectory prediction benchmarks [34]. This contradiction might be explained by over-fitting the many trainable parameters for *AF* on relatively small datasets here. Meanwhile, the logistic regression (*LR*) model is often the most promising approach for binary prediction models (best *AUC* value in 18 of 32 cases, and best *TNR-PR* value in 5 out of 8 cases), especially when tested on random samples, where it has the best *AUC* in all possible sixteen cases. Together, those results indicate that increasing the complexity of such models and their number of parameters might not be a panacea, with simpler models being more promising, especially if datasets are relatively small.

When we compare binary models against trajectory prediction models, we can observe differing behavior for different scenarios. On the one hand, the best performance on the lane-change datasets is generally achieved by a binary prediction model (the best *AUC* values come from binary models in all 16 cases), while on the other hand, similar performance can be observed on the other two datasets. One main difference here is the prediction horizon ($\Delta t \approx 10\,\mathrm{s}$ and $\Delta t < 4\,\mathrm{s}$ respectively, as shown in Table 2.2), which might indicate that trajectory prediction models are more impaired by such longer prediction horizon than binary models. That the average displacement errors are much more noticeable in the lane change scenario also supports this explanation, further showing the difficulties of using trajectory prediction under those conditions. However, a deeper analysis into the causes for these observations when comparing models would likely require a number of ablation studies, which is outside the scope of this work.

Lastly, when evaluating the promise of including binary prediction models into trajectory prediction models (relying on the transformation function T_2 from Table Table 2.1),

we can hold that the benefits are mostly negligible if existent at all, except at roundabout and left turns, where some improvements can be seen in average displacement errors. Nonetheless, due to the problems with that metric, more than those results are needed to render a final judgment. However, due to the small size of datasets and the low number of models, the results discussed here should be treated with care.

2.6 Conclusion

We proposed a framework that connects previously disparate datasets, models, and metrics in the benchmark for testing behavior prediction models in gap acceptance scenarios. We demonstrated its potential and flexibility by comparing two state-of-the-art trajectory prediction models with several binary gap acceptance models. Additionally, we showed that relying on the characteristic time points of gap acceptance scenarios to select the most unintuitive samples in the splitting module is a promising approach to analyzing model generalization, as seen by the general decrease in performance for models trained on those samples. Our framework is open-source and specifically designed in a modular way to simplify adding new datasets, splitting methods, models, and evaluation metrics, which allows researchers to easily expand it in future. This can speed up the testing of new models, only requiring the adaption of the models' implementation to the format of our framework, while previously one would need to write separate code for every model applied to every scenario.

One particularly important addition to the benchmark would be datasets containing more critically accepted gaps, as this would allow for an increased meaningfulness of metrics applied to last useful predictions. Additionally, a metric better aligned with the main purpose of a prediction model as a part of an autonomous vehicle is still needed. Likewise, currently there exists no method for calculating the similarity between testing and training set ζ; the future addition of this would permit a quantitative comparison of a model's robustness against unintuitive test samples. Lastly, one could expand the framework to provide scenario-independent inputs similar to $t_{\underline{C}}$, which would enable training a model on two unrelated datasets simultaneously, leading to a better estimation of the models generalizability.

We acknowledge that testing a model in gap acceptance scenarios alone is necessary, but not sufficient for justifying its usage in actual vehicles. Consequently, expanding the framework to non-gap-acceptance scenarios is an important avenue for future research. This will enable more holistic testing, although only for models predicting (and metrics evaluating) trajectories. Nonetheless, we argue that performance of models on non-gap acceptance scenarios should still be given lower priority compared to gap acceptance scenarios which are more safety-critical.

Our results resonate with the recent literature on hybrid AI [122, 123], showing that including binary prediction models in specific scenarios might make data-driven trajectory prediction models more reliable, especially in accurately predicting dangerous situations. However, especially for the unintuitive and safety-critical edge cases, most models often performed only slightly better than a random predictor at best. Therefore, there currently seems to be no model that a trajectory-planning algorithm can rely on to substantially increase the effectiveness of an autonomous vehicle's driving style in every scenario, necessitating further research into such models.

3

Structured Training and Evaluation Platform for Benchmarking Trajectory Prediction Models

Expanding on the previous chapter, we here present STEP, an general framework for evaluating behavior prediction models. While recent efforts have aimed to unify dataset formats and model interfaces for easier comparisons, they often fall short in supporting heterogeneous traffic scenarios, joint prediction models, or user documentation. In contrast, STEP provides a unified interface for multiple datasets, enforcing consistent training and evaluation conditions, while enabling research to not only assess a model's performance on typical benchmarks, but also its generalizability and robustness. This is supported by a wide range of metrics, which can be extended to classifiable behavior as well. We demonstrate the capabilities of STEP in a number of experiments which reveal for instance the limitations of widely-used testing procedures and the vulnerability of current state-of-the-art models against both distribution shifts and targeted attacks by adversarial agents. With STEP, and its detailed documentation, we aim to shift the focus from the "leaderboard" approach to deeper insights about model behavior and generalization in complex multi-agent settings.

This chapter is based on *J. F. Schumann*, A. Mészáros*, J. Kober, A. Zgonnikov. "STEP: Structured Training and Evaluation Platform for benchmarking trajectory prediction models" IEEE Transactions on Intelligent Transportation Systems (2026, under review)* [124]

3.1 Introduction

Modern automated vehicle (AV) systems are typically organized into modular software stacks, consisting out of components such as perception, prediction, and planning. This modular structure allows for an independent development and replacement of these components, enabling greater transparency and scalability. The development of new prediction modules has been receiving increased interest in the research community [15, 125], with many attempts at improving the accuracy and reliability of predictions.

Rigorous evaluation of prediction models is paramount for ensuring reliable AV performance. However, in the current literature there is a lack of established benchmarks for trajectory prediction models. Despite (or possibly due to) the increasing availability of diverse large-scale datasets, there are no widely accepted evaluation approaches and guidelines. This results in two practical issues that model developers face. First, besides implementing a new model, the developers currently need to implement model evaluation from scratch, such as the coding of data-loaders providing both training and testing data in the format their model requires. This introduces much overhead during the model development. Second, the lack of established routines can lead to splintered evaluation, preventing equitable comparisons with state-of-the-art models (especially if they use different data interfaces). For instance, the calculation of performance metrics used for model comparison can differ from paper to paper (*e.g.*, with respect to which parts of the dataset are included), while even slight differences can have major impact on the reported results.

Motivated by these challenges, recent work has taken steps towards unified evaluation of trajectory prediction models, such as a unified interface to different datasets [67] and frameworks allowing more streamlined model comparisons [16, 126–128]. However, such frameworks still lack many features that are essential for a holistic and fair comparison of trajectory prediction models. *Trajdata* [67], for example, only provides a unified interface between datasets, with no way to ensure that all models are evaluated under the same conditions. TrajNet++ [126] and ATLAS [127], meanwhile, made first steps towards developing a complete benchmarking framework for trajectory prediction. However, these approaches are not widely applicable, as they only allow pedestrians. Correspondingly, there is also a neglect of features more relevant in heterogeneous traffic, such as lane information and agent information like type and size. The gap acceptance benchmarking framework (GAP) [16], being designed to also evaluate models predicting different classes of behaviors, allows evaluation beyond just predicted trajectories. However, it is limited to gap acceptance scenarios only, and thus unable to evaluate models in other scenarios common in large datasets [72, 125]. *UniTraj* [128] addresses many of the limitations mentioned above, but still lacks the support for models that predict the future trajectory of all agents in a scene jointly and simultaneously (which is the case for many state-of-the-art models [75–77, 82]), as well as out-of-the-box evaluation of model robustness against adversarial attacks, which is an increasingly important consideration for trajectory prediction [84]. Importantly, existing benchmarks also lack supporting documentation for the addition of new datasets or models, resulting in a high entry barrier for model developers.

To address these issues, this paper introduces STEP (Structured Training and Evaluation Platform), which provides a unified interface across numerous datasets and enabling flexible formatting of the data to create diverse testing scenarios. STEP enables comparisons with a broad selection of current state-of-the-art models, including single agent tra-

Table 3.1: Comparison between existing benchmarks for trajectory prediction. An asterisk * next to a checkmark (✓) indicates that the feature is available to a limited extent.

Framework	trajdata [67]	TrajNet++ [126]	ATLAS [127]	UniTraj [128]	GAP [16]	STEP (Ours)
Unified data format	✓	✓	✓	✓	✓	✓
Large datasets supported	✓			✓		✓
Evaluation with unified metrics		✓	✓	✓	✓	✓
Controllable dataset splitting				✓*	✓	✓
Environment image provided	✓		✓	✓	✓	✓
Scene graph provided		✓		✓		✓
Agent size provided	✓			✓		✓
Includes pedestrian labels	✓	✓	✓	✓	✓	✓
Includes vehicle labels	✓			✓	✓	✓
Includes motorcycle labels	✓					✓
Includes bicycle labels	✓			✓		✓
Adjustable input/output length	✓		✓	✓	✓	✓
Adjustable trajectory frequency	✓		✓		✓	✓
Supporting scene-level models	✓					✓
Includes behavior classification		✓*		✓*	✓	✓
Validation of model's robustness			✓*			✓
Extensive documentation	✓		✓*	✓*		✓

jectory prediction models [35, 37, 38, 129] and scene-level joint trajectory prediction models [34, 75–77]. Importantly, STEP ensures that all evaluated models are trained and tested under the same conditions, making those comparisons truly equitable. The framework is supported by a detailed documentation, enabling easy addition of new datasets, models, and metrics.

3.2 Related work

In Table 3.1 we provide a high-level overview of the features of existing benchmarking frameworks, highlighting the added functionalities of our approach. First, STEP is the only approach that provides meaningful flexibility over the mechanism used for dividing a dataset into the training and testing subsets. For example, the drone datasets (such as highD [118] and rounD [119]) have recordings from multiple locations. Practitioners who use these datasets split them in multiple different ways, such as following an

N-cross-validations paradigm across the whole dataset [38], or taking a fraction of the recordings from each location [91], or simply via a single random split [130–132]. Alternatively, one may wish to split the data by the specific locations following a leave-one-out paradigm [133] (often used for the ETH/UCY dataset [134, 135]) where all but one location is used for training and the held out location is used for testing. Among existing frameworks, only STEP provides all those functionalities without the need to implement them from scratch.

Second, model developers usually evaluate models on multiple different datasets, but typically training and testing subsets belong to the same dataset. At the same time, combining different datasets together for training and testing can provide more generalized evaluation [128]. While this is theoretically possible in UniTraj [128], one would still only test the model on the part of one selected dataset pre-labeled as testing data. STEP supports and generalizes this approach by giving the user a large number of out-of-the-box options for splitting composite datasets, such as testing only on specific locations picked from any of the composite datasets.

Third, the existing frameworks give the user only limited control over the observation horizon and especially the frequency of the input. Previous research has shown that the duration of the input can play an important role in model performance [136, 137]; similarly, the frequency of the input might also affect performance. Additionally, keeping in mind the bigger picture of the AV stack and the fact that different perception modules may operate at different frequencies, it would be beneficial for model developers if they could train their models already accounting for the operational frequency of the upstream perception module. However, of the existing frameworks, only ATLAS [127] give users control over both duration and frequency of the input. STEP supports this but in addition also allows the user to ensure that varying the input data does not change the selection of included scenario, guaranteeing that differences in performance actually stem from the differences in the input data and not from corresponding changes in the size or structure of the dataset.

Fourth, scene-level predictions, *i.e.* joint predictions for all agents in the scene, are receiving increasing attention [75–77, 82]. The reason for this is that single-agent prediction models are unequipped to properly account for future interactions between agents. To facilitate the development of such models, the evaluation framework must not only provide past and future information of all relevant agents in a scene during training, but also make it possible to evaluate those joint predictions together. Furthermore, such joint prediction models can profit from the use of interaction-based metrics (*i.e.*, metrics accounting for the scene-level predictions). However, only TrajNet++ [126] provides interaction-based metrics, but limits them to two-agent interactions and does not account for the complete scene-level distribution. As such, none of the existing frameworks are suitable for developing scene-level (joint) prediction models, while STEP fully supports these.

Finally, model robustness is becoming increasingly important within the field [84]. While ATLAS [127] can test the robustness of models against white noise, this only represents a small part of potential perturbations of input data. Specifically, targeted attacks by adversarial agents [71, 86, 138–140] are a concern, and STEP is currently the only framework that allows users to test their models (and potential countermeasures) against such attacks out-of-the-box.

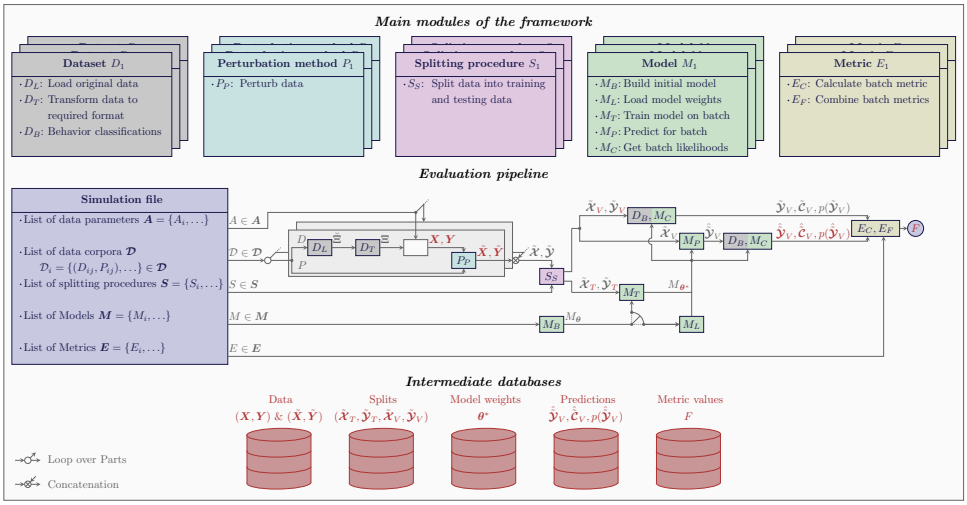


Figure 3.1: A high level overview of STEP (Structured Training and Evaluation Platform for benchmarking trajectory prediction models): the five main modules (top row, with a more detailed explanation in Section 3.3.1), the pipeline for training and evaluating models (middle row, discussed in Section 3.3.2), and the databases that the framework uses to save intermediate steps (bottom row). In a simulation file, the user specifies the target data parameters, datasets, perturbation methods, splitting procedures, models, and metrics. The framework then follows the training/testing pipeline to calculate the performance value F for each potential combination of these.

3.3 STEP

Our proposed framework STEP (Figure 3.1), developed as a generalization of a previously proposed gap acceptance benchmarking framework [16], is designed to fill in the gaps in the existing literature (Table 3.1), providing

- a unified dataset format
- control over the data partitioning
- control over the observation length and frequency
- access to all relevant agent types and their features
- semantic scene information through images and scene graphs
- support for scene-level models during training and testing
- robustness testing
- easy community-based expansion of all relevant modules (datasets, perturbation method, splitting procedure, models, and metrics) with comprehensive documentation.

There are two main use cases for our framework. First, users can easily integrate their own models, metrics, or datasets in STEP; our framework is highly modular so that can be

achieved simply via *adding a new module*. Second, the framework streamlines *execution of experiments*. Users can run fully automated (yet flexible and easily customizable) training and testing of target models on target datasets, as well as calculation of the target metrics.

3.3.1 Adding new modules

Datasets are the fundamental building block for our framework, providing the scenarios in which behavior prediction models are trained and evaluated. Currently, our framework includes 12 state-of-the-art datasets, ranging from the large-scale general datasets (*Argoverse2* [72], *Lyft Level5* [141], *Waymo* [73], *NuScenes* [142], *INTERACTION* [143]) to those more focused on either specific driving contexts (roundabouts in *rounD* [119], highways in *highD* [118], or intersections in *inD* [144]) or specific agent types such as pedestrians (*ETH/UCY* [134, 135]). To the best of our knowledge, STEP contains the most comprehensive library of datasets available, which includes all datasets that are part of the previous frameworks (Table 3.1).

To add a new dataset to the Framework, a user has to implement two main functionalities as well as one optional functionality (cf. the pipeline in Figure 3.1):

- D_L to load the dataset's information $\breve{\Xi}$ in its original form. In its most basic form, this will be information on the agents' states (positions, velocities, etc.), but can also include their type (vehicle, pedestrian, etc.) or size, as well as their environment, either based on top-down view images or graphs containing information such as lane positions.
- D_T to transforms the original data into the framework-compliant format (*i.e.*, Ξ).
- In case the dataset supports high-level behavior classification (*e.g.*, drivers approaching roundabouts making a discrete decision whether to yield to an oncoming car): D_B to assign a provided scene (*i.e.*, arbitrary observation $\boldsymbol{p}$, which might also be only the future trajectory $\boldsymbol{y}$) a high-level behavior classification $\boldsymbol{c}$.

Perturbation methods can be applied to datasets to test robustness of trajectory prediction models, both against natural variability in the data and deliberate attacks [84]. The framework provides two perturbation methods based on state-of-the-art adversarial attack approaches, focused on perturbing either predicted vehicle's positions [71] or control states [86, 140]. Users can also add their own perturbation methods by implementing a function P_P which takes a set of past observations $\boldsymbol{X}$ (including both agent's states and ancillary information) and corresponding future observations $\boldsymbol{Y}$ and perturbs them.

Splitting procedures split the dataset into the training and testing part. The framework provides a range of out-of-the-box splitting procedures:

- Random splitting
- Splitting by dataset and/or locations, to allow testing of models on a location outside the training domain
- Random cross validation splitting, to test the influence of the splitting on the final performance metric.

- Splitting by safety critically to test models on the most important scenarios.
- Predefined splitting to fit to public challenges, such as on the *Waymo* dataset.

Users can implement their own splitting methods by adding a module implementing the function S_S.

Models include implementations of prediction methods. This can be both trajectory prediction models (*i.e.*, a conditional probability distribution $P(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta})$ over future trajectories $\boldsymbol{y}$) or – similar to [16] – behavior classification models (*i.e.*, a distribution $P(\boldsymbol{c} \mid \boldsymbol{x}, \boldsymbol{\theta})$ over future behavior classes $\boldsymbol{c}$). The framework has out-of-the-box implementations for a variety of state-of-the-art trajectory prediction models (*ADAPT* [76], *AgentFormer* [34], *AutoBot* [75], *FJMP* [77], *FloMo* [37], *MTR* [145], *Trajectron++* [30], *TrajFlow* [38], *Wayformer* [35]) as well as several behavior classification models [49, 51, 132].

To integrate a new model to the framework, the following functions should be implemented:

- M_B to initialize the model with random parameters.
- M_T to adjust the model parameters $\boldsymbol{\theta}$ to better fit the presented training data (passed in batches) and save the final parameters. The framework allows to automatically split a subset from the training set for validation, but as this is model-dependent, this has to be implemented for each model separately.
- M_L to load and assign previously saved model parameters $\boldsymbol{\theta}^*$ to the model.
- M_P to take the past observations $\boldsymbol{x}_V$ from the test set and sample a number of potential future trajectories for trajectory prediction models ($\hat{\boldsymbol{y}}_V \sim P(\boldsymbol{y} \mid \boldsymbol{x}_V, \boldsymbol{\theta}^*)$) or the predicted behavior probabilities $\hat{\boldsymbol{c}}_V \sim P(\boldsymbol{c} \mid \boldsymbol{x}, \boldsymbol{\theta})$.
- (optional) M_C to calculate the likelihoods $p(\boldsymbol{y}) = P(\boldsymbol{y} \mid \boldsymbol{x}_V, \boldsymbol{\theta}^*)$ of a given future trajectory $\boldsymbol{y}$ according to the trained model and given the past observations $\boldsymbol{x}$. For models with explicitly defined probability function, it is recommended that users specify M_C. This is, however, optional, as the framework can estimate the likelihoods of sampled trajectories using robust multi-modal density estimation [146].

Metrics assign a model a single scalar value given a specific combination of dataset and splitting method, and are generated automatically to facilitate comprehensive model comparison. STEP's large library of metrics covers all metrics commonly used in the recent literature, ranging from simple distance metrics such as average displacement error (ADE) to likelihood based ones like the negative logarithmic likelihood (NLL) or behavior based ones such as area under curve (AUC). An up to date list of metrics can be found in the documentation.

To include a new metric in the framework, a user need to implement two functions:

- E_C to take the ground truth data (trajectories $\boldsymbol{y}$, behavior classes $\boldsymbol{c}$, or trajectory likelihoods $p(\boldsymbol{y})$) batch-wise and compare it to the corresponding model predictions, resulting in a batch metric f.
- E_F to combine the batch-wise values f into a combined value F.

3.3.2 Running an experiment

To set up an experiment, users need to create a *simulation file* that specifies which modules are used in the experiment.

- **Data corpus** $\mathscr{D} = \{(D_1, P_1), ...\}$ that assembles data from multiple *datasets* D, where each dataset can be optionally paired with a *perturbation method* P (which could be empty in case the data is to remain unperturbed).
- **Data parameter setting** $A = \{n_I, n_O, \delta t\}$ to set the number of input (past observed) and output (future) time steps, and the time step value (in seconds). These are used consistently across different $\mathscr{D}$, facilitating apples-to-apples comparisons across datasets.
- **Splitting method** S specifying the size and characteristic (*e.g.*, the test location or locations when splitting based on scenario locations), or the inclusion/exclusion of (potentially perturbed) data in training or testing set.
- **Model** M to be trained/tested; this can include the setting of specific model hyperparameters, allowing the comparison of different model variants.
- **Metric** E determining the formulas used for the calculation of the final performance values F.

In case the simulation file includes multiple modules of a given type (*e.g.*, multiple datasets, models, and metrics), the framework will evaluate each possible combination, provided that those are feasible (*e.g.*, splitting the data into training and testing set based on dataset requires that the used data corpus $\mathscr{D}$ contains at least two datasets). These final performance metrics F are then determined by the framework in a multi-step process (Figure 3.1).

While our framework allows the addition of both trajectory prediction and behavior classification models predicting the probabilities with which an agent (or group of agents) will perform certain classifiable maneuvers (such as which of two agents will cross an intersection first), for simplicity's sake, the following description of the evaluation pipeline is focused on the former.

As the first step, for each dataset module D included in data corpus $\mathscr{D}$, the underlying raw data $\breve{\Xi}$ is loaded and transformed into the internal format of our framework Ξ (detailed description provided in the online documentation) using the functions D_L and D_T, respectively. From each scenario in this dataset – based on the dataset parameters A – the framework extracts one or more samples (differing in the prediction time point t_0) with the respective input and output data ($\boldsymbol{X}$ and $\boldsymbol{Y}$). In the framework, one can set different methods for determining t_0, such as (1) using the first available point in time with sufficient past trajectories available, (2) a specific point in time before certain behaviors are initialized/completed (see [132] for more details), or (3) regular spacing throughout the given scenario. Importantly, while $\boldsymbol{Y}$ only contains the future (*i.e.*, $t > t_0$) trajectories, $\boldsymbol{X}$ can contain not only the past observed trajectories but also additional predictive information such as agent type and size, or environment information (in image and/or graph format). The perturbation method is then applied to the extracted data (via the function P_P), resulting in the final perturbed values $\tilde{\boldsymbol{X}}$ and $\tilde{\boldsymbol{Y}}$ for the dataset module. After iterating

over all pairwise combinations of D and P, the resulting (potentially perturbed) datasets (with input $\tilde{\boldsymbol{X}}$ and output $\tilde{\boldsymbol{Y}}$) are then concatenated into the combined input $\tilde{\mathcal{X}}$ and output $\tilde{\mathcal{Y}}$.

In the next step, the combined dataset is then split based on the chosen splitting method (function S_S), resulting in training set ($\tilde{\mathcal{X}}_T$ and $\tilde{\mathcal{Y}}_T$) and testing set ($\tilde{\mathcal{X}}_V$ and $\tilde{\mathcal{Y}}_V$). The former is then repeatedly split into batches, which are fed to the initialized (function M_B) model's training function M_T, resulting in a trained model.

Afterwards, in a similar manner, the testing set is split into batches as well. The input data $\boldsymbol{x}_V$ in each batch is presented to the trained model, resulting in predicted trajectories $\hat{\boldsymbol{y}}_V$. For the datasets which include information on high-level behavior classification, the framework extracts the high-level behaviors ($\hat{\boldsymbol{c}}_V$ and $\boldsymbol{c}_V$) using the function D_B; this is done both for the predicted ($\hat{\boldsymbol{y}}_V$) and the ground truth ($\boldsymbol{y}_V$) trajectories. Importantly, on this step M_C calculates the likelihoods of those trajectories according to the underlying predicted trajectory distribution. These are then fed to the batch-wise-evaluation function E_C, resulting in the batch metric values f. Combining those over the whole testing set using the function E_F results in the final performance metric value F.

Whenever the framework saves the results of some intermediate steps, it will preemptively check for the existence of those results and if so, load the existing data instead. Due to the vide variety in model structures, loading model weights requires the model-specific function M_L.

Additionally, while the framework will attempt to calculate a performance metric value for every combination of five main building blocks, this might not always be possible. For example, a metric based on high-level behaviors will only be calculated if the testing set contains at least one scenario coming from a dataset for which behavior extraction is defined (*i.e.*, there is a usable definition for function D_B).

Finally, on top of the basic training/testing pipeline, the framework supports additional processes such as fine-tuning already trained models on new datasets.

3.4 Experiments

To demonstrate the flexibility of the proposed framework, we conducted a number of experiments. In these experiments, we used four state-of-the-art models (ADAPT [76], FJMP [77], Autobot [75], Wayformer [35]). We focus on Argoverse2 [72] as the main large-scale dataset. Additionally we also use rounD [119] (a dataset focused on German roundabouts) to investigate behavior classification performance of the models. To this end, we extracted *gap acceptance* scenarios [16] from rounD, where a vehicle entering a roundabout can either yield to a vehicle inside the roundabout (reject the gap) or go first (accept the gap).

As trajectory prediction metrics, we used mean ADE and mean FDE for single agents and joint distributions, as well as miss rate for single agents, all based on 6 randomly sampled trajectories. As the behavior classification metric we used area under curve (AUC) and the expected calibration error (ECE). Given that some models (*e.g.*, ADAPT) only predict a limited number of trajectories (*i.e.*, not samples from a distribution), we did not use metrics based on probabilities (BrierminFDE [72], NLL [147], etc.).

Table 3.2: Sensitivity of model performance to input data parameters (models trained on Argoverse2). In each cell, the rows correspond to the different timestep lengths $\delta t \in \{0.5s, 0.25s, 0.1s\}$, while the columns represent different observation horizons $T_I \in \{1.5s, 4.5s\}$. Hence, in each cell the upper left number corresponds to the shortest and least detailed input, while the the lower right number represents the longest, most detailed input. The model with the best value for each metric and input parameter combination is shown in bold. The best value achieved over the different input conditions by each model is underlined.

	Metric	minADE		minADE_J		minFDE		minFDE_J		MR	
Model	δt / T_I	1.5 s	4.5 s	1.5 s	4.5 s	1.5 s	4.5 s	1.5 s	4.5 s	1.5 s	4.5 s
ADAPT [76]	0.5 s	**0.78**	0.88	2.35	2.28	1.76	1.73	5.39	4.69	0.15	0.14
	0.25 s	0.82	**0.73**	2.04	1.75	1.92	1.45	4.95	4.39	0.16	0.12
	0.1 s	1.07	**0.86**	2.18	1.94	1.69	2.21	5.40	4.40	0.18	0.13
FJMP [77]	0.5 s	0.85	0.98	**1.73**	1.98	1.80	1.84	**3.77**	4.49	0.20	0.21
	0.25 s	**0.74**	0.77	**1.45**	**1.62**	1.70	1.65	**3.61**	**3.73**	0.20	0.20
	0.1 s	1.00	0.97	1.85	**1.68**	2.35	1.91	4.87	4.27	0.27	0.23
AutoBot [75]	0.5 s	1.31	1.93	2.13	2.78	2.11	2.60	4.13	4.54	0.21	0.34
	0.25 s	1.48	2.32	2.26	3.21	2.33	2.92	4.32	4.83	0.25	0.42
	0.1 s	1.61	1.63	2.40	2.46	2.58	2.50	4.66	4.67	0.28	0.28
Wayformer [35]	0.5 s	0.89	**0.83**	1.95	**1.81**	**1.43**	**1.34**	4.08	**3.87**	**0.09**	**0.09**
	0.25 s	0.92	0.83	1.87	1.73	**1.50**	**1.37**	4.08	3.87	**0.09**	**0.09**
	0.1 s	**0.94**	0.88	**1.76**	1.71	**1.62**	**1.49**	**4.06**	**3.96**	**0.10**	**0.09**

3.4.1 Evaluating model sensitivity to input data parameters

Ideally, prediction models should be able to achieve good prediction performance with minimal observation horizons, to ensure an AV can react adequately even if a road user was just detected, or to ensure real-time computations. At the same time, longer observation horizons may contain subtle cues, which could improve prediction accuracy. Consequently, we test the impact of different observation horizons on the final prediction performance, while also considering different observation frequencies (*i.e.* timestep length δt of past observations), which vary considerably between different benchmarks [72, 142].

Generally in its benchmarks, the Argoverse2 dataset uses $n_I = 50$ input timesteps and $n_O = 60$ output timesteps with a timestep length of $\delta t = 0.1s$. To evaluate the influence of the given input, we still predict the same $6s$, but now with a timestep length of $\delta t \in \{0.5s, 0.25s, 0.1s\}$. To ensure that under each of these conditions the first observation is identical (so one setting does not has access to more observations) the time $T_I = \delta t(n_I - 1)$ from first observation to prediction time t_0 needs to be divisible by all tested δt. Therefore, we cannot use the common value of $T_I = 4.9$ s, and instead use $T_I = 4.5$ s as our large observation horizon and $T_I = 1.5$ s as the small one.

3.4.2 Evaluating the variability of the training process

The performance of most machine learning models is stochastic due to random initialization of weights or the random order in which training samples are presented. However, in many benchmarks, this randomness is often not acknowledged. Therefore, we investigated variability in model performance by training the same model multiple times[1].

In addition to varying the random seed of the model, we replaced evaluation on the predefined train/test partition of Argoverse2 with nine-fold cross-validation. Specifically, the whole dataset (train and test set combined) was randomly divided into nine equally large parts (matching the size of the predefined testing set in Argoverse2, which includes approx. 11% of all samples), each of which served as the test set in one of the nine training runs (with the other eight parts serving as the train set).

3.4.3 Evaluating generalizability

For a prediction model to be truly useful, it needs to be generalizable over different geographic locations. Therefore, we evaluated the performance of models trained on the Argoverse2 dataset (collected mostly in US) on a random subset of the rounD dataset (collected in Germany). We compared the resulting performance of the models to the same models but trained only on the remaining rounD data. Furthermore, we compared it to the models trained on Argoverse2 and then fine-tuned on rounD, using only a twentieth of the training epochs and 0.4 times the original learning rate compared to the model trained on rounD exclusively.

3.4.4 Evaluating robustness

Besides generalizability, another important aspect of model performance is robustness to input perturbations [84]. To illustrate how our framework can be used to test model robustness, we generated perturbations using adversarial attacks against the models trained exclusively on rounD in the previous experiment, with varying limits on allowed perturbations $d_{\max} \in \{0.25\,\mathrm{m}, 0.5\,\mathrm{m}, 1\,\mathrm{m}\}$. Here, we used attacks on control actions with limits on the perturbed future trajectory to ensure that comparisons between the ground truth future trajectories and the predictions resulting from the perturbed data are meaningful [140]. However, as these attacks rely on repeated calculations of the gradient through the target prediction model, we were unable to generate attacks on FJMP due to its intensive data preprocessing in numpy. All other models from the previous experiments were evaluated on respective perturbed datasets.

3.5 Results

After running the experiments described in the previous section, we will now discuss the results[2] for each of the three main aspects we investigated: sensitivity to input data parameters, variability of the training process, generalizability, and robustness against adversarial attacks.

[1] In our evaluation of different timestep lengths δt and observation horizons T_I, the best results were most often achieved for $\delta t = 0.25s$ and $T_I = 4.5s$ (Table 3.2). Consequently, in this and following experiments, we used these settings.

[2] The source code producing and the data underlying the show results can be found at `https://github.com/DAI-Lab-HERALD/General-Framework/tree/main`.

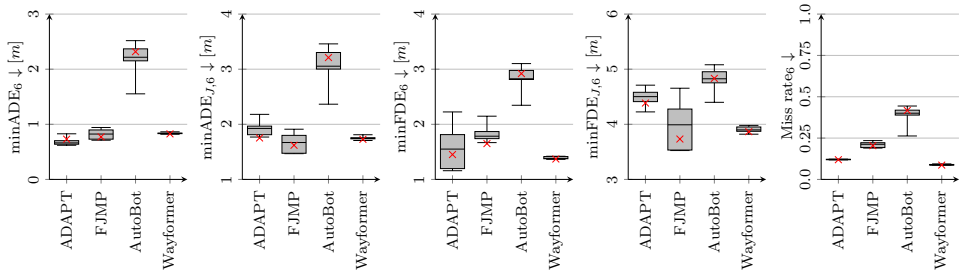


Figure 3.2: Variability of model performance due to variations in train/test split. Boxplots represent metric values resulting from the nine-fold cross validation on Argoverse2 with $\delta t = 0.25s$ and $T_I = 4.5s$. Red crosses show the values achieved under the same settings on the predefined test-set (Table 3.2).

3.5.1 Sensitivity to input data parameters

We found that out of the three tested δt values, $\delta t = 0.1s$ generally has the poorest overall results (Table 3.2) for both tested observation horizons. For $T_I = 4.5s$ specifically, only AutoBot achieves its best performance values across most of the metrics at $\delta t = 0.1s$, despite this setting being the closest to the standard setting for the Argoverse2 dataset. For all other models, the best results when using an input duration of $T_I = 4.5s$ are generally achieved with the timestep length of $\delta t = 0.25s$.

In the case of an input duration of $T_I = 1.5s$, the results are not as clear as with an input duration of $T_I = 4.5s$. The best performance is generally achieved at $\delta t = 0.5s$, however similar performance is often achieved with the timestep length of $\delta t = 0.25s$ as well.

These results suggest that inputs with small δt provide too detailed information to models, resulting in overfitting to minute differences and in turn a deterioration of performance. At the same time, there is no single combination of δt and T_I that consistently yields the best performance across models. In the case of an input duration of $T_I = 1.5s$, the lack of clear difference between $\delta t = 0.25s$ and $\delta t = 0.5s$ can be attributed to the short input duration, where an input timestep length of $\delta t = 0.25s$ does not provide much more additional information compared to an input timestep length of $\delta t = 0.5s$, particularly in urban driving scenarios where behavior changes over a time period of $T_I = 1.5s$ are generally minor.

Overall, our experiments demonstrate that the interplay between timestep length and observation horizon can substantially affect model performance, but is not trivial and thus warrants further investigation. At the same time, our framework can aid researchers in analyzing the sensitivity of models to these parameters.

3.5.2 Variability of the training process

Our experiment demonstrated that random variations in training data can have a major effect on model performance (Figure 3.2). Hence, evaluating models' performance based on a single training run on a predefined test-set (which is a common practice in the field) can be misleading.

For example, based on Table 3.2 it would appear that FJMP has superior performance on the joint metrics. However, Figure 3.2 shows that FJMP's performance in joint metrics

is highly variable depending on the variations in the train/test split and can be better or worse compared, for instance, to the performance of Wayformer.

Furthermore, some models exhibit higher levels of variability than others, with AutoBot especially showing consistently wide spreads in results, whereas Wayformer displays a consistently narrow spread. The low variance of Wayformer indicate that the high performance variability of models such as AutoBot is a result of both the model structure and training process, as opposed to being caused only by the random splitting of the dataset. As such, reporting the results for a instance of the model architectures – as is common in the field – can lead to a skewed perception of model performance. Reporting the results for a single instance of the model architectures is particularly an issue when the selection method for the used instances is not detailed.

Overall, our results suggest that the field would benefit from more transparent evaluation of model performance variability. Model developers can better illustrate the benefits of their models by training multiple model instances with varying random seeds and data splits – something that our framework can streamline as well. Additionally, designing benchmarks in which there is only a single predefined testing set, might introduce an artificial bias in model evaluation.

3.5.3 Generalizability

We found that models trained only on Argoverse2 were substantially worse on all performance metrics compared to models trained on rounD as well as Argoverse2-trained models fine-tuned on rounD (Table 3.3). This is as expected, as the models trained only on Argoverse2 are dealing with a scenario previously underrepresented or nonexistent in the training data. However, our results show that pre-training on a large dataset like Argoverse2 and then fine-tuning on a small subset of the target dataset (rounD) can result in performance comparable (and in some cases even superior) to a model trained on rounD from scratch with orders of magnitude more computational resources.

When comparing different models, we observed that for trajectory prediction metrics, Wayformer tends to outperforms other models in cases when the training data included (parts of) the rounD dataset. While FJMP was comparable to Wayformer when training and testing on Argoverse2 (Figure 3.2), it proved to generalize much worse to rounD.

Meanwhile, for those models trained only on Argoverse2, the best performance is generally produced by ADAPT, which seems to be the most generalizable model in the absence of fine-tuning.

3.5.4 Robustness

Lastly, we tested the four models' robustness against adversarial attacks, with the results shown in Table 3.4. On the whole, model performance generally deteriorates when perturbations are added to the data; this is to be expected of an adversarial attack. There are two main factors that can help mitigate the effect of adversarial attacks based on the obtained results; increased size and diversity of the training set, and the model's architecture. In the case of FJMP, Autobot and Wayformer, models trained on the larger, more diverse Argoverse2 dataset exhibit a lower degree of performance degradation than those trained only on the smaller, roundabout focused rounD dataset. While these models had previously been the worst-performing ones in the unperturbed case, the lower degree of

Table 3.3: Evaluation of generalizability of models tested on rounD. In each cell, the upper value is is the result for the model instance trained only on Argoverse2 (A), the middle value is the result for the model trained on Argoverse2 before being fine-tuned on a small subset of rounD (AR), and the lower value is the result for the model trained only on rounD (R). The model with the best value for each metric and training process is shown in bold, while the best value achieved over the different training process by each model is underlined.

Model	Source	minADE	minADE_J	minFDE	minFDE_J	MR	AUC	ECE
ADAPT [76]	A	**3.06**	**4.08**	**6.34**	**10.03**	0.69	**0.98**	0.42
	AR	1.82	2.75	3.59	6.73	0.54	0.98	0.50
	R	1.74	2.58	3.62	6.07	0.51	**0.99**	0.42
FJMP [77]	A	5.81	7.26	15.07	18.49	0.84	0.97	**0.42**
	AR	2.61	3.22	6.99	8.87	0.70	**0.99**	**0.34**
	R	2.02	2.45	5.78	7.16	0.65	**0.99**	**0.26**
AutoBot [75]	A	5.00	6.21	12.46	15.87	0.83	0.97	0.50
	AR	1.88	2.34	3.18	**4.72**	0.39	0.99	0.50
	R	1.65	2.14	3.99	5.34	0.44	0.99	0.42
Wayformer [35]	A	3.21	4.36	8.07	12.02	**0.62**	0.98	0.50
	AR	**1.27**	**1.88**	**2.68**	4.83	**0.19**	0.99	0.50
	R	**1.36**	**1.95**	**2.98**	**4.91**	**0.26**	0.97	0.50

performance deterioration in the case of perturbations could be explained by the increased size and diversity of the Argoverse2 dataset.

Nevertheless, the training data does not seem to be the only factor affecting a model's robustness, as illustrated by the ADAPT model. When attacked with perturbations optimized against itself, ADAPT' s performance decrease is roughly 34% when allowing perturbations $d_{\max} = 1\,\text{m}$, compared to the more than 6 to 7 fold increase in displacement error for Autobot and Wayformer respectively. However, we can generally observe that larger allowed perturbations when attacking ADAPT will lead to worse performance of the same model. Interestingly, when the perturbation are generated by attacks on other models, the performance of ADAPT stays fairly comparable to that on the unperturbed data (especially for displacements $d_{\max} \leq 0.5\,\text{m}$), and even improves in some cases. This could be due to the fact that as part of the perturbation, the trajectories are smoothed to ensure reasonable curvatures and accelerations. Combined with fairly small perturbations, this smoothing could positively contribute to the final performance of ADAPT. Overall ADAPT's architecture seems to innately be more robust compared to other methods, as was already observed previously when testing against distribution shifts. A deeper analysis of the model aspects as well as how the structure of the training datasets contribute to a model's robustness are valuable lines for future research.

Table 3.4: A comparison of different models on rounD with different training procedures and different perturbation settings for the metric minADE. In each cell, the upper value is is the result for the model trained only on Argoverse2 (A), the middle value is the result for the model trained on Argoverse2 before being finetuned on rounD (AR), and the lower value is the result for the model trained only on rounD (R). Meanwhile, the columns display the different attacked models and the maximum allowed perturbation of each attack. The model with the best value for each attack is shown in bold.

Model	Perturbation Source / $d_{\max}$	None	ADAPT [76]			AutoBot [75]			Wayformer [35]		
			0.25 m	0.5 m	1 m	0.25 m	0.5 m	1 m	0.25 m	0.5 m	1 m
ADAPT [76]	A	**3.06**	**3.24**	**3.40**	**3.61**	**3.14**	**3.15**	**3.19**	**3.14**	**3.18**	**3.23**
	AR	1.82	**1.92**	**2.10**	**2.36**	**1.82**	**1.82**	**1.89**	**1.83**	**1.83**	**1.88**
	R	1.74	**1.82**	**1.99**	**2.32**	**1.62**	**1.73**	**1.79**	**1.66**	**1.73**	**1.83**
FJMP [77]	A	5.81	6.59	6.73	6.88	6.54	6.54	6.54	6.53	6.52	6.54
	AR	2.61	12.1	12.1	12.0	12.1	12.1	12.1	12.0	12.1	12.1
	R	2.02	12.6	12.6	12.5	12.6	12.6	12.6	12.6	12.7	12.7
AutoBot [75]	A	5.00	9.32	9.38	9.45	9.26	9.26	9.30	9.27	9.28	9.30
	AR	1.88	12.3	12.3	12.3	12.3	12.3	12.3	12.3	12.3	12.3
	R	1.65	14.5	14.5	14.5	14.5	14.5	14.5	14.5	14.5	14.5
Wayformer [35]	A	3.21	8.27	8.43	8.53	8.23	8.23	8.26	8.25	8.29	8.33
	AR	**1.27**	9.36	9.46	9.61	9.33	9.33	9.40	9.35	9.50	9.64
	R	**1.36**	10.2	10.2	10.3	10.2	10.2	10.2	10.2	10.2	10.2

Compared to ADAPT, the other models seem to be much more sensitive even to small perturbations, with attacks performed on different models still being similarly successful overall, while the allowed magnitude in displacements only had a minor influence. This highlights a potential vulnerability: evidently, adversarial agents would not need to have access to the specific model an AV might be using to perform a potentially dangerous attack against, as tuning the attack to another model trained on similar data might be sufficient to generate successful attacks.

3.5.5 Comparing model performance

Across the four experiments, no single model could consistently outperform the other models, not even on a single metric. For example, while Wayformer generally achieved better performance in the input sensitivity tests (Table 3.2), exhibited low variability in the presence of random initializations (Figure 3.2), and reasonable generalizability particularly when fine-tuning on the target dataset (Table 3.3) it showed very poor robustness to perturbations (Table 3.4). Conversely, ADAPT – while not always achieving the best results in other experiments – demonstrated superior robustness.

Similarly, differences in model performance could also be observed when comparing trajectory-based and behavior classification metrics in the generalizability experiment on

rounD (Table 3.3). There, despite its poor performance on the trajectory-based metrics, FJMP achieved the best results in terms of behavior classification metrics (AUC and ECE), suggesting that the scene-wise prediction approach of FJMP allows for a superior prediction of high-level interactions (such as the gap acceptance behavior in rounD) compared to usually superior single-agent predictors such as WayFormer.

3.6 Conclusion

With this work, we introduced STEP, our framework for benchmarking trajectory prediction models. We showed that the framework is able to apply the same model to different datasets, provide far-reaching control over the actual structure of the dataset, and is able to create challenging testing regimes using perturbations such as adversarial attacks. Based on those experiments, we found that more and denser input data does not necessarily correspond to better performance, even for the largest state-of-the-art models. Additionally, minute differences in model performance are often not statistically significant, given the noise introduced by the model training process. Furthermore, most models are not directly transferable to scenarios outside their training domain, but can produce acceptable results with simple finetuning. And lastly, especially large prediction models are very susceptible to adversarial attacks, even if those are performed against a different model.

Our findings indicate that a more holistic evaluation of models – such as the one we have performed here with STEP – is needed to determine their applicability in real-world tasks. Furthermore, while state-of-the-art models are achieving rather good performance under the standard setup of training and testing on given datasets, there remains room for future contributions on matters related to generalizability and robustness.

4

Robust Multi-Modal Density Estimator

One part of the evaluation framework introduced in the previous chapter is its ability to use density based metrics to evaluate models which do not allow explicit density calculation. The estimation of such implicit probability density functions is, however, a general problem in science and engineering. As an answer to this, we present ROME (RObust Multi-modal Estimator), a non-parametric approach for density estimation which addresses the challenge of estimating multi-modal, non-normal, and highly correlated distributions – such as found in trajectory prediction. ROME utilizes clustering to segment a multi-modal set of samples into multiple uni-modal ones and then combines simple KDE estimates obtained for individual clusters in a single multi-modal estimate. We compared our approach to state-of-the-art methods for density estimation as well as ablations of ROME, showing that it not only outperforms established methods but is also more robust to a variety of distributions, overcoming the issues of over-fitting and over-smoothing exhibited by other estimators.

This chapter is based on *J. F. Schumann, A. Mészáros, J. Alonso-Mora, A. Zgonnikov, J. Kober. "Rome: Robust multi-modal density estimator" 2024 International Joint Conference on Artificial Intelligence (IJCAI). IJCAI, 2024.* [146]

4.1 Introduction

Numerous processes are non-deterministic by nature, from geological and meteorological occurrences, biological activities, as well as the behavior of living beings. Estimating the underlying probability density functions (PDFs) of such processes enables a better understanding of them and opens possibilities for probabilistic inference regarding future developments. Density estimation is instrumental in many applications including classification, anomaly detection, and evaluating probabilistic AI models, such as generative adversarial networks [148], variational autoencoders [149], normalizing flows [150], and their numerous variations.

When probabilistic models are trained on multi-modal data, they are often evaluated using simplistic metrics (*e.g.*, mean squared error (MSE) between the predicted and ground truth samples). However, such simplistic metrics are unsuited for determining how well a predicted distribution corresponds to the underlying distribution, as they do not capture the fit to the whole distribution. For example, the lowest MSE value between true and predicted samples could be achieved by accurate predictions of the mean of the true underlying distribution whereas potential differences in variance or shape of the distribution would not be penalized. This necessitates more advanced metrics that evaluate the match between the model and the (potentially multi-modal) data. For instance, negative log-likelihood (NLL), Jensen-Shannon divergence (JSD), and expected calibration error (ECE) can be used to evaluate how well the *full distribution* of the data is captured by the learned models [15, 151, 152]. However, most data-driven models represent the learned distribution implicitly, only providing individual samples and not the full distribution as an output. This complicates the comparison of the model output to the ground-truth data distributions since the above metrics require distributions, not samples, as an input. Practically, this can be addressed by estimating the predicted probability density based on samples generated by the model.

To this end, simple methods like Gaussian mixture models (GMM), kernel density estimation (KDE) [153], and k-nearest neighbors (kNN) [154] are commonly used for estimating probability density functions. These estimators however rely on strong assumptions about the underlying distribution, and can thereby introduce biases or inefficiencies in the estimation process. For example, problems can arise when encountering multi-modal, non-normal, and highly correlated distributions (Section 4.2). While more advanced methods such as manifold Parzen windows (MPW) [155] and vine copulas (VC) [156] exist, they have not been thoroughly tested on such problems, which raises questions about their performance.

To overcome these limitations, we propose a novel density estimation approach: the RObust Multi-modal Estimator (ROME). ROME employs a non-parametric clustering approach to segment potentially multi-modal distributions into separate uni-modal ones (Figure 4.1). These uni-modal sub-distributions are then estimated using a downstream probability density estimator (such as KDE). We test our proposed approach against a number of existing density estimators in three simple two-dimensional benchmarks designed to evaluate a model's ability to successfully reproduce multi-modal, highly-correlated, and non-normal distributions. Finally, we test our approach in a real-world setting using a distribution over future trajectories of human pedestrians created based on the Forking Paths dataset [157].

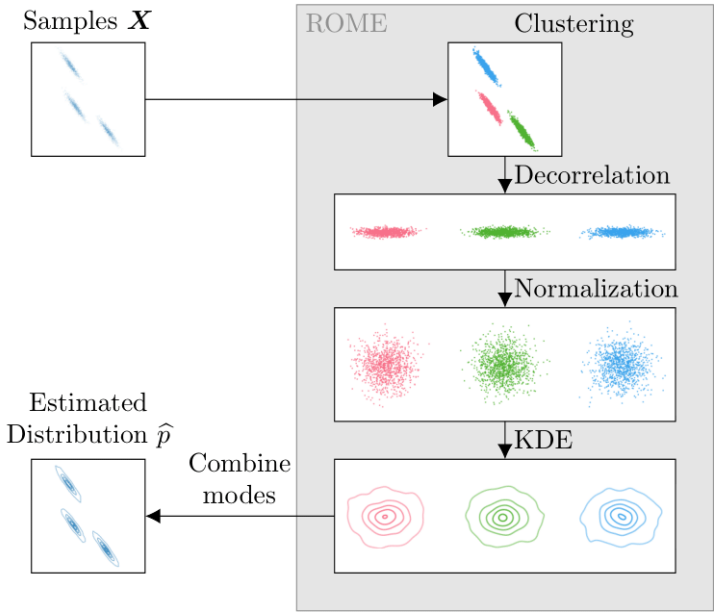


Figure 4.1: ROME takes samples from unknown distributions and estimates their densities to enable further downstream applications.

4.2 Related work

The most common class of density estimators are so-called Parzen windows [158], which estimate the density through an aggregation of parametric probability density functions. A number of common methods use this approach, with KDE being a common non-parametric method [159]. Provided a type of kernel – which is oftentimes a Gaussian but can be any other type of distribution – KDE places the kernels around each sample of a data distribution and then sums over these kernels to get the final density estimation over the data. This method is often chosen as it does not assume any underlying distribution type [159]. However, if the underlying distribution is highly correlated, then the common use of a single isotropic kernel function can lead to over-smoothing [155, 160]. Among multiple approaches for overcoming this issue [160, 161], especially noteworthy is the MPW approach [155]. It uses a unique anisotropic kernel for every datapoint, estimated based on the correlation of the k-nearest neighbors of each sample. However, it has not been previously tested in high-dimensional benchmarks, which is especially problematic as the required memory scales quadratically with the dimensionality of the problem.

Another common subtype of Parzen windows are GMMs [153], which assume that the data distribution can be captured through a weighted sum of Gaussian distributions. The parameters of the Gaussian distributions – also referred to as components – are estimated through expected likelihood maximization. Nonetheless, especially for non-normal distributions, one needs to have prior knowledge of the expected number of components to achieve a good fit without over-fitting to the training data or over-smoothing [162].

Besides different types of Parzen windows, a prominent alternative is kNN [154] which

uses the local density of the k nearest neighbors of every data point to estimate the overall density. While this method is non-parametric, it cannot be guaranteed that the resulting distribution will be normalized [159]. This could be rectified by using Monte Carlo sampling to obtain a good coverage of the function's domain and obtain an accurate estimate of the normalization factor, which, however, becomes computationally intractable for high-dimensional distributions.

When it comes to estimating densities characterized by correlations between dimensions, copula-based methods are an often favored approach. Copula-based methods decompose a distribution into its marginals and an additional function, called a copula, which describes the dependence between these marginals over a marginally uniformly distributed space. The downside of most copula-based approaches is that they rely on choosing an adequate copula function (*e.g.*, Gaussian, Student, or Gumbel) and estimating their respective parameters [163]. One non-parametric copula-based density estimator [164] aims to address this limitation by estimating copulas with the help of superimposed Legendre polynomials. While this can achieve good results in estimating the density function, it may become computationally intractable as the distribution's dimensionality increases. Another approach involves the use of VC [156], which assume that the whole distribution can be described as a product of bivariate Gaussian distributions, thus alleviating the curse of dimensionality. Its convergence, however, can only be guaranteed for normal distributions. Elsewhere [165], a similar approach was pursued, with changes such as using logarithmic splines instead of univariate KDEs for estimating the marginals. However, both of these approaches are not designed for multi-modal distributions and have not been thoroughly tested on such problems.

4.3 ROME

The problem of density estimation can be formalized as finding a queryable $\hat{p} \in \mathscr{P}$, where

$$\mathscr{P} = \left\{ g : \mathbb{R}^M \longrightarrow \mathbb{R}^+ \,|\, \int g(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} = 1 \right\},$$

such that $\hat{p}$ is close to the non-queryable PDF p underlying the N available M-dimensional samples $\boldsymbol{X} \in \mathbb{R}^{N \times M}$: $\boldsymbol{X} \sim p$.

A solution to the above problem would be an estimator $f : \mathbb{R}^{N \times M} \longrightarrow \mathscr{P}$, resulting in $\hat{p} = f(\boldsymbol{X})$. Our proposed estimator f_{ROME}[1] (Algorithm 1) is built on top of non-parametric cluster extraction. Namely, by separating groups of samples surrounded by areas of low density – expressing the mode of the underlying distribution – we reduce the multi-modal density estimation problem to multiple uni-modal density estimation problems for each cluster. The distributions within each cluster then become less varied in density or correlation than the full distribution. Combining this with decorrelation and normalization, the use of established methods such as KDE to estimate probability densities for those uni-modal distributions is now more promising, as problems with multi-modality and correlated modes (Section 4.2) are accounted for. The multi-modal distribution is then obtained as a weighted average of the estimated uni-modal distributions.

[1]The source code and data underlying the reported results can be found at `https://github.com/anna-meszaros/ROME`.

Algorithm 1 ROME

function TRAINROME($\boldsymbol{X}$)
▷ *Clustering (OPTICS)*
$\boldsymbol{X}_{I,N}, \boldsymbol{R}_N \leftarrow$ REACHABILTYANALYSIS($\boldsymbol{X}$)
$\boldsymbol{C}, S \leftarrow \{\{1,\ldots,N\}\}, -1.1$
for all $\epsilon \in \boldsymbol{\epsilon}$ **do**
$\boldsymbol{C}_\epsilon \leftarrow$ **DBSCAN**$(\boldsymbol{R}_{I,N}, \epsilon)$
$S_\epsilon \leftarrow$ **SIL**$(\boldsymbol{C}_\epsilon, \boldsymbol{X}_{I,N})$
if $S_\epsilon > S$ **then**
$\boldsymbol{C}, S \leftarrow \boldsymbol{C}_\epsilon, S_\epsilon$
end if
end for
for all $\xi \in \boldsymbol{\xi}$ **do**
$\boldsymbol{C}_\xi \leftarrow \boldsymbol{\xi}$**-clustering**$(\boldsymbol{R}_{I,N}, \xi)$
$S_\xi \leftarrow$ **SIL**$(\boldsymbol{C}_\xi, \boldsymbol{X}_{I,N})$
if $S_\xi > S$ **then**
$\boldsymbol{C}, S \leftarrow \boldsymbol{C}_\xi, S_\xi$
end if
end for
for all $C \in \boldsymbol{C}$ **do**
▷ *Decorrelation*
$\overline{\boldsymbol{x}}_C \leftarrow \text{MEAN}(\boldsymbol{X}_C)$
$\overline{\boldsymbol{X}}_C \leftarrow \boldsymbol{X}_C - \overline{\boldsymbol{x}}_C$
$\boldsymbol{R}_C \leftarrow PCA(\overline{\boldsymbol{X}}_C)$
▷ *Normalization*
$\widetilde{\boldsymbol{\Sigma}}_C \leftarrow \text{STD}(\overline{\boldsymbol{X}}_C \boldsymbol{R}_C^T)$
$\boldsymbol{T}_C \leftarrow \boldsymbol{R}_C^T \widetilde{\boldsymbol{\Sigma}}_C^{-1}$
▷ *PDF Estimation*
$\hat{p}_C \leftarrow f_{\text{KDE}}\left(\overline{\boldsymbol{X}}_C \boldsymbol{T}_C\right)$
end for
return $\boldsymbol{C}, \left\{\hat{p}_C, \overline{\boldsymbol{x}}_C, \boldsymbol{T}_C \mid C \in \boldsymbol{C}\right\}$
end function

function QUERYROME($\boldsymbol{x}$, $\boldsymbol{X}$)
$\boldsymbol{C}, \left\{\hat{p}_C, \overline{\boldsymbol{x}}_C, \boldsymbol{T}_C \mid C \in \boldsymbol{C}\right\} \leftarrow$ TRAINROME($\boldsymbol{X}$)
$l = 0$
for all $C \in \boldsymbol{C}$ **do**
$\widehat{\boldsymbol{x}} \leftarrow (\boldsymbol{x} - \overline{\boldsymbol{x}}_C)\, \boldsymbol{T}_C$
$l \leftarrow l + \ln(\hat{p}_C(\widehat{\boldsymbol{x}})) + \ln(|C|) - \ln N + \ln(|\det(\boldsymbol{T}_C)|)$
end for
return $\exp(l)$
end function

4.3.1 Extracting clusters

To cluster samples $\boldsymbol{X}$, ROME employs the OPTICS algorithm [166] that can detect clusters of any shape with varying density using a combination of reachability analysis – which orders the data in accordance to reachability distances – followed by clustering the ordered data based on these reachability distances.

In the first part of the algorithm, the reachability analysis is used to sequentially transfer samples from the set of not yet included samples $\boldsymbol{X}_{U,i}$ to the set of included and ordered samples $\boldsymbol{X}_{I,i}$, starting with a random sample $\boldsymbol{x}_1$ ($\boldsymbol{X}_{I,1} = \{\boldsymbol{x}_1\}$ and $\boldsymbol{X}_{U,1} = \boldsymbol{X}\backslash\{\boldsymbol{x}_1\}$). The samples $\boldsymbol{x}_{i+1}$ are then selected at iteration i based on the reachability distance r:

$$\boldsymbol{x}_{i+1} = \underset{\boldsymbol{x}\in\boldsymbol{X}_{U,i}}{\arg\min}\, r(\boldsymbol{x}, \boldsymbol{X}_{I,i}) = \underset{\boldsymbol{x}\in\boldsymbol{X}_{U,i}}{\arg\min}\ \underset{\widetilde{\boldsymbol{x}}\in\boldsymbol{X}_{I,i}}{\min}\ d_r(\boldsymbol{x}, \widetilde{\boldsymbol{x}})\,. \tag{4.1}$$

This sample is then transferred between sets (to the explored set $\boldsymbol{X}_{I,i+1} = \boldsymbol{X}_{I,i} \cup \{\boldsymbol{x}_{i+1}\}$ from the unexplored set $\boldsymbol{X}_{U,i+1} = \boldsymbol{X}_{U,i}\backslash\{\boldsymbol{x}_{i+1}\}$), while expanding the reachability set $\boldsymbol{R}_{i+1} = \boldsymbol{R}_i \cup \{r(\boldsymbol{x}_{i+1}, \boldsymbol{X}_{I,i})\}$ (with $\boldsymbol{R}_1 = \{\infty\}$). The reachability distance d_r in Equation (4.1) is defined as

$$d_r(\boldsymbol{x}, \widetilde{\boldsymbol{x}}) = \max\left\{\|\boldsymbol{x} - \widetilde{\boldsymbol{x}}\|,\ \min_{k_c}_{\widehat{\boldsymbol{x}}\in\boldsymbol{X}\backslash\{\boldsymbol{x}\}} \|\boldsymbol{x} - \widehat{\boldsymbol{x}}\|\right\},$$

where $\min_{k_c}$ is the k_c-smallest value of all the available $\widehat{\boldsymbol{x}}$, used to smooth out random local density fluctuations. We use

$$k_c = \min\left\{k_{\max}, \max\left\{k_{\min}, \frac{NM}{\alpha_k}\right\}\right\}, \tag{4.2}$$

where $k_{\min}$ and $k_{\max}$ ensure there are sufficient but not too many points for this smoothing to ensure stability against random fluctuations while still relying on sufficiently local information. Meanwhile, the term NM/α_k adjusts k_c to the number of samples and dimensions, allowing for the higher number of samples needed in higher-dimensional spaces.

The second part of the OPTICS algorithm – after obtaining the reachability distances $\boldsymbol{R}_N$ and the ordered set $\boldsymbol{X}_{I,N}$ – is the extraction of a set of clusters $\boldsymbol{C}$, with clusters $C = \{c_{\min}, \ldots, c_{\max}\} \in \boldsymbol{C}$ (with $\boldsymbol{X}_C = \{\boldsymbol{x}_{I,N,j} \,|\, j \in C\} \in \mathbb{R}^{|C|\times M}$). As the computational cost of creating such a cluster set is negligible compared to the reachability analysis, we can test multiple clusterings generated using two different approaches (with $r_{\text{bound}} = \min\{r_{N,c_{\min}}, r_{N,c_{\max}+1}\}$), which means that we are not dependent on a pre-selected parameterization of the clustering algorithm:

- First, we use **DBSCAN** [167] for generating the clustering $\boldsymbol{C}_\epsilon$ based on an absolute limit ϵ, where a cluster must fulfill the condition:

$$r_{N,c} < \epsilon \leq r_{\text{bound}}\ \forall\, c \in C\backslash\{c_{\min}\}. \tag{4.3}$$

  In practice, this assumes that a cluster is defined by the fact that all of its samples are closer together than a predefined distance, which is the same for all clusters in a dataset. However, this assumes that clusters generally have a similar density, which might not always be the case.

- Second, we use **ξ-clustering** [166]. Compared to **DBSCAN**, which is based on absolute density, we here select clusters based on relative density (*i.e.*, the points of a cluster are sufficiently closer than the surrounding points). Based on the proportional limit ξ, we generate the clustering $\boldsymbol{C}_\xi$, for which a cluster fulfills the condition:

$$\xi \leq 1 - \frac{r_{N,c}}{r_{\text{bound}}} \; \forall\, c \in C \backslash \{c_{\min}\}. \tag{4.4}$$

  However, this approach is more sensitive to noise, making it more difficult to separate noise from small clusters.

In both cases, each prospective cluster C also has to fulfill the condition $|C| \geq 2$. However, it is possible that not every sample can be fit into a cluster fulfilling the conditions above. These samples are then kept in a separate noise cluster C_{noise} that does not have to fulfill those conditions ($C_{\text{noise}} \in \boldsymbol{C}_\epsilon$ or $C_{\text{noise}} \in \boldsymbol{C}_\xi$ respectively). Upon generating multiple sets of clusters $\boldsymbol{C}_\epsilon$ ($\epsilon \in \boldsymbol{\varepsilon}$) and $\boldsymbol{C}_\xi$ ($\xi \in \boldsymbol{\xi}$), we select the final set of clusters $\boldsymbol{C}$ that achieves the highest silhouette score[2] $S = \mathbf{SIL}(\boldsymbol{C}, \boldsymbol{X}_{I,N}) \in [-1, 1]$ [168]. The clustering then allows us to use PDF estimation methods on uni-modal distributions.

4.3.2 Feature decorrelation

In much of real-life data, such as the distributions of a person's movement trajectories, certain dimensions of the data are likely to be highly correlated. Therefore, the features in each cluster $C \in \boldsymbol{C}$ should be decorrelated using a rotation matrix $\boldsymbol{R}_C \in \mathbb{R}^{M \times M}$. In ROME, $\boldsymbol{R}_C$ is found using principal component analysis (PCA) [169] on the cluster's centered samples $\overline{\boldsymbol{X}}_C = \boldsymbol{X}_C - \overline{\boldsymbol{x}}_C$ ($\overline{\boldsymbol{x}}_C$ is the cluster's mean value). An exception are the noise samples in C_{noise}, which are not decorrelated (*i.e.*, $\boldsymbol{R}_{C_{\text{noise}}} = \boldsymbol{I}$). One can then get the decorrelated samples $\boldsymbol{X}_{\text{PCA},C}$:

$$\boldsymbol{X}_{\text{PCA},C}^T = \boldsymbol{R}_C \overline{\boldsymbol{X}}_C^T.$$

4.3.3 Normalization

After decorrelation, we use the matrix $\widetilde{\boldsymbol{\Sigma}}_C \in \mathbb{R}^{M \times M}$ to normalize $\boldsymbol{X}_{\text{PCA},C}$:

$$\widehat{\boldsymbol{X}}_C = \boldsymbol{X}_{\text{PCA},C} \left(\widetilde{\boldsymbol{\Sigma}}_C\right)^{-1} = \overline{\boldsymbol{X}}_C \boldsymbol{R}_C^T \left(\widetilde{\boldsymbol{\Sigma}}_C\right)^{-1} = \overline{\boldsymbol{X}}_C \boldsymbol{T}_C.$$

Here, $\widetilde{\boldsymbol{\Sigma}}_C$ is a diagonal matrix with the entries $\widetilde{\sigma}_C^{(m)}$. To avoid over-fitting to highly correlated distributions, we introduce a regularization with a value $\sigma_{\min}$ (similar to [155]) that is applied to the empirical standard deviation $\sigma_{\text{PCA},C}^{(m)} = \sqrt{\mathbb{V}_m\left(\boldsymbol{X}_{\text{PCA},C}\right)}$ ($\mathbb{V}_m$ is the variance along feature m) of each rotated feature m (with $\boldsymbol{C}' = \boldsymbol{C} \backslash \{C_{\text{noise}}\}$):

$$\widetilde{\sigma}_C^{(m)} = \begin{cases} \max\left\{ \sum\limits_{\mathscr{C} \in \boldsymbol{C}'} \frac{\sqrt{\mathbb{V}_m(\boldsymbol{X}_{\mathscr{C}})}}{|\boldsymbol{C}|-1}, \sigma_{\min} \right\} & C = C_{\text{noise}} \\ \left(1 - \frac{\sigma_{\min}}{\max\limits_m \sigma_{\text{PCA},C}^{(m)}}\right) \sigma_{\text{PCA},C}^{(m)} + \sigma_{\min} & \text{otherwise} \end{cases}.$$

[2] The silhouette score measures the similarity of each object to its own cluster's objects compared to the other clusters' objects.

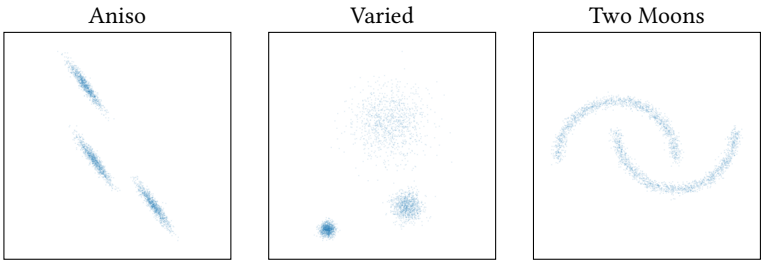


Figure 4.2: Samples from the two-dimensional synthetic distributions used for evaluating different PDF estimators.

4.3.4 Estimating the probability density function

Taking the transformed data (decorrelated and normalized), ROME fits the Gaussian KDE f_{KDE} on each separate cluster C as well as the noise samples C_{noise}. For a given bandwidth b_C for data in cluster C, this results in a partial PDF $\hat{p}_C$.

$$\hat{p}_C(\hat{\boldsymbol{x}}) = f_{\text{KDE}}\left(\hat{\boldsymbol{X}}_C\right)(\hat{\boldsymbol{x}}) = \frac{1}{|C|}\sum_{\boldsymbol{\chi}\in\hat{\boldsymbol{X}}_C} \mathcal{N}\left(\hat{\boldsymbol{x}}|\boldsymbol{\chi}, b_C\boldsymbol{I}\right).$$

The bandwidth b_C is set using Silverman's rule [159]:

$$b_C = \left(\frac{M+2}{4}n_C\right)^{-\frac{1}{M+4}},\ n_C = \begin{cases} 1 & C = C_{\text{noise}} \\ |C| & \text{else} \end{cases}.$$

To evaluate the density function $\hat{p} = f_{\text{ROME}}(\boldsymbol{X})$ for a given sample $\boldsymbol{x}$, we take the weighted averages of each cluster's $\hat{p}_C$:

$$\hat{p}(\boldsymbol{x}) = \sum_{C\in\boldsymbol{C}} \frac{|C|}{N}\hat{p}_C\left((\boldsymbol{x}-\overline{\boldsymbol{x}}_C)\boldsymbol{T}_C\right)|\det(\boldsymbol{T}_C)|.$$

Here, the term $|C|/N$ is used to weigh the different distributions of each cluster with the size of each cluster, so that each sample is represented equally. As the different KDEs $\hat{p}_C$ are fitted to the transformed samples, we apply them not to the original sample $\boldsymbol{x}$, but instead apply the identical transformation used to generate those transformed samples, using $\hat{p}_C((\boldsymbol{x}-\overline{\boldsymbol{x}}_C))$. To account for the change in density within a cluster C introduced by the transformation $\boldsymbol{T}_C$, we use the factor $|\det(\boldsymbol{T}_C)|$.

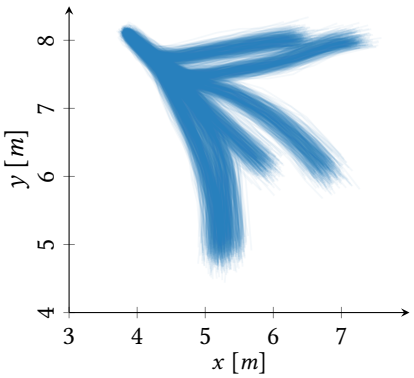


Figure 4.3: Samples from the multi-modal pedestrian trajectory distribution [157] used for evaluating different PDF estimators. Trajectories span 12 timesteps recorded at 2.5 Hz.

4.4 Experiments

We compare our approach against two baseline ones from the literature (VC [156] and MPW [155]) in four scenarios, using three metrics. Additionally, we carry out an ablation study on our proposed method *ROME*.

For the hyperparameters pertaining to the clustering within ROME (Section 4.3.1), we found empirically that stable results can be obtained using 199 possible clusterings, 100 for DBSCAN (Equation (4.3))

$$\boldsymbol{\varepsilon} = \left\{ \min R_N + \left(\frac{\alpha}{99}\right)^2 (\max(R_N \backslash \{\infty\}) - \min R_N) \,|\, \alpha \in \{0, \dots, 99\} \right\}$$

combined with 99 for ξ-clustering (Equation (4.4))

$$\boldsymbol{\xi} = \left\{ \frac{\beta}{100} \,|\, \beta \in \{1, \dots, 99\} \right\},$$

as well as using $k_{\min} = 5$, $k_{\max} = 20$, and $\alpha_k = 400$ for calculating $k_{\boldsymbol{c}}$ (Equation (4.2)). A justification for these choices can be found in Figure 4.4.

4.4.1 Distributions

In order to evaluate different aspects of a density estimation method f, we used a number of different distributions.

- Three two-dimensional synthetic distributions (Figure 4.2) were used to test the estimation of distributions with multiple clusters, which might be highly correlated (Aniso) or of varying densities (Varied), or express non-normal distributions (Two Moons).
- A multivariate, 24-dimensional, and highly correlated distribution generated from a subset of the Forking Paths dataset [157] (Figure 4.3). The 24 dimensions correspond to the x and y positions of a human pedestrian across 12 timesteps. Based on 6

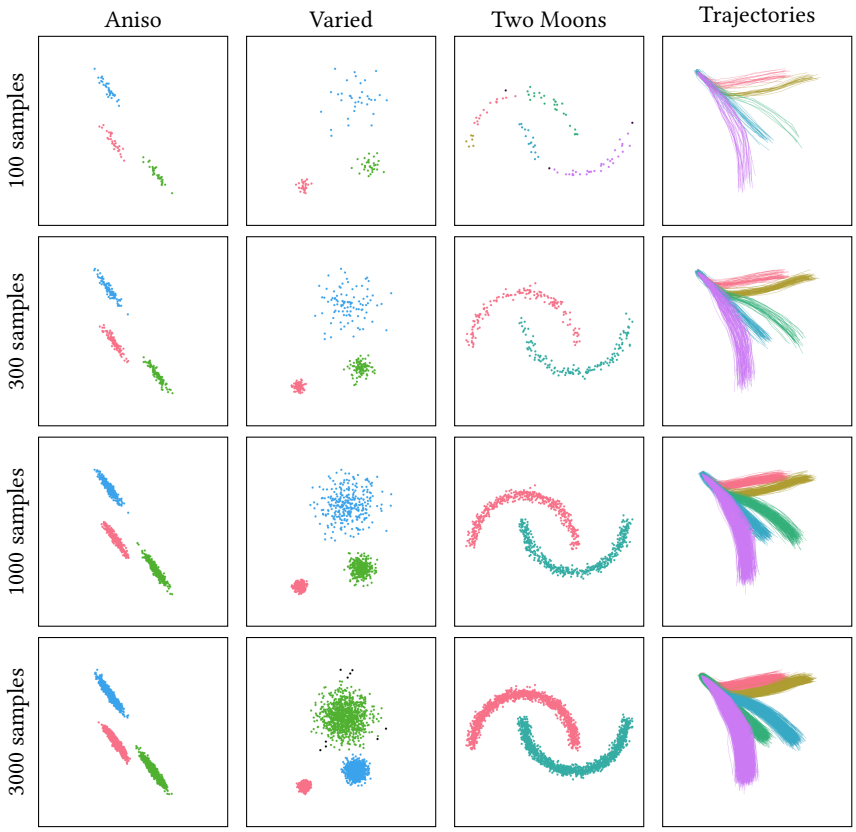


Figure 4.4: Clustering results of OPTICS on 2D distribution datasets and the trajectories when using the silhouette score for the selection criterion, as used in *ROME*.

original trajectories ($\boldsymbol{x}_i^* \in \mathbb{R}^{12\times 2}$), we defined the underlying distribution p in such a way, that one could calculate a sample $\boldsymbol{x} \sim p$ with:

$$\boldsymbol{x} = s\boldsymbol{x}_i^* \boldsymbol{R}_\theta^T + \boldsymbol{L}\boldsymbol{n}, \text{ with } i \sim \mathcal{U}\{1,6\}.$$

Here, $\boldsymbol{R}_\theta \in \mathbb{R}^{2\times 2}$ is a rotation matrix rotating $\boldsymbol{x}_i^*$ by $\theta \sim \mathcal{N}(0, \frac{\pi}{180})$, while $s \sim \mathcal{N}(1, 0.03)$ is a scaling factor. $\boldsymbol{n} \sim \mathcal{N}(0, 0.03\boldsymbol{I}) \in \mathbb{R}^{12\times 2}$ is additional noise added on all dimensions using $\boldsymbol{L} \in \mathbb{R}^{12\times 12}$, a lower triangular matrix that only contains ones.

4.4.2 Evaluation and metrics

When estimating density $\hat{p}$, since we cannot query the distribution p underlying the samples $\boldsymbol{X}$, we require metrics that can provide insights purely based on those samples. To this end we use the following three metrics to quantify how well a given density estimator f can avoid both over-fitting and over-smoothing.

- To test for over-fitting, we first sample two different datasets $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ (N samples

each) from p (*i.e.*, $\boldsymbol{X}_1, \boldsymbol{X}_2 \sim p$). We then use the estimator f to create two queryable distributions $\hat{p}_1 = f(\boldsymbol{X}_1)$ and $\hat{p}_2 = f(\boldsymbol{X}_2)$. If those distributions $\hat{p}_1$ and $\hat{p}_2$ are similar, it would mean the tested estimator does not over-fit; we measure this similarity using the Jensen-Shannon divergence [170]:

$$D_{\mathrm{JS}}(\hat{p}_1 \| \hat{p}_2) = \frac{1}{2N \ln(2)} \sum_{\boldsymbol{x} \in \boldsymbol{X}_1 \cup \boldsymbol{X}_2} h_1(\boldsymbol{x}) + h_2(\boldsymbol{x})$$

$$h_i(\boldsymbol{x}) = \frac{\hat{p}_i(\boldsymbol{x})}{\hat{p}_1(\boldsymbol{x}) + \hat{p}_2(\boldsymbol{x})} \ln\left(\frac{2\hat{p}_i(\boldsymbol{x})}{\hat{p}_1(\boldsymbol{x}) + \hat{p}_2(\boldsymbol{x})} \right)$$

- To test the goodness-of-fit of the estimated density, we first generate a third set of samples $\widehat{\boldsymbol{X}}_1 \sim \hat{p}_1$ with N samples. We then use the Wasserstein distance W [171] on the data to calculate the indicator $\widehat{W}$:

$$\widehat{W} = \frac{W(\boldsymbol{X}_1, \widehat{\boldsymbol{X}}_1) - W(\boldsymbol{X}_1, \boldsymbol{X}_2)}{W(\boldsymbol{X}_1, \boldsymbol{X}_2)}$$

  Here, $\widehat{W} > 0$ indicates over-smoothing or misplaced modes, while $-1 \leq \widehat{W} < 0$ indicates over-fitting.

- Not every density estimator f has the ability to generate the samples $\widehat{\boldsymbol{X}}_1$. Consequently, we need to test for goodness-of-fit without relying on $\widehat{\boldsymbol{X}}_1$. Therefore, we use the average log-likelihood

$$\hat{L} = \frac{1}{N} \sum_{\boldsymbol{x} \in \boldsymbol{X}_2} \ln\left(\hat{p}_1(\boldsymbol{x})\right),$$

  which, according to Gibbs' inequality [172], would be maximized only for $p = \hat{p}_1$ as long as $\boldsymbol{X}_2$ is truly representative of p. However, using this metric might be meaningless if $\hat{p}_1$ is not normalized, as the presence of the unknown scaling factor makes the $\hat{L}$ values of two estimators incomparable, and cannot discriminate between over-fitting and over-smoothing.

  However, given two log likelihoods $\hat{L}_A$ and $\hat{L}_B$ of two density estimation methods A and B, we can calculate a so-called *factor* $\mathcal{F}$ that expresses how much more likely method A is at reproducing the underlying distribution p compared to method B:

$$\mathcal{F} = \exp\left(\hat{L}_A - \hat{L}_B\right),$$

- If one cannot only samples from p, but also evaluate samples, we can compare the estimated distribution directly to the ground truth distribution to test for systematic biases. For this, we use

$$D_{\mathrm{JS}_{\mathrm{true}}} = D_{\mathrm{JS}}(\hat{p}_1 \| p).$$

For each candidate method f, we used $N = 3000$, and every metric calculation was repeated 100 times to take into account the inherent randomness of sampling from the distributions, with the standard deviation in the tables being reported with ±. For the comparative factor F, we can also measure the statistic significance of $\mathcal{F} > 1$ using a standard pairwise t-test based on the 100 individual values.

4.4.3 Ablations

To better understand the performance of our approach, we investigated variations in four key aspects of ROME:

- *Clustering approach.* First, we replaced the silhouette score (Section 4.3.1) with density based cluster validation (DBCV) [173] when selecting the optimal clustering out of the 199 possibilities. Furthermore, we investigated the approach of no clustering ($\boldsymbol{C} = \{\{1, \dots, N\}\}$).

- *Decorrelation vs No decorrelation.* We investigated the effect of removing rotation by setting $\boldsymbol{R}_C = \boldsymbol{I}$.

- *Normalization vs No normalization.* We studied the sensitivity of our approach to normalization by setting $\widetilde{\boldsymbol{\Sigma}}_C = \boldsymbol{I}$.

- *Downstream density estimator.* We replaced f_{KDE} with two other candidate methods. First, we used a single-component Gaussian mixture model f_{GMM}

$$f_{\text{GMM}}(\boldsymbol{X})(\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{x} | \hat{\boldsymbol{\mu}}_{\boldsymbol{X}}, \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{X}}\right)$$

  fitted to the observed mean $\hat{\boldsymbol{\mu}}$ and covariance matrix $\widehat{\boldsymbol{\Sigma}}$ of a dataset $\boldsymbol{X}$. Second, we used a k-nearest neighbor approach f_{kNN} [154]

$$f_{\text{kNN}}(\boldsymbol{X})(\boldsymbol{x}) = \frac{k}{N \, \mathcal{V}_M \min_{k, \hat{\boldsymbol{x}} \in \boldsymbol{X}} \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^M}$$

  where $\mathcal{V}_M$ is the volume of the M-dimensional unit hypersphere. We used the rule-of-thumb $k = \lfloor \sqrt{N} \rceil$. However, this estimator cannot generate samples.

While those four factors would theoretically lead to 24 estimators, f_{KDE} as well as f_{kNN} being invariant against rotation and f_{GMM} being invariant against any linear transformation means that only 14 of ROME's ablations are actually unique.

4.5 Results

4.5.1 Baseline comparison

We found that ROME avoids the major pitfalls displayed by the two baseline methods on the four tested distributions (Table 4.1). Out of the two baseline methods, the manifold Parzen windows (MPW) approach has a stronger tendency to over-fit in the case of the two-dimensional distributions compared to ROME, as quantified by lower D_{JS} values achieved by ROME. MPW does, however, achieve a better log-likelihood for the Two Moons distribution compared to ROME. This could be due to the locally adaptive non-Gaussian distributions being less susceptible to over-smoothing than our approach of using a single isotropic kernel for each cluster if such clusters are highly non-normal. Lastly, in the case of the pedestrian trajectories distribution, ROME once more achieves better performance than MPW, with MPW performing worse both in terms of D_{JS} and $\hat{L}$. Additionally, for

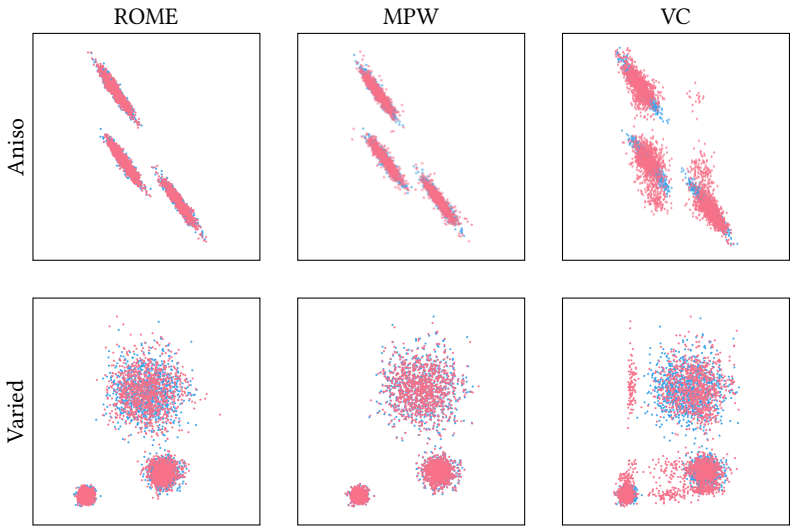


Figure 4.5: Samples obtained with ROME, MPW and VC (pink) contrasted with samples from p (blue).

Table 4.1: Baseline Comparison – marked in red are cases with notably poor performance; best values are underlined. As only Aniso and Varied have a queryable p, $D_{\mathrm{JS}_{\mathrm{true}}}$ is only available for those two datasets.

Dataset	Method	$D_{\mathrm{JS}} \downarrow_0^1$	$\widehat{W} \to 0$	$\hat{L} \uparrow$	$D_{\mathrm{JS}_{\mathrm{true}}} \downarrow_0^1$
Aniso	ROME	0.010±0.001	−0.13±0.31	−2.53±0.02	$0.007^{\pm 0.001}$
	MPW	0.026±0.002	−0.60±0.13	−2.57±0.02	$0.016^{\pm 0.001}$
	VC	0.005±0.001	1.91±0.91	−3.19±0.02	$0.143^{\pm 0.003}$
Varied	ROME	0.011±0.001	−0.13±0.20	−4.10±0.03	$0.078^{\pm 0.003}$
	MPW	0.025±0.002	−0.49±0.11	−4.12±0.03	$0.084^{\pm 0.003}$
	VC	0.008±0.001	1.27±0.53	−4.29±0.03	$0.087^{\pm 0.003}$
Two Moons	ROME	0.002±0.001	1.40±0.52	−1.02±0.01	-
	MPW	0.023±0.002	−0.52±0.10	−0.36±0.01	-
	VC	0.008±0.002	1.36±0.51	−0.95±0.01	-
Trajectories	ROME	0.008±0.002	1.03±0.22	29.32±0.02	-
	MPW	0.016±0.001	1.24±0.24	26.09±0.02	-
	VC	0.743±0.005	9.30±1.30	−215.2±17.6	-

those datasets where we can evaluate $D_{\mathrm{JS}_{\mathrm{true}}}$, ROME achieves the best performance, indicating that it is able to capture the underlying distribution more accurately than both MPW and VC.

Meanwhile, the vine copulas (VC) approach tends to over-smooth the estimated den-

Table 4.2: Ablations ($\widehat{W} \to 0$, Varied): Clustering is essential to prevent over-smoothing. ROME highlighted in gray. Note that the differences between Silhouette and DBCV are not statistically significant.

Cluster.	Norm. Decorr.	Norm. No decorr.	No norm.	f_{GMM}
Silhouette	−0.13±0.20	−0.13±0.20	−0.13±0.19	−0.14±0.20
DBCV	−0.09±0.23	−0.09±0.23	−0.07±0.23	−0.03±0.24
No clus.	2.28±0.72	2.26±0.72	−0.17±0.26	10.53±2.54

Table 4.3: Ablations ($\hat{L} \uparrow$, Aniso): When clustering, decorrelation and normalization improve results for distributions with high intra-mode correlation. ROME highlighted in gray.

Cluster.	Norm. Decorr.	Norm. No decorr.	No norm.
Silhouette	−2.53±0.02	−2.70±0.01	−2.79±0.01
DBCV	−2.56±0.02	−2.69±0.01	−2.83±0.02

sities (large positive $\widehat{W}$ values), and even struggles with capturing the different modes, as it less able to distinguish between them (Figure 4.5). This is likely because VC uses KDE with Silverman's rule of thumb, which is known to lead to over-smoothing in the case of multi-modal distributions [174]. Furthermore, on the pedestrian trajectories distribution, we observed both high D_{JS} and $\widehat{W}$ values, indicating that VC is unable to estimate the underlying density; this is also indicated by the poor log-likelihood estimates.

Overall, while the baselines were able to achieve better performance in selected cases (*e.g.*, MPW better than ROME in terms of $\widehat{W}$ and $\hat{L}$ on the Two Moons distribution), they have their apparent weaknesses. Specifically, MPW achieves poor results for most metrics in the case of varying densities within the modes (Varied), while Vine Copulas obtain the worst performance across all three metrics in the case of the multivariate trajectory distributions. ROME, in contrast, achieved high performance across all the test cases.

4.5.2 Ablation studies

When it comes to the choice of the clustering method, our experiments show no clear advantage for using either the silhouette score or DBCV. But as the silhouette score is computationally more efficient than DBCV, it is the preferred method. However, using clustering is essential, as otherwise there is a risk of over-smoothing, such as in the case of multi-modal distributions with varying densities in each mode (Table 4.2).

Testing variants of ROME on the Aniso distribution (Table 4.3) demonstrated not only the need for decorrelation, through the use of rotation, but also normalization in the case of distributions with highly correlated features. There, using either of the two clustering methods in combination with normalization and decorrelation (our full proposed method)

Table 4.4: Ablations ($\widehat{W} \longrightarrow 0$, Two Moons): Excluding normalization or using f_{GMM} as the downstream estimator is not robust against non-normal distributions. ROME highlighted in gray.

Cluster.	Norm.		No norm.	f_{GMM}
	Decorr.	No decorr.		
Silhouette	1.40±0.52	1.41±0.52	3.65±0.99	4.37±1.14
DBCV	1.59±0.56	1.59±0.56	4.24±1.13	4.77±1.23
No clus.	1.82±0.60	1.84±0.60	3.17±0.89	5.29±1.34

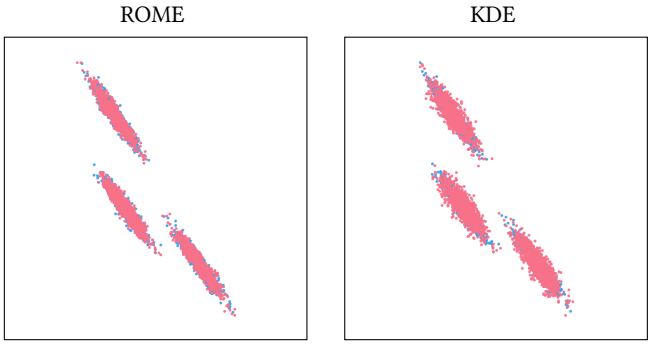


Figure 4.6: Samples generated by ROME and KDE – equivalent to ROME without clustering, decorrelation and normalization – (in pink) contrasted with samples from p (in blue); Aniso. Note that the samples by KDE are more spread out, indicating over-smoothing.

Table 4.5: Ablations ($D_{\text{JS}} \downarrow_0^1$, Trajectories; values are multiplied by 10 for easier comprehension): Using f_{kNN} as the downstream estimator tends to lead to over-fitting. ROME highlighted in gray.

	Clustering	Silhouette	DBCV	No clusters
Normalization & Decorrelation	f_{KDE}	0.084±0.016	0.090±0.015	0.009±0.004
	f_{kNN}	1.045±0.064	1.119±0.073	0.453±0.051
Normalization & No decorrelation	f_{KDE}	0.777±0.116	0.897±0.154	0.015±0.012
	f_{kNN}	1.808±0.112	1.937±0.109	1.044±0.104
No normalization & No decorrelation	f_{KDE}	0.015±0.011	0.017±0.012	0.005±0.003
	f_{kNN}	1.887±0.118	1.934±0.116	1.478±0.132
	f_{GMM}	0.032±0.007	0.043±0.010	0.017±0.011

is better than the two alternatives of omitting only decorrelation or both decorrelation and normalization. In the case of clustering with the silhouette score, the full method is significantly more likely to reproduce the underlying distribution p by a factor of 1.19 (with

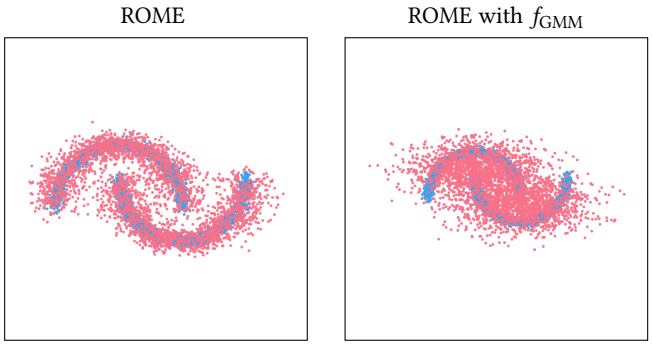


Figure 4.7: Samples generated by ROME, and ROME with f_{GMM} as the downstream estimator (in pink) contrasted with samples from p (in blue); Two Moons. Note that the samples from f_{GMM} are more spread out, which clearly displays over-smoothing.

a statistical significance of 10^{-50}) as opposed to omitting only decorrelation, and by 1.30 compared to omitting both decorrelation and normalization. Similar trends can be seen when clustering based on DBCV, with the full method being more likely to reproduce p by a factor of 1.14 and 1.31 respectively. Results on the Aniso distribution further show that KDE on its own is not able to achieve the same results as ROME, but rather it has a tendency to over-smooth (Figure 4.6). Additionally, the ablation with and without normalization on the Two Moons distribution (Table 4.4) showed that normalization is necessary to avoid over-smoothing on non-normal distributions.

Lastly, investigating the effect of different downstream density estimators, we found that using ROME with f_{kNN} instead of f_{KDE} leads to over-fitting (highest D_{JS} values in Table 4.5). Meanwhile, ROME with f_{GMM} tends to over-smooth the estimated density in cases where the underlying distribution is not Gaussian (high $\widehat{W}$ in Table 4.4). The over-smoothing caused by f_{GMM} is further visualised in Figure 4.7.

In conclusion, our ablation studies confirmed that using f_{KDE} in combination with data clustering, normalization and decorrelation provides the most reliable density estimation for different types of distributions.

4.6 Conclusion

In our comparison against two established and sophisticated density estimators, we observed that ROME achieved consistently good results across all tests, while manifold Parzen windows (MPW) and vine copulas (VC) were susceptible to over-fitting and over-smoothing. For example, while MPW is superior at capturing non-normal distributions, it produces kernels with too small of a bandwidth (hence the over-fitting), which is likely caused by the selected number of nearest neighbors used for the localized kernel estimation being too small. Meanwhile, compared to VC, ROME is numerically more stable and does not hallucinate new modes in the estimated densities (Figure 4.5). Furthermore, as part of several ablation studies, we found that ROME overcomes the shortcomings of other common density estimators, such as the over-fitting exhibited by kNN or the over-smoothing by GMM. In those studies, we additionally demonstrated that our approach

of using clustering, decorrelation, and normalization is indispensable for overcoming the deficiencies of KDE.

Future work can further improve on our results by investigating the integration of more sophisticated density estimation methods, such as MPW, instead of the kernel density estimator in our proposed approach to enable better performance on non-normal clusters.

Overall, by providing a simple way to accurately estimate distributions based on samples, ROME can help in better handling and evaluating probabilistic data as well as enabling more precise probabilistic inference.

5

Realistic Adversarial Attacks for Robustness Evaluation of Trajectory Prediction Models

Another important aspect of the evaluation framework introduced in Chapter 3 is its ability to test trajectory prediction models against meaningful adversarial attacks. Evaluating the robustness of prediction models against such adversarial attacks is essential to ensure their reliability in real-world traffic. However, previously presented approaches tend to focus on perturbing the past positions of the adversarial agent, which can generate unrealistic scenarios. In this chapter, we demonstrate that perturbing not just past but also future states of adversarial agents can provide a more rigorous evaluation of model robustness, by making the comparison of model prediction against the recorded ground truth more meaningful. Specifically, we try to preserves the observed tactical behaviors, while ensuring that attacks are dynamically feasible. Testing our method on state-of-the-art prediction models revealed significant increases in prediction errors and collision rates under adversarial conditions. Qualitative analysis further showed that our attacks can expose critical weaknesses, such as the inability of the model to detect potential collisions in what appear to be safe predictions.

This chapter is based on *J. F. Schumann, J. Hagenus, F. B. Mathiesen, A. Zgonnikov. "Realistic Adversarial Attacks for Robustness Evaluation of Trajectory Prediction Models via Future State Perturbation". ACM Journal on Autonomous Transportation Systems (2026)* [140]

5.1 Introduction

Trajectory prediction – the task of forecasting the future positions of traffic participants over time – is a critical component of autonomous vehicle (AV) systems, allowing them to adapt to the actions of nearby road users (Figure 5.1A). As AVs become more prevalent, ensuring the robustness and reliability of the prediction models is essential for public safety [84, 85]. The use of adversarial trajectories – specifically generated to deceive the model prediction as much as possible – has emerged as a method for evaluating robustness of trajectory prediction models, helping to uncover vulnerabilities that could lead to dangerous situations on the road, particularly in the presence of adversarial agents [71, 86–88, 138, 175]. The ability of a model to withstand such attacks is known as adversarial robustness [176].

Recent advances in adversarial attack generation have shown promise in exposing system vulnerabilities [71, 86, 138, 177, 178]. Existing attack generation methods have traditionally focused on perturbing the observed position of the target vehicle. These methods can create scenarios that challenge trajectory prediction models, leading to prediction errors or simulated collisions (Section 5.2). State-of-the-art approaches demonstrate two key capabilities. First, they generate trajectories that are dynamically feasible, ensuring that attacks can actually be reproduced in a real-world setting [71]. Second, the generated trajectories adhere to traffic norms, ensuring that resulting attacks cannot be trivially detected [71, 86], and ensuring that evaluations of prediction models on the perturbed data are meaningful. However, a key limitation of existing methods is their exclusive focus on perturbing *observed* (past) states of the target vehicle, disregarding the effect of perturbations on the vehicle's future states. This means that existing attack methods can result in adversarial trajectories that restrict the agent to a different *tactical behavior* (maneuver) than the original trajectory, potentially negating the real-life effect of the attack.

As an example, consider an intersection scenario where an adversarial agent intends to turn left, but aims to deceive the ego vehicle (AV across the intersection) into believing that it would *not* turn left (Figure 5.1B). If the future states of the adversarial agent are not taken into account when generating the attack (as is the case for existing methods), the perturbation of the past trajectory might put the agent in the state from which the left turn is kinematically infeasible. As a result, the adversarial prediction would be consistent with the actual behavior of the agent, meaning that the attack would not actually deceive the ego agent (Figure 5.1B).

In summary, disregarding future states when generating trajectory perturbations limits the ability of existing attack methods to represent adversarial behaviors that AVs could encounter in the real world. This in turn limits the effectiveness of adversarial testing of trajectory prediction models. The challenge thus lies in generating adversarial trajectories that not only deceive trajectory prediction models, but also remain consistent with the target agent's original tactical behavior throughout the entire trajectory, *i.e.*, for both past and future states.

In this paper, we present a novel method for generating adversarial attacks that accounts for both observed and future states. Our approach improves on existing techniques by introducing constraints on future states, ensuring that the entire adversarial trajectory maintains realistic driving behavior (Figure 5.1C). This allows us to create more subtle and potentially dangerous attack scenarios, such as slight modifications to the path of a

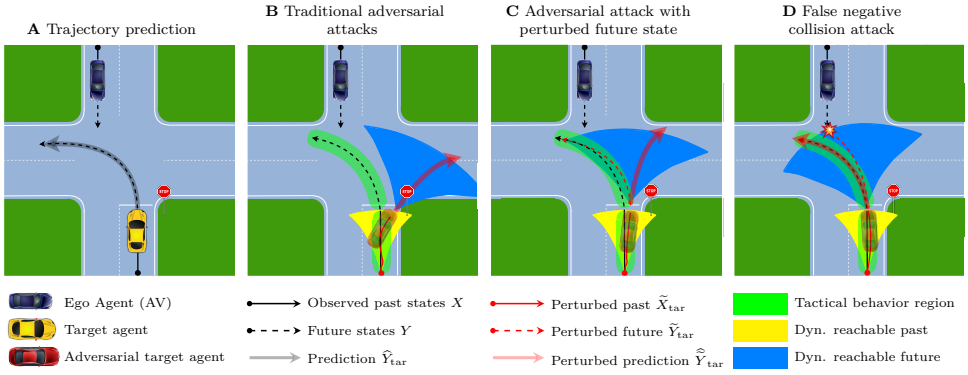


Figure 5.1: Scenarios of adversarial attacks by the target agent on trajectory prediction model of the ego agent (AV). **A**: In the baseline, non-adversarial scenario, based on the observed past trajectory $X = \{X_{\text{ego}}, X_{\text{tar}}\}$ of the target agent, the ego agent makes a prediction $\widehat{Y}$ that aims to closely capture the (unknown) future states Y. **B**: Adversarial (target) agent aims to attack the trajectory prediction model of the AV with the goal for that model to produce an erroneous prediction of the target agent. Existing approaches in the literature typically generate adversarial attacks by perturbing past states of the target vehicle $\widetilde{X}$ in such a way that the corresponding prediction $\widehat{\widetilde{Y}}$ deviates as much as possible from the ground-truth future states Y. However, existing methods do not guarantee that these future states Y actually lie in the reachable set of the target agent after the perturbation, which can result in an overall unsuccessful attack where $\widehat{\widetilde{Y}}$ is actually consistent with the actual future of the adversarial agent. **C**: In our proposed approach, we generate (and constrain) perturbations not only for the past states $\widetilde{X}$ but also future states $\widetilde{Y}$ of the target agent. This ensures that the observed future Y stays dynamically feasible, which ensures that prediction errors resulting from the perturbation are meaningful. **D**: Our approach allows us to generate a novel type of adversarial attacks: *false negative collision attack*, where the adversarial agent takes a future trajectory that collides with the AV but deceives the AV into predicting a non-colliding trajectory. In **B** - **D**, the yellow area represent the dynamically reachable set starting from the first observation, while the blue region represents the same starting from the prediction point. The green region represents a simple distance constraint often employed in adversarial attacks in the literature.

vehicle that lead to unexpected collisions while appearing benign to the prediction model (Figure 5.1D). By considering future states during attack generation, we ensure that adversarial trajectories remain plausible and reflective of real-world driving scenarios, leading to a more comprehensive and realistic evaluation of the robustness of trajectory prediction models. The contributions of this paper can be summarized as follows:

- We demonstrate that current methods of generating adversarial trajectories are dynamically infeasible and thus cannot be implemented in real vehicles, and, consequently, have limited value in improving the robustness of trajectory prediction models.

- We introduce a new method to generate dynamically feasible adversarial attacks that are consistent with the observed tactical behavior, allowing for a more meaningful evaluation of the robustness of a prediction model. Building on the approach by Cao *et al.* [86] to perturb not the trajectory in itself, but rather extracted control inputs of a dynamical model, we extend this approach to cases where only position data is available, while considering the agents ability to reverse. Importantly, we

increase the meaningfulness of such attacks by constraining the planned perturbed trajectory as well.

- We propose a new type of adversarial attack, the *false negative collision attack*. This goes beyond that standard approach of maximizing the prediction error, and instead aims to cause a collision by fooling the ego agent into discarding such a possibility until it is too late.
- We present novel measures to assess the effectiveness and realism of adversarial attacks on trajectory prediction models, based on the distance between ground truth and perturbed past trajectory as well as the amplitude of the control actions required to execute the perturbed trajectory.

5.2 Related work

5.2.1 Adversarial attack generation methods

Existing methods of adversarial attack generation for trajectory prediction models almost exclusively generate adversarial scenarios by perturbing real traffic scenarios [71, 86, 138, 177, 179] (however, see [180] for an approach using generative models to sample adversarial scenarios or [181] for backdoor attacks on a models training data). Perturbation-based attack generation methods aim to find the "worst" possible perturbation for a given scenario and trajectory prediction model. They can be divided into black-box attacks, which only use the output of the model, and white-box attacks, which use gradient information to find perturbations.

Black-box attacks like particle swarm optimization [71], do not require model gradient information but can be slow and unreliable [86]. In contrast, white-box attacks, using some form of gradient descent [71, 86, 138, 177], are generally effective but require the adversarial agent to have access to the model gradient. In cases where access to gradients is restricted, for example due to intellectual property concerns, this challenge can be addressed by using surrogate models and generating transferable attacks [182]. While our work only focuses on utilizing a white-box approach, all those methods could be used to pursue varying attack objectives.

5.2.2 Adversarial attack objectives

Creating adversarial attacks involves specifying the attack objective for robustness evaluation, with different lines of attack being proposed in the literature.

- **Average or final displacement error (ADE/FDE) attack**: Maximize either the average or final distance between the observed ground truth and the prediction made by the behavior model based on the perturbed past [71, 71, 86, 87, 175, 183].
- **Lateral/longitudinal attack**. Maximize the displacement projected either along the trajectory (longitudinal) or perpendicular to it (lateral), to falsely suggest behaviors such as large accelerations or lane changes [71, 139, 175, 183].
- **Targeted attack**. Minimize the distance between the prediction and a user-defined target trajectory [138]. While this method allows for manipulation of the output of

the model, the range of possible outcomes is often more constrained compared to other attack methods.

- **Collision attack**. Minimize the distance between the prediction and the ground truth trajectory of the ego vehicle, to deceive the model into predicting a collision [87, 139, 179].

5.2.3 Constraints used during adversarial attacks

As simple attack objectives like ADE/FDE could be maximized by using very large (and therefore unrealistic and easily detectable) perturbations, previous work generally adopted the approach of limiting the perturbation magnitude to maintain the dynamic feasibility of the attacks. Two approaches to constraining perturbation magnitude have been used in the literature:

- **Constraining vehicle positions**: Zhang *et al.* [71] use a method referred to as a "Search attack" to constrain trajectories by rescaling an initial perturbation so that velocity, accelerations and jerk lie within fixed bounds based upon the spread of those properties in the original dataset. Yin *et al.* [177] enhanced this approach by substituting the data statistics method with a continuous curvature model called an "SA-attack" using a pure pursuit method (a path-tracking algorithm that generates a trajectory by continuously pursuing a point on the path at a fixed look-ahead distance from the current position [184]). Namely, a dynamic model is fit between initial perturbations to generate trajectories that ensure dynamic feasibility at specific speeds. As an alternative approach, Cao *et al.* [86] did not use hard constraints, but instead added a penalty to the attack objective for overly large perturbations.
- **Constraining vehicle control actions**: Cao *et al.* [86] proposed generating adversarial trajectories by perturbing control actions (with hard constraints) derived from the agent's spatial position, heading angle, and velocity using a kinematic bicycle model. This ensures both that the trajectories adhere to the physical constraints associated with a specific vehicle and that the perturbed control actions remain consistent with those of the unperturbed trajectory. Meanwhile, Dong *et al.* [139] added a regularization term to the attack objective to minimize curvature and its derivative in the perturbed trajectories.

Fusing these two approaches, Fan *et al.* [178] used the discriminator of a generative adversarial network (GAN) to keep the perturbed trajectories similar to the unperturbed one, after which the control inputs of bicycle model were optimized to create a kinematically feasible trajectory matching the generated one as closely as possible.

Our proposed method could generate attacks by perturbing both spatial positions [71] or control actions [86], however, we will focus on the more novel, latter approach in the following sections.

5.3 Background

5.3.1 Trajectory prediction

The goal of trajectory prediction is to forecast the future movements of a traffic agent i based on their past observed trajectory $X_i = \{\boldsymbol{x}_i(t) \mid t \in \{-H+1, \ldots, 0\}\}$, that contains H

positions with a time step size Δt. Similarly, we can also collect the future trajectory of the agent $Y_i = \{\boldsymbol{y}_i(t) \mid t \in \{1, \dots, T\}\}$ with the prediction horizon T. In one scene, $X = \{X_i, \dots, X_N\}$ is the combined history for all N agents in the current scene (similarly, Y is the combined future).

A probabilistic model $P_{\boldsymbol{\theta}}$ with trainable parameters $\boldsymbol{\theta}$ then generates K equally likely trajectories $\widehat{Y}^k \sim P_{\boldsymbol{\theta}}(\cdot|X)$ with $\widehat{Y} = \{\widehat{Y}^k \mid k \in \{1, \dots, K\}\}$ from the same initial state X. Generally, such models are trained to minimize the distance of those trajectories to the ground truth observation Y, or maximize the likelihood $P_{\boldsymbol{\theta}}(Y|X)$ of Y.

While there are models that actually predict the trajectories of all agents simultaneously [34, 75–77], most models predict agents independently, that is, they have the form $P(Y_i \mid X_i, X \setminus X_i)$.

5.3.2 Adversarial attacks

In the case of an adversarial attack, an adversarial agent, also referred to as a target agent with trajectory X_{tar}, attempts to deceive the prediction model employed by the autonomous vehicle (ego agent X_{ego}) into making incorrect predictions (Figure 5.1B).

5

As discussed in Section 5.2, most existing attack approaches generate adversarial trajectories by modifying an existing non-adversarial trajectory. Specifically, the observed (past) states of the target agent are modified using perturbations δ_X, resulting in the perturbed past trajectory $\widetilde{X}_{\text{tar}} = X_{\text{tar}} + \delta_X$. The attacked model then makes the (generally probabilistic) predictions $\widehat{\widetilde{Y}}_{\text{tar}} \sim P_\theta(\cdot|\widetilde{X}_{\text{tar}}, X_{\text{ego}})$, which are then used to evaluate the robustness of the models.

5.3.3 Projected gradient descent

To generate adversarial attacks using a white-box attack strategy, we use Projected Gradient Descent (PGD). PGD is an extension of traditional gradient descent methods for constrained optimization problems, iteratively refining the perturbations using alternating gradient and projection steps. Given an initial perturbation $\delta^0 = 0$, we perform two steps in each iteration $m \in \{1, 2, hdots, M_{\max}\}$. First, a gradient step based on the loss function $\mathscr{L}$ utilizes a step size α to update the perturbation δ^{m-1}, producing an intermediate perturbation $\hat{\delta}^{m-1}$:

$$\hat{\delta}^{m-1} = \delta^{m-1} - \alpha^{m-1} \nabla_\delta \mathscr{L} \tag{5.1}$$

Second, a projection step finds an updated perturbation δ^m by projecting $\hat{\delta}^{m-1}$ onto the feasible set C (Equation 5.2).

$$\delta^m = \underset{v \in C}{\operatorname{argmin}} \left\| v - \hat{\delta}^{m-1} \right\| \tag{5.2}$$

5.3.4 Adversarial attack objectives

In this paper, we assume that the loss function $\mathscr{L}$ of the attacked model can be a sum of multiple objective or constraint functions l. The following attack objectives l are common in the literature:

- **ADE/FDE attack** tries to maximize the average or final distance between ground truth Y and adversarial predictions $\widehat{\widetilde{Y}}$ [71, 86, 175, 183].

$$
\begin{aligned}
l_{\text{ADE}}(Y_{\text{tar}}, \widehat{\widetilde{Y}}_{\text{tar}}) &= -\frac{1}{KT}\sum_{k=1}^{K}\sum_{t=1}^{T}\left\|\hat{\tilde{\boldsymbol{y}}}^{k}_{\text{tar}}(t) - \boldsymbol{y}_{\text{tar}}(t)\right\| \\
l_{\text{FDE}}(Y_{\text{tar}}, \widehat{\widetilde{Y}}_{\text{tar}}) &= -\frac{1}{K}\sum_{k=1}^{K}\left\|\hat{\tilde{\boldsymbol{y}}}^{k}_{\text{tar}}(T) - \boldsymbol{y}_{\text{tar}}(T)\right\|
\end{aligned}
\tag{5.3}
$$

- **Collision attack** aims to mislead the ego agent into predicting a collision (*i.e.*, a false positive collision prediction) by minimizing the distance between the predicted positions of the target agent $\widehat{\widetilde{Y}}_{\text{tar}}$ and the ground truth future positions of the ego agent Y_{ego} [87, 179]:

$$
l_{\text{Col}}(Y_{\text{ego}}, \widehat{\widetilde{Y}}_{\text{tar}}) = \frac{1}{K}\sum_{k=1}^{K}\min_{t\in\{1,\dots,T\}}\left\|\hat{\tilde{\boldsymbol{y}}}^{k}_{\text{tar}}(t) - \boldsymbol{y}_{\text{ego}}(t)\right\| \tag{5.4}
$$

5

5.4 Method

5.4.1 Perturbation strategy

To improve on the realism of the traditional approach of perturbing the past states of the target vehicle X_{tar}, in our method we also perturb the future states Y_{tar} with perturbation δ_Y: $\widetilde{Y}_{\text{tar}} = Y_{\text{tar}} + \delta_Y$. Because the resulting perturbed trajectories $\widetilde{X}_{\text{tar}}$ and $\widetilde{Y}_{\text{tar}}$ need to be similar to the original trajectory, perturbation magnitudes are constrained to $\delta_X \in C_X$ and $\delta_Y \in C_Y$. Furthermore, to ensure dynamic feasibility of the perturbed trajectories, we follow Cao *et al.* [86] and perturb not the recorded positions X_{tar} (which may result in trajectories that cannot be executed by real vehicles), but instead the control inputs $\boldsymbol{u} = \{a, \kappa\}$ underlying this trajectory. Here, control inputs include longitudinal acceleration a and curvature κ.

5.4.2 Dynamic model

To allow us to perturb control actions, we must first be able to extract these control actions from a given trajectory $P = \{\boldsymbol{p}(t) \mid t \in \{T_0, \dots, T_1\}\}$ (with positions $\boldsymbol{p}(t) = \{x(t), y(t)\}$), and transform the perturbed control actions into the perturbed trajectory. To this end, we use the dynamic unicycle model Φ, previously utilized by [86]. This dynamic model allows us to advance the state $\boldsymbol{s} = \{x(t), y(t), \theta(t), v(t), \mathscr{D}(t)\}$ with heading angle θ and longitudinal velocity v:

$$
\boldsymbol{s}(t+1) = \Phi(\boldsymbol{s}(t), \boldsymbol{u}(t))\,. \tag{5.5}
$$

Specifically, we get

$$
\begin{aligned}
v(t+1) &= v(t) + a(t)\Delta t \\
\theta(t+1) &= \theta(t) + v(t)\kappa(t)\Delta t \\
x(t+1) &= x(t) + v(t+1)\cos(\theta(t+1))\Delta t \\
y(t+1) &= y(t) + v(t+1)\sin(\theta(t+1))\Delta t\,.
\end{aligned}
\tag{5.6}
$$

The dynamic model can also be inverted to give us the control actions, with $\boldsymbol{u}(t) = \Phi^{-1}(\boldsymbol{s}(t+1), \boldsymbol{s}(t))$, with

$$\begin{aligned} a(t) &= \frac{v(t+1) - v(t)}{\Delta t} \\ \kappa(t) &= \frac{\theta(t+1) - \theta(t)}{v(t)\Delta t}. \end{aligned} \tag{5.7}$$

Many existing vehicle datasets only provide position data but not control input data. Consequently, to be able to use our dynamic model, we need to make the assumption that in the given unperturbed trajectory, intermediate states like velocities and heading angles are constant over the first two timestep (while this might not be true, there is no previous data to contradict this assumption). This allows us then to extract the initial state using Equation (5.6), with $v_x(t) = \frac{x(t+1)-x(t)}{\Delta t}$ and $v_y(t) = \frac{y(t+1)-y(t)}{\Delta t}$:

$$\begin{aligned} v(T_0) &= \sqrt{v_x(T_0)^2 + v_y(T_0)^2} \\ \theta(T_0) &= \operatorname{atan2}\left(v_y(T_0), v_x(T_0)\right) \end{aligned} \tag{5.8}$$

Additionally, in the definition of Φ^{-1} (Equation (5.7)), we cannot use $v(t+1)$ and $\theta(t+1)$, as (starting from $t = T_0$) we only have access to $v(t)$ and $\theta(t)$, requiring us to first calculate those (again using Equation (5.6)):

$$\begin{aligned} v(t+1) &= \mathscr{D}_t \sqrt{v_x(t)^2 + v_y(t)^2} \\ \theta(t+1) &= \operatorname{atan2}\left(\mathscr{D}_t v_y(t), \mathscr{D}_t v_x(t)\right). \end{aligned} \tag{5.9}$$

Here, $\mathscr{D}_t$ describes the current direction of travel (*i.e.*, if the agent is reversing or driving forward), with

$$\mathscr{D}_t = \begin{cases} \operatorname{sgn}(v(t)) & \left|\operatorname{atan2}\left(v_y(t), v_x(t)\right) - \theta(t)\right| \leq \frac{\pi}{2} \\ -\operatorname{sgn}(v(t)) & \text{otherwise} \end{cases}. \tag{5.10}$$

This ensures that reversing the direction of travel is correctly extracted as a (sign) change in velocity brought about by realistic accelerations or decelerations and not as a turn that would require an impossibly high curvature.

5.4.3 Generating dynamically feasible attacks

With the above dynamic model, we can generate dynamically feasible attacks in an iterative procedure (Figure 5.2).

1. In the first iteration ($m = 0$), we extract the control actions $U_{\text{tar}} = \{\boldsymbol{u}_{\text{tar}}(t) \mid t \in \{-H+1, \dots, -1\}$ and $V_{\text{tar}} = \{\boldsymbol{u}_{\text{tar}}(t) \mid t \in \{0, \dots, T-1\}$ from the unperturbed trajectories X_{tar} and Y_{tar} respectively. After initializing the added states (Equation (5.8)), we can use alternately the inverse model Φ^{-1} (equations (5.7) and (5.9)) to get the current control actions, before using the forward model Φ (Equation (5.6)) to the get the next values of the added states (Figure 5.2.1). We also initialize our control action perturbations $\delta^0_{U_{\text{tar}}} = 0$ and $\delta^0_{V_{\text{tar}}} = 0$.

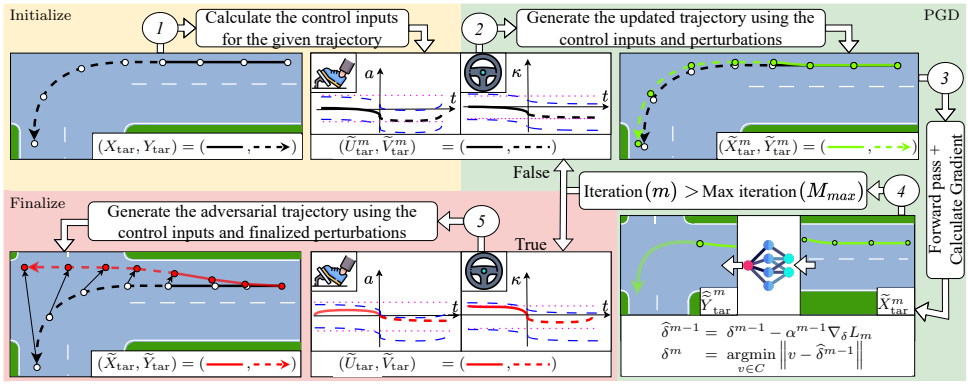


Figure 5.2: Generating adversarial trajectories with constraining future states using control actions. 1) During initialization, we extract the control actions ($U_{\text{tar}} = \widetilde{U}^0_{\text{tar}}$, $V_{\text{tar}} = \widetilde{V}^0_{\text{tar}}$) from the unperturbed trajectory parts X_{tar} and Y_{tar} respectively. 2) Afterwards, we iteratively perturb those control actions, where, starting with the given perturbed control actions $\widetilde{U}^m_{\text{tar}}$ and $\widetilde{V}^m_{\text{tar}}$, we get the corresponding perturbed trajectories $\widetilde{X}^m_{\text{tar}}$ and $\widetilde{Y}^m_{\text{tar}}$. 3) Based thereon – together with the prediction $\widehat{\widetilde{Y}}^m_{\text{tar}}$ from updated states $\widetilde{X}^m_{\text{tar}}$ – we can calculate the adversarial loss L_m and its derivative, with which the perturbation can be updated. The constraints C in this update are represented by the purple dotted line for the absolute control limits and the blue dashed line for the relative bounds. 4) This procedure is repeated for M_{max} iterations. 5) In the end, the final perturbed trajectories are extracted.

2. We get the perturbed control states $\widetilde{U}^m_{\text{tar}} = U_{\text{tar}} + \delta^m_{U_{\text{tar}}}$ and $\widetilde{V}^m_{\text{tar}} = V_{\text{tar}} + \delta^m_{V_{\text{tar}}}$. These can then be used to compute the corresponding perturbed trajectories $\widetilde{X}^m_{\text{tar}}$ and $\widetilde{Y}^m_{\text{tar}}$, using the full initial state $\mathbf{s}_{\text{tar}}(-H+1)$ (Equation (5.8)) and the given control inputs together with the dynamic model Φ (Figure 5.2.2).

3. We use the prediction model to generate the corresponding predictions $\widehat{\widetilde{Y}}^m_{\text{tar}}$ based on the perturbed input. Given an adversarial loss function $L^m = \mathscr{L}\left(X, Y, \widetilde{X}^m_{\text{tar}}, \widetilde{Y}^m_{\text{tar}}, \widehat{\widetilde{Y}}^m_{\text{tar}}\right)$, we can get the derivative of the loss in regard to the perturbation $\delta^m_{U_{\text{tar}}}$ and $\delta^m_{V_{\text{tar}}}$ using differentiable programming. Specifically, for $\delta_{U_{\text{tar}}}$, we can determine that

$$\frac{dL}{d\delta_{U_{\text{tar}}}} = \frac{dL}{d\widetilde{U}_{\text{tar}}}\frac{d\widetilde{U}_{\text{tar}}}{d\delta_{U_{\text{tar}}}} = \frac{dL}{d\widetilde{X}_{\text{tar}}}\frac{d\widetilde{X}_{\text{tar}}}{d\widetilde{U}_{\text{tar}}} = \left(\frac{\partial L}{\partial \widetilde{X}_{\text{tar}}} + \frac{\partial L}{\partial \widetilde{Y}_{\text{tar}}}\frac{d\widetilde{Y}_{\text{tar}}}{d\widetilde{X}_{\text{tar}}} + \frac{\partial L}{\partial \widehat{\widetilde{Y}}_{\text{tar}}}\frac{d\widehat{\widetilde{Y}}_{\text{tar}}}{d\widetilde{X}_{\text{tar}}}\right)\frac{d\widetilde{X}_{\text{tar}}}{d\widetilde{U}_{\text{tar}}}. \tag{5.11}$$

Here, with $\frac{d\widetilde{U}_{\text{tar}}}{d\delta_{U_{\text{tar}}}} = 1$ dropping out, the calculation of the derivatives $\frac{\partial L}{\partial \cdot}$ depends on the definition of $\mathscr{L}$, the calculation of $\frac{d\widetilde{Y}_{\text{tar}}}{d\widetilde{X}_{\text{tar}}}$ and $\frac{d\widetilde{X}_{\text{tar}}}{d\widetilde{U}_{\text{tar}}}$ on the dynamical model Φ, and the calculation of $\frac{d\widehat{\widetilde{Y}}_{\text{tar}}}{d\widetilde{X}_{\text{tar}}}$ on the prediction model P. Similarly, we can do the same for $d\delta_{V_{\text{tar}}}$:

$$\frac{dL}{d\delta_{V_{\text{tar}}}} = \frac{dL}{d\widetilde{V}_{\text{tar}}}\frac{d\widetilde{V}_{\text{tar}}}{d\delta_{V_{\text{tar}}}} = \frac{\partial L}{\partial \widetilde{Y}_{\text{tar}}}\frac{d\widetilde{Y}_{\text{tar}}}{d\widetilde{V}_{\text{tar}}}. \tag{5.12}$$

Again, as $\frac{d\widetilde{V}_{\text{tar}}}{d\delta_{V_{\text{tar}}}} = 1$, the gradients in the last line depend on the loss function $\mathscr{L}$ and dynamic model Φ. Therefore, with careful design of $\mathscr{L}$, Φ, and P, we can compute the required gradients, allowing us to use projected gradient descent to update the perturbations $\delta^m_{U_{\text{tar}}}$ and $\delta^m_{V_{\text{tar}}}$ (Figure 5.2.3).

In regard to the constraints C_U and C_V (Equation (5.2)), we choose the following values to restrict the perturbations of acceleration $a(t)$ and curvature $\kappa(t)$

$$\begin{aligned}
\delta_a(t) &\geq \max\{-2\,\text{ms}^{-2}, a_{\min} - a(t)\} \\
\delta_a(t) &\leq \min\{2\,\text{ms}^{-2}, a_{\max} - a(t)\} \\
\delta_\kappa(t) &\geq \max\{-0.05\,\text{m}^{-1}, -0.2\,\text{m}^{-1} - \kappa(t)\} \\
\delta_\kappa(t) &\leq \min\{0.05\,\text{m}^{-1}, 0.2\,\text{m}^{-1} - \kappa(t)\}
\end{aligned} \tag{5.13}$$

This approach combines both absolute bounds (*i.e.*, bounds on the resulting perturbed control inputs $\tilde{a}(t)$ and $\tilde{\kappa}(t)$) as well as relative bounds (directly on perturbations $\delta_a(t)$ and $\delta_\kappa(t)$, taken from [86]). For the absolute bounds $a_{\min}$ and $a_{\max}$, we use the lowest and highest acceleration values observed in the dataset.

4. If $m > M_{\max}$, we complete the attack, having arrived at our final perturbed trajectories $\widetilde{X} = \widetilde{X}^{M_{\max}}$ and $\widetilde{Y} = \widetilde{Y}^{M_{\max}}$ (Figure 5.2.5).

5.4.4 Preserving tactical behavior

As discussed in Section 5.1, in order for the adversarial attack to be effective, perturbed trajectories need to provide for the same tactical behavior as the ground truth, *i.e.*, the should be close to the original observations. However, our choice of perturbing control actions instead of positions makes it impossible to implement this via the constraints C_U and C_V (Equation (5.13)). Therefore, we chose to instead implement such position constraints by adding a regularizing term l to the adversarial loss $\mathscr{L}$. Here, the regularizing term l constrains a whole trajectory by combining costs l_p that constrain each individual perturbed position $\tilde{\boldsymbol{p}}(t)$ of the original trajectory $P = \{\boldsymbol{p}(t) \mid t \in \{T_0, \dots, T_1\}\}$.

Regularizing individual points in the perturbed trajectory

Inspired by negative log-barrier constraint encoding used in interior point methods [185], we use the following regularization term:

$$l_{p,\text{Method}}\left(\tilde{\boldsymbol{p}}(t), P\right) = -\ln\left(d_{\max} - d_{\text{Method}}(\tilde{\boldsymbol{p}}(t), P)\right) \tag{5.14}$$

Importantly, $l_{p,\text{Method}}\left(\tilde{\boldsymbol{p}}(t), P\right)$ grows to infinity as $d_{\text{Method}}(\tilde{\boldsymbol{p}}(t), P)$ approaches $d_{\max}$. Therefore, with sufficiently small steps, the constraint $d_{\text{Method}}(\tilde{\boldsymbol{p}}(t), P) \leq d_{\max}$ will never be violated. Here, the subscript *Method* denotes one of the two options for the choice of the distance function:

- **Time-specific**: The perturbed position is bounded by the corresponding unperturbed position:

$$d_{\text{Time}}(\tilde{\boldsymbol{p}}(t), P) = \|\tilde{\boldsymbol{p}}(t) - \boldsymbol{p}(t)\| \tag{5.15}$$

- **Trajectory-specific**: The perturbed trajectory is bounded in its shortest distance to the whole ground truth trajectory; only lateral perturbations are penalized.

$$d_{\text{Traj}}(\tilde{\boldsymbol{p}}(t), P) = \min_{t'} d_{\text{proj}}(\tilde{\boldsymbol{p}}(t), \boldsymbol{p}(t'), \boldsymbol{p}(t'+1)) \tag{5.16}$$

Here d_{proj} is the closest distance of point onto a line segment spanned by two other points:

$$d_{\text{proj}}(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) = \begin{cases} \|\boldsymbol{a} - \boldsymbol{c}\| & r(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \leq 0 \\ d_{\perp}(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) & 0 < r(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) < 1 \\ \|\boldsymbol{a} - \boldsymbol{b}\| & \text{otherwise.} \end{cases}$$

with

$$d_{\perp}(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) = \frac{\|(\boldsymbol{a} - \boldsymbol{c}) \times (\boldsymbol{b} - \boldsymbol{c})\|}{\|(\boldsymbol{b} - \boldsymbol{c})\|}$$

$$r(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) = \frac{(\boldsymbol{a} - \boldsymbol{c}) \cdot (\boldsymbol{b} - \boldsymbol{c})}{\|(\boldsymbol{b} - \boldsymbol{c})\|^2}$$

Regularizing perturbed trajectories

Based on different combinations of the point-wise constraints discussed above, we propose three different approaches for regularizing full trajectories P (Figure 5.3):

- **Time-specific**: only use the time-specific constraints (Equation (5.15)):

$$l_{\text{Time}}(\widetilde{P}, P) = \frac{1}{|\widetilde{P}|} \sum_{\tilde{\boldsymbol{p}}(t) \in \widetilde{P}} l_{p,\text{Time}}(\tilde{\boldsymbol{p}}(t), P) \tag{5.17}$$

- **Trajectory-specific**: only use the trajectory-specific constraints (Equation (5.16)):

$$l_{\text{Traj}}(\widetilde{P}, P) = \frac{1}{|\widetilde{P}|} \sum_{\tilde{\boldsymbol{p}}(t) \in \widetilde{P}} l_{p,\text{Traj}}(\tilde{\boldsymbol{p}}(t), P) \tag{5.18}$$

- **Time-trajectory-specific**: While the trajectory-specific approach allows for more flexibility in longitudinal deviations, thereby enabling the exploration of a broader range of realistic adversarial scenarios, this also poses a risk. Namely, perturbations may lead to substantial deviations from the ground truth states at critical moments, resulting in unrealistic or unsafe behaviors not aligned with the natural flow of the trajectory. For example, when making a left turn (Figure 5.1A), it is crucial that, at prediction time $t = 0$, the target agent can still feasibly complete the turn without risking a collision with the ego agent. Therefore we introduce a combined regularization function

$$l_{\text{Time-traj}}(\widetilde{P}, P) = l_{\text{Traj}}(\widetilde{P}, P) + l_{p,\text{Time}}(\tilde{\boldsymbol{p}}(T_1), P)\,. \tag{5.19}$$

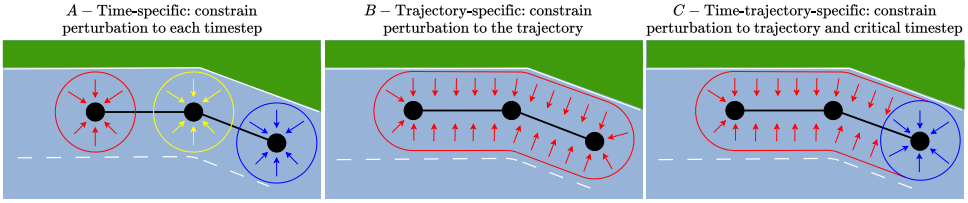


Figure 5.3: Three regularization approaches for preserving tactical behavior. The black line represents the ground truth trajectory, with the colored arrows showing the direction in which the regularization penalties are applied. In A, the different colors represent that the different penalties are applied to different timesteps.

5.4.5 Adversarial attack objectives

For our new attack generation approach, we implemented the attack objectives common in the literature: ADE/FDE and collision rate (Section 5.3.4). Furthermore, we also propose a novel attack objective: *false negative collision*. In contrast to the traditional (*false positive*) collision attack (which tries to deceive the AV into predicting a collision), our novel attack tries to conceal an intended collision from the AV. Compared to the false positive collision loss in Equation (5.4), the attack objective here is

$$l_{\text{FNC}}(Y_{\text{ego}}, \widetilde{Y}_{\text{tar}}, \widehat{\widetilde{Y}}_{\text{tar}}) = \min_{t\in\{1,\dots,T\}} \left\| \tilde{y}^k_{\text{tar}}(t) - y_{\text{ego}}(t) \right\| + \frac{1}{T}\sum_{t=1}^{T} \left\| \frac{1}{K}\sum_{k=1}^{K} \hat{\tilde{y}}^k_{\text{tar}}(t) - \hat{y}^k_{\text{tar}}(t) \right\| \tag{5.20}$$

The first term here is designed to create a perturbed future that seeks a collision, while the second term tries to create a perturbation for which the attacked model's predictions are similar to the prediction resulting from the ground truth (*i.e.*, the prediction model should not be able to differentiate between them). The approach of deceiving the prediction model into making specific predictions was originally proposed by Tan *et al.* [138]. The value of this novel attack is that it presents a far more dangerous scenario for an AV in actual traffic than the traditional collision attack. Namely, in the false positive collision case, the reaction of the AV would be an unnecessary braking maneuver, while a successful false negative collision attack can be far more severe, especially if the vehicles are close enough that there is not enough time left for a safe evasion upon the AV recognizing the attack.

5.5 Experimental setting

We design a number of experiments to test different attack profiles, using an evaluation Framework designed by Schumann *et al.* [124].[1]

5.5.1 Trajectory prediction model

For our primary experiments, we applied our attack generation approach to one of the most popular and arguably state-of-the-art trajectory prediction models: *Trajectron++* [30].

[1]The code for running these experiments and the resulting data presented here can be found online at `https://github.com/DAI-Lab-HERALD/General-Framework-Adversarial-Attack/tree/main`.

Trajectron++, predicting not positions directly, but instead control states of a unicycle model, has the advantage of always producing dynamically feasible predictions, and thus should produce predictions most robust to our dynamically feasible adversarial attacks. We set the model to generate $K = 100$ predictions for each model input, *i.e.*, in each scene.

Additionally, to show that our method is not uniquely suited to just this one model, we also use *ADAPT* [76], a recent state-of-the-art trajectory prediction model. Compared to *Trajectron++*, it predicts positions directly, further emphasizing the versatility of the attacks.

5.5.2 Dataset

As a source of unperturbed trajectories for our primary experiments, we chose L-GAP [46], a urban driving dataset obtained in a driving simulator focusing on an intersection scenario. There, sixteen human drivers (yellow vehicle in Figure 5.1A) executed a combined 1839 valid turns (*e.g.*, no collision), choosing to either go in front of or behind an oncoming AV (blue vehicle), which was programmed to always go straight with constant velocities. We selected the left-turning vehicle as the target agent for which we perturb the trajectory. The oncoming vehicle approaching the intersection was designated as the ego vehicle, which would make a prediction about the behavior of the target agent. Because of noisy measurement of vehicle coordinates in the original dataset, we used the Savitzky-Golay filter [186] to smooth the trajectories.

However, this focus on left turns may introduce a data bias (the actions are either left turn or waiting), a prediction model trained on this dataset alone would have difficulties properly handling different manoeuvres which might arise from adversarial attacks. To address this issue, we trained *Trajectron++* simultaneously on both L-GAP and NuScenes [142], with the latter being larger and including a far more varied range of scenarios. However, adversarial attacks were only generated using trajectories from L-GAP to facilitate interpretation of evaluation results.

In this work, we used an observation period $H = 12$ time steps, and a prediction horizon $T = 12$ time steps, with a time interval of $\Delta t = 0.1s$. The prediction time was set so that the computer-controlled ego agent was $1.2s$ away from the intersection, to ensure that the prediction of the human (adversarial vehicle) would be significant to the ego agent's own path planning (*i.e.*, a wrong prediction would make the ego agent vulnerable to a potentially unavoidable collision). To ensure that the predictions are meaningful, we remove those scenarios where the target agent already initiated the turn (we have 53 scenarios where the agent turned and 971 scenarios where the agent chose to wait remaining).

To test the generalizability of our proposed attack, we also used a second dataset to first train and then attack and evaluate models on, specifically *rounD* [119]. This dataset covers real world roundabouts in Germany. There, we assumed that the ego vehicle was the vehicle inside the roundabout (*i.e.*, similar to the previous dataset, it has priority), while the predicted and perturbed agent was the one entering the round about. Similar to *L-GAP*, this vehicle can either go first or second. We extracted the evaluation scenarios from this datasets with the same parameters H, T, and Δt (see [16] for a discussion on the equivalence of those two scenarios). In this dataset, we evaluated on 174 scenarios where the target agent chose to enter the roundabout first and 2809 scenarios where it chose to wait fro the ego agent to pass.

5.5.3 Comparison to established attack approaches

We compared our method of perturbing control actions with constraints on future states to two alternative approaches commonly used in the literature [84]. In the first approach, one will simply perturb positions, without any constraints regarding dynamic feasibility. The second approach is the *search attack* proposed by Zhang *et al.* [71], which also perturbs spatial positions but limits the perturbation using data statistics. Unlike our approach, both of these approaches impose constraints only on past states.

For each of the three attack strategies (positions; positions with data statistics [71]; actions with future state constraints (ours)), we tested all possible combinations of four attack objectives (ADE; FDE; false positive collision; false negative collision) and two constraint types (time-specific; time-trajectory-specific). This resulted in eight adversarial attack loss functions for each attack approach.

Our approach includes perturbing not only observed (past) but also future states; for perturbing the latter, we used the trajectory-specific displacement constraints from Equation (5.18). The reason for using only this constraint type is that the future perturbed trajectories $\widetilde{Y}_{\text{tar}}$ should mostly preserve the actually chosen maneuver, for which this looser constraint is enough.

Finally, to highlight the added value of perturbing control actions with trajectory-specific constraints on future states, we compared our approach to its ablated version attacking control actions without constraining future states.

5.5.4 Adversarial attack parameters

For all loss functions, we set the maximum displacement to $d_{\text{Max}} = 0.9\,\text{m}$. This choice is based on a typical lane width of 3.5 m and vehicle width of 1.7 m, so that even after perturbations, a vehicle should not leave its lane.

To generate attacks with projected gradient descent, we used an initial learning rate $\alpha^0 = 0.01$ (Equation (5.1)), with a exponential decay of $\gamma = 0.99$ (*i.e.*, $\alpha^m = \gamma\alpha^{m-1}$) for $M_{\text{max}} = 100$ iterations. Crucially, α can be adjusted if the perturbed observed and future states fall outside the tactical behavior constraint region (*i.e.*, in Equation (5.14), we see $d(\tilde{\boldsymbol{p}}(t), P) > d_{\text{max}}$). In such cases, we repeatedly halved α until this condition was met. This learning rate scaling was applied only within each iteration, so that the following iteration would still use the same learning rate as if no adjustment was necessary on the previous iteration.

5.5.5 Performance measures

We employed eight performance measures to evaluate our proposed attack approach as well as established methods. We used measure values *averaged* across all the samples in the dataset.

Attack efficiency

We adopted three measures that are widely used in the literature [84]:

- Average displacement error:

$$\text{ADE} = \frac{1}{KT}\sum_{k=1}^{K}\sum_{t=1}^{T}\left\|\hat{\tilde{\boldsymbol{y}}}^{k}_{\text{tar}}(t) - \boldsymbol{y}_{\text{tar}}(t)\right\| \tag{5.21}$$

- Final displacement error:

$$\text{FDE} = \frac{1}{K}\sum_{k=1}^{K}\left\|\hat{\hat{\boldsymbol{y}}}^{k}_{\text{tar}}(T) - \boldsymbol{y}_{\text{tar}}(T)\right\| \tag{5.22}$$

- Collision rate:

$$\text{CR}_{\text{T}} = \frac{1}{K}\sum_{k=1}^{K}\max_{t\in\{1,\dots,T\}} f_{\text{coll}}\left(\hat{\hat{y}}^{k}_{\text{tar}}(t), y_{\text{ego}}(t)\right) \tag{5.23}$$

with

$$f_{\text{coll}}(\boldsymbol{a},\boldsymbol{b}) = \begin{cases} 1 & \text{BB}(\boldsymbol{a}) \cap \text{BB}(\boldsymbol{b}) \neq \emptyset \\ 0 & \text{otherwise} \end{cases}, \tag{5.24}$$

where BB(·) represents the bounding box of a vehicle.

To accompany our novel *false negative collision* attack, we propose a new measure

- Rate of collisions along the perturbed path:

$$\text{CR}_{\text{F}} = \max_{t\in\{1,\dots,T\}} f_{\text{coll}}\left(\tilde{y}_{\text{tar}}(t), y_{\text{ego}}(t)\right). \tag{5.25}$$

Perturbation magnitude
We used two measures to measure the extent of the produced perturbations, with larger values corresponding to more noticeable attacks:

- Maximum displacement:

$$D_{\text{max}} = \max_{\{t\in -H+1,\dots,0\}} \|\widetilde{\boldsymbol{x}}_{\text{tar}}(t) - \boldsymbol{x}_{\text{tar}}(t)\| \tag{5.26}$$

- Mean displacement:

$$D_{\text{mean}} = \frac{1}{H}\sum_{t=-H+1}^{0} \|\widetilde{\boldsymbol{x}}_{\text{tar}}(t) - \boldsymbol{x}_{\text{tar}}(t)\| \tag{5.27}$$

Dynamic feasibility
We used two measures to evaluate the dynamic feasibility of the perturbed past observations $\widetilde{X}_{\text{tar}}$ using the perturbed control inputs in $\widetilde{U}_{\text{tar}}$:

- Acceleration magnitude:

$$a = \frac{1}{H-1}\sum_{t=-H+1}^{-1} |\tilde{a}(t)| \tag{5.28}$$

- Curvature magnitude:

$$\kappa = \frac{1}{H-1}\sum_{t=-H+1}^{-1} |\tilde{\kappa}(t)| \tag{5.29}$$

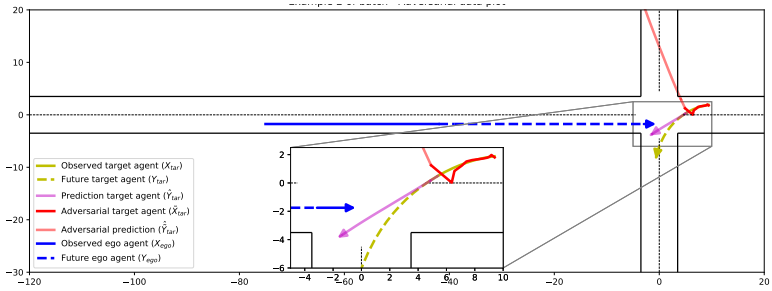


Figure 5.4: Perturbing *positions* with attack loss $\mathscr{L} = l_{\text{ADE}}(Y_{\text{tar}}, \hat{\widetilde{Y}}_{\text{tar}}) + l_{\text{Time}}(\widetilde{X}_{\text{tar}}, X_{\text{tar}})$ - ADE attack without dynamic constraints creates an unrealistic trajectory, with large deviations in the final states of the observed states.

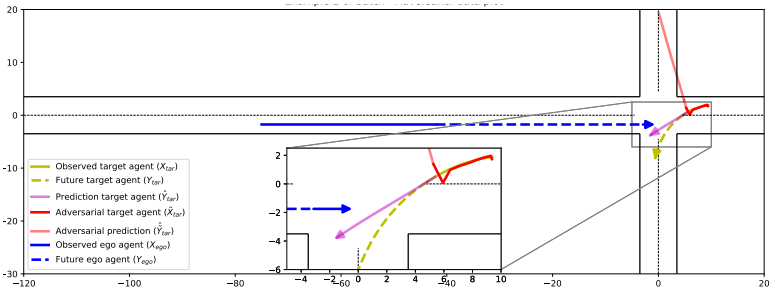


Figure 5.5: Search attack with attack loss $\mathscr{L} = l_{\text{ADE}}(Y_{\text{tar}}, \hat{\widetilde{Y}}_{\text{tar}}) + l_{\text{Time}}(\widetilde{X}_{\text{tar}}, X_{\text{tar}})$ - ADE attack using "Search" dynamic constraints creates an unrealistic trajectory, with large deviations in the final states of the observed states.

5.6 Results

We evaluated attack efficiency, the magnitude of generated perturbations, and dynamic feasibility of adversarial trajectories for three attack approaches: position-based attack (Table 5.1), search attack perturbing positions using data statistics-based constraints (Table 5.2), and our approach of perturbing control action with constraints on future states (Table 5.3).

5.6.1 Position-based attacks produce dynamically impossible trajectories

We found that both the position attack and the search attack yielded significantly higher ADE and FDE values compared to our approach. For example, ADE attack with time-specific position constraints resulted in the ADE value of 4.045 m for the position-based attack, 3.073 m for the search attack, and only 1.030 m/0.250 m (without/with constraints on future positions) for an attack on control actions, compared to 0.115 m that the model achieves on unperturbed inputs.

However, the high efficiency of position-based attacks was achieved by producing trajectories that are dynamically impossible. For both established methods (Tables 5.1 and 5.2), the *average* acceleration magnitude a exceeds $30\,\text{ms}^{-2} \approx 3g$ for all attack losses with average trajectory curvature of at least $20\,\text{m}^{-1}$ (requiring a turn radius lower than 0.05 m).

Table 5.1: Evaluation of *Trajectron++* on *L-GAP* against perturbed positions (without using constraints on future behavior). ADE, FDE, and perturbation magnitudes D are given in m, while acceleration a is given in ms^{-2} and curvature κ in m^{-1}.

Attack loss	Position constraints (observed)	Attack efficiency ADE	FDE	CR_T	CR_F	Perturbation magnitude D_{max}	D_{mean}	Dynamic feasibility a	κ
Unperturbed	-	0.043	0.100	0.635	-	-	-	0.812	0.433
ADE	Time	4.045	3.447	0.007	-	0.770	0.195	53.407	20.860
	Time-traj	3.447	6.203	0.011	-	0.681	0.168	47.066	25.222
FDE	Time	3.983	7.158	0.008	-	0.766	0.186	51.868	23.368
	Time-traj	4.035	7.262	0.008	-	0.846	0.197	56.280	22.338
False Positive	Time	1.371	2.495	0.020	-	0.467	0.119	36.695	27.427
Collision	Time-traj	1.389	2.530	0.020	-	0.495	0.125	39.004	29.138
False Negative	Time	0.562	1.020	0.019	0.736	0.423	0.116	36.427	25.177
Collision	Time-traj	0.598	1.085	0.020	0.741	0.467	0.126	40.322	25.739

Trajectories with such extreme acceleration and curvature values cannot be reproduced in real vehicles (*e.g.*, as shown in Figure 5.4 and 5.5) and hence do not provide a valid means to assess adversarial robustness of trajectory prediction models.

5.6.2 Constraining future states generates efficient and realistic control action-based attacks

The above results demonstrate that the established attacks achieve high efficiency but are by far not realistic. In contrast, our approach of perturbing control actions (instead of positions directly) produces actions that are similar in magnitude to the unperturbed data (Table 5.3), and therefore can be implemented in a real vehicle. Although the impact of the attacks generated by this approach is much lower than for established approaches (as indicated by lower attack efficiency measures), these attacks still lead to substantially decreased accuracy of the attacked model. For instance, the ADE and FDE values achieved by attacks targeting these measures are still substantially higher than for the predictions based on unperturbed trajectories. Importantly, false positive collision attacks for our approach generate similar (compared to position-only constraints) or higher (compared to the search attack) predicted collision rates.

Further investigating the approach of perturbing control actions, we analyzed the effect of using future position constraints on attack performance (Table 5.3). We found that imposing constraints on future positions (in addition to constraining observed positions) results in perturbed trajectories which are far more similar to the ground truth. For instance, we see an decrease in acceleration magnitude of around 40% for ADE and FDE attacks, and 10% for a false positive collision attack.

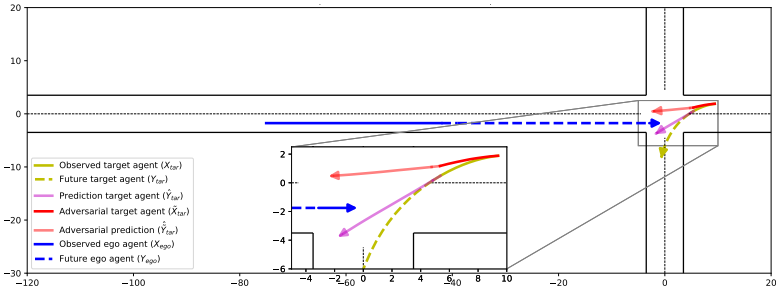


Figure 5.6: Perturbing *control actions* with attack loss $\mathscr{L} = l_{\text{ADE}}(Y_{\text{tar}}, \hat{\hat{Y}}_{\text{tar}}) + l_{\text{Time}}(\tilde{X}_{\text{tar}}, X_{\text{tar}})$ - ADE attack with control action dynamic constraints creates a realistic trajectory, with a gradual increase in deviation from the ground truth.

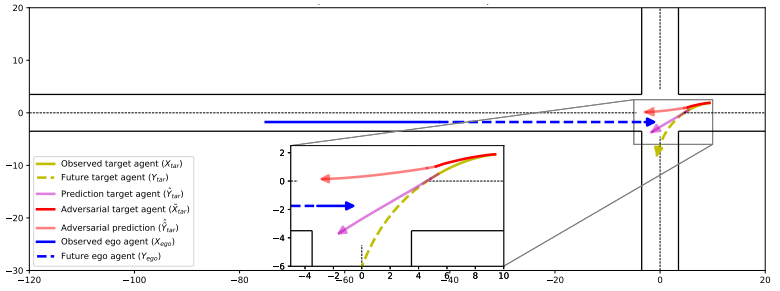


Figure 5.7: Perturbing *control actions* with attack loss $\mathscr{L} = l_{\text{FDE}}(Y_{\text{tar}}, \hat{\hat{Y}}_{\text{tar}}) + l_{\text{Time}}(\tilde{X}_{\text{tar}}, X_{\text{tar}})$ - FDE attack with control action dynamic constraints creates a realistic trajectory, with a gradual increase in deviation from the ground truth.

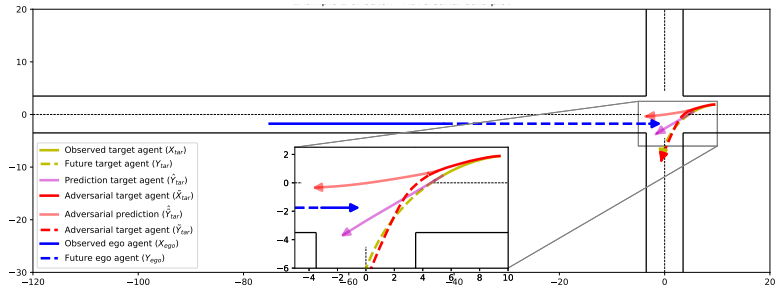


Figure 5.8: Perturbing *control actions* with attack loss $\mathscr{L} = l_{\text{ADE}}(Y_{\text{tar}}, \hat{\hat{Y}}_{\text{tar}}) + l_{\text{Time}}(\tilde{X}_{\text{tar}}, X_{\text{tar}}) + l_{\text{Traj}}(\tilde{Y}_{\text{tar}}, Y_{\text{tar}})$ - Feasible ADE attack with control action dynamic constraints creates a realistic trajectory, with a gradual increase in deviation from the ground truth. The adversarial observed states smoothly transition to the adversarial future states that closely resemble the future ground truth states.

Table 5.2: Evaluation of *Trajectron++* on *L-GAP* against the position-based "search" attack [71] (without using constraints on future behavior). ADE, FDE, and perturbation magnitudes D are given in m, while acceleration a is given in ms^{-2} and curvature κ in m^{-1}.

Attack loss	Position constraints (observed)	Attack efficiency ADE	FDE	CR_T	CR_F	Perturbation magnitude D_{max}	D_{mean}	Dynamic feasibility a	κ
Unperturbed	-	0.27	0.55	0.72	-	-	-	0.81	0.43
ADE	Time	3.07	5.52	0.01	-	0.71	0.18	49.39	20.10
	Time-traj	2.03	3.67	0.01	-	0.54	0.13	38.74	27.57
FDE	Time	2.28	4.12	0.01	-	0.59	0.15	43.67	23.85
	Time-traj	2.31	4.17	0.01	-	0.62	0.15	45.15	23.81
False Positive	Time	1.79	3.24	0.01	-	0.51	0.12	36.49	27.94
Collision	Time-traj	1.82	3.29	0.01	-	0.55	0.13	38.99	25.84
False Negative	Time	0.88	1.59	0.02	0.71	0.45	0.12	38.11	24.99
Collision	Time-traj	0.93	1.69	0.02	0.71	0.48	0.13	40.91	24.47

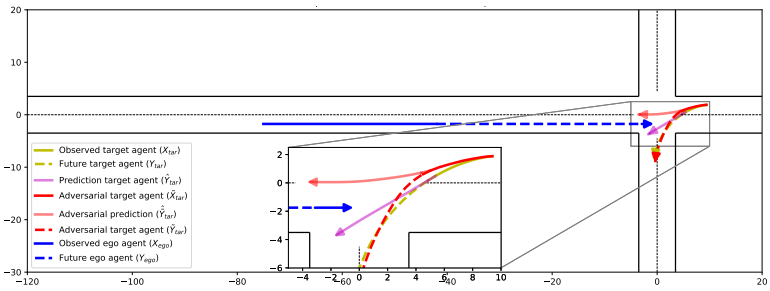


Figure 5.9: Perturbing *control actions* with attack loss $\mathscr{L} = l_{FDE}(Y_{tar}, \widehat{\widetilde{Y}}_{tar}) + l_{Time}(\widetilde{X}_{tar}, X_{tar}) + l_{Traj}(\widetilde{Y}_{tar}, Y_{tar})$ - Feasible FDE attack with control action dynamic constraints creates a realistic trajectory, with a gradual increase in deviation from the ground truth. The adversarial observed states smoothly transition to the adversarial future states that closely resemble the future ground truth states.

Similar effects were observed for the displacement measures. For past-only constraints, trajectories attacking ADE and FDE consistently approached the maximum allowed displacement of 0.9 m (with values of $D_{max} = 0.841$ m). For attacks generated by constraining future states however, the displacement values are consistently and substantially lower, resulting in adversarial trajectories that closely align with original, unperturbed ones. This is likely because adding the constraints on future perturbed trajectories impose further limits on the latter timesteps of the observed trajectory (compare Figures 5.6 and 5.8 as well as Figures 5.7 and 5.9 for illustrative examples). The direct consequence of these smaller perturbation magnitudes however is the reduced efficiency, resulting in lower ADE and

Table 5.3: Evaluation of *Trajectron++* on *L-GAP* against perturbed control actions with trajectory-specific constraints on future states (Ours). Rows with "-" under "Future" display results for perturbing control actions without constraints on future states. ADE, FDE, and perturbation magnitudes D are given in m, while acceleration a is given in ms^{-2} and curvature κ in m^{-1}.

	Position constraints		Attack efficiency				Perturbation magnitude		Dynamic feasibility	
Attack loss	Observed	Future	ADE	FDE	CR_T	CR_F	D_{max}	D_{mean}	a	κ
Unperturbed	-	-	0.12	0.28	0.02	-	-	-	0.51	0.01
ADE	Time	-	1.03	1.60	0.01	-	0.84	0.53	1.66	0.01
	Time	Trajectory	0.25	0.47	0.02	-	0.15	0.05	0.90	0.01
	Time-traj	-	0.74	1.20	0.02	-	0.53	0.15	1.40	0.01
	Time-traj	Trajectory	0.22	0.44	0.02	-	0.11	0.04	0.88	0.01
FDE	Time	-	1.06	1.65	0.02	-	0.86	0.27	1.79	0.01
	Time	Trajectory	0.58	0.94	0.02	-	0.46	0.16	1.11	0.01
	Time-traj	-	0.84	1.36	0.02	-	0.60	0.16	1.64	0.01
	Time-traj	Trajectory	0.50	0.85	0.02	-	0.35	0.11	1.11	0.01
False Positive Collision	Time	-	0.45	0.72	0.02	-	0.36	0.13	1.07	0.01
	Time	Trajectory	0.30	0.50	0.02	-	0.21	0.08	0.92	0.01
	Time-traj	-	0.33	0.57	0.02	-	0.22	0.07	1.01	0.01
	Time-traj	Trajectory	0.26	0.46	0.02	-	0.15	0.05	0.91	0.01
False Negative Collision	Time	-	0.12	0.22	0.02	0.72	0.07	0.03	0.85	0.01
	Time-traj	-	0.11	0.20	0.02	0.72	0.05	0.02	0.86	0.01

FDE values across all attack objectives.

These results highlight a trade-off between attack efficiency (how likely an attack is to deceive a prediction model) and obscurity (how difficult it is to detect the attack). However, surprisingly, we found that attacking the collision rate does not conform to this trade-off. Under the false positive collision attack, constraining future states improved the dynamic feasibility and reduced attack magnitude but resulted in similar CR_T values. Even more surprisingly, for ADE and FDE attacks, collision rates actually increased when constraining future states. The reason for this is that in the left turn scenario, our attack has to deceive the model into predicting a turn that is either sharper or wider than the one pursued originally. With curvature values being consistent with the dataset, it is possible that both options are regularly chosen. For wider turns, those can then be wide enough to avoid the oncoming vehicle, with the target agent staying in its own lane. However, if future perturbations and therefore prediction errors are constrained, achieving this is less likely, which makes predicted collision more likely (refer again to Figures 5.6 and 5.8 for an example).

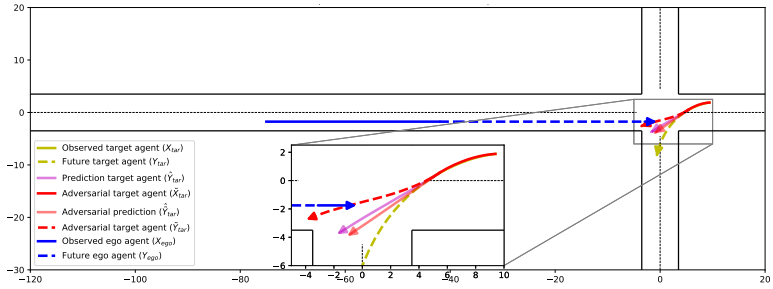


Figure 5.10: Perturbing *control actions* with attack loss $\mathscr{L} = l_{\text{FNC}}(Y_{\text{ego}}, \widetilde{Y}_{\text{tar}}, \widehat{\widetilde{Y}}_{\text{tar}}) + l_{\text{Time}}(\widetilde{X}_{\text{tar}}, X_{\text{tar}})$ The prediction on the adversarial observed states is directed toward the nominal setting prediction. The perturbed future states are directed toward the ego agent, causing a collision.

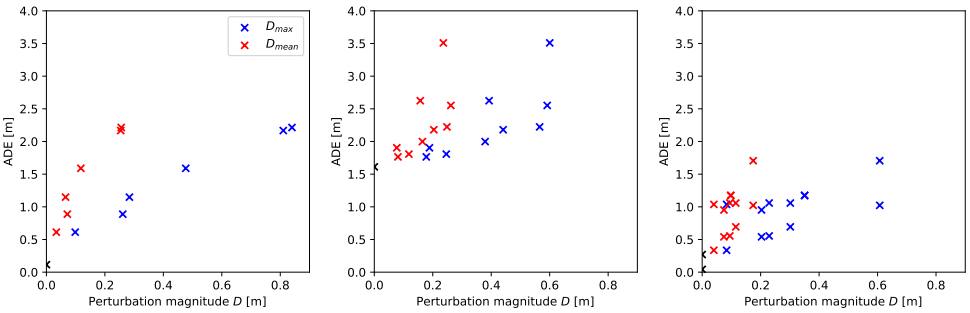
5.6.3 Attacking the false negative collision rate is highly efficient

Our novel false negative collision attack was remarkably successful in deceiving the model into believing that a perturbed trajectory is not on a collision course, while it actually was. Across all three approaches (Tables 5.1, 5.2, and 5.3), at least 70% of generated trajectories were on a collision course while the model believed they were not. Moreover, for the control action-based perturbations (Table 5.3), the adversarial attacks effectively deceived the model into making predictions $\widehat{\widetilde{Y}}$ that roughly resemble the predictions based on the ground truth $\widehat{Y}$. Here, trajectories generated using the false negative collision loss were indistinguishable from the ground truth, with ADE and FDE values similar or lower than on unperturbed inputs, as can be seen, for instance, in Figure 5.10. Importantly, highlighting the danger of this attack, the model was deceived only 1.2 s before a potential collision, leaving little if not no time to correct that assessment.

5.6.4 Regularization approach affects performance of control action-based but not position-based attacks

For the control action-based attacks (Table 5.3), using the time-trajectory specific constraints (Equation (5.19)) generally resulted in lower attack efficiency measures compared to the time-specific approach, while sometimes slightly increasing the collision rates, corresponding to a drop in perturbation magnitude. For example, under the ADE attack with constrained future trajectories (Table 5.3), the ADE measure drops from 0.250 m to 0.222 m and FDE drops from 0.470 m to 0.435 m, while maximum perturbation decreases from 0.150 m to 0.113 m and average acceleration magnitude from 0.901 ms^{-2} to 0.878 ms^{-2}. On first glance, this might seem somewhat counterintuitive, as the use of the full trajectory-based constraint should allow greater freedom. However, this is not unexpected, as we effectively doubled the constraint cost imposed on the last time step here (Equation (5.19)), which, due to the constraints on the control actions for dynamic feasibility also constrains the previous time steps, preventing them from exploiting their increased freedom.

This assumption is supported by the fact that we did not observe similar trends in the position-based attacks (Tables 5.1 and 5.2), and both constraint types could produce larger or smaller perturbations. Here, the attack can exploit these greater freedoms to produce larger perturbations, as the positions at different time steps are not linked to each other



a) L-GAP *b) HighD* *c) RounD*

Figure 5.11: The relationship between the allowed perturbation magnitude and the resulting ADE values on multiple datasets.

via the dynamic model.

5.6.5 Attack performance is consistent with the choice of attack objective

Furthermore, we analyzed the effect of attack objective on performance measures (Table 5.3). Intuitively, one would expect that targeting, *e.g.* ADE would result in largest effect on the ADE measure compared to other attack objectives, yet understanding how other measures (including magnitude and feasibility measures) are affected can provide a better insight into implications of different attacks.

We observed that the FDE loss resulted in larger perturbations than the ADE loss, with the only exception being the mean displacement D_{mean} when not constraining future perturbation and constraining observed ones in a time-specific way. This then also results in higher ADE and FDE values, as well as a higher predicted collision rate CR_{T}. This is somewhat expected, as the gradient under the FDE loss is driven solely by the time step which generally has the largest displacement, and not averaged over previous time steps with lower displacements.

Meanwhile, the picture for the false positive collision attack is somewhat different. There, attack efficiency in terms of ADE and FDE measures is lower than under the FDE attack, but compared to the ADE attack, these measures are only lower if there is no constraint used for the perturbed future trajectory. If such a constraint is used, the false positive collision attack results in larger ADE and FDE measures compared to the ADE attack. The same relationships can be seen in the perturbation magnitude measures.

5.6.6 Influence of perturbation magnitude

To consider the relationship between the displacement magnitude and the impact on model performance, we consider only those attacks that are actually meaningful (*i.e.*, from Table 5.3, we consider those attacks where future trajectories where constrained). For those, the relationship between ADE and the perturbation magnitude (both D_{max} and D_{mean}) seem to generally follow a roughly linear relationship (Figure 5.11). Specifically, the coef-

Table 5.4: Evaluation of *ADAPT* on *highD* with perturbing control actions, with attacks being performed with different displacement thresholds $d_{\max}$. ADE, FDE, $d_{\max}$, and perturbation magnitudes D are given in m, while acceleration a is given in ms^{-2} and curvature κ in m^{-1}.

Attack & constraint (Observed / Future)	$d_{\max}$	Attack efficiency ADE	FDE	CR_T	CR_F	Perturbation magnitude $D_{\max}$	D_{mean}	Dynamic feasibility a	κ
Unperturbed	(0.0)	1.61	3.74	0.87	-	-	-	0.58	0.00
ADE	0.3	1.90	4.29	0.90	-	0.19	0.08	0.58	0.00
(Time-traj / Trajectory)	0.6	2.62	5.56	0.91	-	0.39	0.16	0.62	0.00
	0.9	3.51	7.06	0.90	-	0.60	0.24	0.67	0.00
False Positive Collision	0.3	1.81	3.94	0.90	-	0.25	0.12	0.63	0.00
(Time-traj / Trajectory)	0.6	2.18	4.43	0.90	-	0.44	0.20	0.67	0.00
	0.9	2.55	4.91	0.91	-	0.59	0.26	0.71	0.00
False Negative Collision	0.3	1.76	3.98	0.90	0.954	0.18	0.08	0.58	0.00
(Time-traj / -)	0.6	2.00	4.26	0.91	0.941	0.38	0.17	0.60	0.00
	0.9	2.22	4.51	0.91	0.920	0.57	0.25	0.61	0.00

ficient of determination between perturbation magnitude and ADE are $R^2_{\max} = 0.986$ and $R^2_{\text{mean}} = 0.962$ on the *L-GAP* dataset, respectively. For the *HighD* dataset, meanwhile, we can observe $R^2_{\max} = 0.781$ and $R^2_{\text{mean}} = 0.705$. The linear correlation is less pronounced here than on *L-GAP*, but still there. The main reason is that due to the longer prediction horizon, the overall model performance is worse, and correspondingly, there is more room for variance in the ADE values.

Lastly, on the *RounD* dataset, we observe $R^2_{\max} = 0.777$ and $R^2_{\text{mean}} = 0.766$. While those values are again lower than in the ones from *L-GAP* (mostly because we also had the significant influence of the attacked model), they are still significant. This – together with Section 5.6.7 – strengthens the assumption that the allowed maximum displacement $d_{\max}$ is likely the deciding factor in determining the impact of the adversarial attack on the model performance. However, as increasing this makes the connection to the ground truth future weaker, a certain limit is still of utmost importance.

5.6.7 Attack performance is consistent across scenarios and scales with perturbation threshold

In the Table 5.4, we can see the results for our proposed attack with meaningful constraints against a different model (*ADAPT*) on the *highD* dataset, with varying constraints on the perturbation threshold $d_{\max}$ (Equation (5.14)). As can be seen here, the attacks still work, with the corresponding objectives consistently worsening compared to the unperturbed scenario (*e.g.*, the ADE values under the ADE attack are consistently higher). This confirms that our proposed attack works for different models on different datasets.

Table 5.5: Evaluation of *Trajectron++* on *rounD* with perturbing control actions, with attacks being performed against both *Trajectron++* and *ADAPT*. If the attacked model is the one that is evaluated, it is set in bold. ADE, FDE, and perturbation magnitudes D are given in m, while acceleration a is given in ms^{-2} and curvature κ in m^{-1}.

Attack & constraint (Observed / Future)	Attacked model	Attack efficiency ADE	FDE	CR_T	CR_F	Perturbation magnitude D_{max}	D_{mean}	Dynamic feasibility a	κ
Unperturbed		0.04	0.10	0.64	-	-	-	0.81	0.43
ADE	***T++***	1.06	1.83	0.75	-	0.30	0.12	1.43	0.04
(Time-traj / Trajectory)	*ADAPT*	1.17	1.92	0.75	-	0.35	0.10	1.26	0.04
False Positive Collision	***T++***	1.06	1.84	0.74	-	0.23	0.10	1.46	0.04
(Time-traj / Trajectory)	*ADAPT*	1.02	1.70	0.75	-	0.61	0.18	1.25	0.04
False Negative Collision	***T++***	0.95	1.73	0.74	0.79	0.20	0.07	1.26	0.03
(Time-traj / -)	*ADAPT*	1.04	1.83	0.75	0.79	0.08	0.04	1.05	0.03

Meanwhile, we can clearly see that d_{max} has an important influence on the performance of our attack. Specifically, we can observe a clear trend, where for each attack, a higher perturbation budget allows for increasingly more efficient attacks. This is especially noticeable for the ADE attack, where a linear increase in budget leads to increasingly large degradations in model performance (d_{max} : 0.0 m $\longrightarrow$ 0.3 m increases ADE by 0.29 m, while d_{max} : 0.3 m $\longrightarrow$ 0.6 m results in an addition to the ADE of 0.72 m, with 0.89 m for d_{max} : 0.6 m $\longrightarrow$ 0.7 m). Meanwhile, for the false positive collision attack, it must be noted increasing the perturbation thresholds leads to increasingly marginal increases in CR_{pred}. However, this is likely caused by the fact that the collision rate is already close to its maximum in the unperturbed case.

A much closer linear relationship can meanwhile be seen in the relation ship between the perturbation threshold d_{max} and average maximum perturbation D_{max}, with each of the attack producing high R^2 values: R^2_{ADE} = 0.9998, R^2_{FPC} = 0.994, and R^2_{FNC} = 0.9998. This is an indication that out attack actual tries to use the given displacement budget as much as possible.

5.6.8 Attacking a model trained on the same data is still successful

When now looking at the results on the *rounD* dataset (Tables 5.5 and 5.6), one can see that the when attacking the evaluated model directly, the attacks are effective in drastically reducing the model performance, for both *Trajectron++* and *ADAPT*. Similar to *L-GAP*, the false positive collision attack results in worse ADE and FDE values than the ADE attack. It could be argued that this might be the result of the collision attack pushing the predicted trajectory in one specific direction (*i.e.*, towards the ego vehicle), while the ADE attack just pushes them away from the ground truth future, which if the unperturbed prediction was well aligned, could be in different directions, minimizing the magnitude of the attack

Table 5.6: Evaluation of *ADAPT* on *rounD* with perturbing control actions, with attacks being performed against both *Trajectron++* and *ADAPT*. If the attacked model is the one that is evaluated, it is set in bold. ADE, FDE, and perturbation magnitudes D are given in m, while acceleration a is given in ms^{-2} and curvature κ in m^{-1}.

Attack & constraint (Observed / Future)	Attacked model	Attack efficiency ADE	FDE	CR_T	CR_F	Perturbation magnitude D_{max}	D_{mean}	Dynamic feasibility a	κ
Unperturbed		0.27	0.55	0.72	-	-	-	0.81	0.43
ADE	*T++*	0.69	1.06	0.79	-	0.30	0.12	1.43	0.04
(Time-traj / Trajectory)	***ADAPT***	1.18	1.97	0.80	-	0.35	0.10	1.26	0.04
False Positive Collision	*T++*	0.56	0.88	0.81	-	0.23	0.10	1.46	0.04
(Time-traj / Trajectory)	***ADAPT***	1.71	2.64	0.82	-	0.61	0.18	1.25	0.04
False Negative Collision	*T++*	0.54	0.87	0.81	0.79	0.20	0.07	1.26	0.03
(Time-traj / -)	***ADAPT***	0.34	0.65	0.81	0.79	0.81	0.04	1.05	0.03

(this effect can be seen very pronounced in Section 5.6.6 which shows a strong correlation between the displacement D and measures like ADE).

The important difference is, however, that the false negative collision, while still able to produce collision future perturbations that result in collisions, is much less successful in disguising these intentions, with the predicted collision rate being similar. However, this is expected given the design of the attack, as the predictions are designed to match the unperturbed predictions, which already exhibit a very high collision rate.

Importantly, we can also see that especially *Trajectron++* is still very sensitive against attacks targeted against *ADAPT*, with basically every evaluation measure being in a similar range or even worse than those resulting on a attack against *Trajectron++* itself. This effect is less pronounced on *ADAPT*, where the attacks against *Trajectron++* result in measures which, while still worse than the unperturbed data, are still relatively benign.

5.7 Discussion

Our results demonstrate that perturbing control actions with constraints on future positions enables the generation of realistic, feasible, yet still efficient adversarial trajectories. More generally, when testing trajectory prediction models for adversarial robustness, evaluating them just based on attack efficiency measures such as ADE, FDE, or collision rate is insufficient: attack magnitude and dynamic feasibility need to be considered.

Virtually all existing work on adversarial attacks constrains maximum displacements of the perturbed observed trajectories. However, based on our investigation of dynamic feasibility measures, the typically reported resulting drop in model performance is likely not caused by a failure in the prediction model. Instead, such given perturbed inputs lie far outside not only the training domain, but also the realm of physical feasibility, especially when positions – rather than control actions – are perturbed. Consequently,

such attacks (which, in addition, could easily be detected in the real world by simple filters on control states) do not provide a viable way of evaluating the robustness of prediction models, unless some effective bounds on the control states are added. Therefore, they do not represent a point of reference to compare the effectiveness of our method against.

Along this line, we have also demonstrated the importance of constraining the perturbations along the future trajectory in addition to the past observation. This has a significant influence on the generated trajectories: while resulting in perturbations of the observed trajectory more comparable to the ground truth, it ensures the consistency between the adversarial agent's past and future behavior. Without it, comparing the ground truth future Y with the perturbed predictions $\widehat{\widetilde{Y}}$ would not actually give any meaningful information on the robustness of the prediction model, as the ground truth Y no longer has any connection to the perturbed past $\widetilde{X}$.

One of the key insights of our work is that keeping the adversarial trajectories realistic and dynamically feasible can dramatically reduce attack efficiency in terms of distance-based measures. While our approach is an important step towards finding efficient attack strategies for evaluating adversarial robustness of AVs' trajectory prediction models, future research needs to focus on improving efficiency of dynamically feasible, realistic, and meaningful attacks.

Complementing our contribution of the new attack approach, our novel attack objective, the *false negative attack*, demonstrated that even with a state-of-the-art prediction model, AVs could still be deceived into a false "sense of safety", potentially allowing collision to be caused by adversarial agents. Therefore, more research is required to improve robustness of prediction models towards this kind of attacks.

Meanwhile, the standard, (*i.e.*, *false positive*) collision attack did not lead to significant increases in predicted collision rates compared to the ADE and FDE attacks, especially when we constrained the perturbations of future states. While this is somewhat explained by the specific scenario chosen, comparisons to the unperturbed predictions show that further research into this attack is likely needed to improve its potential without compromising its relation to the ground truth data.

Importantly, significant decreases in model performance were shown to be achieved not by attacking the model itself or a surrogate model trained from queries of the original model [187], but instead by simply attacking a model trained on similar data. This means that keeping a (trained) model proprietary may not be a sufficient defense against such attacks, and more robust methods need to be pursued instead.

Limitations and future work

In our experiment, the search attack [71] did not improve dynamic feasibility of the basic position-based attacks, still requiring extreme control actions inconsistent with dynamical model of an actual vehicle. We believe this was mainly caused by the fact that it only uses absolute bounds on perturbed control actions, and does not impose relative bounds. As absolute bounds are dataset-dependent, the use of the L-GAP dataset limits our conclusions here. Namely, it includes several collisions which are characterized by high accelerations and curvatures that fall outside normal driving patterns. Consequently, both for for the search attack and our approach, it would likely an improvement to use a more dynamical

bound, which could be set to represent some quantile value instead, only increasing for samples where this new bound is surpassed.

Besides this limitation, there are also a number of other improvement and directions that we deem valuable as future work. First, given the observed close correlation between the perturbation magnitude and the performance measures ADE and FDE, it is worthwhile to study more closely the influence of those attack hyperparameters that could influence the perturbation magnitude. In particular, future work should evaluate effects of the maximum allowed displacement $d_{\max}$ (Equation (5.14)), the bounds on control action perturbations δ_a and δ_κ (Equation (5.13)), the length of past observations H, and the number of predicted timesteps. These hyperparameters could also have interdependencies and can affect different models differently. Future research thus needs to systematically investigate their individual and joint effects using a variety of metrics (especially relevant here is predicted collision rate).

Second, given our refined attack, it would be beneficial to study the effectiveness of various responses to this approach, such as fine tuning models on the perturbed input data, or the use of randomized smoothing [188] to increase robustness. Additional, testing it against methods proposed for the detection of such attacks might be beneficial [189].

Third, up to our knowledge, all current research on adversarial attacks on trajectory prediction models is performed in simulation (including the present paper). To truly understand the implications of such attacks on automated vehicles, future research needs to evaluate them in real vehicles. However, there are multiple challenges associated with this, including the lack of publicly available information on prediction models used in real-world AVs, the lack of existing methodology for implementing and evaluating attacks on real vehicles, as well as safety risks related to real-world testing. Still, we believe such research is necessary to improve robustness of real-world AVs.

5.8 Conclusion

In this paper, we introduced a novel strategy for generating dynamically feasible, realistic, and meaningful adversarial attacks on trajectory prediction models. Our approach successfully deceives the attacked state-of-the-art prediction model, resulting in decreased model accuracy while maintaining plausible behavior of the adversarial agent. Our results highlight the propensity of traditional attacks to produce trajectories which would not be executable in the real world or would not allow the adversarial agent to proceed with its initially intended maneuver. Consequently, we argue for the need of a comprehensive discussion in the field about the requirements for adversarial attack needed to make them an integral part of the evaluation of the robustness of prediction model. Our work is only the first step in this direction, and we suggest areas of future research that would be beneficial to the evaluation of robustness of autonomous vehicles.

II

Explaining Human Behavior

As shown in the previous part of this thesis, despite their improvements on clearly defined benchmarks, current data-driven behavior prediction models still suffer from problems with generalizability (Chapter 3), robustness (Chapter 5), and performance in systematically underrepresented [67] safety-critical edge cases (Chapter 4). Additionally, these models often function as black boxes, making it difficult to understand the reasoning behind their predictions and, therefore, the causes of their failures. Consequently, the exploration of more interpretable and generalizable models of human behavior is a growing area of research. One approach that could be used to achieve those goals might be found in knowledge-based models using first principles and cognitive theory [48].

Explanatory and cognitive models of human behavior have a long history in prior research, particularly within cognitive modeling and human factors research. However, early approaches in this area were often relatively simplistic, relying on abstractions such as basic game-theoretic formulations [43] or simple kinematic principles [190]. While these models provided valuable initial insights and proved the viability of the knowledge-based approach, their expressiveness and explanatory power were still limited.

Since then, a large range of increasingly sophisticated approaches has been proposed that aim to ground human behavior models more firmly in cognitive theory. This includes work focusing on evidence accumulation mechanisms to explain decision-making and its timing [46, 47], as well as approaches that incorporate more realistic perceptual representations, for example by modeling human perception in terms of visual angles rather than abstract positional information [48, 191]. These developments represent an important step toward more cognitively plausible models of human behavior.

Despite this progress, many such models remain limited in scope. They are often applied only to narrowly defined scenarios, such as pedestrian street crossing [49, 50] or highway lane changes [51, 52]. In many cases, the structure of the model is explicitly designed around a specific scenario, which makes generalization to other traffic situations difficult or impossible. In addition, some models are restricted to particular types of road users, focusing exclusively on pedestrians [103] or exclusively on vehicles [104, 107]. Other approaches model only isolated aspects of human behavior, such as braking responses in critical situations [53, 192], driver distraction [193], or the takeover behavior of humans in automated vehicles [54, 55]. Consequently, models that provide a more holistic account of human decision-making in traffic are still relatively rare. Additionally, while the knowledge-based nature of those models gives them explanatory power, their evaluation is often not sufficiently thorough (e.g., testing on held-out data) to properly assess their predictive power.

Nevertheless, recent work has begun to address some of these limitations by proposing more general and flexible cognitive modeling frameworks [194–197]. One particularly promising direction for achieving greater generality is the active inference framework (a theory of general behavior in organisms based on minimizing free energy [57]). While this has recently been successfully applied to human driving behavior [198, 199], those applications were limited to simple scenarios such as car-following. Still, active inference, offering a unified account of perception and decision-making, has emerged as an attractive candidate for modeling complex human behavior across a wide range of scenarios.

A particularly interesting case in which models such as those based on active inference and related cognitive frameworks have not yet been evaluated would be highly safety-

critical scenarios such as collision avoidance. Such scenarios are characterized by a high complexity of available avoidance maneuvers and high stakes, and therefore provide a unique testbed for theories and models based on cognitive principles that have so far largely been validated in controlled laboratory settings [200–202]. Additionally, while cognitively inspired models are often claimed to be more robust due to their grounding in human cognition and reasoning [122], these claims remain largely untested. Typically, such models are evaluated on small datasets and within a single scenario [16], and systematic comparisons with deep learning-based models in challenging settings such as safety-critical edge cases are still missing.

Importantly, such knowledge-based models of human behavior in traffic offer a rare domain in which the influence of underlying principles on human behavior can be studied in complex, real-world behavior, providing an opportunity for testing cognitive theories beyond simplified laboratory paradigms. Therefore, the development of such models not only has practical implications for the field of autonomous driving, but can also contribute to the understanding of human cognition in complex, dynamic environments.

Thesis contribution

Consequently, in the second part of the thesis, we present work that aims to address some of the aforementioned limitations in the field of cognitive modelling of human behavior in traffic. In Chapter 6, we start by expanding a cognitively plausible model of interactions between vehicles and pedestrians at street crossings [48], trying to turn it into a general model of human gap acceptance behavior (a more detailed description of those scenarios can be found in Chapter 2). Specifically, we try to expand the underlying mechanisms such as noisy perception and evidence-accumulation-based decision-making to be valid on a larger set of scenarios. We then compare this model in both benign and safety-critical cases of gap acceptance problems from multiple scenarios against a state-of-the-art trajectory prediction model.

Meanwhile, in Chapter 7, we pursue a more ambitious approach. Specifically, we use active inference as a framework to explain human driving behavior, focused on collision avoidance problems. To allow us to reproduce realistic human behavior – especially in regard to key metrics such as response times – we introduce novel concepts such as evidence-accumulation-based delays in policy selection and norm-conditioned behavior prediction of surrounding agents. We evaluate this model in multiple collision avoidance scenarios, using detailed ablation studies to validate the separate cognitive principles included. Importantly, when fitting the model's parameters, we leave some of the scenarios out of this process to also evaluate the model's ability to generalize to unseen situations. Lastly, we also apply a state-of-the-art deep learning-based behavior prediction model to one of those scenarios as a comparison.

Finally, in Chapter 8, we expand this active inference model of human behavior in traffic to model human interactions in conflict (*i.e.*, space-sharing scenarios). Specifically, we explore how the addition of explicit communication and deontic cues such as stop signs or priority rules can influence the effectiveness with which human participants are able to resolve such scenarios. We also investigate the risk introduced by malign agents (using, for example, deceptive communication, *i.e.*, lying about their intentions) in such interactions.

6

Cognitively Inspired Modeling of Human Gap Acceptance Behavior in Traffic

As discussed in the previous introduction, the second part of the thesis explores the potential of cognitive theory to inspire models that can fullfil the need of autonomous vehicles for reliable human behavior prediction. Importantly, such models are theorized to bess less vulnerable to safety-critical edge cases than data-driven approaches. However, such knowledge-based models are commonly developed only for explanatory purposes, with their effectiveness in behavior prediction remaining largely untested. In this chapter, we investigate the usefulness of the Commotions *model – a novel cognitively plausible model incorporating the latest theories of human perception, decision-making, and motor control – for predicting human behavior in gap acceptance scenarios, which entail many important traffic interactions such as lane changes and intersections. Using evaluation methods developed in the first part of the thesis, we show that this model can compete with or even outperform well-established data-driven prediction models across several naturalistic datasets.*

This chapter is based on *J. F. Schumann, A. R. Srinivasan, J. Kober, G. Markkula, A. Zgonnikov. "Using models based on cognitive theory to predict human behavior in traffic: A case study." 2023 IEEE International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2023.* [132]

6.1 Introduction

Automated vehicles have become a major focus of the car industry in recent years due to their potential to revolutionize transportation. The promised benefits of automated vehicles include fewer accidents caused by human errors, increased accessibility of mobility solutions, and more efficient use of time while traveling [8–10]. However, despite significant investments [2], there are still only prototypes of automated vehicles on the street, and they are not yet widely available to the public [12, 13]. One major challenge to the widespread adoption of automated vehicles is ensuring that they are both efficient and safe, traveling in a timely and efficient manner while also maintaining a level of safety that is at least equivalent to human driving [12, 14]. However, many automated vehicles currently focus on ensuring safety, avoiding any action that could potentially lead to an accident. While this approach may reduce the risk of traffic participants being harmed, it misses out on travel efficiency and acceptance, requiring further efforts to make automated vehicles truly useful [12, 13]. One potential solution is to incorporate prediction models to reduce uncertainty about future human behavior and allow for more actions to be classified as safe [15, 43].

Accurate predictions of human behavior are especially critical in scenarios involving gap acceptance [16], which form a significant subset of space-sharing conflicts in traffic, including situations such as crossing an intersection or changing lanes [89]. Many models for predicting human behavior in these scenarios have been developed, including trajectory prediction models [30, 34] and models predicting the binary choice of either accepting or rejecting the gap [47, 49]. However, most of these include few assumptions about human decision-making – using a mainly data driven approach known for being unreliable in safety-critical edge cases [16, 122].

Meanwhile, there is a separate literature of cognitive theory developed to explain human decision-making in traffic [46, 48]. Inclusion of such theory into predictive models might help overcome the unreliability issues of purely data-driven approaches [122]. However, current cognitively plausible models have a number of limitations which hinder their use for behavior prediction. In particular, most such models are limited to a specific scenario [46, 47] and cannot handle complex inputs which prevents their applications to naturalistic datasets. As a result, it is currently unknown if incorporating cognitive theories in behavior prediction models could actually yield any benefits in terms of prediction accuracy and robustness.

This study aims to explore the potential of one possible approach of incorporating cognitive theory into prediction models: the adaption of a specific existing explanatory model [48] to function as a prediction model, using gap acceptance as target scenario type. Adaptation of this model for prediction purposes is non-trivial, and does in itself represent a significant contribution to the field (Section 6.3). Furthermore, we also conduct an ablation study to find the most promising configurations for the model (Section 6.4). Finally, we compare the performance of the resulting configurations of this model to state-of-the-art data-driven prediction models (Section 6.5).

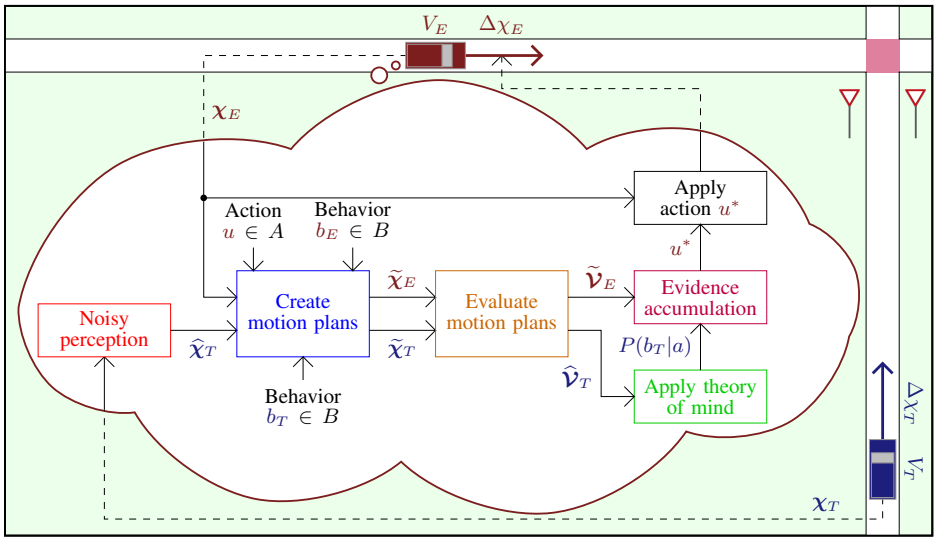


Figure 6.1: A depiction of the *Commotions* model [48] and its high-level parts, showing how the position of ego vehicle V_E is updated at one point in time. Simultaneously, the target vehicle V_T also updates its kinematic state by using the same mechanics – only with mirrored inputs.

6

6.2 Background

This section provides a description of the general type of gap acceptance scenarios addressed here, a brief overview of the tested cognitive model, and an introduction to a framework that facilitates unbiased comparisons of the model's predictive performance against existing benchmarks. We also discuss our changes of the tested model that enable its use as a predictive model.

6.2.1 Gap acceptance

Gap acceptance problems are a type of traffic interaction that involves a space-sharing conflict between two agents with intersecting paths, such as intersections, pedestrian crossings, and lane changes on highways [16, 89]. There, these two agents can be differentiated by the possession of the right of way, with the vehicle with priority being referred to as the ego vehicle V_E. In such a situation, the other agent, designated as the target vehicle V_T, must then decide whether to cross V_E's path in front of V_E (*i.e.*, accepting the offered gap) or to wait until V_E has passed, thereby rejecting the gap. For example, if V_T approaches an intersection via a secondary road, it needs to decide whether the gap to the vehicle coming from the perpendicular direction is large enough to cross the intersection without waiting for that car to pass (Figure 6.1). Accurately predicting V_T's decision in such scenarios is crucial for V_E, as V_T's future behavior could limit V_E's options, such as V_E being forced to slow down to prevent a collision by V_T accepting the gap.

6.2.2 The *Commotions* model

Markkula *et al.* [48] proposed a cognitive framework for modeling road user interactions in gap acceptance scenarios. Their framework includes a wide range of cognitive mechanisms, such as decision-making based on evidence accumulation [47], the use of noisy perception [203] and applying a theory of mind [204] (Figure 6.1). They implemented this framework in models for interactions between vehicles and/or pedestrians on straight crossing paths, *i.e.*, including gap acceptance scenarios between two vehicles. What we will refer to here as the *Commotions* model (after the name of the project in which the model was developed) is the most successful model variant identified in [48], applied to such scenarios.

As illustrated in Figure 6.1, the proposed model postulates that at each time step, both ego vehicle V_E and target vehicle V_T concurrently determine their current control inputs. This decision-making process of each agent is subject to sensory noise and Bayesian filtering during the perception of the position of the other agent. Based on their own short-term control input u (A is a discrete set) and both vehicles' long-term behavior b_E and b_T (*i.e.*, preference for going first or second through the contested space), corresponding pairs of future trajectories – represented by pairs of $\widetilde{\boldsymbol{\chi}}_E$ and $\widetilde{\boldsymbol{\chi}}_T$ – are generated, with the constraint that the resulting interactions are safe. Each pair of trajectories is then evaluated (punishing large control inputs, time delays, and traffic rule violations), resulting in the value $\widetilde{\mathscr{V}}_E$ representing the agent's own opinion and the value $\widehat{\mathscr{V}}_T$, which is the value the agent assumes that the other agent assigns to each trajectory pair for each possible combination of behaviors and control inputs. Each agent then weighs the evaluation $\widetilde{\mathscr{V}}_E$ of their own trajectory based on the probability of the other vehicle behaving accordingly, assuming per the theory of mind that this probability is correlated with the respective value $\widehat{\mathscr{V}}_T$. Evidence accumulation is used to ensure no abrupt and seemingly arbitrary changes in behavior, by combining the weighted values with previous evaluations of a potential action u and only changing the applied control input u^* if this entails a sufficiently substantial improvement in this accumulated value, *i.e.*, control is intermittent. Based on the currently chosen control input u^* each agent's states are then projected forward to the next time step. By repeatedly using this process for both agents, the model can generate a pair of simulated trajectories on the perpendicular intersection. To represent the models randomness, n_p different trajectory pairs are generated in repeated simulations.

6.2.3 The framework for benchmarking gap acceptance models

To compare several prediction models in a fair and unbiased manner, we utilize a framework previously developed by Schumann *et al.* [16]. This framework facilitates the comparison of such models in any gap acceptance scenario according to a wide selection of metrics. Moreover, it grants precise control over the timing of the evaluated predictions and the allocation of individual samples to training and testing sets.

The framework also permits the conversion of different types of predictions, including between binary and trajectory predictions, increasing the number of metrics that can be employed to compare models. For instance, the benchmark enables models that originally predict only gap acceptance probabilities to also generate predictions of full trajectories. Specifically, to transform a predicted probability $a_{\text{pred}} \in [0,1]$ of accepting the gap into a set of predicted trajectories for a given sample, the framework uses two instances of a state-

of-the-art trajectory prediction model [30]. One of these models is trained exclusively on samples with accepted gaps, while the other is trained on samples with rejected gaps. Both models are utilized to predict a set of trajectories based on the given sample's input, from which the final set is sampled with weights adjusted by a_{pred} [16].

6.3 Commotions as general model of gap acceptance behavior

Although the *Commotions* model's capability of expressing a number of empirically observed human interaction phenomena was demonstrated successfully in the original paper [48], it was not developed for use as a prediction model. As such, it has many limitations compared to existing models developed for this purpose. For once, the computational efficiency of its existing implementation makes training and testing on most datasets infeasible. In this paper, we address this problem by implementing parallel processing of multiple model predictions on a GPU and using analytical instead of numerical integration inside the model. This achieves a speed increase of roughly four orders of magnitude.

Another problem with the *Commotions* model is that it is constrained to the scenario of perpendicular intersections with straight trajectories seen in Figure 6.1, which is incongruent to most real world situations. We utilize an expansion of the benchmarking framework 6.2.3 allowing us to project real-world two-dimensional trajectories onto the quasi-one-dimensional-scenario required as the input data. Namely, for each agent, we define a method for determining the most probable path from their current location towards the contested space, where the trajectories of the ego vehicle V_E and the target vehicle V_T intersect. The length of this path is then assumed to be equal to the distances of those agents to the contested space (the purple square in Figure 6.1) along the respective perpendicular streets.

While it might be possible to use the same approach to project predicted trajectories from the quasi-one-dimensional scenario to the original two-dimensional space, they could solely be projected onto the aforementioned predefined most probable paths. As this would drastically limit the solution space, we instead use scenario-independent information from the predicted trajectories. For each pair of trajectories from simulation p we can determine if the contested space was reached first by V_T ($a_{\text{pred},p} = 1$ represents an accepted gap) or V_E ($a_{\text{pred},p} = 0$). Simultaneously, the time $t_{A,\text{pred},p}$ of V_T reaching the contested space can be extracted as well. Averaging over all predictions allows us to calculate the probability $a_{\text{pred}} \in [0, 1]$ of V_T accepting the gap. Combined with the predicted time of acceptance t_A, which the framework accepts as another type of prediction [16], generating predicted trajectories in the original space then becomes possible (6.2.3).

Finally, the *Commotions* model is able to process merely the current position and velocity of only the two principal actors in a gap acceptance scenario as *initial model input*, *i.e.*, V_E and V_T, and not any other agents in the scene. However, as this only hinders but does not prevent the model's predictive usage, this issue remains currently unaddressed.

6.4 Evaluating Commotions' configurations

In this section, we investigate the predictive performance of several configurations of the *Commotions* model stemming from a number of design decisions that have to be made when using the *commotions* model to predict human behavior. For example, the modeling of the interaction between V_E and V_T can utilize either an *interactive* approach, where both agents utilize all aspects of the *Commotions* model (Figure 6.1) to determine their current control inputs u^* (*IM*), or a *non-interactive* approach (*NM*), where only the behavior of V_T is predicted by the model, with V_E set to maintaining its original velocity. Meanwhile, another decision pertains to selecting the form of short-term control inputs u, with the options being the application of either a *constant acceleration* (*AC*) or *constant jerk* (*JC*).

As important parts of the model such as the creation of the trajectories $\boldsymbol{\chi}_E$ and $\boldsymbol{\chi}_T$ (Figure 6.1 and 6.2.2) are non-differentiable, we use Bayesian optimization [205] to fit the *Commotions* model's parameters. However, regarding the optimization procedure, some open questions still remain. First, the user must decide whether to train the model in a *single optimization* round (*1O*) or use a *two-stage optimization* (*2O*), wherein the second stage of optimization is carried out over a reduced parameter search space surrounding the optimized parameters obtained in the first stage. Second, a choice between the two available *loss functions* $\mathscr{L}_1$ and $\mathscr{L}_2$ used to fit the *Commotions* model's parameters must be made. $\mathscr{L}_1$ is adapted directly from the work of Zgonnikov *et al.*[46] (with t_C being the time when V_E reaches the purple intersection in Figure 6.1) and evaluates every prediction p for each sample i, while $\mathscr{L}_2$ expands upon this by enforcing more varied predictions:

$$
\begin{aligned}
\mathscr{L}_1 &= \sum_i \frac{1}{n_p} \sum_{p=1}^{n_p} 4\left|a_i - a_{\text{pred},i,p}\right| + \left(t_i - t_{A,\text{pred},i,p}\right)^2 \\
\mathscr{L}_2 &= \mathscr{L}_1 + \sum_i 100\,V_i - 20\sqrt{V_i} + 1 \\
\mathrm{t_i} &= \min\left\{t_{A,i}, \max\left\{t_{C,i}, t_{A,\text{pred},i,p}\right\}\right\} \\
V_i &= \min\left\{\mathrm{V}_p\left(t_{A,\text{pred},i,p}\right), \frac{1}{100}\right\}
\end{aligned}
\tag{6.1}
$$

6.4.1 Setup

While we only provide a brief overview of the experimental setup here, more details can be found in Appendix A.

Datasets

The predictive performance of the different model configurations is compared using three datasets, each focusing on a different scenario.

- *L-GAP* [46], a driving simulator dataset, contains scenarios in which V_T must decide whether to turn left in front of or behind V_E approaching on the opposite lane.
- *rounD* [119], a real-world dataset captured by a drone, covers roundabouts where V_T must decide whether to enter the roundabout in front of or behind V_E which is already inside the roundabout.

Table 6.1: The number of accepted gaps N_A and rejected gaps $N_{\neg A}$ in the implemented datasets, in the form: $N_A - N_{\neg A}$. The numbers depend on the method for choosing the time of prediction t_0 (Chapter 2.3.1).

Dataset	Gaps at opening	Fixed size gaps	Critical gaps
L-GAP	700/716	375/205	227/397
rounD	789/2164	381/556	147/2810
UDISS	99/942	41/108	0/942

- The *UDISS* dataset [206], created in a driving simulator, focuses on a perpendicular intersection where V_T must cross either in front of or behind V_E, which is driving along the other road with the right of way.

While the latter two datasets include other agents besides V_E and V_T, in this paper we ignore those due to the aforementioned limitations of the *Commotions model*, with the resulting datasets being referred to respectively as $rounD_{2V}$ and $UDISS_{2V}$. We also restrict the provided input trajectories to two input time steps ($n_I = 2$), as this provides sufficient information to extract the two agents' current positions and velocities, which are the only inputs the *Commotions* model is able to process. In Table 6.1, we summarize the sizes of the datasets after filtering, depending on the method for choosing the time of prediction t_0 (Section 6.4.1).

Train/test splits

On each dataset, we perform eleven training-and-testing cycles for each configuration. In ten of these, the split between training and testing set is random. In the last split however, we place the 20 % of samples that exhibit the most unintuitive human behavior – respectively smallest accepted gaps and largest rejected gaps – into the *critical* testing set. This latter approach allows us to evaluate the robustness of the model's predictive capabilities against the most challenging and safety-critical cases.

Metrics

To evaluate the models' predictions made on the testing set, we employ three metrics which have previously been used to assess different aspects of gap acceptance predictions [16, 30, 49]. First, the area under the receiver-operator curve (*AUC*) assesses binary predictions (accept/reject gap) at two different time points: the initial opening of the gaps and the time corresponding to a fixed (dataset-specific) characteristic gap size [16]. Second, the average displacement error (*ADE*) metric evaluates full predicted trajectories at the characteristic gap size. Third, we use the true negative rate under perfect recall [16] (*TNR-PR*), a metric that rates the usefulness of binary predictions made on the smallest possible gaps at the last point in time when they can aid in adjusting V_E's planned path accordingly. However, due to a lack of gaps accepted after this point in time on the *UDISS* dataset, the *TNR-PR* cannot be calculated on that scenario, resulting in eleven viable combinations of metrics and datasets we can use to compare model configurations.

Furthermore, when we transform binary predictions into trajectory predictions, so that for example *ADE* metric can be applied to the *Commotions* model, we use *Trajec-*

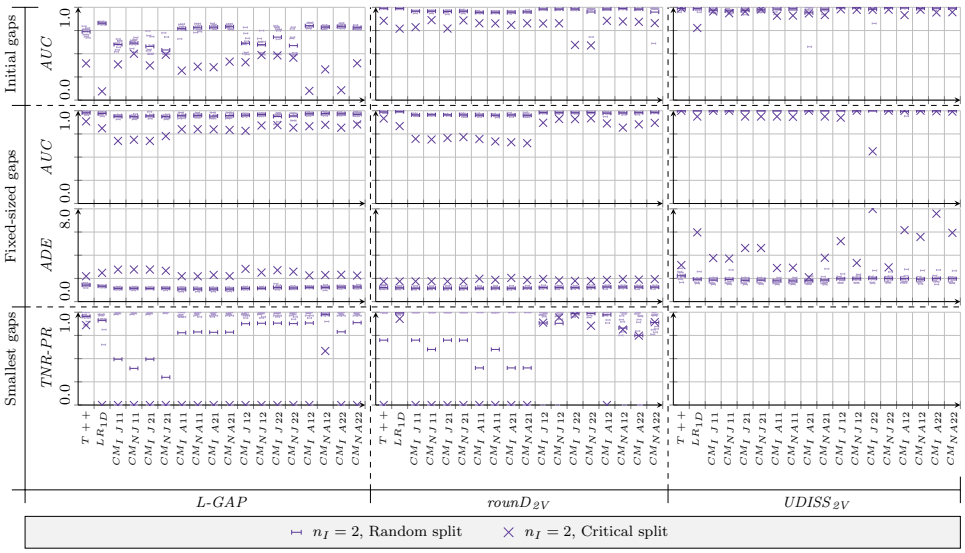


Figure 6.2: Here, the smaller and lighter symbols for random test sets show the results for each individual run (they can be matches between different models by their relative position along the x-axis), while the bigger and darker line markers symbolize the average results over all ten cases with random splitting. Regarding the models, the model CM_{NJ21} would be the *Commotions* model configuration $NM-JC-2O$ trained on loss function $\mathscr{L}_1$, while CM_{IA12} would be the *Commotions* model configuration $IM-AC-1O$ trained on loss function $\mathscr{L}_2$.

tron++ [30], a state-of-the-art trajectory prediction model, in accordance to the method laid out in Section 6.2.3.

6.4.2 Results

Following the setup described above, we test 16 configurations of the *Commotions* model (resulting from four independent design choices) on the eleven combinations of datasets and metrics, resulting in 88 comparisons for each design choice on both random and critical split test sets[1] For example, on the *L-GAP* dataset, the *AUC* averaged over the ten random test sets for predictions made at the fixed-size gap (a size of 3.36 s) ranges from 0.936 to the value 0.970 produced by CM_{NA12}, which utilizes the non-interactive modeling approach (*NM*) and acceleration control (*AC*) and was trained in one round of optimizing (*1O*) $\mathscr{L}_2$.

Comparison between the configurations of the Commotions model (Table 6.2) indicates that there was no consistently better alternative for any of the four design choices. Still, we are able to make some recommendations. For example, the non-interactive modeling approach (*NM*) appears to be more likely to outperform its opposite on the critical test set, while having the added advantage of faster evaluations by obviating half of the *Commotions* model's calculations updating $\boldsymbol{\chi}_E$ (Figure6.1). Similarly, using acceleration control (*AC*) produces better predictions slightly more often, possibly by enabling the model to

[1]The source code and data underlying the reported results can be found at `https://github.com/DAI-Lab-HERALD/Commotions-model-evaluation/tree/main`.

Table 6.2: Assessing the influence of the binary configuration choices in the *Commotions* model (6.4) on the predictive performance, based on the results shown in Figure 6.2. The first number in each cell represents percentage of cases on the randomly split testing sets, whereas the number in the parentheses corresponds to the critical split (*i.e.*, the testing set including the most unintuitive samples). In the last row, results are split by metric.

Binary choice $\mathscr{C}$	$\mathscr{C}_1$ vs $\mathscr{C}_2$	Cases	$\mathscr{C}_1$ better than $\mathscr{C}_2$	$\mathscr{C}_2$ better than $\mathscr{C}_1$
Ego agent modeling	Interactive (*IM*) vs Not interactive (*NM*)	88	7% (7%)	2% (22%)
Control action	Jerk (*JC*) vs Acceleration (*AC*)	88	20% (23%)	26% (25%)
Optimization method	Single round (*1O*) vs Two rounds (*2O*)	88	2% (15%)	6% (11%)
Loss function	$\mathscr{L}_1$ vs $\mathscr{L}_2$	24 (*ADE*) 64 (else)	79% (29%) 0% (11%)	0% (13%) 52% (30%)

Statistical significance of the differences in metrics is tested with a paired Student t-test (significance level α = 0.05). For the critical split, for this purpose, we use the varaince from the random splits.

predict faster human reactions. Although the number of optimization rounds appears to be largely irrelevant, using only one round of optimization (*1O*) may make the model even more robust on the critical test sets, with faster training being another benefit. Comparatively, the most significant factor seems to be the choice of the loss function – as long as one differentiates by metric. Specifically, $\mathscr{L}_1$ is a better choice when minimizing *ADE*, while $\mathscr{L}_2$ is superior on the other three metrics. This is expected, as the regularization achieved by $\mathscr{L}_2$ enforcing some variance in the predictions also leads to a larger spread of predicted trajectories, resulting in a larger average displacement error.

When seeking the best configuration of the *Commotions* model, rather than comparing the binary choices, we can compare the 16 configurations among themselves as well, either by the average result over the ten random test sets or the result on the critical test sets. As model performance mainly depends on the chosen metric, here we discuss *ADE* separately from other metrics. Specifically, we found that the CM_{NA11} configuration is best, having a lower *ADE* in 79% of all the 90 possible comparisons (i.e, on two types of results, three datasets, and against 15 other configurations). Using the same approach on the remaining metrics, we find the most promising configuration to be CM_{NA12} with better metric values in 70% of all cases. These results further support CM_{NA11} and CM_{NA12} (non-interactive modeling, acceleration vehicle input, single-round optimization) as the optimal configurations of the *Commotions* model.

6.5 Comparing Commotions to established models

In this section, we assess the potential of the *Commotions* model by comparing the predictive performance of two of its configurations (CM_{NA11} for *ADE* and CM_{NA12} for other

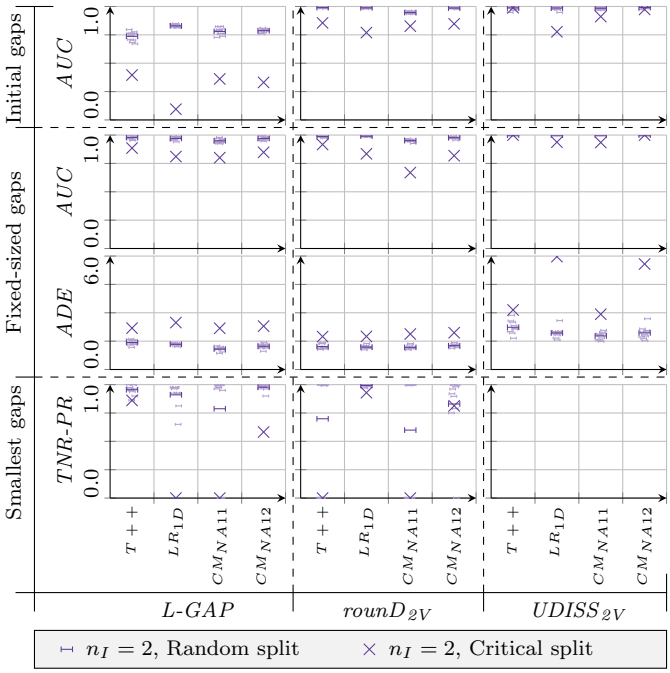


Figure 6.3: Behavior prediction performance of the two *Commotions* model (*CM*) configurations compared to *Trajectron++* (*T++*) and logistic regression (*LR*) across three datasets (L-GAP, rounD, and UDISS) according to considered metrics (AUC, ADE, TNR-PR). Large markers indicate the average over repeated simulations. This is in practice a more readable reproduction of Figure 6.2.

metrics) against established prediction models. Besides the *Trajectron++* model (*T++*) introduced in Section 6.4.1, we also used a logistic regression model (*LR*) as a baseline, with both methods having previously demonstrated good performance on similar gap acceptance problems [16]. While these models have far fewer restrictions on the type of input data they can process, we artificially constrain the used input data to the *Commotions* model's limitations to allow for an equitable comparison. The only exception is the dimensionality of the input for *T++*, as this model can only process the original two-dimensional trajectories, but not the projected quasi-one-dimensional inputs of the *Commotions* model.

Here, the datasets, metrics, as well as testing and training splits are identical to the ablation study (6.4.1).

6.5.1 Results

Comparison of the models (Figure 6.3 and Table 6.3) demonstrates that the *Commotions* model can compete with established models, although variations were observed depending on the metric and dataset. Notably, the *Commotions* model routinely outperforms the other models in terms of *ADE*, consistently on the random test sets and, when compared to *LR*, even on the critical test sets. This may be attributed to the model's capacity to forecast both the probability of accepting a gap and the time at which it may be accepted, with the

Table 6.3: Percentage of cases in which the *Commotions* model performed significantly better or worse compared to the other tested models, based on the results shown in Figure 6.3.

Model M	Cases	*CM* better than M	*CM* worse than M
T++	3 (*ADE*)	100% (0%)	0% (0%)
	2 (Other, *UDISS*)	0% (0%)	0% (0%)
	6 (Other, Other)	17% (0%)	33% (33%)
LR_{1D}	3 (*ADE*)	67% (67%)	0% (0%)
	8 (Other)	13% (75%)	25% (0%)

Notation similar to Table 6.2. The configuration CM_{NA11} is used for *ADE* and CM_{NA12} for the other metrics. The results are split by metric, and for *T++* partially by dataset.

additional information being used to filter out the most aberrant trajectories suggested by the transformation function (6.2.3).

However, on the other metrics, the *Commotions* model's performance is mostly similar to the other two models (no significant difference on 10/16 random and 8/16 critical splits). Nonetheless, it appears to be more robust than *LR* when predicting unintuitive human behavior, with consistently better outcomes on the critical test. This suggests that constraining a model's predictions using cognitive theory to make it less susceptible to out-of-domain edge cases is a viable way to improve the model's reliability.

The *Commotions* model's worst performance can be observed on the *L-GAP* and *rounD* datasets when compared to *T++* on metrics other than *ADE*. While this might indicate a superiority of the *T++* model, this deviation in performance may be at least partly explained by the aforementioned differences in the inputs provided to the models.

To investigate the extent of potential impact of this difference on our results, we compared the second *LR* model taking two-dimensional inputs to the original *LR* model processing the one-dimensional inputs. The results of the comparison (Table 6.4) show that, at least for the *LR* model, processing the two-dimensional original inputs (as *T++* does) appears to simplify the prediction task compared to using the quasi-one-dimensional inputs that the *Commotions* model relies on. This seems plausible, as the projection employed to transform the input data from two-dimensional to one-dimensional likely leads to information loss, leaving fewer cues for the models to make accurate predictions. However, more research is required to accurately assess the impact of input dimensionality on predictions. Thus, a final verdict on the comparative advantage of the *Commotions* model or *T++* is still pending.

6.6 Conclusion

This study evaluates the predictive performance of the different configurations of the *Commotions* model, which integrates state-of-the-art theories of human perception, decision-making, and motor control, in gap acceptance scenarios, comparing the best configurations with other established models. We demonstrate that the *Commotions* model can

Table 6.4: Evaluating the impact of the input dimensionality on the predictive performance of a logistic regression model.

Dataset	Cases	LR_{2D} better than LR_{1D}	LR_{1D} better than LR_{2D}
UDISS	3	0% (33%)	0% (0%)
Other	8	38% (13%)	0% (13%)

Notation similar to Table 6.2

compete with or even outperform state-of-the-art behavior prediction models, as long as the same input information is provided. Notably, the average displacement error of predicted trajectories is most often significantly lower than the one achieved by other tested models.

We also seek to assess the potential impact of the *Commotions* model's restriction to the quasi-one-dimensional scenario of a perpendicular intersection on its performance. Unable to overcome this restriction, we instead compare two versions of the logistic regression model for this investigation. Our findings suggest that allowing *Commotions* model to instead process two-dimensional trajectories as inputs would be beneficial. As an added benefit, this expansion could also enable the model to function as a dedicated trajectory prediction model. Consequently, such an expansion of the *Commotions* model is likely worthwhile, even if it comes at the cost of more expensive computations. In addition, investigating the impact of other limitations, such as the number of processable input time steps, or the number of interacting road users, should be addressed in future research, as it would provide benefits for model designing even beyond the *Commotions* model.

However, due to its theoretical basis, the *Commotions* model will always be restricted to scenarios such as gap acceptance, where a small number of potential behaviors, like accepting or rejecting a gap, make it feasible to create and evaluate all distinct future trajectories $\widetilde{\chi}$. This limits the model's general applicability compared to models like *Trajectron++*. Additionally, as the model itself is non-differentiable, the resulting need for gradient-free optimization makes the model's training process relatively cumbersome, hampering its feasibility further. Nevertheless, our findings provide encouraging evidence supporting the usefulness of the *Commotions* model, at least for predicting human behavior in gap acceptance scenarios, justifying further research into both this specific model and the general approach of integrating cognitive theory into prediction models. For example, it would be worthwhile to investigate how the cognitive assumptions in the *Commotions* model (or other cognitive models) might be leveraged in model architectures that are specifically designed for use in the prediction context.

7

Active Inference as a Model of Human Collision Avoidance Behavior

While the previous chapter showed the promise of knowledge-based models for predicting human behavior, that approach was limited to gap acceptance scenarios only. There exist many safety-critical scenarios, for which such models are required. On such critical example is collision avoidance, where a both the threat detection and and the execution of an appropriate evasive maneuver need to reproduced. However, existing models of human collision avoidance behavior are fragmented, focusing on specific scenarios or only describing certain aspects of the avoidance behavior, such as response times. This chapter addresses these gaps by proposing a novel computational cognitive model of human collision avoidance behavior based on active inference. Active inference provides a unified approach to modeling human behavior: the minimization of free energy. Building on prior active inference work, our model incorporates established cognitive mechanisms such as evidence accumulation to simulate human responses in three distinct collision avoidance scenarios: front-to-rear lead vehicle braking, lateral incursion by an oncoming vehicle, and another vehicle failing to yield at an intersection. We demonstrate that our model explains a wide range of empirical findings on human collision avoidance behavior. Specifically, the model closely reproduces both aggregate results from meta-analyses previously reported in the literature and detailed, scenario-specific effects observed in two recent driving simulator studies, including response timing, maneuver selection, and execution. This suggests the potential of active inference as a generalizable framework for understanding and modeling human behavior in complex real-life driving tasks.

This chapter is based on 🗎 *J. F. Schumann, J. Engström, L. Johnson, M. O'Kelly, J. Messias, J. Kober, A. Zgonnikov. "Active inference as a model of collision avoidance behavior in human drivers." Nature Communications (2026)* [207]

7.1 Introduction

Collision avoidance is a critical skill for human drivers. It involves the rapid detection of threats (such as a vehicle ahead suddenly braking) and deciding on an appropriate evasive maneuver (for instance, braking or swerving). These maneuvers are complex, requiring not only precise execution but also continuous adjustments as the situation evolves. Furthermore, drivers need to account for the uncertainty in the future behavior of other road users: for example, will an oncoming vehicle encroaching into my lane continue across or move back to its own lane? Understanding how humans avoid collisions in traffic can provide insights into high-stakes, split-second decision making which has substantial implications for traffic safety.

Behavior models play a key role both in understanding the mechanisms of human collision avoidance and in improving traffic safety. These models are applied in diverse contexts, such as collision risk estimation [208], understanding effects of driver distraction [193], modeling take-over behavior [54], representing human agents in simulated test environments [18], and providing behavioral benchmarks for automated vehicles [20, 209]. Besides immediate practical applications, modeling human collision avoidance behavior is an interesting subject of study in its own right. Being a highly complex and dynamic task with extremely high stakes, collision avoidance provides a unique testbed for theories and models of cognition previously not validated in the real world [200–202].

Most existing computational models of human collision avoidance are mechanistic, that is, are based on the explicit modeling of cognitive mechanisms underlying response timing and evasive maneuvering. However, they are typically fragmented, focusing either on specific scenario types (*e.g.*, front-to-rear conflicts [53, 210, 211] or merging [212]), specific explanatory factors (such as off-road glances [208, 213] or cognitive load [193]), or only reproduce certain aspects of human behavior (such as response times [53–55, 209, 210] or the extent of steering [214]). Altogether, these models cover a wide range of scenarios and diverse aspects of collision avoidance behavior. Yet, each one of these mechanistic models on its own is highly specific: they are not designed to generalize to multiple scenarios or describe multiple aspects of human behavior.

Recently, machine learning models based on large datasets of human driving have demonstrated the ability to generalize across a wide range of traffic scenarios [34, 38, 40, 76, 215, 216]. Because such models typically generate full motion trajectories, they also have the potential to represent multiple aspects of collision avoidance behavior and not just a single metric of interest. However, a key challenge is that safety-critical behavior such as collision avoidance is often under-represented in the datasets used for training [67], which makes it hard to achieve representative human-like collision avoidance behavior solely based on learning from data [16].

Thus, there is currently a lack of models that can capture the key aspects of human collision avoidance behavior (response selection, timing, and execution) all at the same time and across different scenarios. This limits both practical applications (due to the need to develop a new model for every new scenario) and fundamental understanding of cognitive mechanisms underlying the behavior of humans in safety-critical situations in traffic (due to the lack of a unified explanation for multiple aspects of behavior).

To address this gap, here we present a model of human collision avoidance behavior based on active inference. Originating in computational neuroscience, active inference

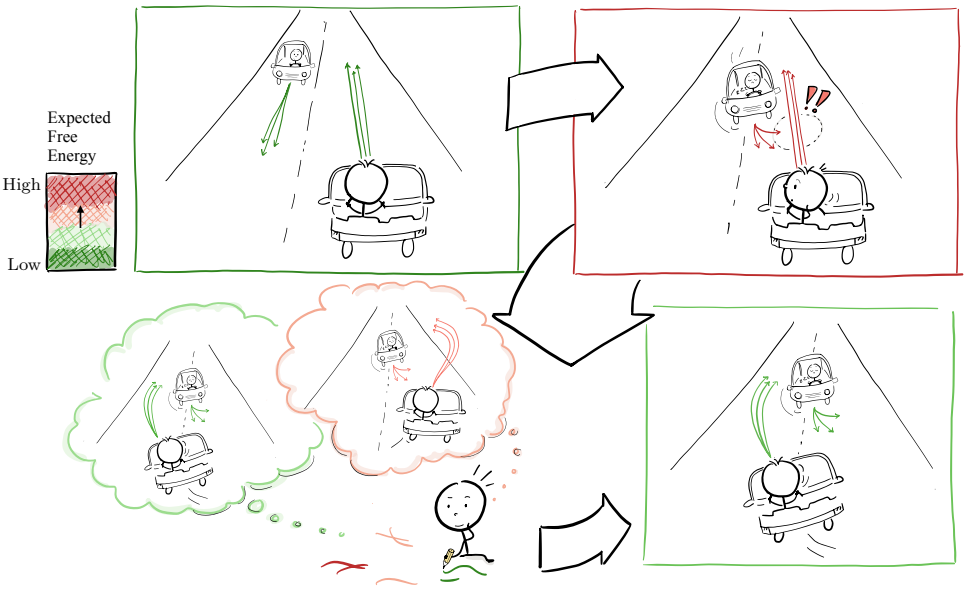


Figure 7.1: An illustration of the key principles of our active inference model in the opposite-direction lateral incursion scenario. Upper left panel: The modeled agent (at the bottom, facing forward) is continuously generating beliefs (represented by the arrows) about how the current driving situation will play out and what future observations it will make as a consequence of its own actions and the actions of the other vehicle. In "normal", non-conflict situations, the situation typically plays out as expected. The agent thus initially believes with high certainty that both itself and the oncoming vehicle will remain in their lanes and pass each other safely. This yields low expected free energy (EFE; green arrows) since the agent's expected (and thus preferred) observations (*e.g.*, making progress, avoid collisions) are aligned with the actual observations. Upper right panel: When the oncoming vehicle suddenly encroaches into the driver's lane, the EFE increases rapidly (as indicated by the red arrows) because this new situation is likely to result in an observed collision in the future, which is strongly disfavored by the agent. In this situation, the agent's belief about the oncoming vehicle's future trajectory also becomes more uncertain, because the other vehicle can no longer be trusted to follow established traffic norms. The agent thus needs to imagine a better future with lower EFE – in this case, a scenario where it avoids collision by swerving left (lower left panel) and take action to make that better future come true (lower right panel).

is a versatile general framework for understanding and modeling sentient behavior in living systems [57, 217, 218] that has been previously used to model human behavior in diverse contexts [219–223], including the modeling of human driver behavior [224] such as car following [225], responses to driving automation failures [55], and managing uncertainty around occlusions and non-driving-related tasks [199]. Building on the model of Engström *et al.* [199], our active inference model is designed to reproduce a wide spectrum of human behavior (*e.g.*, chosen maneuver, reaction timing, collision likelihood, etc.) in response to potential collisions. To this end, our model incorporates several well-known cognitive mechanisms to represent the dynamics of human decision making in response to sudden stimuli, such as looming perception [226, 227] and evidence accumulation [44, 45, 228, 229]. We evaluated our model against a range of previously reported empirical findings in three paradigmatic collision avoidance scenarios: 1) the front-to-rear

scenario, where a driver needs to respond to a suddenly braking vehicle in front, 2) the opposite-direction lateral incursion scenario, where an oncoming vehicle suddenly cuts across the driver's path, and 3) the intersection right-turn-into-path scenario, where a vehicle coming from the right on a perpendicular road does not yield when turning right into the driver's lane. Testing the model in three markedly different scenarios allowed us to investigate the potential of active inference as a framework for generalizable modeling of human collision avoidance behavior.

7.2 Methods

In this work, we use active inference as an overarching framework to guide the modeling. The key premise of active inference is that all behavior and cognition can be understood based on the single principle of minimizing *free energy*. An agent minimizing free energy can be conceptually understood as the agent sensing, and acting upon, the world in such a way as to minimize its surprise over time. In the active inference framework, this amounts to seeking out observations that are *expected* (unsurprising) and preferred given the type of creature the agent is, reflecting its adaptation to its particular environment or niche (*e.g.*, a fish expects and prefers the sensation of being immersed in water). Agents capable of planning into the future and modeling the consequences of their actions (such as human car drivers), select plans that minimize the *expected* surprise, or more generally the *expected free energy* (EFE) [57].

Our model follows the architecture of the original model initially formulated by Engström *et al.* [199], and combines some of the original elements of that model with multiple new mechanisms. In particular, our model retains the representation of probabilistic beliefs using sets of samples and the correspondingly sample-based estimation of the expected free energy (Equation (7.8)) as well as the cross entropy method used for policy selection. In addition to these mechanisms, we introduced looming perception, evidence accumulation, the norm-conditioned particle filter, and the constraints on acceleration profiles (Figure 7.6 for the analysis of the impact of these new mechanisms on model behavior). We also modified the Bayesian belief update (Equation (7.2)) and the calculation of the epistemic value (Equation (7.9)).

In general, the environment is represented by the *generative process*, with a state of the world $\boldsymbol{\eta} \in \mathscr{E}$ which is partially observable by the agent. The agent can influence the environment with its actions $\boldsymbol{a} \in \mathscr{A}$ (according to the state transition probability $\hat{p}(\boldsymbol{\eta}'|\boldsymbol{\eta}, \boldsymbol{a})$), resulting in observations $\boldsymbol{o} \in \mathscr{O}$, whose dependence on the world state is described in $\hat{p}(\boldsymbol{o}|\boldsymbol{\eta})$ (the $\hat{p}$ is used to show these function be part of the *generative process*).

The agent has an internal model of the world, the *generative model*, with the corresponding partially observable state $\boldsymbol{s} \in \mathscr{S}$, and the corresponding state transition function $p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$ and observation probability $p(\boldsymbol{o}|\boldsymbol{s})$. As the *generative model* is an abstraction of the real world described by the *generative process*, $\boldsymbol{\eta}$ and $\boldsymbol{s}$ do not necessarily have to represent the same state spaces ($\mathscr{E} \neq \mathscr{S}$). The agent is uncertain about the state of the generative model, represented by the probability density function $q(\boldsymbol{s})$ (referred to as the agent's belief about the state $\boldsymbol{s}$). Following [199], in our model the agent represents its belief distributions $q(\boldsymbol{s})$ in a non-parametric way as a set of N samples $\boldsymbol{S} = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N\}$ (*i.e.*, a particle filter).

The agent then tries to minimize its *free energy*, both in the belief $q(\boldsymbol{s})$ it forms (*vari-*

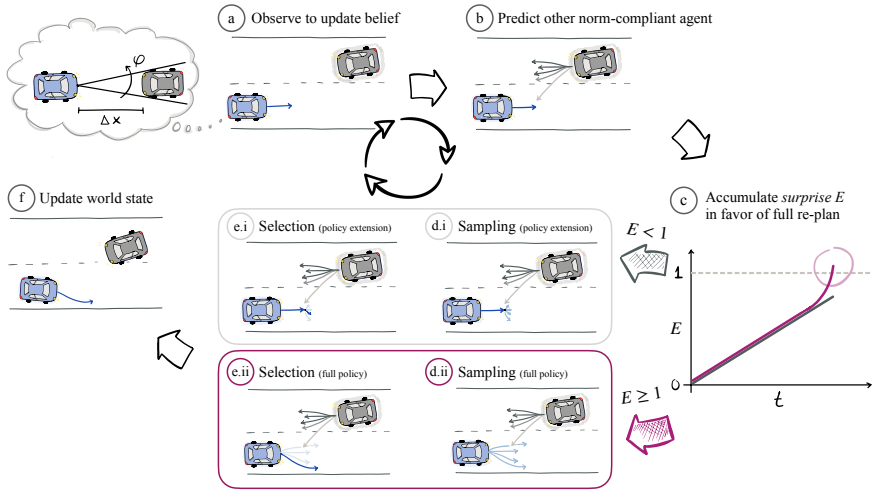


Figure 7.2: Main functional components of our active inference collision avoidance model. At each time step, the ego agent observes the world state, updates its belief about the world (a), and makes probabilistic predictions about the actions of the other agent (where future trajectories that adhere to traffic norms are prioritized) (b). At every time step, the agent accumulates evidence (as measured by surprise associated with the current state of the world) on the unsuitability of the current policy (c). If the accumulated evidence reaches a threshold, the agent re-plans the entire policy, selecting the one that minimizes the expected free energy (EFE) (d.ii and e.ii). Otherwise, it continues using its current policy in an extend form – *i.e.*, only sampling and then selecting the policy's last time step (d.i and e.i). The first action in the new or extended policy is then applied to the environment (f).

ational free energy) and the actions $\boldsymbol{a}$ it chooses (*expected free energy*). The action and policy selection is based on the existence of a preference prior $p(\boldsymbol{o})$ depicting preferred observations.

7.2.1 Perception

At time t, the agent perceives the world state $\boldsymbol{\eta}_t$, resulting in the observation $\boldsymbol{o}_t$:

$$\boldsymbol{o}_t \sim \hat{p}(\boldsymbol{o}'|\boldsymbol{\eta}_t) \tag{7.1}$$

With this observation, the agent updates its old belief $q(\boldsymbol{s}_{t-1})$ to $q(\boldsymbol{s}_t)$. The active inference framework postulates that this update is based on the minimization of the *variational free energy* [57], but in our case we use standard Bayesian updating:

$$q(\boldsymbol{s}_t) \propto p(\boldsymbol{o}_t|\boldsymbol{s}_t)\mathbb{E}_{q(\boldsymbol{s}_{t-1})}p(\boldsymbol{s}_t|\boldsymbol{s}_{t-1}, \boldsymbol{a}_{t-1}). \tag{7.2}$$

Here, the particle filter first advances the individual samples in $\boldsymbol{S}_{t-1}$ (representing $q(\boldsymbol{s}_{t-1})$) using the transition function to get the expected states $\boldsymbol{S}_{A,t}$ (where we approximate $q_A(\boldsymbol{s}_t) = \exp\left(\mathbb{E}_{q(\boldsymbol{s}_{t-1})} \ln p(\boldsymbol{s}_t|\boldsymbol{s}_{t-1}, \boldsymbol{a}_{t-1})\right)$), with $\boldsymbol{s}_{A,t,n} \sim p(\boldsymbol{s}'|\boldsymbol{s}_{t-1,n}, \boldsymbol{a}_{t-1})$ being randomly sampled.

Then, a kernel density estimate [153, 230] based on $\boldsymbol{S}_{A,t}$ is used to approximate $q_A(\boldsymbol{s}_t)$ as a Gaussian Mixture Model (GMM). Assuming that $p(\boldsymbol{o}_t|\boldsymbol{s}_t)$ is Gaussian as well, $q(\boldsymbol{s}_t)$ can

also be represented as a GMM; this allows straightforward sampling of S_t. Compared to the approach in the original model [199] of weighting samples $s_{A,t,n} \in S_{A,t}$ with $p(o_t|s_{A,t,n})$ and then resampling based thereon, this approach allowed us to represent large shifts in beliefs outside the extreme values of the initial set S_0.

The agent's visual perception in our model is based on looming (Section B.2.2) – the optical information immediately available to humans about the relative motion of objects straight ahead in the direction of travel [227]. Specifically, the relative position and motion of objects along the forward path of the agent is perceived in terms of the visual angle φ subtended by the object at the retina of the observer, and its derivative, the angular rate $\dot{\varphi}$ (looming) [226, 231]. Due to the nonlinear relationship between the observer's distance to the observed vehicle (in Figure 7.2**a** referred to as Δx) and φ (with increasing distances, equal variations $\delta\Delta x$ lead to increasingly smaller $\delta\varphi$), perception errors for φ and $\dot{\varphi}$ will lead, respectively, to increasing inference errors and therefore uncertainty about the position and speed of the leading other vehicle at increasing distances. We represent this aspect in the model by setting the noise in $p(o|s)$ to be constant across φ and its derivatives.

At a certain distance, $\dot{\varphi}$ is no longer perceptible to the human eye, which implies a minimal threshold for looming detection [232, 233]. To account for this, the model incorporates a detection threshold $\dot{\varphi}_0$ on $\dot{\varphi}$ (*i.e.*, for small $|\dot{\varphi}|$, the means of the observation probabilities $p(o|s)$ for velocities and acceleration were set to the values corresponding to $\dot{\varphi} = 0$). This means that changes in the other vehicle's velocity cannot be perceived by a human driver beyond a certain distance, resulting for example in a delayed recognition if the lead vehicle suddenly brakes.

7

7.2.2 Behavior prediction

After observing the current state of the world, the agent applies the state transition function $p(s'|s, a)$ starting from its current belief $q(s_t)$ in order to generate a predicted belief about the future states of the situation $\tilde{q}_s(s_\tau|\pi_t, q(s_t))$ (with $\tau \in \{t+1, \dots, t+H\}$, an arbitrary policy π containing the ego agent's sequence of planned actions, and prediction horizon H) of all the agents in the scene, as well as the corresponding belief over possible future observations $\tilde{q}_o(o_\tau|\pi_t, q(s_t)) = \mathbb{E}_{\tilde{q}_s(s_\tau|\pi_t, q(s_t))} p(o_\tau|s_\tau)$. During this prediction process, the agent adds noise to the control inputs of the other agents to represent the possibility that they might change their behavior.

Practically, the agent advances the belief $\tilde{q}_s(s_\tau|\pi_t, q(s_t))$ (which we represent using the set of samples $\tilde{S}_\tau = \{s_{\tau,1}, \dots, s_{\tau,N}\}$) to the next time step by using the state transition function (in our case based on a bicycle model [234], with a detailed discussion in Section B.2.1) on each of the sample states $s_{\tau,n} \sim p(s_\tau|s_{\tau-1,n}, a_{\tau-1})$ (similarly to forming the expected belief $q_A(s)$ in the belief update). The belief $\tilde{q}_o(o_\tau|\pi_t, q(s_t))$ is then represented by the set $\widetilde{O}_\tau = \{o_{\tau,1}, \dots, o_{\tau,N}\}$, where $o_{\tau,n} \sim p(o_\tau|s_{\tau,n})$.

This belief propagation scheme represents *unsophisticated* inference in the technical sense that it does not condition the beliefs about future states on corresponding counterfactual observations. While it would theoretically be possible to use a form of *sophisticated* inference [235] that accounts for such conditioning, we here used the unsophisticated approach for simplicity, similar to the original model [199] and most existing active inference models.

Importantly, our state transition function $p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$ function is composed out of two parts, with

$$p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a}) \propto \check{p}_{\mathrm{n}}(\boldsymbol{s}')p_o(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a}). \tag{7.3}$$

Here, $p_o(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$ is the standard kinematic likelihood function (as used in [199]), while the *projected normative probability* $\check{p}_{\mathrm{n}}(\boldsymbol{s}')$ biases the agent to predict future states which currently, and over the near future, are aligned with the norm-compliant behavior of other vehicles. This represents the human tendency to avoid overly pessimistic predictions about the behavior of other traffic participants by instead assuming that they will adhere to road rules and other established traffic norms [236, 237]; for example, assuming that oncoming vehicles will stay in their lane or yield the right of way, without evidence to the contrary, is necessary for fluent driving.

In practice, we define $\check{p}_{\mathrm{n}}$ with

$$\check{p}_n(\boldsymbol{s}_\tau) \propto \min\left\{p_{\mathrm{n}}(\boldsymbol{s}_\tau), 2\frac{p_{\mathrm{n}}(\check{\mathbf{s}}_{\tau+1})\,p_{\mathrm{n}}(\check{\mathbf{s}}_{\tau+H_{\mathrm{n}}})}{p_{\mathrm{n}}(\check{\mathbf{s}}_{\tau+1}) + p_{\mathrm{n}}(\check{\mathbf{s}}_{\tau+H_{\mathrm{n}}})}\right\}, \tag{7.4}$$

applying the *normative probability* p_{n} to the current time as well as short- and medium-term predictions $\check{\mathbf{s}}$ (H_n is setting the medium-term prediction horizon) to evaluate their norm compliance. This ensures that the model's predictions are not only biased against actions which lead to immediate norm violations, but also against those that put the other vehicle into a position where norm compliance becomes kinematically less likely over time (*e.g.*, initiating a turn while in a straight lane does not immediately lead to the vehicle leaving the road, but might put the vehicle in a position where staying on the road becomes kinematically impossible). Crucially, the use of the current norm-compliance as an upper bound ensures that once norm-violating behavior is observed, the model no longer places trust into norms as a way of constraining predictions about other agents' behavior. In collision avoidance scenarios initiated by the unexpected, norm-violating behavior of other agents, this allows the model to account for all kinematically possible but low-probability "extreme" (long-tail) behaviors. For details on sampling from $p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$, refer to Section B.2.1.

This approach – which we call *norm-conditioned particle filter* – helps aligning the model's predictions with realistic driving scenarios while accounting for norm violations when necessary. While there is generally a large set of normative expectations dependent on cultural influence, we chose to only implement specific norm-influencing behaviors in each scenario (*e.g.*, staying in one's lane or yielding to the right of way, with the detailed implementation shown in Figure B.1). A generalized, scenario-independent implementation of such norms is feasible, but with the goal of computational efficiency was not pursued here.

7.2.3 Policy sampling

Based on the predicted behavior of the other vehicle, the agent selects its policy $\boldsymbol{\pi}$. Here, policies are defined as sequences of H future actions; following the bicycle model, the actions are defined in terms of acceleration a_{long} and steering rate ω. These actions are selected using the cross-entropy method [199, 238], which iteratively resamples sets of M candidate policies K times, where the resampling focuses around the subset of the

βM best samples from the previous iteration. Specifically, the agent takes the mean and variance of this promising subset (ignoring potential correlations), and then samples a new set with size M from the corresponding normal distribution. In contrast to the original model [199] that randomly picked the final sampling output from the last iteration's set, here we specifically choose the best sample.

To better represent human behavior, the model incorporates a pedal constraint to filter out unrealistic changes in acceleration (Section B.2.3). As human drivers generally control acceleration by operating the brake and gas pedal with a single foot and cannot move the foot between pedals instantaneously, the model imposes a constant holding time interval of 0.2 s at $a_0 \lesssim 0\,\mathrm{ms}^{-2}$ (similar to [239]), such that if the model transitions from a current acceleration $a_{\mathrm{long}} > a_0$ to a target acceleration $a_{\mathrm{long},f} < a_0$, it must first decelerate to a_0, maintain that value for 0.2 s, and then proceed to $a_{\mathrm{long},f}$. The same applies for acceleration changes in the other direction.

7.2.4 Policy roll-out and evaluation

To evaluate a policy π, the agent first uses the bicycle model to roll out its possible future states based on $\boldsymbol{\pi}$. In the general case, the possible future states of the agent and the other vehicle are coupled, hence the agent rolls out its policies jointly with the predicted futures of the other vehicle. For the reason of simplicity, however, our model assumes that in $p(\boldsymbol{s'}|\boldsymbol{s},\boldsymbol{a})$ the other vehicle is unresponsive to the actions of the ego vehicle, allowing us to separate behavior prediction and policy roll-outs.

Active inference then rests on the fundamental assumption that an agent prefers policies which minimize its expected free energy (EFE) G over the prediction horizon of H time steps; in other words, the best policy is the one with the lowest EFE. This EFE combines the desire of the agent to, on the one hand, observe itself in states that it prefers (maximize pragmatic value g_{pragm}) and, on the other hand, seek information that may reduce uncertainty about the world (maximize epistemic value g_{epist}):

$$G(\boldsymbol{\pi}_t, q(\boldsymbol{s}_t)) = \sum_{\tau=t+1}^{t+H} -g_{\mathrm{pragm}}\left(\tilde{q}_o(\boldsymbol{o}_\tau|\boldsymbol{\pi}_t, q(\boldsymbol{s}_t))\right) - g_{\mathrm{epist}}\left(\tilde{q}_s(\boldsymbol{s}_\tau|\boldsymbol{\pi}_t, q(\boldsymbol{s}_t))\right). \tag{7.5}$$

The pragmatic values g_{pragm} is defined as

$$g_{\mathrm{pragm}}\left(\tilde{q}_o(\boldsymbol{o})\right) = \mathbb{E}_{\tilde{q}_o(\boldsymbol{o})} \ln p(\boldsymbol{o}), \tag{7.6}$$

while we use the Shannon entropy $\mathscr{H}$ to calculate the epistemic value [57]

$$\begin{aligned} g_{\mathrm{epist}}\left(\tilde{q}_s(\boldsymbol{s})\right) &= \mathbb{E}_{\tilde{q}_o(\boldsymbol{o})} D_{KL}\left[\tilde{q}(\boldsymbol{s}|\boldsymbol{o})|\tilde{q}_s(\boldsymbol{s})\right] \\ &= \mathscr{H}(\mathbb{E}_{\tilde{q}_s(\boldsymbol{s})} p(\boldsymbol{o}|\boldsymbol{s})) - \mathbb{E}_{\tilde{q}_s(\boldsymbol{s})} \mathscr{H}(p(\boldsymbol{o}|\boldsymbol{s})). \end{aligned} \tag{7.7}$$

The first term in Equation (7.7) is known as the *posterior predictive entropy* and represents the extent to which a policy is expected to yield a variety of different observations. The second term, known as *expected ambiguity* represents the diversity of observations expected for a given state, that is, the extent to which the observations of that state are unreliable (*e.g.*, due to reduced visibility).

In our particle-based representation, we represent $\tilde{q}_s(\boldsymbol{s})$ with the set $\tilde{S}$ of N particles and $\tilde{q}_o(\boldsymbol{o}) = \mathbb{E}_{\tilde{q}_s(\boldsymbol{s})} p(\boldsymbol{o}|\boldsymbol{s})$ with the set $\widetilde{\boldsymbol{O}}$, resulting in the approximations

$$\tilde{g}_{\text{pragm}}\left(\widetilde{O}\right) = \frac{1}{N} \sum_{\boldsymbol{o} \in \widetilde{\boldsymbol{O}}} \ln p(\boldsymbol{o}) \tag{7.8}$$

and

$$\tilde{g}_{\text{epist}}\left(\tilde{S}, \widetilde{\boldsymbol{O}}\right) = -\frac{1}{N} \sum_{\boldsymbol{o} \in \widetilde{\boldsymbol{O}}} \ln\left(\frac{1}{N} \sum_{\boldsymbol{s} \in \tilde{S}} p(\boldsymbol{o}|\boldsymbol{s})\right) - \frac{1}{N} \sum_{\boldsymbol{s} \in \tilde{S}} \mathscr{H}\left(p(\boldsymbol{o}|\boldsymbol{s})\right), \tag{7.9}$$

under the assumption that the entropy $\mathscr{H}\left(p(\boldsymbol{o}|\boldsymbol{s})\right) : \mathcal{S} \longrightarrow \mathbb{R}$ can be calculated analytically. The original model [199] had calculated g_{epist} by using KDE methods based on $\widetilde{\boldsymbol{O}}$ to approximate $\tilde{q}_o(\boldsymbol{o}) = \mathbb{E}_{\tilde{q}_s(\boldsymbol{s})} p(\boldsymbol{o}|\boldsymbol{s})$, compared to us using $\frac{1}{N} \sum_{\boldsymbol{s} \in \tilde{S}} p(\boldsymbol{o}|\boldsymbol{s})$. We opted for this approach, as the original estimation would result in an over-approximation of $\tilde{q}_o(\boldsymbol{o})$, given that we evaluate the KDE exclusively at the center of each kernel. Consequently, we use the following approximation $\tilde{G}$ for the EFE:

$$\tilde{G}(\boldsymbol{\pi}_t, q(\boldsymbol{s}_t)) = \sum_{\tau=t+1}^{t+H} -\tilde{g}_{\text{pragm}}\left(\widetilde{\boldsymbol{O}}_\tau\right) - \tilde{g}_{\text{epist}}\left(\tilde{S}_\tau, \widetilde{\boldsymbol{O}}_\tau\right). \tag{7.10}$$

When evaluating the pragmatic value, we assume that the preference (*i.e.*, the distribution of desired observations) $p(\boldsymbol{o})$ accounts for four aspects of the agent's objective: a) to maintain its desired longitudinal velocity, b) to minimize the magnitude of control inputs, c) to remain on the road and within its current lane (*e.g.*, avoiding lane markers or opposite lanes), d) prevent collisions, and e) avoiding hazardous situations. Each of these aspects is represented by an individual preference function (for details refer to Section B.2.4), which are multiplied together:

$$p(\boldsymbol{o}) = \mathcal{N}(v_{\text{ego}}|\mu_v, \sigma_v)\, \mathcal{N}(a_{\text{long,ego}}|0, \sigma_a) \mathcal{N}(\omega_{\text{ego}}|0, \sigma_\omega)\, p_{\text{lat}}(y_{\text{ego}})\, p_{\text{coll}}(\boldsymbol{o})\, p_{\text{safe}}(\boldsymbol{o}). \tag{7.11}$$

For $p_{\text{safe}}(\boldsymbol{o})$, the model specifically seeks to avoid states where, under the assumption that the other vehicle begins braking suddenly with a deceleration of $a_{\text{OV,min}}$, the ego vehicle would fail to avoid a collision despite initiating maximum braking after a response time of 1 s.

7.2.5 Surprise-based re-planning

Our model assumes that on every time step, the agent performs planning incrementally, unless it observes a surprising event, in which case it re-plans the full policy. Practically, on every time step the agent takes the policy chosen in the previous time step, disregards its first action (as it was already executed) and assumes that the remaining actions constitute all but the last actions of the new policy. The above policy selection mechanism is then used to generate the last action of the new policy.

In parallel with incremental policy updates, on every time step the agent accumulates evidence in favor of a full re-plan, using the accumulation rate λ:

$$E_t = E_{t-1} + \lambda \epsilon_t. \tag{7.12}$$

Traditionally in evidence accumulation literature, the accumulation rate is referred to as drift rate, which is interpreted as the quality of the incoming evidence [240] and can be linked to the decision-maker's efficiency of processing the perceptual information [241, 242]. Here, we follow Engström *et al.* [209] in accumulating *surprise* $\epsilon \geq 0$, defined as the *residual information* of the pragmatic value [243]:

$$\epsilon_t = H \max_{\boldsymbol{o}} \ln p(\boldsymbol{o}) - \sum_{\tau=t+1}^{t+H} g_{\text{pragm}}\left(\tilde{q}_o(\boldsymbol{o}_\tau | \boldsymbol{\pi}_t, q(\boldsymbol{s}_t))\right). \tag{7.13}$$

Under this definition, the *surprise* is the difference between the highest pragmatic value possible and the actual pragmatic value of a policy. For instance, if a policy would result in the most preferred observation, the agent would not accumulate any additional evidence. However, if a policy would lead to an undesired (*i.e.*, *a priori* unlikely) observation, much evidence for a re-plan would be gained.

If the accumulated evidence is below the threshold of 1, the model then follows the extended policy. Otherwise, it generates a completely new policy, applying the above policy selection mechanism to every action in the policy.

Finally, after the policy is determined, its first action $\boldsymbol{a}_t$ is used to update the state of the actual world. This is done using the state transition function of the *generative process*:

$$\boldsymbol{\eta}_{t+1} \sim \hat{p}(\boldsymbol{\eta}' | \boldsymbol{\eta}_t, \boldsymbol{a}_t) \tag{7.14}$$

7.2.6 Model parameters

Each of the key model mechanisms is associated with a number of parameters, with 26 of the most essential ones shown in Tables 7.1 and 7.2. For 12 of them, the exact values have been directly adopted from the literature, in particular the original model [199]. Another thirteen model parameters were treated as free parameters, and were manually tuned to qualitatively match the empirical observations.

Table 7.1: An overview of the most essential parameters in the *generative model*. For tuned parameters, we normally only tested three to four different parameters.

Parameter	Description	Source
State transition function $p(\boldsymbol{s}'\|\boldsymbol{s},\boldsymbol{a})$		
$\Delta t = 0.2\,\mathrm{s}$	Time step size	Routine driving model [199]
$4.2\,\mathrm{m} \times 1.72\,\mathrm{m}$	Vehicle size	Matching [244]
$w = 3.65\,\mathrm{m}$	Lane width	Matching [244]
$\sigma_{a,0} = 3\,\mathrm{ms}^{-2}$	Noise applied to other vehicle's acceleration during belief update	Tuned parameter
$\sigma_{\omega,0} = 0.4575\,\mathrm{s}^{-1}$	Noise applied to other vehicle's steering rate during belief update	Tuned parameter
$\sigma_a = 0.6\,\mathrm{ms}^{-2}$	Noise applied to other vehicle's acceleration during behavior prediction	Tuned parameter
$\sigma_\omega = 0.0915\,\mathrm{s}^{-1}$	Noise applied to other vehicle's steering rate during behavior prediction	Tuned parameter
$H_\mathrm{n} = 20$	Prediction horizon for the *projected normative probability*	Tuned parameter
Observation probability $p(\boldsymbol{o}\|\boldsymbol{s})$		
$\dot{\varphi}_0 = 0.002\,15\,\mathrm{s}^{-1}$	Looming threshold	Taken from [233]

Table 7.2: An overview of the most essential parameters used in the full model for the policy selection. For tuned parameters, we normally only tested three to four different parameters, with the drift rate λ being the noticeable exception.

Parameter	Description	Source
Preference function $p(\boldsymbol{o})$		
$\mu_v = v_0$	Mean of preference distribution of agent's velocity v	Routine driving model [199]
$\sigma_v = 0.5\,\mathrm{ms}^{-1}$	Standard deviation of preference distribution of agent's velocity v	Tuned parameter
$\sigma_a = 0.1\,\mathrm{ms}^{-2}$	Standard deviation of preference distribution of agent's acceleration a	Tuned parameter
$\sigma_\omega = 0.02\,\mathrm{s}^{-1}$	Standard deviation of preference distribution of agent's steering rate ω	Tuned parameter
$g_{LL} = -15000$	Pragmatic value for leaving the road	Tuned parameter
$g_{LC} = -1000$	Pragmatic value for driving on a lane boundary or in opposing lanes	Tuned parameter
$g_C = -10000$	Pragmatic value after collision with relative velocity of $10\,\mathrm{ms}^{-1}$	Tuned parameter
Policy selection		
$H = 30$	Prediction horizon	Routine driving model [199]
$N = 75$	Number of particles	Routine driving model [199]
$K = 20$	Number of iterations of policy sampling	Routine driving model [199]
$M = 100$	Number of sampled policies	Tuned parameter
$\beta = 0.1$	βM samples with lowest EFE are used as base for next iteration	Routine driving model [199]
$\mathcal{N}\left(0, 5\,\mathrm{ms}^{-2}\right)$	Distribution to sample accelerations a_τ in first CEM iteration	Routine driving model [199] (mean) and Tuned parameter (std)
$\mathcal{N}\left(0, 0.1\,\mathrm{s}^{-1}\right)$	Distribution to sample steering rates ω_τ in first CEM iteration	Routine driving model [199] (mean) and Tuned parameter (std)
$a_0 = -0.1\,\mathrm{ms}^{-2}$	Acceleration applied when no pedal is pressed	Logical consideration
$\lambda = 10^{-5.95}$	Evidence accumulation drift rate	Tuned parameter

7.3 Results

We evaluated the model against human data in three scenarios[1]. First, in the *front-to-rear* scenario, we compared the model to the results of a meta-analysis of brake response times [245] in experimental driving simulator and field studies (*e.g.*, Brookhuis *et al.* [246] and Lee *et al.* [247]), and an analysis of deceleration magnitudes reported in [213] based on two real-world driving datasets captured across the US (the SHRP2 dataset [248]) and Africa (the ANNEXT dataset [249]).

Second, in the *opposite-direction lateral incursion* scenario, the model was compared to human data from a driving simulator study conducted in UK [244] which was conducted in parallel to the present work. Thus, by contrast to the model evaluation for the front-to-rear scenario, which was based on aggregated data, we were here able to run our model on nearly identical scenarios for which the human data was collected, allowing for more detailed comparisons. We manually fitted the model parameters to reproduce human behavior in the first two scenarios; same parameter values were used for both scenarios.

Third, to test the ability of the model to generalize to new scenarios, we tested it, with the same parameter settings, in an unseen third scenario conducted in Canada [250], an intersection right-turn-into-path scenario where another vehicle failed to yield the right of way. Similarly to the lateral incursion scenario, we were able to closely reproduce the scenarios faced by the human participants.

7.3.1 Front-to-rear scenario

In the front-to-rear scenario, the ego vehicle trails the other vehicle which is driving in the same lane; both vehicles have the same initial velocity which was systematically varied in our simulations together with the initial time gap between the vehicles. After a short time of driving at constant speed, the other vehicle starts braking with a high constant deceleration until coming to a stop. Depending on the initial kinematics of the vehicles, the ego agent can avoid collision either solely by braking (Figure 7.3**a**) or combined braking and swerving (Figure 7.3**b**). The model sometimes avoided a collision by swerving only (*i.e.*, without braking). However, since existing results in the literature on front-to-rear scenarios typically only report on braking response performance, we only compared our model's brake response times to the human response data (*i.e.*, for the analysis of those, steering only responses were not considered).

At lower speeds and larger time gaps, the model typically avoids collisions by braking only. In a representative simulation (Figure 7.3**a**), after the leading agent started braking at $t = 0.8\,\text{s}$ (first dashed line), it took the model 0.6 s to perceive this (see the ego agent's belief in the acceleration plot). At that point ($t = 1.4\,\text{s}$, second dashed line), the model recognized the imminent conflict resulting from this change in perceived behavior of the other vehicle, as can be seen by the sudden decrease (increase in negative value) in the collision part of the pragmatic value. This leads to a corresponding increase in the rate of accumulation of the evidence for a re-plan (*i.e.*, surprise). However, it still took another 0.6 s until the accumulated evidence E reached the threshold, delaying the first noticeable agent reaction to $t = 2\,\text{s}$ (third dashed line), where the agent started to execute a new policy – emergency braking. This policy was selected because it allowed the agent to drastically reduce its EFE

[1]The simulation data underlying the presented results can be found at `https://osf.io/gs4bu`.

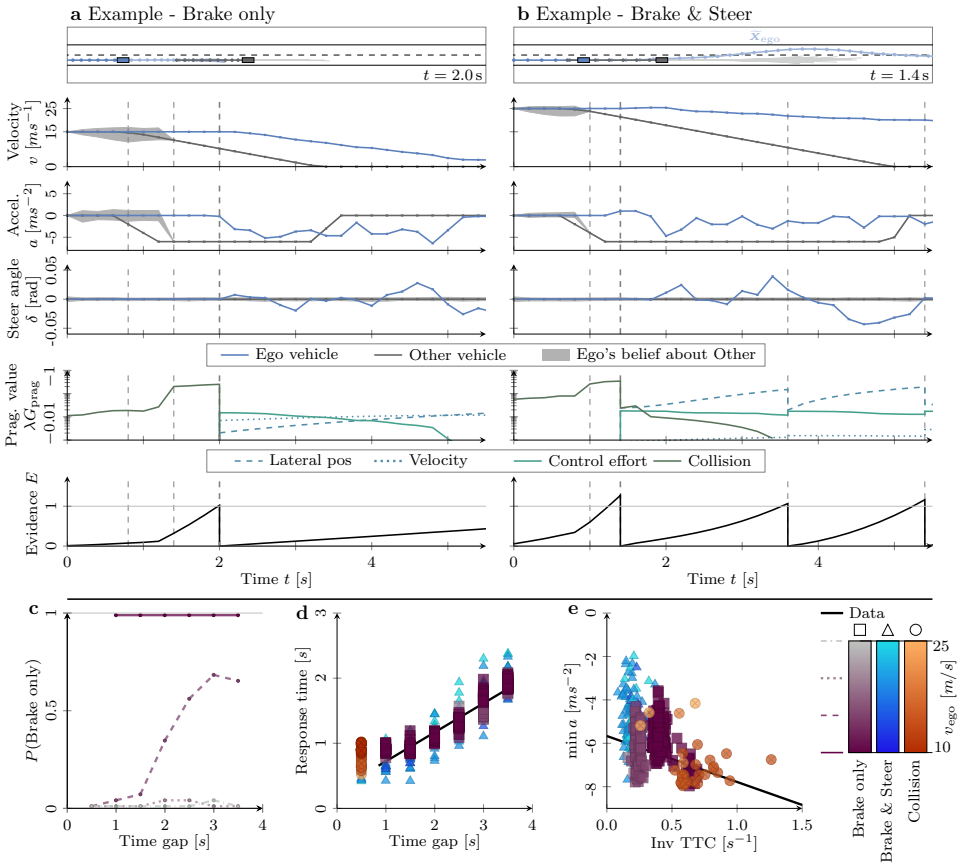


Figure 7.3: Evaluation of the model in the front-to-rear scenario. **a)** In a scenario where initial pre-conflict kinematics are such that collision can be avoided by braking (initial vehicle velocities $v_0 = 15\,\mathrm{ms}^{-1}$, a bumper-to-bumper time gap of 3 s), the model typically produces braking-only behavior. The sub-panels visualize the top-down view of the scenario, the time dynamics of both vehicles' velocity v, acceleration a, steering angle δ and (for the ego vehicle) the components of the pragmatic value G_{prag} (scaled with the evidence accumulation gain λ) that underlie the accumulated evidence towards a full policy re-plan. **b)** In a scenario where braking on its own might not be sufficient to avoid collision ($v_0 = 25\,\mathrm{ms}^{-1}$, 1 s time gap), the model selects a policy that involves both braking and steering towards the opposite lane (see the model's selected policies in the topmost plots). **c)** The likelihood of the ego vehicle deciding to brake and stay behind the leading other vehicle (without any steering) as a function of time gap, with the four lines representing four different velocities (ranging from $10\,\mathrm{ms}^{-1}$ to $25\,\mathrm{ms}^{-1}$). This data include those samples where the agent is only steering. **d)** The relationship between the time gap and the brake response time of the ego agent in 896 model simulations (Section B.3.1). Regression line based on a meta-analysis of human brake response times [245] is shown for reference. **e)** The relationship of the inverse time-to-collision (TTC) at brake onset (defined as $\frac{\max\{0, v_{\mathrm{ego}} - v_{\mathrm{OV}}\}}{\Delta x}$) and the lowest observed deceleration. Regression line based on the human deceleration magnitude data [213] is shown for reference. Video replays of the example simulations are available in the Supplementary Material.

via reaching future states with high pragmatic value, in particular thanks to low collision probability. This policy was preferred over the alternative policy of braking and steering into the adjacent lane because the pragmatic value associated with lane changing was more negative than that associated with lower velocity, as the desired (corresponding to the initial) velocity is comparatively low. However, even under this policy, at $t = 2\,\mathrm{s}$, the acceleration has still not changed, due to the pedal constraints. Consequently, the first deceleration of the model can be observed at $t = 2.2\,\mathrm{s}$, resulting in a total response time of 1.4 s.

Smaller time gaps and higher initial velocities can make it kinematically impossible to avoid collision by braking only. In such situations, the model typically opts to steer and brake at the same time (Figure 7.3**b**). Here, due to the smaller initial distance, the model's perception delay was noticeably shorter (being only about 0.2 s, with the modeled agent perceiving the braking at $t = 1\,\mathrm{s}$, the first dashed line). However, due to the more challenging initial kinematics, it was more difficult for the model to avoid collision, despite the shorter perception delay. In particular, the model switched between different policies three times. Initially, the model recognized the need to slightly brake and then steer around the other vehicle: already at the time of the first re-plan ($t = 1.4\,\mathrm{s}$, second dashed line) the collision component of EFE was substantially reduced by the chosen policy. This policy was mostly sufficient in improving the collision component of the pragmatic value, but as visible in the top panel (planned trajectory $\widetilde{X}_{\mathrm{ego}}$), the agent intends to stay in between lanes for an extended period. This led to two further re-plans ($t = 3.6\,\mathrm{s}$ and $t = 5.4\,\mathrm{s}$, third and fourth dashed lines respectively) to adjust the trajectory. However, even having passed the other vehicle already, the model is not completely successful, with a noticeable lateral position component remaining in the pragmatic value. As the optimal solution of driving in the center of the current lane on a free road should have maximized the pragmatic value, this remaining lateral error is likely the result of the cross-entropy method not identifying the optimal policy. This represents a bounded planning capacity which, as discussed above, is also present in humans [212, 251–254] and thus an intended feature of our model.

By systematically varying the initial kinematics of the vehicles, we assessed the model behavior across a spectrum of front-to-rear scenarios. Due to the stochasticity of the model (foremost in behavior prediction and policy sampling), for each of the 28 sets of initial conditions we ran the simulation of the front-to-rear scenario 32 times. Based on these simulations, we analyzed the model's chosen evasive maneuver, response time, and deceleration magnitude and compared it to human data reported in the driver behavior literature.

Figure 7.3**c** shows the probability that the model will brake only (without steering) for different velocities and time gaps. Existing studies on how human evasive maneuver choice depends on scenario kinematics in front-to-rear scenarios are limited. Thus, we compared our model results to the few related studies that exist, but these do not provide sufficiently detailed analyses for rigorously evaluating our model's evasive maneuvering decisions. These studies generally suggest that evasive maneuver decisions, similar to response timing, depend on the scenario kinematics (see [244] for a review). Avoiding collisions is generally kinematically more feasible by braking at lower speeds and by steering at higher speeds [255] and human drivers tend to swerve into an adjacent lane if a kine-

matically viable escape path exists [244, 256–258]. In line with these general observations, our model prefers braking at lower velocities and favors combined swerving-and-braking or swerving only responses at higher velocities, regardless of the time gap (Figure 7.3**c**). At medium velocities ($15\,\mathrm{ms}^{-1}$) the model behavior is sensitive to the gap between the vehicles: the longer the time gap, the higher the probability of a brake-only maneuver. This pattern reflects a trade-off between the components of the pragmatic values associated with collisions, control effort (harshness of braking and steering), and lateral position. In more urgent scenarios (short time gaps), the model prefers swerving due to the low pragmatic value of harsh braking. Conversely, in less urgent scenarios (larger time gaps), the model opts for braking since moderate braking avoids collision without incurring the significant reduction of pragmatic value due to leaving the lane (while the deviation from the desired velocity is still tolerable at those medium speeds). These specific dependencies of evasive maneuver choice on scenario kinematics have, to our knowledge, not been empirically tested and represent interesting quantitative predictions from our model.

Regarding brake response times, empirical studies have consistently found a strong dependence on scenario kinematics [54, 193, 209, 213, 245, 259, 260]. Specifically, more kinematically urgent scenarios (*e.g.*, with a small initial time gap and/or hard lead vehicle braking) lead to shorter response times while less urgent scenarios lead to longer response times, with approximately linear relationship between measures of urgency (*e.g.*, the initial time gap) and mean response time [209, 245]. Our model produced behavior that is remarkably consistent with these findings (Figure 7.3**d**). Interestingly, compared to the evasive maneuver decisions, velocities do not seem to significantly influence brake response times, nor does the final chosen evasive behavior. Furthermore, model response times remain consistent (approx. 1 s) over the range of short time gaps (0.5 s to 1.5 s). This last result could be a consequence of the evidence accumulation mechanism requiring certain minimum time to trigger a re-plan even at a very high rate of incoming evidence. Relatedly, the lack of fast enough responses at short time gaps is a likely explanation for the collisions observed in the model (only at the shortest time gap of 0.5 s). However, because the data reported in the literature does not cover such short time gaps, this model prediction remains to be tested in future studies.

While response selection and its timing are comprehensively characterized by the choice of evasive maneuver and response times, response execution is a dynamic process and can be characterized by a variety of metrics. Here we focus on deceleration magnitude, which has been previously shown to depend on the kinematics of the front-to-rear scenario: human drivers brake harder in more kinematically urgent scenarios [213]. Our model captured this phenomenon: the magnitude of its braking increased with the inverse time-to-collision at brake onset (Figure 7.3**e**). This behavior of the model can be explained by the pragmatic value trade-off between avoiding a collision on the one hand and avoiding the effort of hard braking and losing velocity on the other hand. In other words, the model aimed to not brake harder than necessary to avoid collision.

7.3.2 Opposite-direction lateral incursion scenario

For evaluating model behavior in the opposite-direction lateral incursion scenario, we used human data from a driving simulator study at the University of Leeds, UK, reported by Johnson *et al.* [244]. In this study, vehicles initially approached each other in opposite

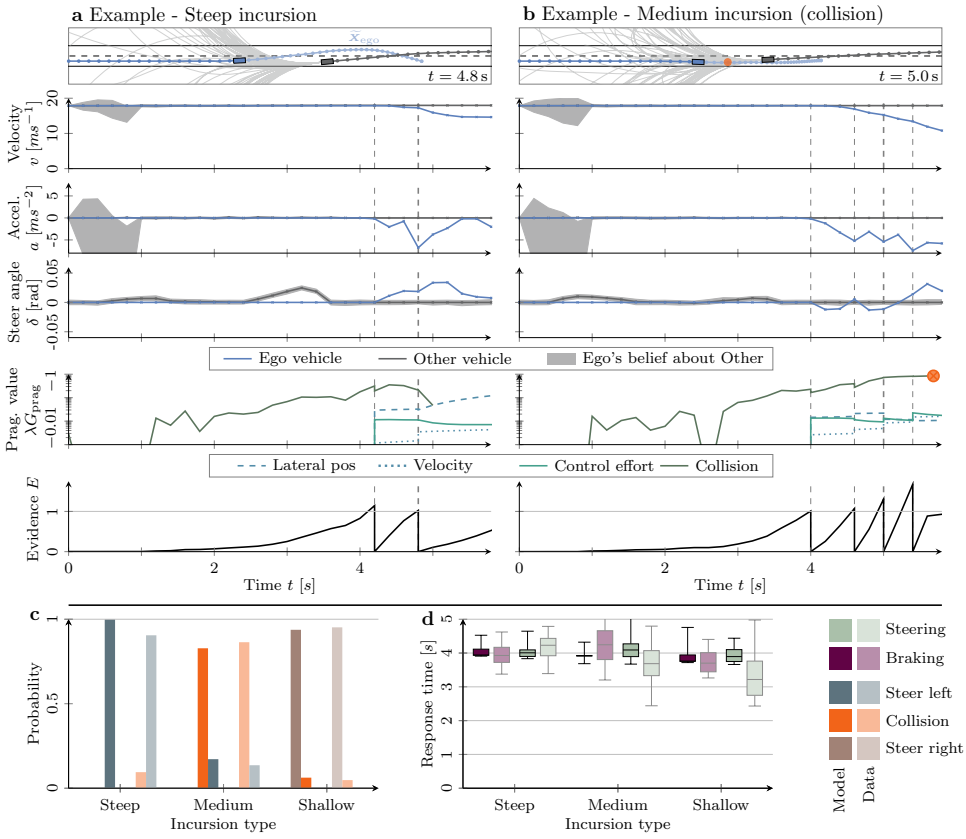


Figure 7.4: Evaluation of the model in the opposite-direction lateral incursion scenario. **a)** A steep, unprompted incursion by the other vehicle leads to the model swerving towards the opposite lane. The sub-panels visualize the dynamics of both vehicles' velocity v, acceleration a, steering angle δ and (for the ego vehicle) the components of the pragmatic value G_{prag} (scaled with the evidence accumulation gain λ) that underlie the accumulated evidence towards a full policy re-plan. **b)** In response to a medium incursion by the other vehicle, the model attempts to avoid collision by braking and steering left, but collides at $t = 5.8\,s$ (crossed circle marker indicates the location of the collision). **c)** The likelihood of the ego vehicle successfully avoiding the other vehicle by steering left (toward the opposite lane) or right (toward the shoulder), or failing to avoid a collision. This is based on 64 model simulations and 21 to 22 human demonstrations for each of the three incursion types. **d)** Brake and steer response times. In panels C and D, human behavior and response times measured in a driving simulator study [244] are shown for reference. Video replays of the example simulations are available in the Supplementary Material.

lanes when the computer-controlled vehicle unexpectedly steered toward the participant's lane along a predefined path. The study implemented three kinematic variants differing in incursion "steepness": (1) a *steep* incursion crossing in front of the participant at a sharp angle and allowing a relatively easy escape by steering toward the opposite lane; (2) a *medium* incursion at a shallower angle, heading directly toward the participant; (3) a *shallow* incursion only partially entering the participant's lane, making it possible to find an escape path by steering slightly toward the shoulder (the original study used left-hand traffic, so we are here avoiding directional terms to prevent confusion).

We re-implemented reversed (*i.e.*, right-hand traffic) versions of these three conditions in our setup (Figure B.2), enabling a direct comparison between human and model behavior under nearly identical conditions. Each of those three conditions was repeated 64 times.

An example of our model implemented in the steep incursion condition is shown in Figure 7.4**a**, while Figure 7.4**b** shows an example of the medium incursion condition. When looking at the example simulations, we can observe again that the model needs multiple re-plans before deciding on the final avoidance maneuver. Compared to the front-to-rear scenario, where this is likely caused by the model's bounded planning capacity, in the lateral incursion scenario this is due to the high uncertainty inherent in the predicted positions of the encroaching vehicle (top panels). In Figure 7.4**a** (steep incursion), a first re-planning is triggered at 4.2 s (first dashed line) but this does not significantly improve the collision component of the pragmatic value. Additionally, the model only initiates minor steering maneuvers and small accelerations/decelerations. Together, this suggests that the high uncertainty about the other vehicle's behavior prevents the agent from finding an evasive policy that is better (has lower EFE) than the current policy, instead delaying the decision, thus gaining time to see how the situation develops. With the collision risk unresolved after the first re-plan, surprise quickly accumulates toward a second re-plan at 4.8 s (second dashed line). By this time, the agents are closer, allowing less space for uncertainty to grow and enabling the model to identify a steering policy that mitigates the collision risk (after the second re-plan, the collision component of the pragmatic value no longer significantly contributes to surprise accumulation). The decision to swerve toward the left is a result of both the other vehicle having crossed sufficiently over to the right side to leave a gap for swerving, and the model's stronger preference for moving into the opposite lane over leaving the road to the right (similar to Figure 7.1). However, it can be seen in Figure 7.4**a** that this trajectory is not optimal. Namely, as can be seen in the top plot, the planned trajectory $\widetilde{X}_{\text{ego}}$ chosen during the second re-plan at $t = 4.6$ s (second dashed line) over-steers when returning to its own lane and consequently risks leaving the road. Thus, slight course corrections will likely be performed later (but this would occur outside the current simulation window).

Figure 7.4**b** shows an example of the medium incursion scenario where the model is unable to avoid a collision. Again, likely due to uncertainty about the other vehicle's behavior, the first re-plan at $t = 4.0$ s (first dashed line) mainly results in a sharp deceleration and minor steering towards the right (*i.e.*, towards the shoulder). While steering towards the right might seem suboptimal given the other agent's current path, it aligns with the influence of the norm-conditioned particle filter, which biases the model towards believing that the other agent will likely return to its original lane (for more details on this,

see our discussion of Figure 7.6**g** below). At the second re-plan (t = 4.6 s, second dashed line), the model still focuses mostly on continued braking. However, this only marginally improves the collision component of the pragmatic value, leaving the imminent collision risk unresolved. At the third re-plan at 5 s (third dashed line), the agent keeps braking to reduce impact severity. At this point, the vehicles, which are still on a direct collision course with nearly no lateral offset, got so close that a collision becomes nearly unavoidable. Consequently, even the fourth re-plan (t = 5.4 s, fourth dashed line) fails to find a satisfactory trajectory and thus a collision occurs shortly thereafter at t = 5.8 s.

We compared our model to the human data collected in the driving simulator study by Johnson *et al.* [244]. We closely replicated the scenarios of that study in our simulations (Section B.3.2), which allowed direct comparisons between the model and the data with respect to the chosen avoidance behavior and both braking and swerving response times (which were extracted using the same method as in [244], which is described in Section 7.4.1 as well). A key finding of Johnson *et al.* [244] was that the participants' evasive maneuvering patterns and collision outcomes were strongly determined by the different scenario kinematics in the the three incursion scenarios. In the steep scenario, most participants avoided collisions by steering toward the opposite lane, while in the shallow scenario, participants typically steered toward the shoulder, passing the other vehicle on the "inside". However, in the medium scenario, the majority of participants collided with the oncoming vehicle. Since the urgency (*i.e.*, time to collision at the initial steering of the oncoming vehicle) was constant across scenarios, the high collision rate in the medium case was attributed to greater uncertainty about the other vehicle ' s future path, leaving no clear escape route.

As shown in Figure 7.4**c**, our model reproduced these results, reliably avoiding collisions in the steep and shallow scenarios through steering respectively left (toward the center) or right (toward the shoulder). Interestingly, the model also reproduced the human propensity for collision in the medium scenario. Johnson *et al.* [244] suggested that, conceptually, a key reason why human drivers tended to collide in the medium incursion scenario is the high perceived uncertainty about the oncoming vehicle's future behavior which prevents the driver from finding a sufficiently certain escape path. As described above, the current model offers a detailed computational account for the possible mechanisms underlying this phenomenon: the wide spread (uncertainty) in the behavior predictions about the oncoming vehicle, as well as a bias towards expecting that it will return to its own lane due to the norm-conditioning of the beliefs, prevents the model to find an evasive policy in time to avoid collision (Figure 7.4**b**). An additional factor behind the observed collision rates was likely also the model's bounded planning capacity. Increasing the model's planning capacity – by increasing the number of evaluated policies in the cross entropy method tenfold – resulted in a significant drop in the model's collision rate (from 82.3% to 51.6% in the medium incursion scenario, and from 6.3% to 3.1% for shallow incursions).

Furthermore, Johnson *et al.* [244] found that braking and steering response times – measured from initiation of the incursion to the first reaction – were generally long, ranging around 3.5 s to 4.5 s. While braking response times did not vary noticeably with incursion level, steering response did decrease consistently from the *steep* towards the *shallow* scenario. Our model generally reproduced the range of response times, with closely

matched median values (Figure 7.4**d**), but did not capture the variation in steering response times (especially for the shallow incursion, the model typically steered at least half a second later). One possible explanation may be that many participants avoided collision in the steep scenario primarily by braking before making a final steering adjustment, delaying their recorded steering response. In contrast, the model relied more on steering for collision avoidance, requiring an earlier response. The underlying reasons for these behavioral differences between the model and human drivers are still unclear but may have to do with varying individual preferences for braking versus swerving. Further work is needed to see if these individual differences can be reproduced by the model by varying its velocity, lane change, and lateral control effort preferences.

7.3.3 Intersection scenario

To test the ability of our model to generalize to more complex scenarios and to unseen data, we evaluated it in an intersection scenario against human data that was not used for tuning model parameters. Specifically, we used human response data obtained in a driving-simulator study conducted in Guelph, Canada, by Ziraldo *et al.* [250] in the context of collision avoidance at intersections. This study implemented a right-turn-into-path scenario, where the participants approached an intersection with the right of way (green light). A computer-controlled vehicle then approached from the right side, on an intersecting road, and performed a right turn on red into the participant's lane. Right turns on red are allowed in US and Canada but the turning vehicle must yield; the participant's vehicle therefore had the right of way. Two variations of this scenario were implemented: In the right-turn-stopped (RS) scenario variant, the other vehicle initially waited at the stop line in front of the intersection and began to accelerate when the modeled agent's time-to-arrival was approximately 4 s. In the right-turn-not-stopped (RNS) scenario variant, the other vehicle was moving towards the intersection at a constant speed, and was controlled such that, under a constant-velocity assumption, it would cross the stop line when the modeled agent was roughly 1.5 s from entering the intersection. In both scenarios, there was a parallel lane to the left of the modeled agent, allowing it to steer left to avoid a collision. A detailed version of our implementation of these scenario variants can be found in Section B.3.3. While Ziraldo *et al.* [250] had each of their 52 participant face three of those scenarios, we only consider the first responses (26 per scenario type), as significant changes in response times and collision rates between repetitions indicated an anticipation of the conflict scenario. In total, we ran each of those scenario variants 96 times, varying the initial velocity of the modeled agent within the limits specified by Ziraldo *et al.* [250].

In the RS scenario (Figure 7.5**a**), after the other vehicle begins to accelerate at $t = 0.6\,\mathrm{s}$ (first dashed line), several time steps pass before the model considers the collision risk to be significant (as reflected in the collision penalty component of the pragmatic value). As evidence (*i.e.*, surprise) thereafter gradually accumulates, the model reaches the re-planning threshold at $t = 3.4\,\mathrm{s}$, selecting a policy that combines braking with a leftward steering adjustment. However, as there is still considerable uncertainty about the other vehicle's future actions – particularly since the other vehicle is already violating normative expectations by not yielding, reducing the bias introduced by norm conditioning – the inferred collision risk decreases only slightly. This necessitates an additional re-planning step one

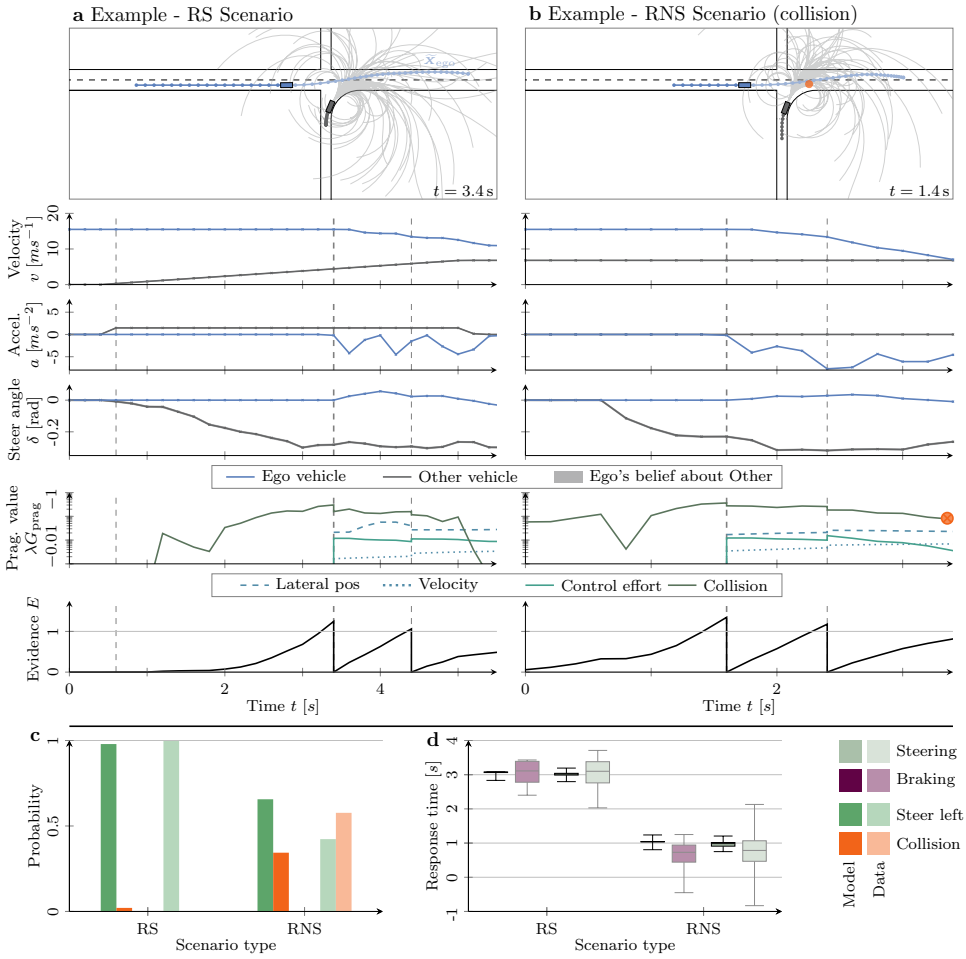


Figure 7.5: Evaluation of the model in the intersection scenario. **a**) Right-turn-stopped (RS): A vehicle initially waiting at the stop line performs a right turn on red without yielding; by accelerating and turning into the ego vehicle's lane it forces the model to steer left into the parallel lane to avoid a collision. The sub-panels visualize the dynamics of both vehicles' velocity v, acceleration a, steering angle δ and (for the ego vehicle) the components of the pragmatic value G_{prag} (scaled with the evidence accumulation gain λ) which constitutes the accumulated evidence towards a full policy re-plan. **b**). Right-turn-not-stopped (RNS): Another agent approaches the intersection with a constant speed, making a right turn into the model's lane without yielding or even slowing down, leading to a collision at t = 3.4 s (crossed circle marker indicates the location of the collision). **c**) The likelihood of the ego vehicle successfully avoiding the other vehicle (by steering to the left or braking) or colliding with it, based on 96 model simulations and 26 human demonstrations for each of the two scenario variants. **d**) Brake and steer response times. In panels C and D, human event outcomes and response times obtained in a driving simulator study [250] are shown for reference. Video replays of the example simulations are available in the Supplementary Material.

second later (third dashed line). Shortly afterward, the ego vehicle passes the other vehicle, resolving the conflict.

In the RNS scenario (Figure 7.5**b**), the situation becomes critical earlier, as the other vehicle is approaching at constant speed. Since the vehicle does not exhibit any observable braking to yield at the stop line, as would be normatively expected, the model considers it increasingly likely that the other vehicle will encroach into its path. As a result, surprise accumulates based on the increasing collision likelihood. At $t = 1.6\,\mathrm{s}$ (first dashed line), the ego vehicle initiates braking and applies a slight leftward steering adjustment. However, this response is largely insufficient to reduce the collision risk (the collision component of the pragmatic value does not change substantially). While the model generally predicts the other agent to make a right turn, the high uncertainty in this prediction – given the norm violations by the other vehicle, it is mostly based on kinematic likelihood – leaves no clear safe escape path. The reaction is also too late for braking (although it is attempted) to prevent a collision. Additional re-planning at $t = 2.4\,\mathrm{s}$ (second dashed line) likewise fails to avert the collision ($t = 3.4\,\mathrm{s}$). Had the model opted for a more extreme steering maneuver at the first re-planning point, it could have avoided the collision. However, in this specific simulation, the added pragmatic value of such an extreme maneuver was unclear at that point due to the high uncertainty about the other agent's future trajectory. Nevertheless, as the policy sampling process is stochastic, the model can occasionally opt for early steep steering and thus avoid collisions in this scenario.

Compared the model to human responses in the same scenario, we found that our model closely reproduced the main observations reported in [250]. As in the human data, the model successfully avoided most collisions in the RS scenario, but often collided in the RNS scenario (Figure 7.5**c**). The slightly lower collision rate of the model in the RNS scenario might be explained by the fact that in this scenario approximately ten percent of human participants collided without any noticeable reaction, which might indicate distraction effects that are not represented in our model. Next to collision rate distributions, our model reproduced the variation in human reaction times in the two scenarios (Figure 7.5**d**): responses in the more critical RNS scenario were markedly faster than in the RS scenario in both human data and model simulations. The model closely matched the timing of human responses in both conditions, with median reaction times generally not being apart more than 0.2 s. This is especially noteworthy, as it was achieved in an unseen scenario without any parameter tuning, indicating that the model generalizes to new scenarios and can account for human data that was not used for parameter fitting/tuning. Similar to the lateral incursion scenario (Figure 7.4**d**), the variances of model's reaction times are smaller compared to the human data. However, this is most likely simply a consequence of using a single model with fixed parameters for all simulations, thereby ignoring potential individual differences between human participants.

7.4 Evaluating individual model mechanisms

Comparison to the empirical data (Figures 7.3, 7.4, and 7.5) revealed that our model captured the key aspects of human collision avoidance performance in three different collision avoidance scenarios. To better understand which of the model mechanisms are essential for capturing the empirical observations, we evaluated the performance of six simpler models which systematically excluded the key mechanisms one-by-one (Figure 7.6).

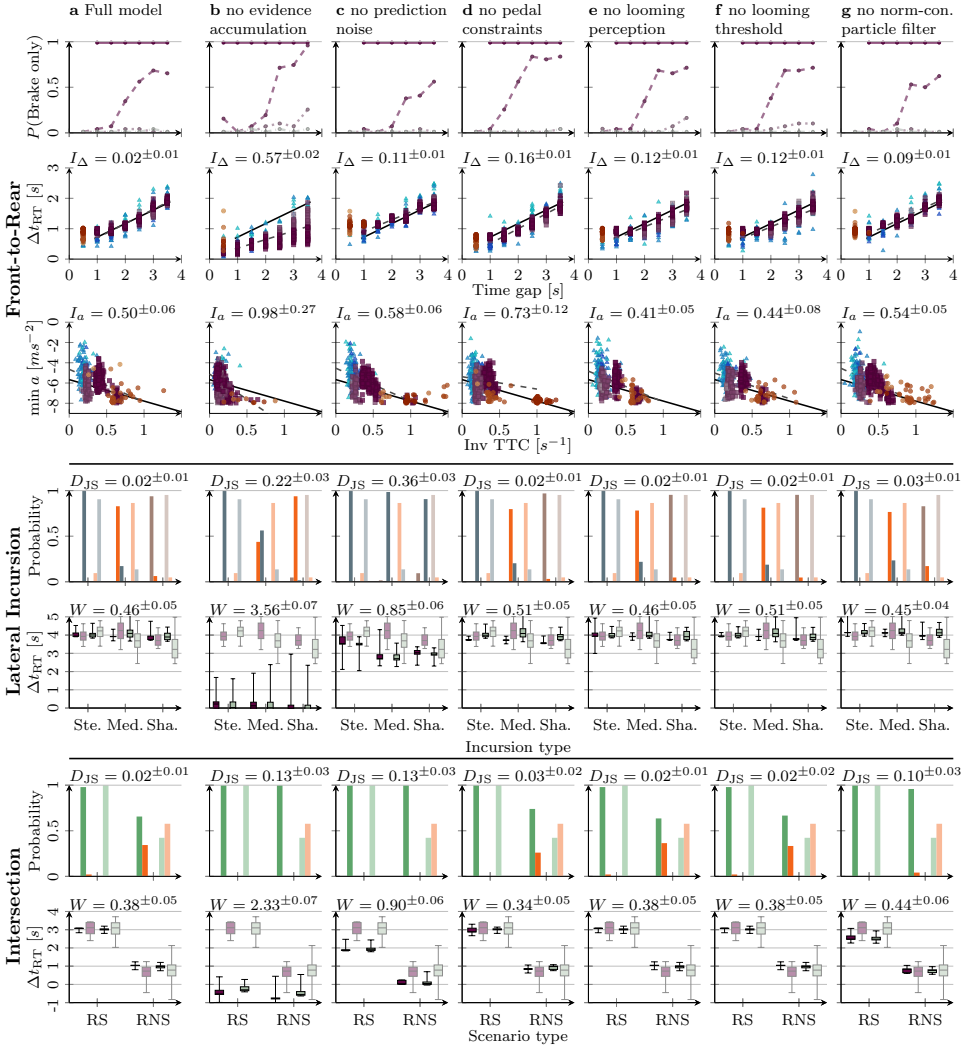


Figure 7.6: Evaluation of variants of the model that systematically exclude its key mechanisms in the front-to-rear scenario (top three rows, refer to Figure 7.3 for the legend), the lateral incursion scenario (middle two rows, with the legend shown in Figure 7.4), and the intersection scenario (bottom two rows; for the legend, refer to Figure 7.5). In the second and third rows, dashed gray lines depict the *maximum a posteriori* linear regression of the model simulation results, with the corresponding mean absolute error I between that line and the ground truth (I_Δ is given in s, while I_a is given in ms^{-2}). In the fourth and sixth row, we used the Jensen-Shannon divergence D_{JS} and in the fifth and seventh row the Wasserstein distance W (in s) to quantify the difference between the model and empirical distributions. For all metrics, expected values ± standard deviation are indicated.

7.4.1 Metrics

To better compare the performance of different model variants, we introduce a metric of the goodness of fit between model predictions and human data for each of the three scenarios.

Front-to-rear scenario

In this scenario, we used the vehicle trajectory and driver input data to extract brake response times and deceleration magnitudes. To that end, we followed Markkula *et al.* [213], extracting response times by fitting a piecewise-linear function to the recorded velocity data; the brake response time was then determined as the time instant when the first constant line switches to one with falling velocity. The slope of this line was used as the estimated deceleration magnitude.

When evaluating the goodness of fit of a model to the ground-truth human data, we used the mean absolute error I. Given a dataset $D_0 = \{(x, y) \mid x \in X_0, y \in Y_0\}$ and a linear approximation of the ground truth $y_{\text{data}}(x) = a_{\text{data}}x + b_{\text{data}}$ over the support interval $[x_0, x_1]$, we first discarded all sample pairs from D_0 for which $x \notin [x_0, x_1]$, resulting in D, after which we computed the residuals

$$E = \{y - y_{\text{data}}(x) \mid (x, y) \in D\}. \tag{7.15}$$

We assumed the residuals can be approximated by a linear function

$$e(x) = \alpha x + \beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma_E), \tag{7.16}$$

where σ_E is the empirical standard deviation of E. We placed Gaussian priors

$$p_0(\alpha) = \mathcal{N}\left(0, \frac{\sigma_E}{2\sigma_X}\right), \quad p_0(\beta) = \mathcal{N}\left(0, \frac{\sigma_E}{2}\right), \tag{7.17}$$

with σ_X denoting the empirical standard deviation of X. The posterior $p(\alpha, \beta \mid D)$ was then obtained using standard Bayesian linear regression [261]. Defining

$$I(\alpha, \beta) = \frac{1}{x_1 - x_0} \int_{x_0}^{x_1} \left|\alpha x + \beta\right| \mathrm{d}x, \tag{7.18}$$

we estimated the posterior mean $\mu_I = \mathbb{E}_{p(\alpha,\beta|D)}[I(\alpha, \beta)]$ and also the standard deviation $\sigma_I = \sqrt{\mathbb{V}_{p(\alpha,\beta|D)}[I(\alpha, \beta)]}$ using Monte Carlo sampling from $p(\alpha, \beta \mid D)$ with 10000 samples.

When analyzing response times (Figure 7.3**d** and second row of Figure 7.6), we set $x_0 = 0.9\,\text{s}$ and $x_1 = 3.6\,\text{s}$. Meanwhile, for the acceleration error (Figure 7.3**e** and third row of Figure 7.6), we set $x_0 = 0\,\text{s}^{-1}$ and $x_1 = 1.0\,\text{s}^{-1}$.

Opposite-direction lateral incursion scenario

In this scenario, we extracted brake and steering response times following Johnson *et al.* [244], by interpolating along the acceleration a_{long} and steering angle δ data to find the time at which $-1\,\text{ms}^{-2}$ and $0.0077\,\text{rad}$ (*i.e.*, a steering wheel angle of 5°) are exceeded, to extract brake and steering response times respectively.

When evaluating model fit of our model variants to the human data in this scenario, we used the Jensen-Shannon divergence (a measure of dissimilarity of two categorical distributions) to compare predicted outcomes and the Wasserstein distance (a proper metric) for reaction times. Given the relatively small size of the ground truth dataset, we applied nonparametric bootstrapping [262] with 10000 resamples to estimate the sampling variability of each metric.

Intersection scenario
To extract response times, here we followed the same approach as in the lateral incursion scenario. We only compared our model to the first collision instance shown to each participant in the experiment of Ziraldo *et al.* [250] to ensure that the underlying data represents natural reactions of human drivers. Importantly, the human steering reaction times were extracted based on steering wheel angles. However, as Ziraldo *et al.* [250] did not report how this relates to the steering angle of the vehicle (which our model works with), there might be some discrepancy here, as we had to rely on the relationship reported by Johnson *et al.* [244]. In this scenario we used the same bootstrapped metrics as in the lateral incursion scenario (*i.e.*, Jensen-Shannon divergence for collision probabilities and Wasserstein distance for response times).

Statistic significance
To determine whether one model variant provided a statistically significant better fit than another, we computed the signal-to-noise ratio of the difference between models, defined as the mean difference in metric values divided by its estimated standard deviation (using sampling from posterior distributions or bootstrapping as described above). A signal-to-noise ratio exceeding 3 was considered indicative of a meaningful difference, following standard statistical heuristics [263]. Here, the difference is computed as the mean metric difference between models, and under the assumption of independence, the standard deviation of the difference is approximated as the square root of the sum of the individual variances.

7.4.2 Removing evidence accumulation

Without evidence accumulation, the model fully re-plans its policy at every time step. Consequently, it starts reacting to changes in the other vehicle's behavior immediately after detecting them. This resulted in a drastic reduction in response times across all scenarios (Figure 7.6**b**). Additionally, the model without evidence accumulation predominantly failed to avoid collisions in the shallow incursion scenario, contrasting sharply with human data and the full model. This was mostly caused by the model initiating braking very early. Given the uncertainty about the other vehicle, neither steering left nor steering right allows safe escape paths, so the agent continues braking, coming to a complete stop. At the point in time where the certainty about the other agent is low enough for safely choosing either left or right, the agent can physically no longer avoid a collision (as it cannot move laterally when stopped). An example simulation of this scenario can be found in the supplementary material. The fact that collisions happened mostly in the shallow incursion scenarios is likely because in the medium and steep incursion scenarios, premature braking leaves enough space in front of the ego agent for the other vehicle to move

across, which is not the case in the shallow incursion. In this scenario, a collision would still occur if the ego agent stayed in its starting lateral position, which makes braking an insufficient avoidance mechanism. Finally, in the intersection scenario, immediate reaction (including both braking and steering) allowed the reduced model to avoid all RNS collisions, in contrast to both the full model and the human data. Based on those observations, the inclusion of evidence accumulation is critical for accounting for empirical data on human response times in all three scenarios, evasive maneuver selection in the shallow lateral incursion scenario, and collision rates in the intersection scenario.

7.4.3 Removing added noise in behavior prediction

The model that uses deterministic predictions about the other vehicle produced response times in the front-to-rear scenario that are only slightly longer than for both the full model and the human data (Figure 7.6**c**), especially given shorter time-gaps. By contrast, in the opposite-direction lateral incursion and intersection scenarios, removing the noise led to substantially shorter response times. The reason for this apparent discrepancy is that in the front-to-rear scenario, prediction noise results in ego agent's believing that the other vehicle may brake more aggressively, prompting the model to react earlier. In the other scenarios, however, removing the prediction noise decreases the ego agent's uncertainty about the other vehicle, which enables it to find a safe escape path earlier, leading a faster response.

In the lateral incursion scenario specifically, this leads to earlier steering responses, which, given how these particular scenarios unfolded, turned out be a more effective avoidance strategy, as evidenced by the absence of collisions in this case. However, this strategy was only more effective after the fact (*i.e.*, with hindsight of how the scenario played out). In another counterfactual scenario, where the oncoming vehicle turns back into its own lane early, swerving could instead result in a collision (see [244] for a discussion of this important point). Prediction noise affected not only response times but also the model's choice of evasive maneuver in the lateral incursion scenario. In particular, the model with deterministic predictions predominantly opted to avoid the other vehicle by steering left regardless of the incursion profile. This allowed the model to avoid collisions in all simulations, but is not consistent with the human data.

Similarly, in the intersection scenario, this model ablation successfully avoided all collisions as well. Taken together, these findings underscore that behavior prediction noise is crucial for accurately reproducing human behavior, particularly in the lateral incursion and intersection scenarios. Specifically, uncertainty in the predictions of the other vehicle's actions seems to be a key explanation for why most human drivers ended up in a collision in the medium lateral incursion scenario and the RNS intersection scenario (as discussed above in relation to Figure 7.4**b** and Figure 7.5**b**, respectively).

7.4.4 Removing pedal constraints

The model not constrained by pedal switching delays resulted in approximately 0.2 s faster brake response times in all three scenarios compared to the full model (Figure 7.6**d**). This matches the delay of 0.2 s implemented to represent the switch of the driver's foot from the gas pedal to the brake pedal. Although the effect appears simple and straightforward, the implemented pedal constraints are essential for capturing the differences between steer-

ing and braking response times observed in human behavior. For example, in the lateral incursion scenario, our model still slightly overestimates steering response times while underestimating braking response times, a discrepancy that could not be corrected by adjusting the surprise threshold for re-planning, as this would shift all response times uniformly.

7.4.5 Removing looming perception

We analyzed a version of the model which directly perceived kinematic variables without relying on looming (Figure 7.6**e**) as well as one which perceived looming but did not have a looming perception threshold (Figure 7.6**f**). In the front-to-rear scenario, both these models differed from the full model and the human data in that they produced faster brake response times at longer time gaps. For the model without the looming threshold, this is due to the fact that greater time gaps (and therefore greater distances) require a larger velocity difference to surpass the looming detection threshold, which delays responses, as it needs more time to accumulate. Furthermore, these results being very similar to the model that did not perceive looming at all (Figure 7.6**e**) indicates that in this scenario, the looming threshold was a key factor behind the ability of the model to capture human response times. Additionally, while not specifically analyzed here, this effect is expected to be exacerbated in high speed freeway scenarios where (given the higher speed) the following distances for a given time gap increase, further emphasizing the importance that the model accounts for this phenomenon. In the other two scenarios, removing the looming threshold (or looming perception entirely) did not have any major effects, as the relevant observations in those scenarios are based on lateral movements.

In summary, models without either looming perception or looming threshold resulted in qualitative differences between the model and human data on brake response times in the front-to-rear scenario. In addition to the strong theoretical support, this highlights the importance of both looming-based perception and looming perception threshold for scenarios with largely longitudinal dynamics.

7.4.6 Removing the norm-conditioned particle filter

Without the norm-conditioned particle filter, the model behavior in the front-to-rear scenario changes only slightly compared to the full model (Figure 7.6**g**), showing marginally longer response times. In the lateral incursion scenario, however, the effect is more pronounced, with collisions being less likely in the medium incursion (avoided by leftward steering) and more likely in the shallow incursion. Relative to the full model, this ablation lacks the bias against predicting steeper incursions earlier, which incentivizes steering towards the left. This yields a small advantage in the medium incursion scenario, where avoiding towards the other vehicles lane is typically effective, but increases collision risk in the shallow incursion, where both the full model and humans consider rightward avoidance the better option. Additionally, the ablated model also shows slightly longer reaction times in the lateral incursion scenario. This again follows from the lack of norm conditioning, as without it, the model often predicts that the other vehicle will leave the road before posing a collision risk, whereas the full model – biased by the normative filter – believes the other vehicle will stay in its lane. Consequently, in this ablation, with the predicted collision likelihood being lower, surprise accumulates more slowly, and the re-planning

threshold is reached later, leading to a slower reaction. In the intersection scenario, removing the norm-conditioned particle filter leads to both faster reaction times and a successful resolution of the conflict in nearly every instance. This is logical, as the expectation that the other vehicle would stop to yield resulted in delayed reaction. Without this bias, potential collisions are taken more seriously and reacted to earlier.

While these differences already highlight the importance of the norm-conditioned particle filter, it has the strongest impact in benign (non-conflict) scenarios. In such scenarios, the model with the simple kinematics-based behavior prediction mechanism is unrealistically conservative due to having to plan for all kinematically possible futures of the other vehicle – a problem that is avoided by the model with norm-conditioned particle filter. An example of this is discussed in Section C.1.

7.4.7 Further analyses

In addition to systematic ablations of the six key mechanisms, we investigated a model variant that ignored the epistemic value when selecting policies. This ablation, however, did not result in any meaningful differences in the three scenarios considered. Nevertheless, the epistemic value does play a role in scenarios where, for instance, observability is limited; there, epistemic value has substantial effect on model behavior (Supplementary Material C.2), in line with previous observations [199]. For this reason, the full model reported here does include epistemic value as one of its key components.

Additionally, in the lateral incursion scenario, we examined the impact of our model's bounded reasoning capacity. Specifically, as noted above, increasing the number of policies evaluated during replanning led to a substantial reduction in observed collision rates —ranging from roughly one-third to one-half across scenarios. This suggests that bounded reasoning is a crucial component for accurately modeling human collision-avoidance failures. However, since not all collisions were avoided, other factor (namely, prediction uncertainty and the expectation of norm-compliant behavior) also play a significant role in driving such outcomes.

Overall, the six ablated model variants analyzed above show qualitative differences compared to the full model, and none reproduce human behavior across all three scenarios as well as the full model. This provides further evidence for the importance of all the analyzed model mechanisms in regard to capturing human collision avoidance behavior.

7.5 Parameter sensitivity analysis

It was not the purpose of this study to obtain the closest possible fit to human data, so no exhaustive parameter optimization was performed. To analyze the impact that changes of model parameters have on its performance, we conducted a sensitivity analysis (Figures 7.7 and 7.8). Specifically, we varied the thirteen tuned parameters from Tables 7.1 and 7.2 one at a time, while keeping the other parameters fixed. For most parameters, this explored deviation by roughly one order of magnitude, if not otherwise constrained. For each of the new parameter sets, we reran all the simulation in the three scenarios described in the Results section, and calculated the goodness-of-fit metrics (Figure 7.6). Using the methods described in the previous Metrics section to estimate the uncertainty of those metrics inherent in the limited number of simulations and human

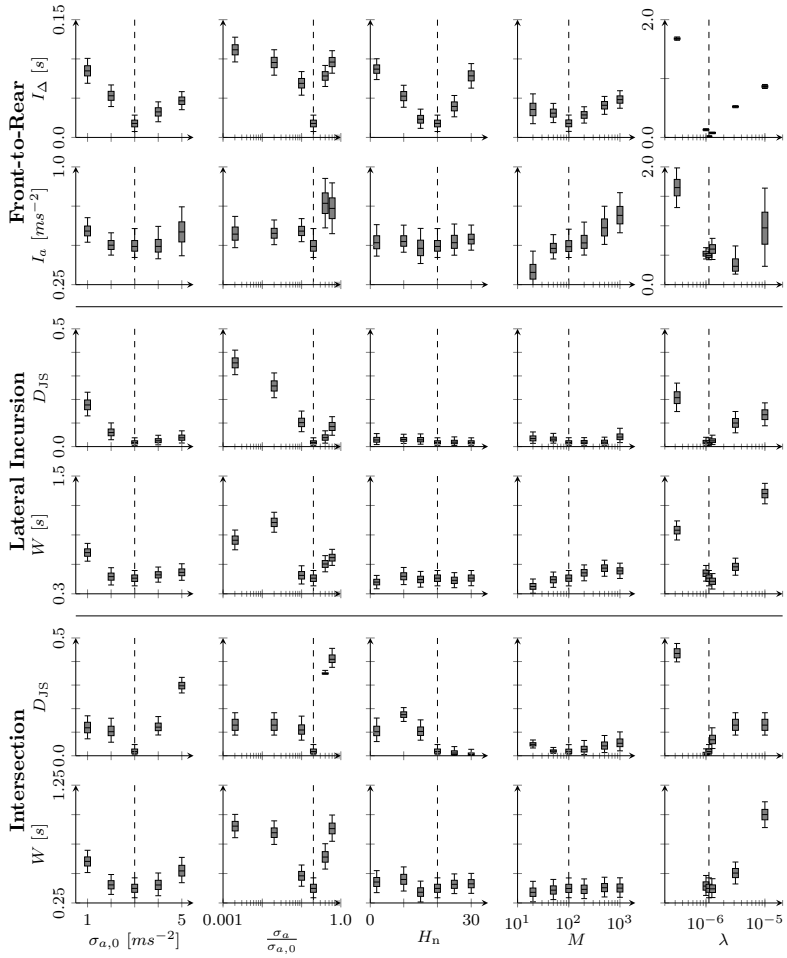


Figure 7.7: Sensitivity analysis for selected model parameters related to the state transition function and policy selection (Tables 7.1 and 7.2). The vertical dashed lines indicate the final values used in the full model. We generally used a scheme of only changing one parameter at a time. However, when changing $\sigma_{a,0}$ (middle column), we changed $\sigma_{\omega,0}$ with the same factor. Meanwhile, in the third column, we ensured that $\frac{\sigma_a}{\sigma_{a,0}} = \frac{\sigma_\omega}{\sigma_{\omega,0}}$, while keeping $\sigma_{a,0}$ and $\sigma_{\omega,0}$ fixed. In the box plots, the lower and upper whiskers indicate the 5th and 95th percentile, respectively. For λ in the rightmost column, note the change in y-axis labels for some metrics.

demonstrations, we express those in the corresponding box-plots.

The most consequential parameter is the evidence drift rate λ (Equation (7.12)), where moderate deviations can lead to large changes in reaction times and therefore substantially worse goodness of fit (the goodness of fit metrics I_a and W in Figure 7.7 can degrade on the order of seconds). Model behavior also deteriorates, as seen in the Jensen–Shannon divergences. This is expected, as for example a sufficient increase in λ to trigger re-planning at (nearly) every timestep would in principle be equivalent to removing the evidence accumu-

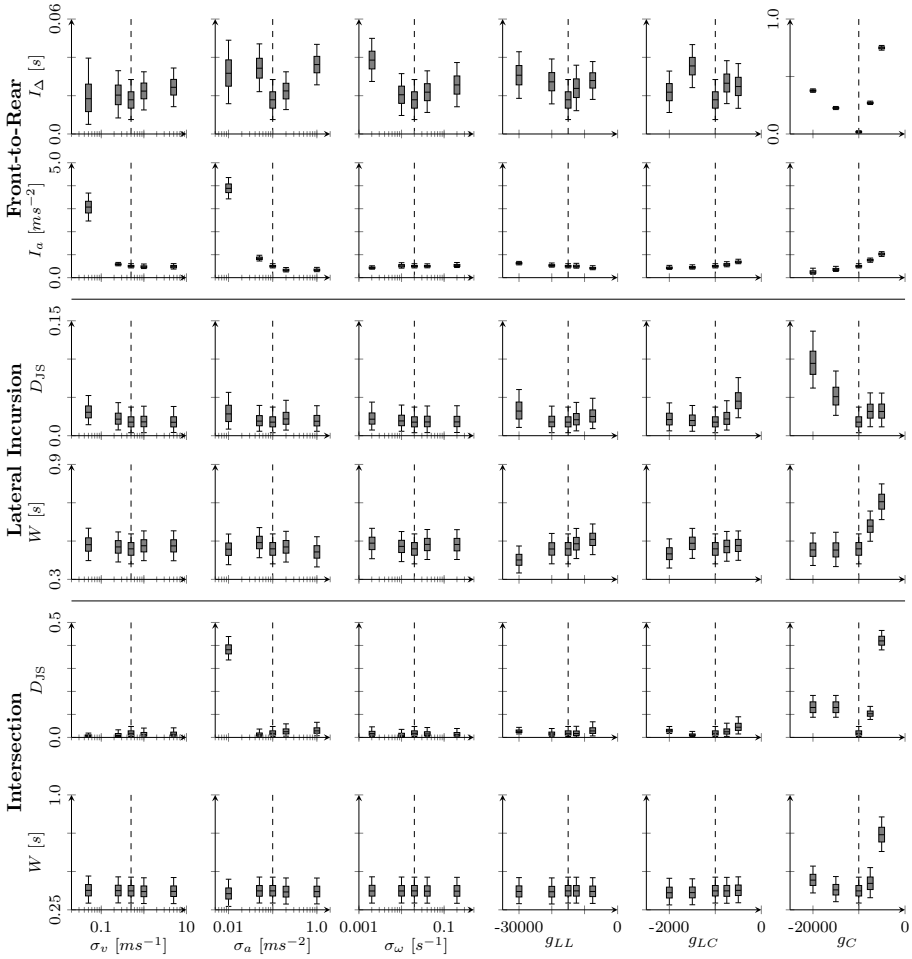


Figure 7.8: Sensitivity analysis for selected model parameters related to the preference function (Table 7.2). The vertical dashed lines indicate the final values used in the full model. Similar to Figure 7.7, we only changed one parameter at a time; the lower and upper whiskers in the box plots indicate the 5th and 95th percentile, respectively. While the metrics used are the same, the scales of the y-axes are different than in Figure 7.7 for illustrative purposes. For g_C in the rightmost column, note the change in y-axis labels for I_Δ.

lation mechanism completely from the model, which was shown to be disadvantageous for reproducing human behavior in Figure 7.6**b**. However, our chosen parameter value appears to involve a slight trade-off. In the lateral incursion and intersection scenarios, small increases in λ can improve the accuracy of reaction times while worsening outcome prediction, whereas small decreases show the opposite trend. Meanwhile, the other hand-tuned policy selection parameter, the number of simultaneously evaluated policies M, had even weaker impact (Figure 7.7). There, for I_Δ and D_{JS}, increasing or decreasing M negatively affects the goodness of fit, but there was no evidence that these differences were statistically significant. Meanwhile, for I_a and W, lower M values did yield better fit to

the data, but the improvements were marginal.

Parameters of the state transition function did have a major effect on model behavior (Figure 7.7). For $\sigma_{a,0}$ and $\sigma_{\omega,0}$ (the noise assigned to the other vehicles' control actions in the generative model's state transition function), we generally observed a U-shaped response across all considered metrics, indicating that deviations from the chosen values led to poorer fit to the data, although not all differences were statistically significant. Similarly, σ_a and σ_ω (the corresponding noise in the policy roll-out during behavior prediction, second column) chosen for the final model generally yielded best goodness-of-fit. The horizon for calculating the projected normative probability H_{n} was less influential: it only impacted model behavior in terms of front-to-rear reaction times (I_Δ) and the prediction of collision outcomes in the intersection scenario (D_{JS}).

For the parameters of the preference function $p(\boldsymbol{o})$ (Equation (7.11)), two primary effects can be observed (Figure 7.8). First, the choice of σ_v and σ_a is important for producing a reasonable fit for the deceleration magnitude applied in the front-to-rear scenario (I_a in Figure 7.8). Second, the choice of collision cost g_C is critical, with slight deviations from the selected value leading to substantial decreases in model performance (Figure 7.8). Consequently, fine-tuning of this parameter is of high importance. Varying the remaining parameters did not result in statistically significant changes, suggesting that their coarse calibration is sufficient. However, this is somewhat expected, as the initial trajectories of the ego agents often are aligned with the preferences expressed in those parameters, which mostly influence the specific avoidance maneuver chosen, but not the specific reaction times. This can be seen in the specific metric expressing such behavior (I_a), where especially σ_v and σ_a have a significant impact.

Especially notable is that – with parameters tuned only on the first two scenarios without any exposure to the intersection scenario – the chosen parameters perform similarly well in the intersection scenario, suggesting good model generalizability. Taken together, this parameter sensitivity analysis indicates that 1) most of the sensitivity of the model is primarily concentrated in λ and g_C; 2) most parameters are robust to minor perturbations; and 3) while not necessarily optimal, our final parameter settings achieve good fit to human data, with no parameter variations leading to substantially better goodness-of-fit across all three scenarios.

7.6 Discussion

Building on previous work [199], we here presented an active inference-based framework for modeling human collision avoidance behavior based on first principles. To our knowledge, this is the first published computational model that offers a detailed account of human collision avoidance behavior across multiple conflict scenarios. Our model reproduces many key results in the rich literature on collision avoidance in front-to-rear scenarios, in particular the dependence of response times and braking magnitude on scenario kinematics. The model also generated several detailed predictions on how evasive maneuver decisions (braking vs. swerving) vary with scenario kinematics which can be tested in future experiments. Furthermore, we compared the model to detailed human behavior data in an opposite-direction lateral incursion scenario with varying kinematics [244] and an intersection scenario with with a right-turning vehicle failing to yield the right of way [250], both obtained in driving simulator studies. This analysis demonstrated that

the model can reproduce human evasive maneuver decisions, collision outcomes, as well as response times in different variations of these two scenarios. The model also shows strong generalization across scenarios, with the model tuned solely on the front-to-rear and lateral incursion scenario making accurate predictions of human responses in the intersection scenario.

7.6.1 Modelling collision avoidance behavior

A key novel feature of our model is its ability to flexibly represent closed-loop collision avoidance through dynamic re-planning of policies, including braking, steering, and accelerating actions, in a way that is qualitatively and quantitatively similar to human collision avoidance behavior. In contrast, existing publicly available collision avoidance models either implement a "one-shot", often predetermined, open-loop evasive maneuver [20, 208, 264] or limited aspects of closed-loop control, such as intermittent braking but no steering [53, 192]. Some commercially available models – the Stochastic Cognitive Model (SCM) from BMW [194, 195] and the driveBOT model from cogniBIT [196, 197] – reportedly implement human-like closed-loop behavior in collision avoidance scenarios. However, although the SCM is openly available, the published evaluations provide only a limited basis for meaningful comparison. For instance, Fries *et al.* [195] evaluates SCM on only a few collision-avoidance cases and compares mainly velocity-over-time profiles, without a systematic analysis of diverse metrics across scenario variations. For driveBOT, the evaluations are more comprehensive [196, 197], but the lack of available information on the detailed working principles of the model precludes a meaningful comparison with our work.

Our model achieves closed-loop behavior by continuously evaluating evasive policies based on their expected free energy, pursuing the policy with the lowest EFE, and allowing for further re-planning if the chosen policy turns out to not yield the preferred (and hence expected) outcomes (*e.g.*, due to limited planning accuracy or unexpected changes in how the scenario unfolds). This, in effect, implements a constraint satisfaction mechanism where different model preferences are traded against each other in finding the policy with the highest overall combined pragmatic value and epistemic value. We observed several examples of this (Figures 7.3) where, for example, the model generally prefers to brake at low speeds (since the loss related to the preferred speed is not as high as the cost of changing lane) and swerve at higher speeds (where the cost of braking in terms of speed loss are higher). This yields detailed predictions of human behaviors in different kinematic situations that could be further tested empirically.

Another important novel aspect of our model is its ability to dynamically account for uncertainty about the future behavior of other agents. This is particularly important in situations like the lateral incursion scenario and the intersection scenario. For instance, in the former, the driver cannot determine *a priori* if the other vehicle will continue crossing the ego vehicle's lane or if it will turn back to its original lane, resulting in a wide range of potential future trajectories of the other vehicle. However, the uncertainty may be reduced as the scenario unfolds, opening up new available escape paths, or *escape affordances* [244], as was the case in the current steep and shallow incursion scenarios. In such cases, an evasive response that is initiated too early may be premature (and non-optimal) and does not properly reflect human behavior. We saw an example of this in Figure 7.6**c**, where in

both the lateral incursion and intersection scenario the model without prediction noise responded much faster than humans and the full model, thus avoiding collision in all cases. However, such overconfident evasive maneuvering behavior could be detrimental in other scenarios, for instance if in our lateral incursion scenario the other vehicle would turn back into its original lane. By representing the uncertainty about the other vehicle's future positions using a vehicle dynamics (bicycle) model and a noisy particle filter, our model is able to handle such situations and successfully reproduce human behavior.

Furthermore, in particular the intersection scenario showed that it is also important that such noisy predictions are biased towards norm-compliant behavior (Figure 7.6**g**). Specifically, without the prior expectation that other vehicle will follow established rules and norms such as yielding the right of way, the model will predict potential collision scenarios earlier, react faster and resolve the conflict safely. However, in many situations this leads to overreactive non-human like collision avoidance behavior. We also demonstrated that the *norm-conditioned particle filter* is necessary to avoid overly conservative (and therefore non-human like) behavior in non-conflict driving scenarios (Section C.1).

In the collision avoidance scenarios simulated in the present study, there is little opportunity for the driver to reduce the uncertainty about the oncoming vehicle's future trajectory through epistemic actions. In other words, there is little the driver can do to obtain further information that could help predict what the other vehicle will do. Thus, as expected, disabling the epistemic value component of the EFE did not significantly change the model behavior. However, as shown in Section C.2 and previous work [199], the epistemic value component in our model does drive policy selection in traffic scenarios with epistemic affordances, that is, situations with opportunities for actions yielding information that may resolve uncertainty. Hence, even if the epistemic value did not have any significant effect on policy selection in the present study, it should still be considered a key aspect of human road user behavior. In particular, in the real world communicative acts such as honking, flashing headlights or initially moving to the right in the lane could reduce uncertainty about the other vehicle driver's intent. Since other agents were here assumed to be non-reactive, such aspects were not addressed in the current work; this is a topic for future development of the model.

While active inference models can be implemented computationally in many different ways, we used a stochastic receding-horizon control architecture as the basis for the implementation, following [199]. This shares similarities with prior work in robotics, particularly decision-theoretic approaches that explicitly account for uncertainty in actuation and sensing by propagating belief states into the future, often using particle-based representations, and then selecting actions that optimize expected outcomes based on this projected belief evolution [265–267]. However, our work diverges from these engineering models in two crucial aspects. First, since our goal is to represent human driver behavior rather than designing a robot/vehicle controller, the modeling focuses on capturing key aspects of human driving, such as response timing and evasive maneuver decision making, rather than optimizing performance based on engineering criteria. Second, our modeled agent minimizes EFE (as prescribed by the active inference framework), while traditional state-based reward functions often employed in the control literature lack such a theoretically-grounded objective derived from first principles.

Our model incorporates an explicit mechanism for evidence accumulation to account

for human response timing, based on existing models [46, 47, 132, 191, 209]. Evidence accumulation models naturally account for the situation-dependency of response timing in traffic situations: the faster a traffic conflict escalates, the faster the human's response to it [46, 209, 213]. Compared to existing evidence accumulation models, our model introduces some key novel aspects. In particular, in contrast to traditional models that accumulate perceptual evidence (*e.g.*, looming [191]), our model rather accumulates *surprise*. Similar ideas have been explored in existing models [54, 192, 209, 243], based on the notion that a traffic conflict by definition is an extremely rare (and hence surprising) event which always takes place against a "default" expectation of how the situation would normally play out (Figure 7.1). This is also the key idea behind the NIEON (Non-Impaired driver with their Eyes ON the conflict) response time model which conceptualizes the onset of the stimulus that human drivers respond to in a traffic conflict as the onset of surprise (*i.e.*, the violation of the initial default expectation) [209]. The current model generalizes this idea in several novel ways. First, instead of triggering a predefined action such as braking when the surprising evidence has reached the threshold, the evidence accumulation here triggers the *re-planning* of a new policy, using the EFE-based policy selection mechanism discussed above. Second, the surprise signal that is being accumulated is not just the difference between a predicted and actual signal but rather the negative pragmatic value of the current policy, that is, how "bad" the currently selected default policy is relative to the agent's preferred observations. This provides a straightforward solution to the known problem of how to determine whether a given surprise signal is *relevant* to the agent [243].

Importantly, our model accumulates surprise deterministically without accumulation noise. Under this assumption, the accumulation rate can be expressed in units of the decision boundary (another key parameter traditionally important in the evidence accumulation literature, interpreted as response caution) and hence we did not need to define a separate decision boundary [240]. Nevertheless, adding such accumulation noise could be seen a future improvement of our model if we want to better match the high variance observed in human reaction times (Figure 7.4**d** and Figure 7.5**d**). Similarly, the addition of surprise decay could also be considered.

While we here model evidence accumulation explicitly, it can in principle be seen as an implicit feature of active inference [268]. However, in practice, it is challenging to accurately represent human response dynamics implicitly in a simplified model like ours. To enable straightforward tuning of the model's response performance to human data, we chose the current explicit modeling of evidence accumulation. As demonstrated by our simulation results (Figure 7.6**b**), the inclusion of this mechanism is essential to account for realistic response times in our model.

7.6.2 Active inference as a model of human behavior

The current model is based on the general model predictive control architecture originally developed for the model of routine driving described in [199] which, in turn, is based on the standard discrete-time active inference EFE model for planning agents formulated as a partially observable Markov decision process [57]. The original routine driving model in [199] focused primarily on the role of epistemic value in resolving uncertainty when driving around visual occlusions and during visual time sharing (drawing from the rat-in-a-T-maze example in [57]). However, that model was not aimed at describing time-critical

behaviors. For instance, in collision avoidance scenarios its responses to sudden stimuli would be overly fast, similar to the version of our model without evidence accumulation (Figure 7.6**b**). Extending the original model [199] with not only evidence accumulation, but also other mechanisms such as looming-based perception and norm-conditioned predictions was essential for capturing human collision avoidance behavior (Figure 7.6). Otherwise, the model we reported here retains the key features of the original routine driving model needed for dealing with uncertainty resolution through epistemic action, thus offering a powerful and general computational framework for modeling driving behavior, also beyond collision avoidance. As in [199], we employed engineering tools like the cross-entropy method and particle filters with discrete and continuous variables. However, while relying on such established methods, our model does not reduce to a combination of them; instead, these methods serve as means for implementing specific cognitive mechanisms within our active inference framework.

Our framework is grounded not only in active inference and the free energy principle [218], but also enactivist approaches to cognitive science [269, 270]. Active inference and enactivist cognitive science share the the foundational notion that behavior in biological agents results from a self-organizing process geared towards sustaining the agent's integrity (existence) over time. This emphasizes a strong continuity between life and mind [270, 271] and offers a natural solution to the problem how the certain aspects of the environment become *meaningful* and *relevant* to an agent [272]. Such an *existential imperative* can be described as the organism striving to maintain itself in a sparse attractive set of preferred agent-environment states, where the attractive set depends on the specific organism in question. Thus, to survive over time, a fish needs to remain in water with the right chemical balance and temperature range, obtain food, avoid being eaten by certain predators etc. This implies that these particular states are meaningful and relevant to the fish which then acts in order to observe these states with a high degree of certainty, thus looking as if it is minimizing free energy over time. By the same token, we endowed our model with a (highly simplified) set of preferred states that are meaningful and relevant to a human car driver (avoid danger of collision, stay on the road, keep up progress, avoid too harsh braking/steering) and defined the model such that it acts (selects and executes policies) in a way that maximizes the chance of observing these preferred states.

Related to this, our model also offers a novel operationalization of the classical notion of *affordance*, traditionally broadly defined by Gibson [227] as what the environment "*offers the animal, what it provides or furnishes, either for good or ill*" (p. 127). Thus, when our model identifies candidate evasive policies associated with low EFE, these can be understood as affordances in the sense of opportunities for actions that are worth pursuing. The active inference framework offers a further distinction between *pragmatic affordances* yielding unsurprising, familiar and preferred observations and *epistemic affordances* yielding information that can resolve uncertainty about pragmatic affordances [199, 273, 274] (see [244]) for a further discussion about this expanded notion of affordances and its relation to the classical affordance concept in ecological psychology).

Thanks to generality of active inference, models based on it have been developed for virtually every facet of biologically-based cognition and behavior, in organisms ranging from single cells [271], plants [275], individual animals and humans [276] to social and cultural phenomena [277], which provides a great source of concepts and ideas for modeling

different aspects of human road user behavior. Thus, importantly, we see active inference not just as a computational approach to model development, but as a general theoretical framework for understanding road user behavior that can guide modeling at a conceptual level and generate novel hypotheses for experimental studies. At the same time, the great majority of existing active inference models use relatively simple toy scenarios, thereby limiting its applicability to real-world human behavior. By operationalizing active inference in the context of a dynamically rich task, that is, collision avoidance and evaluating it against human data from both simulated and real-world driving, this study pushes the boundaries of active inference applications toward the modeling of complex human naturalistic behaviors.

7.6.3 Limitations and future work

While, as discussed above, our model represents a significant advancement in human collision avoidance modeling, it has several limitations that can be addressed in future work. First, in this work we essentially hand-tuned the model parameters to fit the human data from the two collision avoidance scenarios. Whereas this highlights the model's generalizability and interpretability, especially considering its good performance on the third (intersection) scenario not used for tuning, such an approach is impractical at scale. Future research should explore systematic parameter optimization methods, for example using the approach suggested by Wei *et al.* [198]. However, even with better optimization methods, the reliance on reaction times to evaluate model performance could be a limiting factor. Specifically, outside well controlled experimental settings, any definition of reaction time is likely somewhat arbitrary, as the stimulus onset that the driver has to respond to is often imprecise. Similarly, in complex multi-agent scenarios, it may be difficult to match a model's observed reaction to a certain stimulus, as there may be multiple stimuli that could have triggered the response. Thus, future work should explore alternatives that are more applicable to real-world data, while still allowing for an evaluation on how well the model captures the mechanisms underlying human responses.

Second, the current model assumes that other agents are non-reactive (*i.e.*, in the generative model's state transition function, other agents' states are independent from each other and the ego agent). This assumption therefore limits the model to scenarios where interactions are only weakly linked, and is unlikely to hold in the general case. For example, norms about priority – such as at unsignalized intersection – are generally interactive, with one agent ' s norm compliance depending on the other agents ' behavior. Thus, future work should extend the model to include reactive agents by implementing interactive norms and communicative or epistemic actions that allow agents to probe or signal intentions. Such extensions would strengthen the model's applicability and allow simulations of multi-agent interactions under active inference. Moreover, the current model (as well as the original routine driving model in [199]) only reasons at a shallow "unsophisticated" level about how actions bring about future observations with varying degrees of uncertainty. Further work could explore the possibilities of incorporating *sophisticated inference* [235], which enables "deeper" levels of recursive reasoning about how future observations would lead to updated beliefs and corresponding new actions. This may be particularly relevant when modeling interactions between road users.

Third, the model currently only accounts for perceptual limitations in terms of visual

angle and looming. However, looming only represents the relative motion of objects moving longitudinally towards the observer and, strictly speaking, only applies to objects located along a straight path of travel relative to the gaze direction of the observer (in our case, this means objects located straight ahead of the ego vehicle). Thus, additional perceptual variables are needed to represent perceptual thresholds on the detection of perpendicular motion (*e.g.*, a pedestrian moving into the street ahead) or representing relative motion of objects that are otherwise perpendicular from the observer's gaze and heading direction. In addition, the perceptual effects of environmental and stimulus-related factors such as weather (*e.g.*, reduced visibility due to fog, heavy rain or snow), lighting conditions (daylight, dusk, night, effects of headlights etc.), glare and stimulus conspicuity would ideally need to be accounted for. Another way in which the model's perceptual fidelity could be improved is by explicitly modeling the human field of view, including the different functional properties of foveal versus peripheral vision which can be combined with the visual behavior model outlined in the original model [199].

Beyond perceptual limitations, the use of more realistic perceptual variables such as the optically specified longitudinal time-to-collision (τ in [226]), bearing angle (specifying perpendicular collision course [278]) and global optic flow rate or edge rate (specifying self motion relative to the ground [279, 280]) could be further explored. This would allow for defining observation and state spaces in the model which are more aligned with the ecological information available to human drivers.

Fourth, while mechanisms for dealing with partial observability during occlusions and visual time sharing were explored in [199], further work is needed to develop more comprehensive solutions for representing and reasoning about multiple agents hidden behind occluding objects or appearing outside the driver's field of view (with initial work being done in Wei *et al.* [281]); this could help translate the model to more complex and crowded conflict scenarios.

Fifth, the model's generalizability beyond the presented scenarios from the Anglo-American cultural context has not been evaluated. Three types of assumptions are especially noteworthy: (1) the low-level control assumption that drivers use the same foot for acceleration and braking, (2) the higher-level assumptions about other peoples behavior encoded in normative likelihoods (for example, the different rules regarding right turns on red light in North America and Europe), and (3) the own preferences defining acceptable behavior (such as infringing on other agents' lanes). As those assumptions are subject to cultural influences [282, 283], they may not transfer seamlessly to other regions of the world. A future model could be made more robust by making such assumptions explicit and learnable (*e.g.*, by modeling them as additional latent (belief) states or context variables dependent on a country).

Finally, while the current policy sampling and kinematic-based particle filter approach were demonstrated to work well in the current scenarios, this could be a limiting factor in scaling the model to a wider range of scenarios, especially when modeling pre-conflict scenarios with more complex road infrastructures and road user interactions. To overcome this limitation, machine-learning-based data-driven models [30, 35, 75, 145, 284] could be leveraged to generate possible futures in our generative model's state transition function, either stand-alone or in combination with the current kinematics-based predictions, thus potentially further enhancing the generalizability of the model. In principle, such

data-driven models, if employed in a closed-loop manner, could potentially serve as an alternative to our proposed active inference model in its entirety, given their clear advantages in inference time – no expensive policy selection process is needed – as well as their superior ability to learn from large human datasets in complex geometric environments. These characteristics, after all, allowed those models to show remarkable, and increasing accuracy in the prediction of routine driving behavior [124]. However, research on evaluating such models in collision avoidance scenarios is lacking. To understand their current capabilities, we tested one of the state-of-the-art machine-learned models (ADAPT [76]) in the lateral-incursion scenario (Section C.3). We found that the data-driven model did not capture human behavior well, likely due to simplistic learning objectives and underrepresentation of critical scenarios in the training data. Furthermore, the black-box nature of such models restricts them to an exclusively descriptive/predictive role, without providing insight into the underlying cognitive mechanisms driving human behavior.

7.7 Conclusion

Overall, our work contributes to the recent body of research [198, 199, 281, 285] with new evidence that active inference can serve as a general framework for computational road user modeling. Along with the emerging real-world applications of active inference in robotics [286] and artificial intelligence [287], this work can aid in the development of agents used for the simulation-based evaluation of advanced driver assistance systems and automated vehicles [17, 18, 288].

8

Active Inference as a Model of Human Resolution of Space-Sharing Conflicts

*In the previous chapter we showed the potential of active inference as a generalizable framework for modeling human behavior in traffic, focusing on examples of collision avoidance. In this chapter, we extend this framework to model how road users can resolve less critical space-sharing conflicts in a collaborative manner, which is crucial both for general traffic safety and the safe deployment of autonomous vehicles. While existing models have captured specific aspects of such interactions (*e.g.*, explicit communication), an integrated computational framework has been lacking, which we hope to provide with our active inference approach. Specifically, we aim to capture three complementary mechanisms for resolving such interactions via uncertainty reduction: (i) implicit communication via direct behavioral coupling, (ii) reliance on normative expectations (stop signs, priority rules, etc.), and (iii) explicit communication. At a simplified intersection scenario, we show that two simulated crossing agents can increase the likelihood of a successful conflict resolution with both explicit communication and normative cues. However, this relies on agents acting as expected; in situations where another agent (intentionally or unintentionally) violates normative expectations or communicates misleading information, reliance on these cues may induce collisions. These findings illustrate how active inference can provide a novel framework for modeling road user interactions, and point to the importance of communication design and normative context in ensuring safe and efficient coordination between human drivers and autonomous vehicles.*

This chapter is based on *J. F. Schumann, J. Engström, R. Wei, S.-Y. Liu, J. Kober, A. Zgonnikov. "Resolving space-sharing conflicts in road user interactions through uncertainty reduction: An active inference-based computational model". Collective Intelligence (2026,* under review*)* [289]

8.1 Introduction

Understanding interactions between human road users is an important aspect of road safety [290, 291] and key to the safe and efficient deployment of autonomous vehicles [292, 293]. In recent years, the topic has been a key focus of road user behavior research, which has resulted in a significant body of empirical findings as well as conceptual and computational models for understanding the key cognitive and behavioral mechanisms underlying road user interaction.

Following the conceptual framework outlined by Markkula et al. [89], we adopt their definition of a *space-sharing conflict* as "*an observable situation from which it can be reasonably inferred that two or more road users are intending to occupy the same region of space at the same time in the near future*" (page 736). Thus, a space-sharing conflict represents a situation where there is potential for road users to end up in a collision, given the uncertainty about how the situation will play out, but does not necessarily have to involve an imminent collision course, as in the traditional definition of a traffic conflict [294]. Based on this, Markkula et al. [89] further define a (traffic) *interaction* as a "*situation where the behaviour of at least two road users can be interpreted as being influenced by a space-sharing conflict between the road users*" (page 737).

Space-sharing conflicts can generally be resolved by reducing uncertainty [295] and establishing a shared understanding, or common ground [296], of the situation [89]. This can be achieved by various means of communication, where a general distinction can be made between implicit and explicit communication [297]. Markkula et al. [89] define *implicit communication* as "*... behavior which affects own movement or perception, but which can at the same time be interpreted as signaling something to or requesting something from another road user.*" (page 741). Examples of implicit communication involve early yielding at an intersection to signal intent and looking towards another road user to indicate that they have been perceived. Means for implicit communication specifically related to movement and distance have been referred to as kinesics and proxemics respectively [292, 298]. By contrast, *explicit communication* can be defined as "*... behavior which does not affect own movement or perception, but which can be interpreted as signaling something to or requesting something from another road user.*" (Markkula et al. [89], page 742). Examples of explicit communication include using verbal utterances, hand gestures, vehicle lights, turn indicators, and the horn [299]. Empirical results indicate that implicit signaling is the dominant means for communication while explicit communication happens more rarely [89, 300].

Another key feature of road user interaction is interdependence [301, 302] or coupling [303] where the behavior of one actor depends reciprocally on the behavior of another actor. However, as shown by Domeyer et al. [298] and Noonan et al. [302], such interdependent behavior appears to mainly occur in uncontrolled scenarios, whereas the behaviors of road users tend to be less dependent in scenarios where interactions are facilitated by traffic signals or signs. This suggests that traffic controls, such as traffic lights and stop signs, and the associated road rules reduce the need for direct, interdependent communication between road users.

A wide range of computational models have been proposed to explain these phenomena. This includes models based on game theory [101, 297, 304–309], evidence accumulation [46–48, 310–312], coupled dynamical systems [301, 302], social force models [313], and the communication-enabled interaction model [212]. These models bring different

perspectives and have significantly advanced our understanding of road user interactions. Some models, such as the the social force model [313] or Siebinga *et al.*'s communication-enabled interaction model [212, 314], are formulated as general models, applicable across multiple scenarios such as pedestrian crossings or highway merging. Nevertheless, such existing models do not offer an integration into a more general theory of human cognition and behavior. Importantly, none of these models implemented either explicit communication or traffic norms.

In this paper, we explore *active inference* as a framework for modeling the collective behavior of human road users in resolving space-sharing conflicts in traffic interactions. In the spirit of recent work on active inference modeling of interactive behavior in other domains, we assume that road users maintain reciprocal beliefs about each other's future behavior as the basis for selecting their next actions (e.g., braking to yield or accelerating to go first when crossing an intersection). As long as the interacting agents have a shared *and precise* understanding about how the situation will play out (e.g., who will go first), the space-sharing conflict will be resolved smoothly, in a similar way as when two agents are singing a familiar song together [315, 316] or performing other joint actions such as dancing or jointly carrying an object [317]. In such situations, the agents' beliefs and ensuing actions can be seen as guided by a common, agent-neutral, *generative model* [317], typically established through implicit communication (e.g., via proxemics and kinesics [292, 298]).

However, in some traffic situations a sufficiently precise common understanding may be lacking due to uncertainty in the agents' beliefs (e.g., agent A is uncertain about whether agent B will yield), which may preclude efficient resolution of the space-sharing conflict. We here propose two main ways in which uncertainty in road user interaction may be resolved – relying on established *normative expectations* [318, 319] and *explicit communication* – and demonstrate how these two mechanisms can be represented in a computational mechanistic road user behavior model.

In this framing, normative expectations can be seen as a form of *niche construction* [320, 321] where traffic rules and more informal social norms have been established in a traffic culture to guide, or scaffold, expectations about road user behavior, supported by deontic cues [322] such as road signs, traffic signals and lane markings. Explicit communication can be understood as an *epistemic action* [57, 218, 317] with the purpose to obtain new information that could potentially reduce uncertainty in the beliefs about the other agent's intent.

We implement these mechanisms into an active inference model previously proposed for capturing the behavior of individual drivers in non-interactive scenarios [199, 207]. We then use this extended model to show how uncertainty in human interactions can be reduced through (i) implicit communication – a baseline approach (modeled similarly to Siebinga *et al.* [212]), (ii) explicit communication [220] and (iii) normative expectations [318, 319] and explicit deontic cues such as traffic lights, painted road markings and stop signs [236, 322]. We evaluate the model in a simplified intersection scenario (Figure 8.1) with two interacting vehicles, similar to Fox *et al.* [305]. We demonstrate how norms and communication can (jointly or separately) facilitate successful coordination, illustrating how active inference can unify behavioral coupling (through implicit communication), explicit communication, and normative reasoning within a single modeling

framework. Furthermore, we explore how miscommunication or failures to adhere to normative expectations can lead to conflicts and collisions under those principles.

8.2 Method

8.2.1 Model overview

Our model is grounded in active inference – a general framework for understanding and modeling sentient behavior of living systems [57, 218]. It is based on the key idea that agents strive to sustain their existence by forming (probabilistic) beliefs about how observations are caused by the environment and their own actions, and pursuing policies (action sequences) that generate observations that are preferred or expected given the type of creature they are (maximizing *pragmatic value*) or allow the creature a better understanding of their environment (maximizing *epistemic value*). Observations that are not aligned with the preferred expectations are surprising to the agent and, in order to be in tune with its environment, the agent strives to minimize surprise over time, which can be modeled in terms of the minimization of variational free energy. When acting in an uncertain environment, such as road traffic, a key aspect in minimizing surprise or free energy is to control the uncertainty in one' s beliefs about how the situation will play out, for example, whether another road user will yield or cross at an intersection ahead.

We base our model on a framework previously developed for active-inference modeling of human driver behavior [199, 207]. Fundamental to active inference, this framework includes a global environment model (*i.e.*, the *generative process*) as well as each agent's internal *generative model* approximating this environment. The environment in the is described by the state $\boldsymbol{\eta}$, from which observations $\boldsymbol{o}$ can be made according to the *generative process*'s observation probability $\hat{p}(\boldsymbol{o} \mid \boldsymbol{\eta})$. The state of the environment can be influenced by the agents' combined actions $\boldsymbol{a}$ according to the *generative process*'s state transition function $\hat{p}(\boldsymbol{\eta'} \mid \boldsymbol{\eta}, \boldsymbol{a})$. Meanwhile, each agent's *generative model* is based on an internal state $\boldsymbol{s}$, as well as the corresponding observation probability $p(\boldsymbol{o} \mid \boldsymbol{s})$ and a state transition function $p(\boldsymbol{s'} \mid \boldsymbol{s}, \boldsymbol{a}_v)$. A central tenet of active inference is that the agent is unsure about the exact state of the world, (*i.e.*, the agent has an internal probabilistic belief $q(\boldsymbol{s})$ over $\boldsymbol{s}$). While it is possible to assign different generative models to different agents, in this work, both agents use separate instances of the same model (*i.e.*, their understanding about the world as well as their preferences priors for desired observations are identical).

In our model, each of the modeled agents from a set of agents V repeatedly chooses an action based on a multi-step process, following the model of Schumann *et al.* [207]:

1. Each agent makes an observation $\boldsymbol{o}$ and uses it to update its belief $q(\boldsymbol{s})$, which includes both its own state and that of the other agents.

2. The agent generates beliefs about the possible future kinematic states of the other agent, representing the uncertainty about where the other agent may move next.

3. The agent generates candidate policies $\boldsymbol{\pi}$ (*i.e.*, sequences of actions $\boldsymbol{a}_v$ (here defined as acceleration and steering rate, and potentially explicit communication signals) up to a prediction horizon). Using the *generative model's* state transition function, the agent's corresponding beliefs about the future states associated with each policy $\hat{q}(\boldsymbol{s})$ are generated.

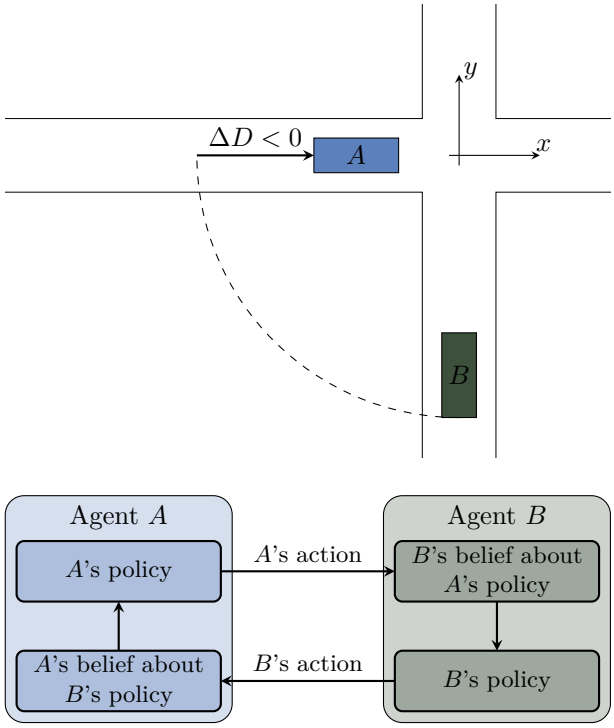


Figure 8.1: The modeled scenario: two agents on crossing paths representing a space-sharing conflict with ΔD denoting agent B' s lead over agent A (*i.e.*, $\Delta D < 0$ if A is closer to the intersection). Below it, the basic action-perception (hermeneutic) cycle resulting from modeling two active inference agents observing each other (based on Friston and Frith [315])

4. Candidate policies are evaluated based on the expected free energy (EFE) of the corresponding predicted beliefs, and the policy with the lowest EFE is selected. The EFE scores both the value of achieving and agent's goals as implemented in the preference prior $p(\boldsymbol{o})$ (maximizing *pragmatic value*) and the value of obtaining new information to resolve uncertainty (maximizing *epistemic value*).

5. The agent only updates its currently pursued policy if the accumulation of surprising evidence E against the current policy has reached a predefined threshold. Otherwise, it extends the current policy by appending a single action at the end of the time horizon. This mechanism allows the model to capture realistic reaction times of human drivers. In practice, the accumulated surprise is linearly dependent on the *pragmatic value* (*i.e.*, a low pragmatic value drives the agent's need for a comprehensive policy update, which is also referred to as "re-planning" by Schumann *et al.* [207]).

The first actions of the currently selected policies for each agent are then executed simultaneously, based on which the environment is updated, and new observations generated. The model of Schumann *et al.* [207] used the above five steps to model human collision

avoidance behavior in non-interactive safety-critical scenarios, where a modeled agent responded to another agent executing a predefined policy. Here, we extended this model to represent the dyadic interaction between two agents A and B on a crossing path at an intersection (Figure 8.1, top panel) aimed at resolving the resulting space-sharing conflict.

In such a scenario, our model implementation leads to the coupled action-perception hermeneutic cycle (Figure 8.1, bottom panel) where each agent maintains, and repeatedly updates, its belief about the other agent's policy. To resolve the space-sharing conflict, the agents can adapt their behavior to one another in a reciprocal fashion, what we here refer to as *direct coupling* (*e.g.*, by slowing down or accelerating), reducing the uncertainty in their beliefs about each other's future actions via *implicit communication*.

In the traffic interaction literature, implicit communication is often understood as *intentional* signaling of intent through kinematics [323–326], for example, a vehicle slowing down early when approaching a zebra crossing to let the pedestrians know they can cross. However, here we use this term in a more general sense, implying communication that de-facto happens through vehicle kinematics regardless whether the agents intend changes (or lack of changes) in own kinematics as a communicative act. For instance, if a driver intends to merge onto a highway lane, a driver of a vehicle already in that lane would likely interpret initial movement of the merging vehicle towards the target lane as an indication of the forthcoming merge. Here, the kinematics of the merging vehicle effectively signals the intent of that vehicle's driver to surrounding road users even if that driver is not aware of their presence. This notion of implicit communication is consistent with the definition of Markkula *et al.* [89]: "...behaviour which affects own movement or perception, but which can at the same time be interpreted as signalling something to or requesting something from another road user" (page 741).

Besides this direct coupling via implicit communication, where inferences about the other agents' intents are made purely based on their kinematics, other methods of uncertainty reduction are also possible. The first mechanism considered in this work is the reliance on *normative expectations* [318, 319], which are shared beliefs within a society about the behavior of other road users in interactive scenarios. Such expectations may be encoded in formal rules, such as traffic lights or traffic signs, as well as in informal conventions. The second mechanism for reducing uncertainty about other agents' policies is *explicit communication*, which enables an agent to directly signal its intent and thereby allows others to more accurately predict its future behavior. In both cases, the resulting reduction in uncertainty facilitates a more reliable resolution of space-sharing conflicts. The following two subsections describe how normative expectations and explicit communication are implemented in our model. A more detailed mathematical description of the model can be found in Appendix B.

8.2.2 Modeling normative expectations

In driving, an agent can rely on established conventions for interactions – closely related to the notion of "interaction rituals" proposed by Goffman [327] and discussed by Markkula *et al.* [89] – as a means for resolving space-sharing conflicts with other road users. Such conventions can be understood in terms of *normative expectations* [318, 319], that is, shared societal beliefs about proper road behavior. These expectations allow an agent to constrain its beliefs about the expected actions of other road users, thus reducing uncertainty and

improving the likelihood of a successful interaction. In traffic, *normative expectations* are partially encoded in formal road rules and partially embodied in informal conventions, and are often reinforced by designed road features such as lane markings, traffic lights, and stop signs (referred to by Constant *et al.* [322] as *deontic cues*).

To implement these normative expectation into our active inference framework, we employ the *norm-conditioned particle filter* introduced by Schumann *et al.* [207]. In this approach, the *generative model's* state transition probability is no longer solely determined by kinematic likelihoods [199], but is instead biased by the *projected normative probability*. The latter probability captures the assumption that a modeled ego agent assigns low probability to actions by another road user $v \in V$ that would – if pursued consistently over short- and medium-term horizons (up to 4 s in our implementation) – lead to norm violations (Equation (B.32)). Norms could then be specified via the *normative probability* p_n, which for instance is low if the other agent is off the road; in that case, when making predictions of other agents, the model of Schumann *et al.* [207] would consequently assign a low *projected normative probability* to large steering maneuvers by other vehicles if these would result in leaving the road.

Importantly, however, the norm-conditioned particle filter as implemented by Schumann *et al.* [207] can only account for "non-interactive" norm violations by a surrounding agent v. Such non-interactive violations depend solely on the violating agent's own behavior but not on other agents. Concretely, Schumann *et al.* [207] assumed that

$$p_n(\boldsymbol{s}) = \prod_{v \in V} p_{n,V}(\boldsymbol{s}_v), \tag{8.1}$$

where $\boldsymbol{s}_v$ denotes the state variables associated with agent v, such as position or velocity. Such non-interactive norms are represented, for instance, by lane following, adhering to speed limits, traffic lights, or stop signs. Under the assumption that agents in a scene are not kinematically dependent on one another as well, the behavior of each agent becomes independent from each other. This factorization of the state transition function is computationally advantageous, as the future trajectories of all other agents need to be predicted only once when the ego agent evaluates multiple policies. This is particularly beneficial in crowded scenes.

In our proposed model, however, we extend this formulation by adopting the more general assumption of *interactive norms*. There, we assume that the norm compliance of a road user's action can only be assessed in relation to the behavior of the ego (*i.e.*, the modeled) agent. Consequently, we modified Equation (8.1), yielding

$$p_n(\boldsymbol{s}) = \prod_{v \in V} p_{n,V}(\boldsymbol{s}_v | \boldsymbol{s}_{\text{ego}}), \tag{8.2}$$

with the *normative probability* of each surrounding agent v explicitly depending on the state of the modeled ego agent. With such interactive norms our model can capture priority rules (which we investigated here), but potentially also other cases, such as rules mandating minimum following distances.

Crucially, in scenarios involving at least one interactive norm, the modeled ego agent can no longer be factorized out of the *generative model's* state transition function. As a result, the future trajectories of other road users must be regenerated for each candidate policy, leading to a substantial increase in computational cost.

Types of norms
In this work, we implemented several types of norms. First, we re-used the lane-following norms of the baseline model [207]: the model expects other agents to stay in their lane. Second, in our model agents always assume that other traffic participants will follow a speed limit, avoiding velocities above $v = 11\,\mathrm{m\,s^{-1}}$ if possible.

Third, we investigate the impact of additional traffic rules, whose application we correspondingly vary. Concretely, we consider the application of stops signs and a "first come, first go" priority rule. The former is implemented as a requirement for an agent to fall below a velocity limit of $v_L = 0.278\,\mathrm{m\,s^{-1}}$ (*i.e.*, $1\,\mathrm{km\,h^{-1}}$) in a short distance in front of the stop sign. Concretely, states where another agent v is moving beyond the stop sign without having previously stopped will result in lower values of that agent's normative weight function $p_{n,V}$ (Equation (8.2)). Meanwhile, for the priority rule, an ego agent considers the other agent v to have priority if they are closer to the intersection. While this is initially reevaluated at every timestep, once the ego agent completed the stop in front of the stop sign, its opinion about the priority becomes fixed. If an agent believes it has priority, the normative weight function $p_{n,V}$ (Equation (8.2)) will assign low likelihoods to observations where the other agent v reaches the intersection without trailing the ego agent by a sufficient amount.

In our model, each agent not only assumes that the other agent follows the above norms, but will also try to follow them itself. The latter was implemented via the model's prior preferences $p(o)$ (Equation (7.11)): policies leading to norm violations are disincentivized via low pragmatic values. The exact implementation of the norms and preference prior adjustments can be found in the Appendices B.3.4 and B.2.4, respectively.

8.2.3 Modeling explicit communication

Another way for agents to reduce the uncertainty about each other's policies is by using explicit communication. To implement this mechanism, we endow each agent with the capacity for two communicative acts $\boldsymbol{\gamma} = \{\gamma_A, \gamma_Y\}$:

- γ_A: **Asking** the other agent for their intent, with the expectation of a reply.
- γ_Y: Signaling one's intention to **yield**, informing the other agent about one's intended behavior

For simplicity, we assume that the intended signal can be directly mapped from the communicative act (see [220] for a more elaborate account of how such mappings are established in the first place). We also model these communicative acts at a high level of abstraction and do not attempt to represent their specific communication modalities. For example, in a real interaction, prompting may be mediated by seeking eye contact and yielding could be signalized with a hand gesture, but in the current model we abstract away from the detailed ways in which prompting and signaling yielding are instantiated.

These signal states are defined as discrete, binary variables (*i.e.*, $\boldsymbol{\gamma} \in \{0,1\}^2$) – with agents either signaling ($\gamma_A, \gamma_Y = 1$) or not ($\gamma_A, \gamma_Y = 0$) – when they are part of the *generative process*'s state $\boldsymbol{\eta}$, of the observations, or of the policies. However, in an agent's belief state $\boldsymbol{s}$, they are defined over a continuous domain (*i.e.*, $q(\boldsymbol{\gamma}) : [0,1]^2 \rightarrow \mathbb{R}^2_+$). This implementation of belief states as continuous variables has two purposes: (1) It captures

perceptual uncertainty (*e.g.*, due to distance between agents), allowing agents to represent uncertainty about whether a signal was observed. (2) Under an assumption of honest communication by the other agent v, the belief $q(\gamma_{Y,v})$ can be interpreted as the ego agent's belief over agent v's *yielding intent*. Importantly though, this belief $q(\gamma_{Y,v})$ is only updated based on the observed yielding signal $\gamma_{Y,v}$ in our current implementation. In reality though, the ego agent's belief about agent v's yielding intent would be influenced by other sources of information (*e.g.*, priority rules or kinematics). Consequently, here $q(\gamma_{Y,v})$ actually represents only the *signalized* yielding intent of agent v, but for simplicity in the rest of the paper we refer to it simply as *yielding intent*.

Furthermore, to allow the agent to react to perceived signals, we select the signaling policy (*i.e.*, prompting and yielding) at every timestep, in contrast to the evidence accumulation-based re-planning [207] used for the kinematic policy. This is supposed to allow for reaction to perceived signals, even when the current kinematic policy is considered sufficient. If required, we could achieve the same effect in the evidence accumulation approach – setting the surprise accumulated by suboptimal yielding policy in a single timestep to be higher than the evidence threshold – but the current implementation is computationally more efficient. Additionally, to allow for more immediate actions, once a yielding signal is perceived, and agent will update its kinematic policy.

Prompting for intent

The primary motivation for the ego agent to prompt the intention of another agent $v \in V$ – that is, to choose the action $\gamma_{A,\text{ego}} = 1$ – is the maximization of *epistemic value*, which – as shown in Equation (7.7) – can be split into *posterior predictive entropy* and *expected ambiguity* [199]. If the ego agent does not prompt ($\gamma_{A,\text{ego}} = 0$), the epistemic value is always approximately zero, reflecting the fact that no new information is expected to be gained. In contrast, when prompting (under the assumption that agent v is cooperative), the ego agent's belief over the yielding signal $q(\gamma_{Y,v})$ collapses to a quasi-binary state through the *generative model*'s state transition function (Equation (B.29)). In particular, the observation model $p(\boldsymbol{o}' \mid \boldsymbol{s})$ for $\gamma_{Y,v}$ follows a Bernoulli distribution $\mathcal{B}(\gamma'_{Y,v}|\gamma_{Y,v})$, resulting in the belief $q(\gamma_{Y,v})$ concentrating around the extreme values $\gamma_{Y,v} = 0$ and $\gamma_{Y,v} = 1$. Under this belief, the ego agent assumes that – given a hypothetical future observation – it will become certain about agent v's yielding intent. This certainty corresponds to the zero value of *expected ambiguity*. Consequently, the epistemic value of the prompting action ($\gamma_{A,\text{ego}} = 1$) reduces to the corresponding *posterior predictive entropy*, which is maximized when uncertainty about agent v's yielding intent is highest. Specifically, the epistemic value of the prompting action is highest when there is an initial belief with $\mathbb{E}_{q(\gamma_{Y,v})}\gamma_{Y,v} = 0.5$. A more detailed mathematical explanation of the epistemic value of prompting can be found Appendix D.

It must be noted that in the preference prior underlying the pragmatic value we implement a base cost for active signaling (Equation (B.70)), as in real life this will involve some amount of effort. Therefore, an agent will only send a prompting signal if the epistemic value of that action outweighs the pragmatic cost.

Signaling intent

The agent's motivation for sending a yielding signal is grounded in the assumption that agents are, in principle, benign and honest. Although signaling yielding, as with prompt-

ing before, incurs a base cost representing the required effort, the penalty for uncooperative behavior is higher. Specifically, if the ego agent believes that another agent v is prompting for its intent (*i.e.*, $q(\gamma_{A,v}) \rightarrow \delta(\gamma_{A,v} - 1)$) and simultaneously intends to yield (*i.e.*, under its current policy, it predicts arriving second at the intersection), then failing to signal yielding (*i.e.*, setting $\gamma_{Y,\text{ego}} = 0$) leads to a substantial decrease in *pragmatic value*.

However, this pragmatic value penalty is not the primary effect of the yielding signal in our model. Instead, its main impact lies in reducing the uncertainty about the other agent's future behavior. This follows from the assumption introduced above that an agent's belief in having perceived a yielding signal directly translates into the belief that the other agent intends to yield. Specifically, if the ego agent would receive a yielding signal from agent v (*i.e.*, $q(\gamma_{Y,v}) \approx \delta(\gamma_{Y,v} - 1)$), it would belief that agent v is going to yield. This expectation is implemented in the *normative probability*, biasing the ego agent towards predicting those behaviors by agent v which correspond to their signaling. Concretely, this bias is adjusted such that, under the belief that agent v is yielding (*i.e.*, $q(\gamma_{Y,v}) \rightarrow \delta(\gamma_{Y,v} - 1)$), predictions in which agent v arrives first at the intersection are suppressed, and the ego agent instead assigns high probability to agent v choosing to manifest the yielding signal with deceleration. In this way, a yielding signal can promote the successful resolution of our modeled interaction at the intersection.

8.2.4 Model parameters

While we mostly reuse the parameters from Tables 7.1 and 7.2 from the previous Chapter 7, some variables are changed, and new ones are added, which can be seen in Table 8.1. . The change that stands out most here is the different value implementing the speed preference σ_v, which was decreased to prevent the potential of deadlocks in scenarios where norms or communication are sufficient to remove the risk of a collision, but where the cross entropy method approach is unable to converge to the superior approach of accelerating across the intersection. This change, however, should not be seen as major, as the parameter sensitivity analysis in the previous model [207] showed no significant difference in performance under such a change.

8.2.5 Simulated scenario

We simulated the model behavior in an interaction between two agents ($V = \{A, B\}$) on a crossing path at an intersection (Figure 8.1, top panel). In this scenario, the two vehicles start at $t = 0$ at respective distances of $D_A(0) = x_A(0) < 0$ and $D_B(0) = y_B(0) < 0$ – defined based on their center of mass position. Specifically, we set that $D_A(0) = -65\,\text{m}$, while varying $D_B(0) \in \{-90\,\text{m}, -80\,\text{m}, -70\,\text{m}, -68\,\text{m}, -66.5\,\text{m}, -65\,\text{m}\}$, resulting in initial distance differences $\Delta D(0) = D_B(0) - D_A(0) \in \{-25\,\text{m}, -15\,\text{m}, -5\,\text{m}, -3\,\text{m}, -1.5\,\text{m}, 0\,\text{m}\}$. Both vehicles start with an initial velocity of $v_1(0) = v_2(0) = 10\,\text{m}\,\text{s}^{-1}$ towards the intersection. To capture the variability due to the stochasticity of our model, we simulated its behavior 50 times for each $\Delta D(0)$ value.

Table 8.1: An overview of the new and changed parameters compared to the original model.

Parameter	Description	Source
State transition function $p(\boldsymbol{s}'\|\boldsymbol{s},\boldsymbol{a})$		
$\sigma_{\gamma,0} = 0.005$	Noise applied to other vehicle's signalling during belief update	Tuned parameter
$\sigma_{\gamma} = 0.001$	Noise applied to other vehicle's signalling during behavior prediction	Tuned parameter
Preference function $p(\boldsymbol{o})$		
$\sigma_v = 0.2\ [\mathrm{m\,s^{-1}}]$	Standard deviation of preference distribution of agent's velocity v	Tuned parameter
$\sigma_a = 0.2\ [\mathrm{m\,s^{-2}}]$	Standard deviation of preference distribution of agent's acceleration a	Tuned parameter
$g_S = -10000$	Pragmatic value of norm violation	Tuned parameter
$g_\gamma = -0.125$	Pragmatic value of an active communication signal	Tuned parameter
$g_W = -10000$	Pragmatic value of uncooperative signalling	Tuned parameter
Policy selection		
$\mathcal{N}\left(0, 10^{-5}\right)\ [\mathrm{s^{-1}}]$	Distribution to sample steering rates ω_τ in first CEM iteration – set very low to avoid steering responses	Baseline model [207] (mean) and Tuned parameter (std)
$\lambda = 10^{-5.9}$	Evidence accumulation drift rate	Tuned parameter

8.3 Results

In this section we first demonstrate how outcomes of the modeled interaction are affected by four different regimes: (1) the baseline using *implicit communication*, (2) reliance on *normative expectations*, (3) using *explicit communication*, and (4) a combination of the latter two. We then zoom in on example simulations representing each regime in respective subsections. Finally, we demonstrate the effects of miscommunication and overtrust in communication or norm-following[1].

8.3.1 Interaction outcomes

Under the baseline regime (implicit communication only), whether a space-sharing conflict is resolved depends on the presence of asymmetries in agents' proxemics (Figure 8.2**a**). Sufficiently large initial asymmetries (*i.e.*, Agent A having a significant lead of $\Delta D \leq -5\,\mathrm{m}$, which is equivalent to an initial time gap between agents of 0.5 s) allow the leading agent to always go first. However, that likelihood decreased with the difference in initial distances, with the trailing agent going first in some cases. Additionally, for small and especially zero

[1]The simulation data underlying the presented results can be found at `https://osf.io/2397s`.

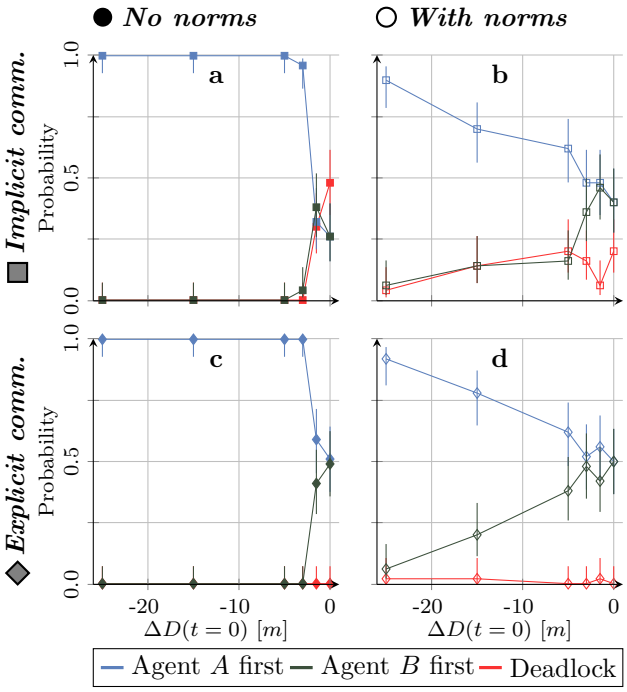


Figure 8.2: The likelihoods of different simulation outcomes under different model setups, for different initial differences in distance to the intersection $\Delta D(0)$. "With norms" refers to models considering both first-come priority rules and stop signs. The vertical lines indicate the 95% confidence intervals of the shown probabilities according to the Wilson score [328].

8

differences in initial distances, the conflict might not be resolved successfully at all. For instance, for $\Delta D = 0$, there is approx. 48% chance of a deadlock – both agents coming to a standstill and failing to proceed through the intersection.

One way to resolve the uncertainty that causes the deadlock in interactions is to introduce social norms (*i.e.*, precedence at an intersection) that the agents can rely on when predicting each other's behavior. If these norms are understood and adopted by both agents, they promote shared beliefs, that is, normative expectations, about who should go first at the intersection. In our model, we implemented this mechanism by endowing the agents with two norms: a priority rule (who arrives first goes first) and the expectation to stop at stop signs. Our simulations demonstrated that normative expectations indeed helped to reduce the likelihood of a deadlock for initially symmetric scenarios (Figure 8.2**b**). Specifically, for $\Delta D > -5\,\mathrm{m}$, the likelihood of a deadlock falls to 20%, compared to nearly every second case in the baseline scenario. Meanwhile, however, for cases where Agent A had a larger lead, both deadlocks and role reversal (*i.e.*, Agent B going first) become possible.

Another way to potentially resolve the uncertainty that causes the standstill is by explicitly communicating one's intent to the other agent. As can be seen in Figure 8.2**c**, the addition of explicit communication allows the agents to successfully resolve every interaction in the uncontrolled intersection scenario. Notably, at small ΔD values the

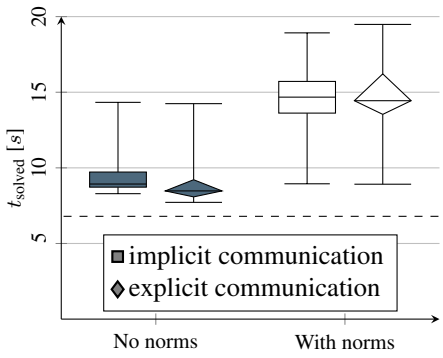


Figure 8.3: Time required to solve the space-sharing conflict in a scenario where both agents have an identical initial distance to the intersection ($\Delta D(t = 0) = 0$). We define t_{solved} as the time elapsed from the start of the simulation till the instant at which one agent crosses the intersection; deadlock results are excluded. For agents neither braking or accelerating one would see $t_{\text{solved}} = 6.796\,\text{s}$ (dashed line added as a lower limit).

leading agent can yield priority to the lagging agent – this is due to random variations in policy sampling; although this is not consistent with the initial conditions, this outcome is not implausible and avoids the deadlock hence here we consider it a successful resolution of the conflict.

Although in the uncontrolled intersection the addition of communication allows the agents to successfully resolve any interaction, intersections with imposed norms are very common in the real world, making them an interesting case study. In our model implementation, we assume that explicit communication supersedes deontic cues such as traffic norms (*e.g.*, if an agent signals yielding, the other agent will no longer assign a low preference prior to policies which result in going first despite not having priority according to the set rules). As can be seen in Figure 8.2**d** when compared to Figure 8.2**b**, adding communication to the model can significantly reduce the number of unsuccessful interactions across all starting conditions (*i.e.*, from 20% to 0% at $\Delta D = 0$). Nevertheless, by contrast to the scenario with only communication, an insignificant number of deadlocks are still present.

To complement the analysis of interaction outcomes, we investigated the efficiency of conflict resolution by determining when the first agent left the intersection behind (Figure 8.3). Here, the effect of communication is much less pronounced, with no statistically significant differences in resolution time. Consequently, communication itself can increase the amount of successful resolution of conflicts, but does not make a successful resolution more efficient. We assume that this is the case because an agent only signals yielding when yielding is the favored policy (because of priority rules or being in a trailing position), which needs time to become apparent, as well as the time needed for the other agent to prompt. Meanwhile, the addition of norms can lead to longer resolution times, as the stop sign requires braking to near stand-still under any condition.

8.3.2 Interaction dynamics

To highlight the mechanisms at play, we analyzed the dynamics of representative simulations of our model in each of the four regimes.

Baseline: Implicit communication

Figure 8.4**a** presents an example of an agent with a larger initial lead using that lead to go first. There, both agents initially predict a collision, as illustrated by low pragmatic values. For the trailing agent B, this is based on the assumption that the leading agent A may slow down; meanwhile, the leading agent assumes that the trailing agent might accelerate. However, as this would violate the normative expectation about following the speed limit, it is treated as unlikely, and therefore the leading agent accumulates evidence much slower. This causes the trailing agent to initiate braking at $t = 3.8\,\mathrm{s}$ (first dashed line). Once the leading agent perceives this braking (second dashed line at $t = 4.4\,\mathrm{s}$), it predicts an increasing lead (thin blue lines in the ΔD plot), thus perceiving an affordance to move ahead through the intersection with no collision risk. Consequently, its (negative) pragmatic value $G_{\mathrm{prag},A}$ approaches zero in all components, meaning there is little evidence for the need of a full re-plan; therefore, A follows its original policy of crossing the intersection without braking. This represents a clear case where an initial difference in distance to the intersection is enough to yield an efficient resolution of the space-sharing conflict.

When the initial distances to the intersection are similar, stochastic variations in agents' policy generation might or might not break down the initial symmetry. For instance, in Figure 8.4**b**, both agents start with an equal distance to the intersection, and therefore accumulate evidence for a full re-plan at a similar rate, before initiating deceleration, with the goal of letting the other vehicle pass. Here, a small difference in the agents' braking policies breaks the symmetry, leading to sufficiently different distances to the intersection ($\Delta D \not\approx 0$). The now trailing agent A decides to slow down even more, which gives the leading agent B enough confidence to cross the intersection first.

However, if the initial braking policies of the agents happen to be similar (Figure 8.4**c**), this leads to $\Delta D \approx 0$, with both agents continuing to brake until coming to a stop (dashed line at t $t = 8.6\,\mathrm{s}$). In this situation, they end up in a standstill, as there is no normative expectation that allows them to disregard the kinematically likely occurrence of the other vehicle accelerating onto the intersection. Specifically, this can be seen in the predictions of increasing velocities at $t = 8.6\,\mathrm{s}$, which indicate that each agent perceives it as likely that the other agent will accelerate into the intersection (see the ΔD plot, where A predicts B taking the lead and the other way around). Thus, the space sharing conflict is not resolved because implicit communication alone does not sufficiently reduce the agents' believed likelihood of a possible future collision.

Normative expectations

Our simulations demonstrated that norms can help resolve uncertainty in an interaction beyond what is possible with implicit communication (Figure 8.2b). Figure 8.5**a** shows an example simulation illustrating how this happens. Here, both agents perform initially similar braking maneuvers (from around $10\,\mathrm{m\,s^{-1}}$ at $t = 2.4\,\mathrm{s}$, first dashed line, to around $v = 2\,\mathrm{m\,s^{-1}}$). This – as can be seen in the parts of the pragmatic value – is mostly driven by the need to follow the rules of the stop sign. Meanwhile, because of the normative

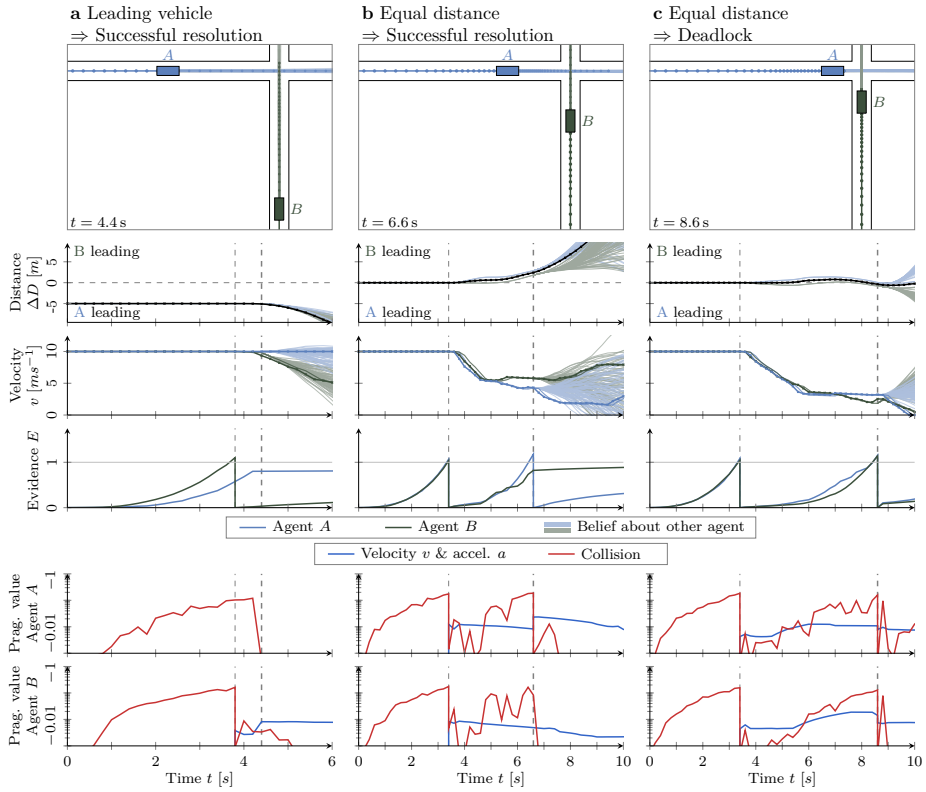


Figure 8.4: Three illustrative examples of model behavior in the baseline scenario: the agents interact based on implicit communication only. For each example simulation, the panels illustrate (from top to bottom) the top-down view at a specified instant t, the difference in distance $\Delta D(t) = D_B(t) - D_A(t)$ ($\Delta D > 0$ m means that agent B is closer to the intersection), the agents' velocities v, their accumulated evidence (surprise) E, and the components of the pragmatic value (Equation (B.16), scaled with the surprise gain factor λ from Equation (7.13)). The top-down view plots illustrate each agent's states as predicted by another agent. For instance, in panel **b** at $t = 6.6$ s, agent B predicts that agent A would on average pursue a constant velocity (while in reality, agent A decided to break shortly thereafter). The distance difference plots show the beliefs the agents have over ΔD (for example, in panel **c**, agent A overestimates ΔD, while agent B underestimates it). In the velocity plots, the thin lines indicate the beliefs about the velocity value held by the other agent (*e.g.*, in panel **a** for $t > 4.4$ s, the agent A overestimates the velocity of agent B). **a)** Early resolution: Agent A starts 5 m closer to the intersection than agent B. When agent B brakes first, agent A becomes sufficiently confident that it can pass without having to also slow down; it crosses the intersection first without braking. Thus, the space-sharing conflict is here resolved by "early yielding" by agent B, which can be interpreted as a form of implicit communication. **b)** Late resolution: The two agents start at an equal distance to the intersection and initially start braking at the same time. However, agent A takes a slightly more cautious approach, which gives agent B confidence that it can go first (*i.e.*, reducing uncertainty about the potential space overlap, opening up an affordance for B to pass). This is then "confirmed" by A through slowing down further. **c)** Unsuccessful resolution: Similar to panel (b), the two agents start at an equal distance to the intersection and start braking roughly at the same time. However, in this case the symmetry is never broken and the vehicles end up in a stand-still where none of the agents are confident enough of a collision-free path to attempt passing, each believing that the other agent may start.

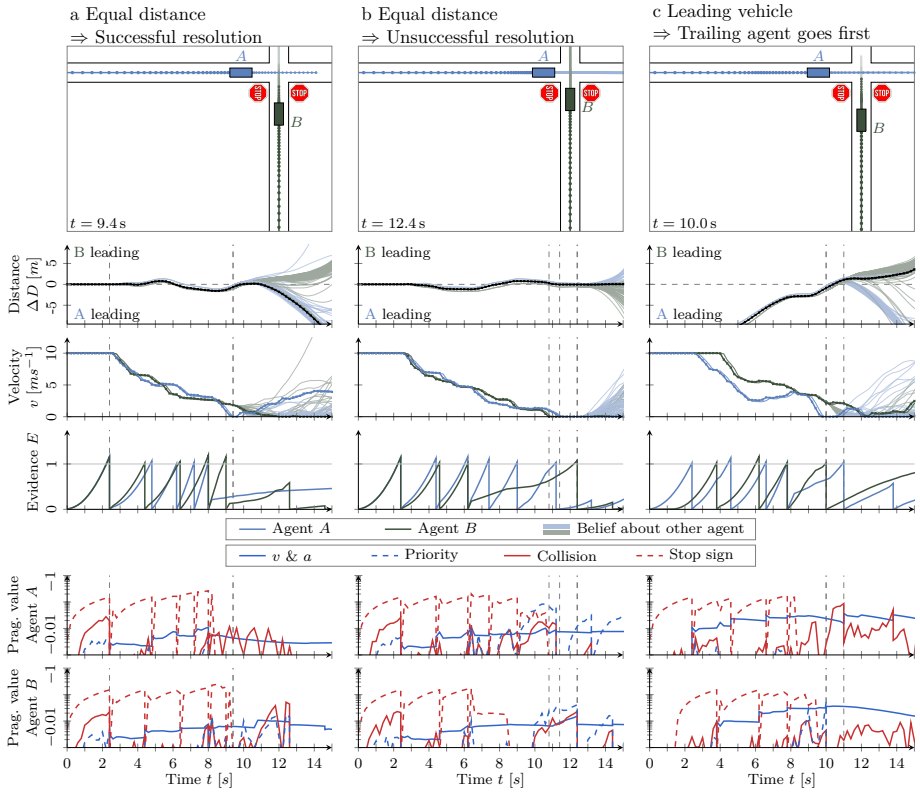


Figure 8.5: Three example interactions of the model endowed with normative expectations (priority rule and stop signs). **a**) The two agents start at an equal distance to the intersection, both agents consider agent A as the leading agent which allows A (after stopping) to pass the intersection first. **b**) The two agents start at an equal distance to the intersection, but due to lagging perception, assign each other priority, leading to a standstill. **c** The required stopping at intersections with priorities can lead to reversal in role, if the trailing agent B decides to brake for the stop sign very late and aggressively.

expectation that other agents will break for the stop sign, collisions are seen as much less likely (at $t = 2.4\,\mathrm{s}$, the pragmatic value associated collisions is only ≈ -0.03, while in the scenarios without normative expectation shown for instance in Figure 8.4**b** it approaches -0.1). Therefore, the risk of collision is not the main driver of surprise accumulation here. However, even though agent A continues to brake (coming to a complete stop at $t = 9.4\,\mathrm{s}$, second dashed line), it is able to maintain a slight lead ($\Delta D < 0$). Consequently, it sees itself in the priority position. Combined with the fact that it assumes that agent B still needs to brake until standstill as well (the thin blue lines for the predictive belief all trending toward negative ΔD after the second dashed line), agent A's belief in a widening gap allows it to follow its plan of going first. This indicates that there are two mechanisms which facilitate the successful resolution of these initially symmetric scenarios. On one hand, the stop signs impose a need for larger decelerations, which leaves more space for the stochasticity of the model to result in diverging policies, thereby breaking the symmetry. One the

other hand, the assignment of priority allows agents to exploit smaller asymmetries than previously, as long as they are big enough to assign priority to oneself.

However, as discussed above, the addition of normative expectations is no panacea, with deadlock still possible – as illustrated in Figure 8.5**b**. Here, the deadlock is mostly caused by the agents believing that the other one has the priority. As described in Section 8.2.2, the priority decision becomes final once an agent completes the deceleration in front of the stop sign. For agent B, this is the case at $t = 10.8\,\mathrm{s}$, at which point it believes the in agent A's priority (B believes that $\Delta D < 0$ and hence that agent A has priority). Meanwhile, a short time later at $t = 11.4\,\mathrm{s}$, agent A sees agent B in the lead (according to A's belief, $\Delta D > 0$). Consequently, both agent predict the other one to take the lead, with agent B predicting a decrease in ΔD and agent A predicting a further increase. This results in both agents standing still, with any advance considered too risky in terms of a potential collision.

However, the addition of stop signs has an additional effect. As can be seen both in Figure 8.2**b** and Figure 8.5**c**, it is now possible for an agent trailing by a large initial distance difference to still cross the intersection first. This happens when the trailing agent B brakes so late and so harshly that at the point where it fulfills the speed requirement set by the stop signs, it is actually leading, and then goes first. This is highlighted by the low priority component in agent B's pragmatic value after its corresponding re-plan at $t = 10.0\,\mathrm{s}$ (the first dashed line), as well as B's confidence that agent A will not increase speed (see the velocity plot). Similarly, as agent A reached this point earlier while still leading, agent A also considers itself as having priority (as can be seen in its prediction that the other agent will decelerate and therefore yield). However, as seen by the sharp increase in the evidence gain and the decrease in the collision component of the pragmatic value after this point, it no longer believes in the norm compliance of agent B, and therefore decides to brake instead at $t = 11\,\mathrm{s}$ (second dashed line). While this scenario of both agent believing to have priority might induce collisions, we only observed collisions in 0.833% of simulations in this scenario. While such behavior might seem as a model artifact, it was shown previously that humans can interpret other traffic participants' cautious approach to interactions as yielding, and therefore self-assign priority and move aggressively [329]. The same mechanism, where the leading agent brakes early while the trailing agent brakes late and hard, is likely the cause for the deadlocks observed in scenarios where one agent has a large initial lead (Figure 8.2**b**).

Explicit communication

We found that explicit communication in our model is highly efficient in enabling successful interactions (Figure 8.2c). An example can be seen in Figure 8.6, where two agents with initially equal distance to the intersection approaching each other. After their first re-plan at $t = 3.4\,\mathrm{s}$ (first dashed line), the agents initially begin to decelerate, while driven by the epistemic benefit of that action prompt each other for their intention (the definition of g_{prompt} can be found in Equation (D.1)). As can be seen in the plot of the prompting signal γ_A, this is not yet fully perceived by the respective other agents. While the uncertainty about the other agent's yielding intention expressed in γ_Y is reduced initially by the negative reply, it starts to build up again (see first the sharp decrease in g_{prompt} after the initial prompting and then the gradual increase). By $t = 4.6\,\mathrm{s}$ (second dashed line), the uncertainty in agent B's belief has increased sufficiently for it to decide on prompt agent

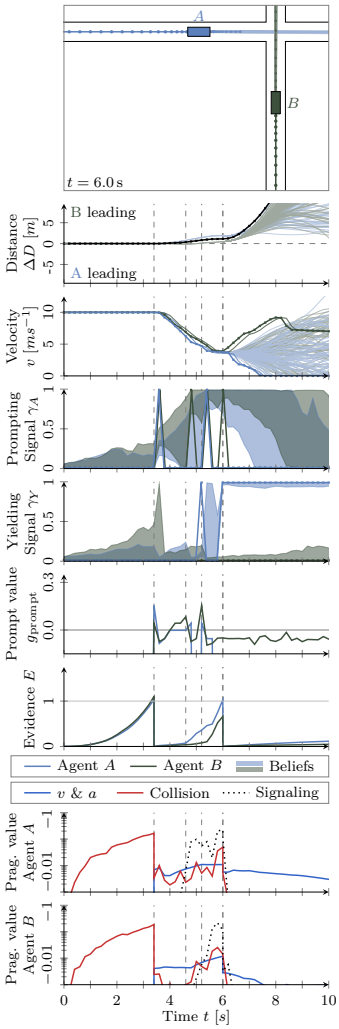


Figure 8.6: An example scenario for the interaction in which agents could use explicit communication. In addition to variables presented in previous examples, the explicit communication signals are shown (asking γ_A and yielding γ_Y); shaded regions denote the other agent's beliefs about each signal. Both agents start at equal distance to the intersection; agent B is braking slightly less sharply than agent A and is thus recognized by both agents as the leading vehicle. Both agents prompt each other for their intention (see γ_A, with the increase in epistemic value being higher than the pragmatic cost g_{prompt} (Equation (D.1))), after which agent A signals yielding (see γ_Y), thereby reducing agent B's uncertainty in its belief about agent A, giving agent B the necessary confidence to cross the intersection first.

A again at the next timestep. This repeated prompting is now sufficient for agent A to understand, and given it considers itself to be trailing, agent A sends a first yielding signal $t = 5.2\,\text{s}$ (third dashed line). After a final prompt by agent B, and with the trailing margin increasing, agent A commits to a continuous yielding signal at $t = 6\,\text{s}$ (fourth dashed line), which makes further prompting by agent B unnecessary.

Being convinced of agent A's intention to yield, agent B considers crossing the intersection to be safe (the gray lines in the ΔD plot indicate that agent B predicts is small lead to increase significantly, while the drastic drop in the collision part of the pragmatic value show that agent B no longer considers collisions to be possible). Consequently, it decides to accelerate and cross the intersection first, avoiding a potential deadlock.

It must also be noted that in this plot, not every case of $g_{\text{prompt}} > 0$ was followed by a prompting signal being send ($\gamma_A = 1$). This can largely be explained by the fact that we recovered g_{prompt} *post-hoc*, and therefore noise introduced by, for instance, the state transition function can lead to slightly different outcomes. Nevertheless, the actual prompting signals still line up consistently with the highest peaks in g_{prompt}, clearly demonstrating that prompting is indeed driven by the epistemic value that this action brings.

Explicit communication and normative expectations

Similar to the case of uncontrolled intersections, in the presence of norms the capability for explicit communication improved the agents' likelihood to successfully resolve space-sharing conflicts (Figure 8.2d).Examination of representative simulations showed that agents engaged in explicit communication only if proxemics and normative priors were insufficient to resolve the conflict. For instance, in Figure 8.7**a**, the initially leading agent A ($\Delta D(0) = -3\,\text{m}$) has a sufficient margin to execute the crossing policy decided upon at $t = 8.0\,\text{s}$ (first dashed line). Specifically, agent A already deciding to cross first at $t = 8.0\,\text{s}$ is evident in two ways. First, by its next policy re-plan at $t = 9.8\,\text{s}$ (third dashed line), it is already accelerating, as seen in the velocity plot. Second, the thin blue trajectories in the ΔD plot indicate that, at $t = 8.0\,\text{s}$, agent A predicts a substantial future lead, which is at that point only achievable by crossing the intersection. Importantly, this confidence does not require explicit communication, as after receiving a negative yielding signal following its prompting at $t = 9.0\,\text{s}$ (second dashed line), agent A assigns low probability to agent B' s (signaled) yielding intent γ_Y. Instead, agent A's confidence is driven by its belief in its own priority, which we can infer from the negligible contribution of the priority component to the pragmatic value after $t = 8.4\,\text{s}$.

If both agents start at equal distances from the intersection, priority assignment might become more ambiguous. Nevertheless, in many such cases the agents can successfully resolve the conflict without having to rely on signaling, with an example shown in Figure 8.7**b**. After stopping – agent B at $t = 10\,\text{s}$ (first dashed line) and agent A at $t = 11\,\text{s}$ (second dashed line) – both agents believe that they are leading (gray and blue curves in the ΔD plot at those respective timesteps). Consequently, each assigns itself priority, planing to cross first (see them already accelerating more or less immediately after coming to a stop) as seen in them immediately accelerating after coming to a stop without a new policy re-plan. However, realizing that this would result in a collision, agent B at $t = 11.6\,\text{s}$ decides to yield, allowing agent A to assert itself and proceed onto the intersection first.

However, there are other cases where communication is utilized extensively by the agents even in the presence of norms. For example, in Figure 8.7**c**, both agents assign

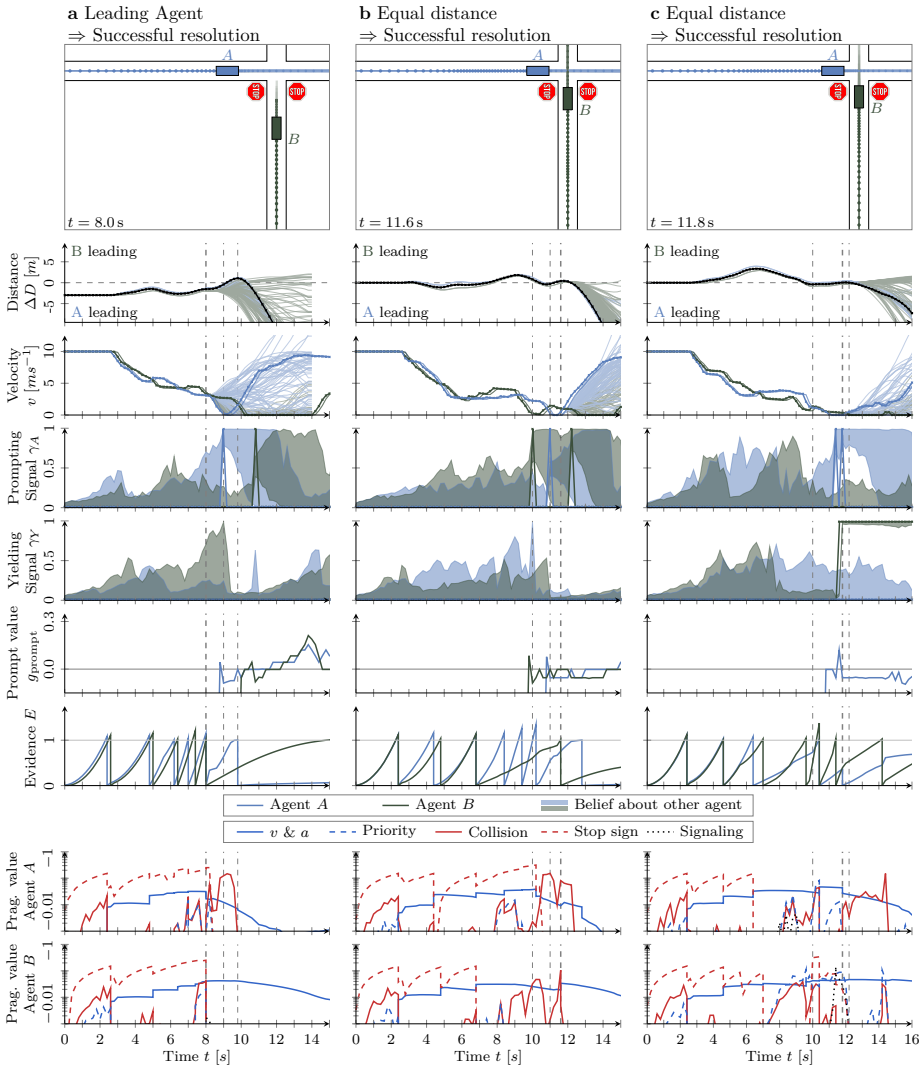


Figure 8.7: Three example scenarios for interactions with communication and different imposed norms. **a)** At an intersection where agent A is leading ($\Delta D(0) = -3\,\mathrm{m}$), it manages to pass first, solely relying on the normative expectation, with communication not necessary. **b)** The two agents start at an equal distance. As both agents assign themselves priority, they do not consider signaling yielding. In the end agent A is faster at asserting its priority, allowing it to go first. **c)** Two agents again start at an equal distance to the intersection. Initially, both agents consider the other one to be leading and therefore having priority. However, with agent B consistently signaling yielding, agent A decides to cross the intersection.

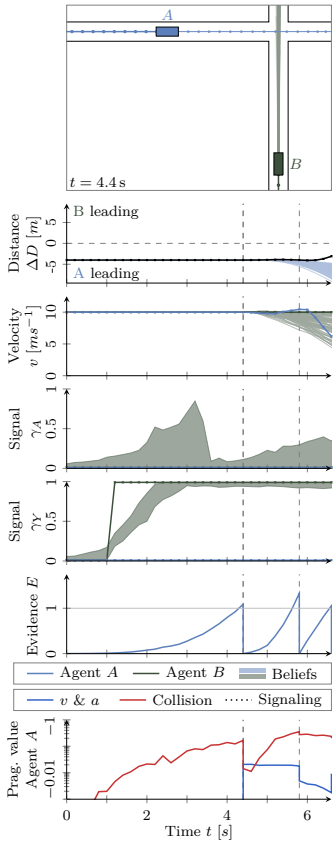


Figure 8.8: Simulation of model behavior in a provoked conflict. The leading agent A ($\Delta D(0) = -4\,\text{m}$) is modeled, while agent B is programmed to drive with constant speed while signaling yielding, successfully deceiving agent A, which results in a collision.

priority to the other one – contrasting Figure 8.5b – with both assuming the other agent to hold at the moment they come to a stop (agent A at $t = 10.0\,\text{s}$ and agent B at $t = 12.2\,\text{s}$ – the first and third dashed lines, respectively). Without communication, this scenario would have most likely resulted in a deadlock (as in Figure 8.5**b**). However, prompted by agent A, agent B decides to start signal yielding. Consequently, once perceiving this signal at $t = 11.8\,\text{s}$ (third dashed line), agent A predicts agent B to stay standing still (see the thin gray lines in the velocity plot), resulting in high confidence that crossing is safe.

8.3.3 Conflict from false signaling

The above scenarios illustrate the interactive behavior of the agents under the assumption that both agents are fully attentive and well-intentioned; hence, the interaction is virtually never safety-critical. To investigate the model behavior in an adversarial setting, we pre-programmed one of the agents (B) to drive with a constant velocity while signaling yielding to mislead the modeled agent A into causing a collision, with initial distances

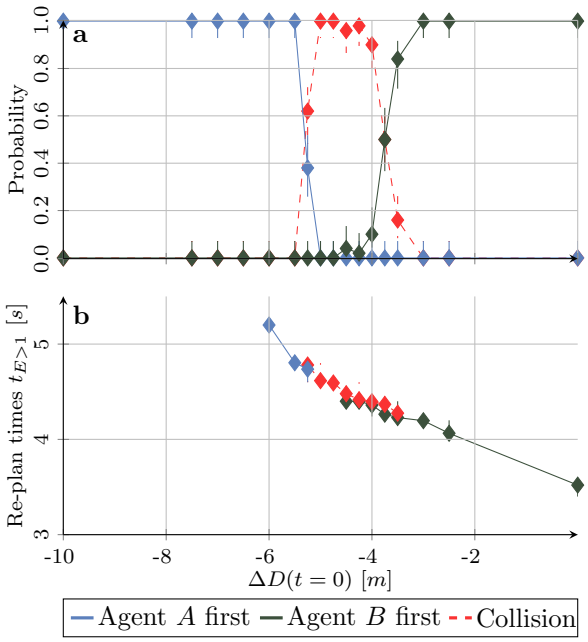


Figure 8.9: **a**) The likelihoods of the different simulation outcomes in the provoked conflict scenario, with error bars indicating 95% confidence intervals. **b**) The time $t_{E>1}$ at which the agent A first triggered a full re-plan of its policy (lower panel). Error bars show the 5th and 95th percentiles.

$\Delta D(0) \in [-10\,\text{m}, 0\,\text{m}]$. Here, the yielding signal gives the modeled agent the belief of being able to pass safely, which is then violated by agent B.

In Figure 8.8, for $\Delta D(0) = -4\,\text{m}$, such an interaction can be seen. There, at the first re-plan ($t = 4.4\,\text{s}$, first dashed line), convinced by the yielding signal of the other vehicle, A considers light acceleration to be sufficient to cross the intersection safely (the collision part of the pragmatic values falls by over an order of magnitude, and A's predictions of ΔD show high confidence that the gap will become wider, with agent B predicted to brake in the velocity plot). However, over the following timesteps, the agent slowly loses trust into the other agent's signaling, with collisions being considered increasingly likely under the current policy (see the collision part in the pragmatic value). Specifically, without observing the increasingly larger (and, according to the state transition function, increasingly improbable) decelerations required by agent B to still yield safely, agent A stops to expect them. This finally convinces A that braking is needed (resulting from the second policy re-plan at $t = 5.8\,\text{s}$, the second dashed line). However, at this point, a collision is no longer avoidable, and braking mainly serves the purpose of minimizing the relative impact velocity.

This pattern of the modeled agent A being unable to avoid a collision seems to be mostly limited to certain values of $\Delta D(0)$, ranging roughly in between $-5.5\,\text{m}$ and $-3.5\,\text{m}$ (Figure 8.9**a**). For larger initial leads, the agent is able to simply go first by maintaining its velocity or slightly accelerating, while for much smaller initial leads ($\Delta D(0) \geq -3\,\text{m}$), it

is able to brake in time. Specifically, the ability of the agent to avoid collisions seems to mostly depend on it being able to respond earlier, as shown in Figure 8.9**b**. Given how the initial reaction time is driven purely by predicted collisions (see pragmatic value in Figure 8.8), shorter reaction times are caused by the agent believing into the possibility of a collision earlier, which is conditioned on them loosing trust into the other agent's signaling. Consequently, with an earlier reaction, and less reliance on the agent B's yielding signal, agent A is able to choose a policy which avoids a collision.

8.4 Discussion

In this paper, we extended an existing active inference-based computational driver model [199, 207] to account for interactive behavior of two road users, mirroring a novel recent framing of social cognition in the active inference literature [220, 293, 315, 317, 320–322]. In this framing, the interacting agents form mutual and reciprocal beliefs about each others' future actions. If the agents' mutual beliefs are well aligned, a common ground is established and the interactive situation can be seen as governed by a single agent-neutral generative model, like when two persons are singing or dancing together. However, such alignment may be precluded by high initial uncertainty in the agents' beliefs, which can be resolved by two key means: (1) reliance on normative expectations and (2) explicit communication.

Our simulation results demonstrate how our model can provide a concrete mechanistic account of this general framing of road user interaction. In the baseline simulations (Figures 8.2**a** and 8.4), without interaction-specific normative expectations, the space-sharing conflicts can still be resolved by implicit communication as long as the initial distances between the agents are large enough (at least 5 m, corresponding to a 0.5 s time gap). In such situations, the trailing agent typically starts yielding which reinforces the leading agent' s policy to proceed first as it, despite the uncertainty, perceives that it is safe to move ahead. However, this resolution mechanism breaks down for smaller initial distances where the mutual uncertainty about the other agent' s future behavior often leads to deadlock.

Adding interaction-specific normative expectations – the "first come, first go" priority rule supported by a stop sign (a deontic cue) requiring agents to come to a stop before proceeding – constrains the beliefs of the agents and usually leads to a successful resolution (Figures 8.2**b** and 8.5). This mechanism, however, typically fails in situations where the agents apply the rule in a non-congruent way (*e.g.*, both agents believe that the other agent arrived first and thus has priority).

Our simulations also show how explicit communication (implemented by simple communicative acts such as signaling yielding in response to initial prompting), offers an efficient way to resolve uncertainty in situations that are not guided by clear normative expectations (Figures 8.2**b** and 8.5), or when the agents' interpretations of who has priority are not aligned (Figure 8.7**c**). Finally, we demonstrate how collisions can occur as the result of false expectations, in this case overly confident beliefs in misleading signaling by the other agent (Figures 8.8 and 8.9).

Our proposed framing and computational model connect road user interaction to broader contemporary accounts of social cognition, in particular the classical notion of common ground in linguistic communication [296] and modern, active inference-based approaches to joint action based on shared, agent-neutral, generative models [315, 317]. In

particular, we leverage the idea that successful road user interaction requires that agents establish a common understanding of the situation and, as a consequence, can be seen as governed by a shared generative model. In our current approach, each agent retains their own generative model which generates possible futures, including their own planned actions as well as the predicted actions of the other agent. If these possible futures are aligned, the agents thus have established a single common expectation of how the situation will play out and the agents can be seen as being governed by a single agent-neutral model, as suggested by Pezzulo *et al.* [317] for other forms of joint action. However, in our framing, such an agent-neutral model should not be seen as separate from the agents' individual generative models, but rather as a phenomenon that is established when two agents enter into a coupled interaction.

Our model also highlights how road user interaction is facilitated by normative expectations and explicit communication as different means for reducing uncertainty in order to establish a shared and precise understanding (generative model) of the traffic situation. Here, normative expectations, facilitated by deontic cues such as road signs, traffic lights and lane markings, can be understood as an example of niche construction [320, 321] as part of a general *traffic culture* that differs between geographical and cultural regions. On the other hand, explicit communication (here prompting the other agent for its intent to yield) can be seen as a specific form of epistemic action (driven by epistemic value) with the purpose to gain uncertainty-resolving information, thus serving a general function similar to that of visually scanning an intersection or checking the rear view mirror before changing lane [199].

Our model and simulations suggest an explanation for the existing empirical finding that close dynamical coupling in road user interactions is primarily observed in uncontrolled scenarios [302]. Such scenarios are similar to the baseline uncontrolled intersection in the present simulations (Figure 8.5), where the decision on who goes first and who yields emerges dynamically based on the scenario kinematics, leading to a basic form of coupling. In this coupling, once the symmetry is broken, both vehicles will act jointly to reinforce the resolution of the space-sharing conflict such that their actions become interdependent (one vehicle will continue at the current speed or accelerate and the other will decelerate to yield). However, in situations where the agents rely mainly on normative expectations (*e.g.*, in our stop sign + precedence rule scenario; Figure 8.5), the agents' behavior will be less interdependent at the sensorimotor level.

The model also accounts for the well-established finding that explicit communication is relatively rare in real-world road user interactions [300]. Since the space-sharing conflict can most often be resolved by implicit communication and direct coupling (as in Figure 8.4) or by relying on established normative expectations (Figure 8.5), explicit communication only becomes necessary when any of these mechanisms fail, for example when mutual uncertainty about each others' future action prevents progress (as in Figures 8.4**c**) or when normative expectations are misinterpreted (as in Figure 8.7**c**).

Compared to existing road user interaction models [46, 47, 101, 212, 297, 301, 302, 304–313] our model adds several novel features similar in spirit to recent computational models of social cognition and behavior outside the traffic domain [220, 315, 317, 320–322]. First, it proposes distinct mechanistic roles for dynamical coupling (*i.e.*, implicit communication), normative expectations and explicit communication in the resolution of space-sharing

conflicts. Second, in contrast to most existing models, our model posits the resolution of uncertainty in the agent's mutual beliefs about each other as the key factor underlying road user interaction behavior. Finally, many existing models are based on the game-theoretic notion of inferring and reasoning about the intent of the other agents and one's own actions in a nested and explicit way (*e.g.*, "if I do *a*, they will do *b*, and then I can do *c*, etc."). While this type of explicit reasoning is certainly possible, it does not seem necessary and may play a less important role in road user interaction than traditionally believed (see Pezzulo *et al.* [317] for a similar argument in the context of joint action).

While the model presented here provides a novel mechanistic account of several key aspects of road user interaction, it is necessarily simplified in many ways and subject to a number of limitations. First, in the current model, the reduction of uncertainty in beliefs is essentially "built into" the model rather than arising from an explicit inference process. As a result, the uncertainty in the explicit belief about the other agent's yielding intent can only be reduced through explicit communication, while implicit communication (*e.g.*, kinematics) and contextual information such as traffic rules do not directly update that belief. This limitation can lead to implausible behavior. For example, an agent's belief about the other agent's yielding intent initially evolves regardless of the agents' proxemics, at least until communication begins. Consequently, an agent may be equally likely to prompt the other for its intention regardless of whether both vehicles arrive at the intersection at nearly the same time or one arrives substantially earlier. In the latter case, however, such prompting is unrealistic, since the probability that the trailing agent proceeds first is negligible – there is little actual uncertainty about the outcome. A more principled approach would allow beliefs about yielding intent to be updated through a richer Bayesian inference process that incorporates kinematic information, traffic rules, and potentially implicit communicative cues. Under such a formulation, a substantial lead by one agent would already reduce uncertainty about yielding intent, rendering the epistemic value of prompting negligible compared to the pragmatic cost of that action (see also Maisto *et al.* [330] for a related approach). However, such an approach would require careful balancing to avoid positive feedback: high confidence in the other agent yielding would bias an agent towards predicting kinematics that reflect yielding behavior, which in turn could provide confirmatory evidence that further reinforces this belief—without any external evidence justifying it.

Second, the model does not account for accidental misunderstandings, and unrealistically assumes that signal observability is independent of distance or relative velocity. Specifically, given reduced visibility through for example car windshields, it would be reasonable to assume that perceiving a human driver's signal would require a close spatial proximity. Additionally, for such a mechanism to work properly, one would also have to imbue the modeled driver with an understanding that its own signals might not always be perceived. Furthermore, the exact method for the discussed signals (prompting and yielding) has been ignored for simplicities sake.

Third, the interaction between kinematic policy and communicative actions is very loose, with signaling actions being chosen at every timestep and a perceived yielding signal automatically triggering re-planning of the kinematic policy independent of the accumulated evidence. While this produces plausible behavior – such as quickly reacting to a prompting signal – it is not well aligned with the underlying framework of surprise

accumulation, which is primarily driven by collision avoidance [207]. More importantly, communication affects behavior through an external trigger (*i.e.*, observing a yielding signal leads to a re-plan of the kinematic policy) rather than through the same inferential process that governs kinematic decision-making. If we wanted to integrate the signaling policy into the evidence accumulation based framework for generating and updating the kinematic policy more closer, multiple changes to the current model would be necessary. While a quick reaction to a prompting signal might be achieved by increasing the pragmatic cost of not signaling yielding when required (and thereby causing a faster re-planing, where a yielding signal could be chosen), more substantial model changes are likely needed to account for timely kinematic reactions to perceived yielding signals without external triggers. One possible direction is to adjust how accumulated surprise is computed. Specifically, instead of comparing the predicted observations under the reference policy to ideal preferred observations (Equation (B.21)), one could compare them to achievable outcomes dependent on the current belief (*i.e.*, evaluating the reference policy relative to the best possible – but still feasible – alternative instead of a collection of individually ideal, but kinematically impossible set of observations). In such a formulation, the agent would not only respond to deterioration in the pragmatic value of the reference policy (*e.g.*, when braking becomes necessary to avoid a collision), but also to environmental changes that enable higher-value policies. Specifically, yielding signals would change the ego agents belief towards considering the crossing of the intersection to be safe. This would make the best possible baseline policy (crossing immediately is the ideal policy once collisions are no longer a concern) pragmatically more valuable, which – given a constant reference policy – increases the accumulated surprise and thereby triggers a timely re-plan. Additionally, by tying the signaling policy to the re-plan mechanism, we would overcome the issue above, where the agent still prompted even in cases where the outcome was already quite certain. As initial re-planning is dependent on an increasing chance of collisions, any scenario where this is deemed unlikely based on a large lead or norms would not trigger any re-planning, and therefore not allow the agent to initiate prompting. However, implementing such an approach would be computationally expensive, as it would require solving the policy optimization problem at every timestep and would likely require careful parameter tuning (such as the evidence accumulation rate λ). For these reasons, this extension was considered out of scope for the present work. Nevertheless, exploring this idea may be a promising direction for future research.

Finally, we did not compare our model to human data. Such data is not readily available for the interactive scenario analyzed here; our results were consistent with the more general observations reported in the literature (*e.g.*, the prevalence of implicit communication in human road user interactions [300]), but future work should focus on bringing the model closer to scenarios for which detailed human data are available or could be collected. Still, even without direct comparisons to human data, our results are valuable, demonstrating that active inference can unify communication, coupling, and normative mechanisms into a general framework of road user interaction while producing results consistent with general intuition. Additionally, our model can explain why communication and norms are needed to increase the efficiency of traffic interactions, while highlighting the danger that adversarial agents can pose (for example by misleading communication). Furthermore, by bringing the model to an interactive scenario, we expand its potential usefulness

in the simulation-based evaluation of automated vehicles [17, 18, 288]. Importantly, our insights about the importance of uncertainty reduction (and the role expectations and communication play in it) could be transferred to other human interactions as well, such as pedestrians on sidewalks [331].

8.5 Conclusion

In this work, we used a novel model, based on active inference and implementing communication and normative expectations, to simulate the interaction between two agents at an intersection. We showed that both communication and traffic norms can be used to increase the likelihood of a successful and timely interaction by reducing the uncertainty in road users' beliefs about each other. However, this depends on the agents being collaborative and acting in good faith, and we show that misleading intention signaling in road user interactions can cause collisions. Our results, although not validated against human data, demonstrate that active inference can provide a unified account of the role of behavioral coupling, communication, and normative mechanisms in road user interaction. Additionally, we highlight the risks introduced by adversarial or misleading agents.

III

Conclusion and future work

9

Conclusion

In this thesis, we have explored the field of modeling human behavior in traffic from two perspectives. First, we focused on the evaluation of state-of-the-art prediction models (Part I). There, we identified several shortcomings in current evaluation methods and proposed a holistic framework able to address those issues. Afterwards, our attention shifted to the development of explanatory models of human behavior in traffic (Part II), in an attempt to overcome a major problem in human behavior modeling: the lack of explainability and generalizability of current data-driven models. Below, we summarize the results obtained in these two parts in greater detail.

9.1 Evaluating Prediction Models

In the first part of this thesis, we focused on the evaluation of human behavior prediction models in traffic. Having identified several shortcomings in current evaluation methods (discussed in detail in the introduction to Part I), our goal was to develop methods for a more holistic evaluation of such models. Additionally, we aimed to apply those methods to a broad range of state-of-the-art models in order to better understand their limitations and the true state of progress in the field.

In Chapter 2, we introduce a novel evaluation approach focusing on gap acceptance scenarios. In such scenarios, we show that current state-of-the-art models often struggle when tested on subsets containing the most safety-critical or unintuitive cases. In the proposed definition of gap acceptance problems – where a traffic agent has to decide whether to pass in front of or wait for an oncoming agent with priority – these scenarios are those with the smallest accepted and longest rejected gaps. Specifically, most state-of-the-art models are frequently unable to outperform very simple classifiers, such as logistic regression, in correctly predicting human acceptance behavior in these scenarios. In fact, their performance often approaches that of a random classifier. Additionally, even at the latest point where reasonably safe avoidance maneuvers are still possible, avoiding false negative predictions (*i.e.*, failing to predict gap acceptance behavior and thus underestimating the risk of a collision) can only be guaranteed by adopting an extremely conservative strategy of assuming that almost all gaps will be accepted. This, in turn, requires a large number of unnecessary avoidance maneuvers.

Meanwhile, in Chapter 3, we introduce the comprehensive open-source framework *STEP* for evaluating human behavior prediction models in traffic. This framework includes most commonly used datasets and implementations of several state-of-the-art prediction models. Furthermore, it enables the comprehensive testing of a model's robustness against input perturbations such as adversarial attacks, where an antagonistic agent – through its chose trajectory – tries to confuse and mislead the prediction model. It also provides a large library of evaluation metrics (leveraging the method proposed in Chapter 4 to extend probability-based metrics to models previously unable to use them). Additionally, it offers precise control over the splitting of training and testing data as well, allowing users to focus on safety-critical scenarios. The increased control helps estimating the stochastic effects of model training on performance as well, which is essential for meaningful evaluation not only of a specific model instance but of the underlying model architecture itself. With this tool, researchers can avoid the effort of reimplementing common datasets and state-of-the-art models, allowing them to focus instead on the development and rigorous evaluation of novel approaches.

In Chapter 4, we presented an improved method for estimating the probability density function underlying a given set of samples. Specifically, this approach combines standard kernel density approaches with non-parametric clustering (using the OPTICS algorithm [166]). To assess this method, the paper also proposes several novel metrics that operate solely on samples drawn from a test distribution, without requiring access to pointwise density evaluations. According to those metrics, the proposed method outperforms previous state-of-the-art approaches, particularly for non-normal distributions, such as trajectories with multi-modal characteristics. While broadly applicable, this method is especially useful in human trajectory prediction, where multi-modal predictions are common, but many models do not provide access to an explicit probability density function.

Expanding upon previous work, Chapter 5 proposes a new approach for generating meaningful and kinematically feasible adversarial attacks against trajectory prediction models (which is also incorporated into *STEP*). An analysis of previous methods shows that much of the performance degradation reported in earlier work was likely not due to a lack of model capability. Instead, the future ground-truth trajectories used for evaluation are kinematically unreachable from the adversarial agent's perturbed past trajectories. Furthermore, these perturbed trajectories are often dynamically infeasible as well, artificially pushing these test cases unreasonably far outside the training domain. Consequently, expecting models to predict the future ground truth under such conditions seems unreasonable. This renders the results in earlier papers meaningless as an evaluation of a trajectory prediction model's robustness. To address this issue, the new approach does not perturb the recorded positions of the past trajectories themselves, but rather the underlying control inputs (*e.g.*, acceleration). This ensures that the generated perturbed trajectories are kinematically feasible and could, in principle, be executed by a real-world vehicle. Furthermore, by imposing additional constraints that also consider the positions reachable from the end of the perturbed trajectory, the proposed method enables meaningful comparisons to the recorded ground-truth future. As demonstrated in the results, these adversarial attacks can still significantly degrade the performance of state-of-the-art human behavior prediction models in multiple scenarios. Importantly for real-world safety, exact knowledge of the attacked model is not required, as perturbations generated against

a different model trained on the same data were nearly as effective. These results are reproduced in Chapter 3 for a larger set of prediction models, reinforcing the demonstrated shortcomings of current state-of-the-art models under adversarial attacks.

Further analysis with *STEP* (Chapter 3) also revealed that these models still struggle to generalize to unseen scenarios. Even benign cases (*i.e.*, neither adversarially generated nor extremely safety-critical) can be problematic, particularly when transitions across cultural or regional boundaries are involved. For example, training on US-centric data and testing on data collected in Germany leads to significant drops in performance. Moreover, further experiments indicate that the minor differences between models observed in many benchmarks [72] are likely not statistically significant under the noise introduced by their initialization and training process. Consequently, any inference made about superior model architectures – beyond general observations such as the benefits of denoising diffusion made across different fields [39, 332–334] – is likely standing on very shaky ground. While there are some researchers who take this into account [38, 335], this point is still underappreciated in the field at large. Lastly, there seems to be room for improvement by optimizing the amount of past observations passed to a trajectory prediction model (*e.g.*, the observed time horizon or the frequency of observations). Specifically, less input information is shown to be beneficial for prediction performance even on large datasets like *Argoverse* [72].

Taken together, the contributions presented in this part are able to address several of the discussed shortcomings [61, 63, 69, 336] in the evaluation of human behavior prediction models. First, we introduced new methods that enable a stronger focus on behavior-level outcomes (Chapters 2 and 3) and improve the selection of evaluation data (Chapter 2), allowing model evaluation to concentrate more directly on safety-critical situations. Second, we enabled the application of density-based metrics to all prediction models (Chapter 4). Finally, we proposed approaches for the generation of more meaningful adversarial attacks (Chapter 5), enabling improved assessment of a prediction model's robustness against realistic perturbations. To make these contributions broadly usable, we integrate them into the open-source framework *STEP* (Chapter 3), which provides a unified infrastructure for datasets, models, evaluation metrics, and robustness testing. While previous works [67, 128] have attempted to introduce similar tools, *STEP* is considerably more comprehensive. In particular, it includes a larger library of datasets, models, and evaluation metrics, as well as far more detailed documentation, improving its accessibility and usability.

Applying these methods revealed that current state-of-the-art models, despite years of gradual improvements on common leaderboards [72, 73, 134, 135, 142], still struggle with reliability and generalizability, particularly in safety-critical scenarios. This finding aligns with previous critiques, both for trajectory prediction specifically [56, 61] and for deep learning systems more broadly [122, 337–339]. However, the key contribution of this work is that these limitations are demonstrated in a unified and systematically controlled evaluation setting.

Overall, these results demonstrate the need for more rigorous and comprehensive evaluation of human behavior prediction models, particularly in safety-critical contexts. At the same time, this work provides a structured foundation for addressing these limitations through improved evaluation design and tooling, rather than isolated benchmark

improvements alone. Nevertheless, addressing these limitations fully likely requires complementary advances in model design. In the following part of the thesis, we explore one possible approach that is found in cognitively inspired modeling.

9.2 Explaining Human Behavior

In the second part of this thesis, we shifted our approach and moved from the evaluation of deep learning-based models to the exploration of models of human driving behavior inspired by cognitive theory. In particular, we aimed to investigate how such models could be constructed to be as generalizable and explanatory as possible. From there, we then conducted initial assessments of their performance compared to state-of-the-art deep learning-based models, to evaluate whether their often-touted potential [48, 122] could be realized in practice.

Extending a cognitively inspired model focusing on pedestrian street crossing behavior [48] to general gap acceptance behavior in traffic demonstrated that even models with a narrow scope can be successfully expanded to a larger set of problems based on the strength of their underlying cognitive mechanisms (Chapter 6). Specifically, the expanded model was able to achieve comparable performance to state-of-the-art deep learning trajectory prediction models [30] in multiple gap acceptance scenarios. This demonstrates that, in principle, cognitively inspired models can achieve similar predictive performance to deep learning models, while also providing explanations of the underlying mechanisms leading to the observed behavior.

Meanwhile, a novel model of human driving based on active inference – a theory positing that all sentient creatures behave according to the principle of surprise minimization – demonstrated even more general applicability (Chapter 7). Focusing on collision avoidance, our model was able to closely match human avoidance behavior in multiple scenarios. Importantly, this was also the case for those scenarios unseen during parameter tuning. This indicates that active inference is a promising framework for generalizable models of human behavior in traffic. Additionally, detailed ablation studies showed that the inclusion of specific cognitive principles (*e.g.*, evidence accumulation, visual-angle-based perception, and norm-conditioned behavior prediction of surrounding agents) is crucial for achieving this performance. It is noteworthy that a state-of-the-art deep-learning behavior prediction model [76] failed to reproduce this human behavior (response times and chosen avoidance behavior). In general, these results demonstrate that knowledge-based models such as those based on active inference can indeed provide a viable alternative to deep learning-based models, at least in safety-critical edge-case scenarios such as the collision avoidance scenarios studied here.

Finally, Chapter 8 demonstrated the adaptability of the active inference model introduced above. Specifically, we extended it to an interactive intersection scenario where two human agents attempt to resolve a space-sharing conflict. This work showed the key role that uncertainty about future actions of other agents plays in human behavior. Specifically, in a baseline scenario – without any indicators allowing the model to reduce this uncertainty – the modeled agents often became stuck at the intersection. However, this could be resolved in two major ways, the first being the introduction of traffic rules such as "first come, first go" priority. This rule allowed the leading agent to discard kinematically possible but norm-violating interference by the other agent, enabling it to proceed

first. The second approach, explicit communication, had a similar effect. For instance, a yielding signal from one agent could give the other agent confidence that collisions are unlikely and that crossing the intersection is safe. Given that in the first method both agents still need to agree on priority, explicit communication was shown to be the more effective method. Importantly, the ability to easily integrate these mechanisms into the existing model of collision avoidance further supports the promise of active inference as a general model of human driving. However, reliance on such mechanisms can also introduce vulnerabilities, as they depend on benign behavior. For example, an adversarial agent misleadingly signalling yielding – thereby exploiting the trust of the other agent – could induce collision outcomes with high probability.

In summary, our work demonstrates that models grounded in well-established cognitive theories can accurately capture complex human behavior in traffic scenarios. This builds upon previous research on cognitive process models showing that they can reproduce certain aspect of human behavior. For instance, drift-diffusion models and related evidence accumulation frameworks [45] have been shown to account for human decision-making in dynamic traffic interactions [46, 47, 340], reproducing observed response time distributions and behavioral variability. Our work also strengthens previous findings [48, 226, 341] showing the crucial role of perceptual limitations that arise from noisy, angle-based visual perception. This further reinforces the view that such principles play a fundamental role in human cognition. Incorporating such well-supported mechanisms allows our models to explicitly represent internal decision-making and perception processes and to capture behavioral patterns that are difficult to reproduce with purely data-driven approaches.

Besides providing interpretable explanations of the underlying mechanisms leading to the observed behavior, cognitively grounded models can also lead to improved performance. Specifically, our results suggest that they often match or even outperform state-of-the-art deep learning-based behavior prediction models. This effect becomes more pronounced in rare safety-critical scenarios, where deep learning models tend to degrade in performance (Appendix C.3). This finding lends empirical support to claims frequently made in the literature that mechanistic models of cognition can provide both competitive predictive performance and improved interpretability compared to "black-box" deep-learning models [48, 57, 122, 342]. In domains such as automated driving, where safety and transparency are critical, such interpretable models may therefore represent a particularly promising direction.

In particular, the active inference framework appears especially promising as a foundation for generalizable models of human behavior, as it provides a unified theory of perception, action, and learning in biological systems [57, 218, 343]. By framing behavior as the minimization of expected free energy under uncertainty, active inference naturally integrates perception, prediction, and decision-making into a single principled framework. This makes it well suited for modeling complex and interactive behaviors in traffic scenarios (Chapter 7). Moreover, the framework is inherently extensible: additional cognitive mechanisms or interaction modalities can be incorporated in a principled manner. For example, as demonstrated in Chapter 8, the addition of explicit communication between agents can reduce uncertainty during interaction. Future extensions could similarly integrate additional aspects of human behavior, such as gaze allocation or attention dynamics.

This could enable increasingly realistic models of multi-agent traffic interactions.

Overall, while our work does not culminate in a universal model of human behavior in traffic, our results show that working towards that goal is a promising direction. In particular, we reaffirm the promise of such an approach in terms of generalizability and robustness, showing that this can be achieved without completely sacrificing the performance of deep learning models. However, the path towards such a complete model is still long, with many steps still to be taken.

10

Extensions and Impact

While the work presented in this thesis has made significant contributions to both the discussed directions, no work is ever complete or perfect. Several simplifying assumptions were necessary, and important open questions remain. These limitations and directions for future work are discussed in more detail in the following sections.

10.1 Limitations

Throughout this thesis, while we tried to be as thorough as possible, trade-offs were necessary, and the resulting findings should therefore be interpreted with caution. While many of these issues were discussed in the respective chapters, we briefly highlight several overarching limitations here. While we believe that *STEP* is an important contribution that enables more comprehensive evaluation of these models, it must be acknowledged that our testing and applications were limited in scale. Although our experiments revealed important insights into the shortcomings of recent state-of-the-art models (Chapters 3 and 5), the rapidly increasing number of models published in the field [56, 59, 344–347] means that our evaluation is far from comprehensive. Because only a relatively small subset of models was tested on a limited number of scenarios, our claims about flaws in current state-of-the-art models should be interpreted with caution. Nevertheless, the fact that we observed such flaws across multiple models and scenarios – with most of the newer models not addressing them directly – suggests that these issues are likely to be widespread in the field.

Furthermore, it should be noted that *STEP* is currently designed for models that use preprocessed input data in a top-down, bird's-eye-view representation. As recent work in the field, however, moves toward the use of raw sensor inputs (*e.g.*, front-view camera images [345]) the applicability of *STEP* might become increasingly limited. Consequently, our claims regarding the (lack of) robustness of current state-of-the-art models are based on an analysis covering only a subset of existing approaches and should therefore not be overgeneralized.

Additionally, the relevance of adversarial attacks has so far been investigated primarily in simulation, and their overall importance – especially for real-world applications – remains debated [348]. In addition, even when made as meaningful as possible, adversarial

robustness represents only one aspect of model robustness [84]. A more comprehensive assessment would therefore also require testing models against other sources of input perturbation, such as sensor noise.

Similarly, while our results highlighting the promise of cognitively inspired models (Part II) are encouraging, they should also be interpreted with caution. In particular, the active inference model presented in Chapters 7 and 8 makes many simplifying assumptions and has only been tested on a limited number of scenarios. For example, the implementation of norm-conditioned behavior prediction (Section 7.2.2) was intentionally minimalistic, incorporating only norms that were directly relevant to the evaluated scenarios. Moreover, the implementation of these norms was heavily optimized computationally by exploiting assumptions about the known scenario geometry (*e.g.*, basing the lane-following norms in Chapter 7 solely on lateral position). As a result, although the findings are promising, substantial additional work would be required to make the model applicable to new scenarios and to test its claimed generalizability in a more systematic manner. The conclusions regarding the potential of cognitively inspired models should therefore be interpreted cautiously, as they are based on a limited set of scenarios and model implementations.

The requirement for more general – and therefore more complex – model implementations also exacerbates the previously disregarded issue of scalability and computational efficiency. Even the comparatively simple model implementations evaluated in this thesis are generally much slower at inference than large deep learning models. On the one hand, this would pose challenges for deployment in real vehicles, where real-time capabilities are required. On the other hand, it is also disadvantageous for large-scale testing of automated vehicles in simulation, where turnaround times are a major concern. Consequently, significant effort would be needed to improve the computational efficiency of such models. This becomes even more critical as attempts to increase generality (*e.g.*, through map-based or dynamic traffic norms) will inevitably introduce additional complexity.

10.2 Future work

Building on the findings presented in Chapter 9, several directions can be pursued for further work. First, we will cover more immediate extensions of the work presented in this thesis. After that, we discussed more long-term prospects that can further advance the field of human behavior prediction in traffic.

10.2.1 Evaluating prediction models

While we believe that the structured evaluation framework presented in Chapter 3 provides a solid foundation for improved evaluation of human behavior prediction models, there remain several avenues for further enhancement:

- **Improved metrics:** One important direction is the development of improved evaluation metrics that more accurately capture the safety implications of prediction errors. Specifically, such metrics could incorporate domain knowledge about traffic scenarios, enabling a more meaningful assessment of model performance in critical situations. For example, instead of focusing purely on kinematic states, aligning

predicted and observed quantities such as response times could help better capture how well a model reflects human-like behavior.

- **Large-scale testing:** An additional avenue is the systematic evaluation of state-of-the-art prediction models – using *STEP* – across a broad spectrum of safety-critical scenarios to better understand their failure modes. Such tests would provide a clearer picture of the limitations of current models and help establish the true state of progress in the field. A more detailed analysis of the optimal number of input observations (which currently varies widely across benchmarks) could also be beneficial. Concretely, it could help us understand how much past information is actually required for accurate predictions.

- **Improved adversarial attacks:** Another direction involves a more detailed exploration of adversarial attacks. Future studies could investigate different attack formulations, including alternative loss functions. Specifically, we should analyze how these impact model robustness (*e.g.*, whether attack effectiveness can be improved independently of the magnitude of perturbations). In parallel, potential defenses against adversarial attacks also deserve closer study, such as randomized smoothing [188] or retraining models on adversarially generated samples.

- **Complete robustness evaluation:** Adversarial attacks are only one source of input perturbations for prediction models. Many other sources of perturbation, such as sensor noise, are relevant for real-world applications but remain even less studied than adversarial attacks [84]. Consequently, we should consider incorporating such perturbations into *STEP* (the interfaces already support this) and testing model robustness against them. This would be an important step toward a more complete evaluation of robustness under real-world conditions.

- **Closed-loop evaluation:** Furthermore, extending *STEP* to closed-loop simulations – where we advance the world according to the model's predictions after each timestep, similar to the approach in *Trajdata* [67] – would allow models to be tested in dynamic, interactive settings. This would enable a more consequential evaluation of model performance. Consequently, it could enhance the framework's ability to provide comprehensive, realistic, and meaningful assessments of human behavior prediction models.

- **Additional scope:** Lastly, incorporating additional datasets, models, and metrics would increase the framework's versatility, relevance, and usefulness for other researchers. This could help it become a standard tool for evaluating human behavior prediction models in traffic.

10.2.2 Explaining human behavior

Meanwhile, the second part of this thesis established that cognitively inspired models, such as those based on active inference, can be highly effective in modeling human behavior in traffic – especially in safety-critical and other interactive scenarios. However, it is important to recognize that the work presented in Part II is only a first exploration of this direction. In practice, many challenges remain before such models can reach the

same level of maturity as current state-of-the-art deep learning-based models. While not exhaustive, a roadmap toward this goal would likely include the following steps:

- **Parameter learning:** A crucial next step is to enable the optimization of model parameters based on large-scale datasets. In the current work, most parameters were hand-tuned, which limits scalability and adaptability. Automatic parameter optimization to better fit human data could improve predictive performance while also enabling faster adaptation to new scenarios and agents. This step becomes particularly important if aspects of these cognitive models are replaced by neural networks. For example, implementing the preference prior that describes the agent's preferred observations in the active inference model as a neural network would allow for greater expressivity while reducing the need for strong modeling assumptions. Such integration would combine the strengths of knowledge-based modeling with data-driven flexibility. In principle, zero-order (*i.e.*, gradient-free) optimization, such as used in Chapter 6, is always viable option. Nevertheless, more efficient gradient-based approaches are likely required to scale such models to larger datasets and higher-dimensional parameter spaces.
- **Cultural influences:** Another key area for future research is the incorporation of cultural and contextual influences on driving behavior. Human traffic behavior is strongly shaped by local traffic norms, enforcement practices, and social conventions. All of these vary significantly across regions and countries. Extending the model to account for these differences would improve generalizability and enable better prediction of behavior in diverse real-world settings.
- **Improved perception modeling:** Enhancing the modeling of human perception also represents a critical point for future work. Current perceptual modeling is relatively simplified (e.g., using angle-based perception only for longitudinal states or assuming a 360-degree field of view). Incorporating features such as gaze direction, attention allocation, and reasoning about partially or fully occluded agents could enable more realistic modeling of human perception. Additionally, the influence of these perceptual factors on decision-making – including how agents reason about potentially hidden traffic participants [281] – should be evaluated as well. Incorporating such perceptual sophistication would improve both the realism and predictive accuracy of the model, particularly in dense or complex traffic environments.

10.2.3 Long term prospects

While the above points are more immediate improvements that could be made to the work presented in this thesis, there are also longer-term prospects that could be pursued. These aim to build upon a more holistic integration of the work presented here.

- **Hybrid models:** Assuming that the parameter learning issues of cognitively inspired models can be overcome sufficiently, a promising direction would be the development of hybrid models. Such models could combine the strengths of both cognitively inspired models (*i.e.*, explainability, generalizability, and reliability) and data-driven approaches. The latter excel in flexibility and in capturing complex, nonlinear relationships in data. By integrating these approaches, it may be possible to

create models that not only perform well across a wide range of scenarios but also provide insights into the underlying cognitive processes driving human behavior in traffic. For example, a hybrid model could use a cognitively inspired framework as its core structure [349, 350], while leveraging deep learning components to capture complex patterns in the data that are difficult to model with traditional knowledge-based approaches. This could reduce the need for hand-crafted assumptions and improve scalability. Of course, other integration strategies are also possible. For instance, one could switch between a cognitively inspired model and a deep learning model depending on the current scenario [96] or use a deep learning model to predict the errors of the cognitively inspired one [100]. Nevertheless, the general idea of combining the strengths of both approaches is likely to be a fruitful avenue for future research. One specific possibility worth mentioning is the use of a deep learning model to replace the *projected normative probability* (Section 7.2.2) that encodes the agent's understanding of traffic rules and similar norms in our active inference model of driving behavior. This would better capture the complex and dynamic nature of traffic norms compared to manually encoding scenario-dependent and often discretionary rules. Alternatively, it should even be possible to replace the *generative model's state transition function* and use a neural network to predict the behavior of the surrounding agents directly, taking into account both norms and vehicle dynamics. Importantly, such an approach would still preserve the overall structure and interpretability of the active inference framework.

- **Extensive testing of cognitive models:** A final challenge is the evaluation and validation of cognitive models in more complex and realistic scenarios. The scenarios considered in this thesis provide a well-defined testbed that allows for controlled evaluation of the proposed models, but real-world traffic involves multiple agents with complex and dynamically evolving interactions. Evaluating whether the model can produce realistic human behavior under such conditions will be crucial to demonstrate robustness, practical applicability, and generalization beyond simplified settings. Ideally, this could be achieved by integrating such models into an improved version of *STEP* (for instance, one with closed-loop testing). Specifically, this would simplify running extensive simulations to compare model behavior against human data across a wide range of scenarios. Furthermore, such an approach would ensure that such models are tested to the same extent as deep learning-based approaches (Part I already provides examples of such experiments). Consequently, this can ensure that such models are not only explanatory, but also robust, generalizable, and safe. Additionally, this would allow us to better understand the limitations of these models and identify areas for further improvement.

As can be seen, while the work presented in this thesis may represent an important step towards more robust and explainable models of human behavior, it is only a first step. Considerable room remains for future research and subsequent work to build upon it.

10.3 Impact

Stepping back towards a wider outlook, this thesis makes multiple valuable contributions to both the assessment and the design of human behavior prediction models in traffic.

On the evaluation side, we show that current state-of-the-art models still tend to struggle in safety-critical and rare scenarios, which could be overlooked in evaluation approaches standard to the field. For instance, these models frequently fail to generalize to unseen situations and remain sensitive to perturbations such as adversarial attacks. By highlighting these limitations, we hope to encourage the field to take these issues more seriously during the development and validation of new models. This should drive the development of more robust and generalizable models of human behavior in traffic. Such a development could be seen as a next step for automated driving systems, as enabling them to better handle the complex and dynamic nature of real-world traffic is essential for ensuring safety and reliability.

In these efforts, our work on introducing novel benchmarks implementing robust evaluation methodologies – culminating in the open-source *STEP* framework – provides the tools necessary to systematically evaluate such new models and screen them for these limitations. Specifically, it proposes methods to better quantify model reliability, gives more control over the scenarios chosen for model validation, and estimates the statistical significance of observed differences between model architectures. This will enable researchers and practitioners to evaluate human behavior prediction models in a more realistic, safety-critical context – an essential step for models intended for deployment in autonomous driving systems. By releasing *STEP* as an open-source tool providing a large library of datasets, models, and evaluation protocols, we hope to enable other researchers to reproduce experiments, benchmark new models, and explore additional evaluation metrics. Ideally, with a wide-ranging adoption of *STEP* in the field, it could become a standard tool for evaluating human behavior prediction models. This should lead to more rigorous and comprehensive assessments of model performance and robustness across the research community. Importantly, *STEP* could allow for the standardization of model validation in the field, making the evaluation of models both more transparent and faster. This could, on the one hand, speed up model development, as researchers would no longer have to spend time on implementing model validation. One the other hand, it would also provide a much clearer overview of the current field, making it easier to discern promising modeling approaches.

This thesis also demonstrates that cognitively inspired modeling can complement advances in the evaluation of data-driven models, with the strengths of the former addressing precisely the weaknesses exposed by the latter. Experiments show that cognitively plausible models are highly robust, predicting human behavior in safety-critical traffic scenarios at levels comparable to – or exceeding – comparatively brittle state-of-the-art deep learning models. Additionally, by integrating principled cognitive mechanisms such as evidence accumulation, perceptual noise, and expectations of norm compliance, our models achieve robust generalization across diverse scenarios. Specifically, we demonstrated successful use of the model in both collision avoidance and space-sharing interactions. At the same time, this structured design allows these models to provide interpretable explanations of the underlying decision-making processes – an increasingly important requirement in safety-critical domains. For example, regulations require autonomous vehicles to perform at least as well as a "competent and careful human driver" [20, 351]. Consequently, any behavior reference model used as a human baseline in simulation would likely need to be explainable, so that its compliance with this standard can be clearly

demonstrated to regulators. Furthermore, broader emerging regulatory frameworks, such as the *EU AI Act* [352], mandate transparency and risk assessment for high-risk AI systems, including autonomous vehicles. This further increases the importance of interpretable models, even though our active inference model currently lacks the sophistication to be viable for direct application in autonomous vehicles. *Waymo* – widely considered a leader in autonomous vehicles [353] with a strong record of safe driving outcomes [354] – serves as an example of the industrial application of such models. In particular, Waymo uses the mechanistic *NIEON* (Non-Impaired human driver with their Eyes ON the conflict) driving behavior reference model [209] as a behavioral benchmark for collision avoidance performance [355, 356] as a human baseline to compare their autonomous vehicles against. Such interpretable insights allow companies like Waymo to further improve safety, identify edge cases, and validate system robustness before deployment [19]. In this role, our proposed active inference model can provide the basis for a new generation of more sophisticated driving behavior reference models able to represent human driver behavior with higher fidelity. This could eliminate the need for manual annotation of human behaviors such as response times and, consequently, broaden model applicability.

While autonomous driving is the central field discussed in this thesis, our work on human behavior models can contribute to a wider range of problems. These examples span areas such as robots assisting humans in industrial applications [357, 358], human-aware navigation for mobile robots in shared spaces such as hospitals [359–361], or collaborative manipulation between humans and robots that requires both safety and intent inference [362]. In such human-robot interaction (HRI) settings, predictive models of human behavior can improve performance and safety by enabling robots to anticipate human actions, adapt their policies online, and coordinate effectively with human partners. In particular, our work on active inference should have broader applicability. Concretely, concepts such as affordance-based policy selection (even if the action space differs), normative expectations reducing uncertainty about other agents, and reaction times controlled by evidence accumulation could also be useful in these applications.

Overall, this work advances the development of human behavior models that are both reliable and interpretable. By providing a new framework for rigorous evaluation methodologies, we enable the identification and mitigation of key vulnerabilities in predictive models while simultaneously reducing the effort required for their validation. This establishes a foundation for deploying human behavior prediction models in real-world autonomous driving systems, where proven reliability in complex interactions is essential. Furthermore, our work on novel cognitively inspired models highlights their potential for enhancing generalizability and robustness in safety-critical scenarios. Concretely, our active inference framework shows that models grounded in first principles – rather than purely data-driven heuristics – can satisfy both performance and transparency requirements. By also offering insight into the mechanisms underlying human behavior, such models have already demonstrated significant potential for industrial applications. In the long term, this thesis aims to pave the way toward an explainable, generalizable, and robust approach to human behavior modeling. By combining the strengths of both data-driven and cognitively grounded methods, such models will be suitable for the most safety-critical driving scenarios.

IV

Appendices

A

Details of the Gap Acceptance Benchmark Framework

In this chapter, we provide additional implementation details for the work from chapters 2 and 6. It uses the notation introduced in chapter 2.

A.1 Extracting 1D data from 2D data

To allow for a transformation from two-dimensional (2D) trajectories to quasi-one-dimensional (1D) ones, it is important to give the respective agents (besides the ego vehicle V_E that offers the gap and the target vehicle V_T that has to decide whether to accept it) certain roles.

V_1: The vehicle driving along V_E's path P_E in front of V_E. Inclusion is justified because of the need to define t_S.

V_2: The vehicle following V_E along P_E. This is justified by studies showing that the size of the next gap can influence the decision making of the target vehicle V_T.

V_3: The agent preceding V_T along its path P_T. It can for example block V_T from accepting a gap, or force a decision as an upcoming roadblock (this can also for instance be the end of the merging lane on highways), but also goad V_T into accepting the gap by following a specific example. Therefore, its inclusion is justified.

Based on those roles, the following one dimensional distances can be defined:

D_C: The distance of V_E to the contested space. It is positive if V_E is still offering the gap. It is measured from V_E's front bumper.

D_A: The distance of V_T to the contested space. It is positive as long the gap is not accepted. It is again measured from V_T's front bumper.

D_1: The distance between V_1 and V_E along P_E from bumper to bumper. It is positive if V_1 is in front of V_E

A

D_2: The distance between V_2 and V_E along P_E from bumper to bumper. It is positive if V_E is in front of V_2.

D_3: The distance between V_3 and V_T along P_T from bumper to bumper. It is positive if V_3 is in front of V_T.

L_E: The size of the contested space along P_E.

L_T: The size of the contested space along P_T. While this concept is fairly easily to understand for clearly crossing trajectories, determining L_T is more difficult if P_E and P_T start to continuously overlap after entering the contested space. Examples are highway lane changes or cases where V_T enters a roundabout. In such situations, the size of L_T might depend on current kinematic information, i.e. $L_T = L_T(t)$, and might be defined as the distance for which V_T would have to accelerate to to prevent being rear-ended by V_E.

Those seven distances than from the input provided to the prediction models unable to deal with two dimensional inputs, such as the *Commotions* model [48]. Their definition in regards to the two dimensional trajectory data has to be defined for each dataset individually. However, those transformations from 2D to 1D data are not invertible, which limits such models using this extracted input data to the prediction of the binary choice (accept/reject a gap) and the time of accepting the gap.

A.2 Datasets

Four datasets were used in the experiments discussed in the relevant chapters. While there are differences in the implementation of those datasets, some aspects of their implementation, which follow below, are general.

- The following parameters are identical across all datasets: $a_{\text{brake}} = 4\,\text{ms}^{-2}$, $\delta t = 0.2\,\text{s}$, $t_\epsilon = 0.01\,\text{s}$, and $n_p = 100$.
- The Δt needed to define constant sized gaps is always created by maximizing the term

$$\min\{N_A,\, N_{\neg A}\}\,, \tag{A.1}$$

  where

$$N_A = \sum_i a_i\,,\; N_{\neg A} = \sum_i (1 - a_i) \tag{A.2}$$

  are the number of samples that fulfill all requirements to be included in the final dataset based on the current choice of Δt, which influences the chosen t_0.
- A common definition of C_S, C_C, and C_A (which are used to extract t_S, t_C, and t_A respectively) is used, based on the extracted 1D data:

$$\begin{aligned} C_S(t) &: D_1(t) - D_C(t) = L_E \wedge \dot{D}_1(t) > \dot{D}_C(t) \\ C_C(t) &: D_C(t) = 0 \wedge \dot{D}_C(t) < 0 \\ C_A(t) &: D_A(t) = 0 \wedge \dot{D}_A(t) < 0 \end{aligned} \tag{A.3}$$

- For all datasets, the following definitions are used for the variables $t_{\underline{C}}$ and t_{brake} required for extracting t_C and t_{crit}:

$$\begin{aligned} t_{\underline{C}}(t) &= \frac{D_C(t)}{\max\left\{-\dot{D}_C(t), 0\right\}} + t \\ t_{\text{brake}}(t) &= \frac{\max\left\{-\dot{D}_C(t), 0\right\}}{2a_{\text{brake}}} \end{aligned} \tag{A.4}$$

- A lane width of $w = 3.5\,\text{m}$ and a vehicle length of $l = 5\,\text{m}$ are always assumed.

- When additional agents V_i are unavailable but required as inputs, vehicles in this role are created. These are placed far away from the ego and target vehicles, so their influence on the prediction model should be minimal. For example, a missing V_1 would be assumed to drive in front of the ego vehicle V_E, but at a very large distance of 500 m.

A.2.1 Lane changes



Figure A.1: The lane change scenario. The black crosses symbolize the position vector $\boldsymbol{x}_i$ belonging to agent V_i.

The first dataset—called highD—is a natural driving set recorded on German highways, captured by using drones [118], which entails lane changes of V_T towards a faster lane to the left. Here, a gap is accepted when V_T merges in front of V_E. The trajectories of additional surrounding vehicles are also recorded (or their existence is assumed outside the camera's scope). These are V_1 in front of V_E as well as V_2 behind and V_3 in front of V_T (Figure A.1).

We did not use this dataset in chapter 6, so we reporting the transformation to one dimensional data is less stringent:

$$\begin{aligned} D_C(t) &= x_T(t) - x_E(t) - l \\ D_A(t) &= -y_T(t) \\ D_1(t) &= x_1(t) - x_E(t) - l \\ L_E(t) &= l \end{aligned} \tag{A.5}$$

For this dataset, the optimization of Equation (A.1) results in $\Delta t = 11.8\,\text{s}$.

When assembling the restricted version of this dataset, only samples satisfying on of the following two requirements are included:

- V_T changes the lane after V_E has overtaken it, indicating that V_T has already been looking out for feasible gaps before but has rejected the one offered by V_E. 1025 rejected gaps fulfill this condition.

A

- V_T is substantially faster than V_3, i.e., $t_{\underline{C}}(t_S) - t_S \geq 2\,(t_{\underline{3}}(t_S) - t_S)$ with

$$t_{\underline{3}}(t) = \frac{x_3(t) - x_T(t) - 5\,\mathrm{m}}{\max\{\dot{x}_T(t) - \dot{x}_3(t), 0\}} + t\,. \tag{A.6}$$

  This condition is satisfied for 133 rejected gaps where it is suggested that V_T has consciously chosen to brake instead of accepting the gap.

As those conditions are not exclusive, only 1075 rejected gaps are included in the final dataset, with the optimization of Equation (A.1) resulting in $\Delta t = 3.36\,\mathrm{s}$.

A.2.2 Left turns

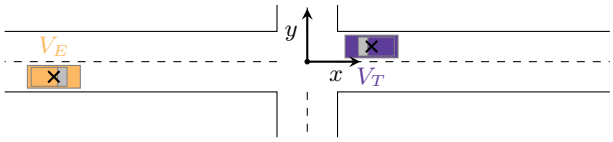


Figure A.2: The left turn. The black crosses symbolize the position vector $\boldsymbol{x}_i$ belonging to agent V_i.

The *L-GAP* dataset covers left turns at intersections, where the target vehicle V_T turns left across the planned trajectory of the ego vehicle V_E, which wants to drive straight ahead across the intersection (Figure A.2). Here, V_E appears when V_T has slowed down sufficiently during its approach to the intersection (t_S is set at this point). As there is no position data for V_E before t_S, special adjustments are needed ($t_0 = t_S + (n_I - 1)\delta t$) to permit prediction at the opening of the gap. However, we only use $n_I = 2$ in this work, so no onset of movement (possibly providing an easy cue) should be observable in V_T due to limited human reaction times.

The following transformation is used to convert the two dimensional position data into the one dimensional form:

$$\begin{aligned}
D_C(t) &= -\left(x_E(t) + w + \frac{1}{2}l\right) \\
D_A(t) &= \mathrm{sgn}\big(y_T(t)\big)\sqrt{y_T(t)^2 + \Delta x_T(t)^2} \\
\Delta x_T(t) &= \min\left\{\frac{y_T(t)\dot{x}_T(t)}{\dot{y}_T(t)} <, x_T(t) + \frac{1}{2}w\right\} \\
D_1(t) &= D_2(t) = D_3(t) = 500\,\mathrm{m} \\
L_E(t) &= 2w \\
L_T(t) &= w
\end{aligned} \tag{A.7}$$

For this dataset, the optimization of Equation (A.1) results in $\Delta t = 3.36\,\mathrm{s}$.

A.2.3 Roundabouts

The *rounD* dataset is a naturalistic dataset covering the scenario of roundabouts. There, V_T wants to enter the roundabout, either in front of or behind the ego vehicle V_E already inside

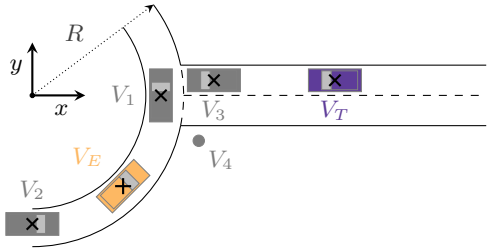


Figure A.3: Entering the roundabout. The black crosses symbolize the position vector $\boldsymbol{x}_i$ belonging to agent V_i.

the roundabout with an outer radius R. Further agents are V_1 and V_2 driving respectively in front of or behind V_E inside the roundabout. V_3 might enter the roundabout directly in front of V_T, while V_4 is the pedestrian most likely to interact with V_4 (Figure A.3).

The following transformation is used to convert the two dimensional position data into the one dimensional form:

$$\begin{aligned}
D_C(t) &= d_{I,E}(t) - \frac{1}{2}l \\
D_A(t) &= d_{O,T}(t) \\
D_1(t) &= d_{I,E}(t) - d_{I,1}(t) - l \\
D_2(t) &= d_{I,2}(t) - d_{I,E}(t) - l \\
D_3(t) &= d_{O,T}(t) - d_{O,3}(t) - l \\
L_E(t) &= w \\
L_T(t) &= \max\left\{w, \frac{1}{2}a_{\text{esc}}t_{\text{esc}}(t)^2 - \dot{D}_A(t)t_{\text{esc}}(t)\right\}
\end{aligned} \tag{A.8}$$

Here, a transformation from a Cartesian to a cylindrical coordinate system is used (when driving in a counter clockwise circle, the angle $\alpha_i \in [-\infty, \infty]$ is not reset but keeps increasing):

$$r_i(t)\exp(i\alpha_i(t)) = x_i(t) + iy_i(t)$$

Meanwhile, one also uses the 1D distance of a vehicle inside (d_I) and outside (d_O) the roundabout

$$\begin{aligned}
d_{I,i}(t) &= \ln\left(\frac{1}{1+\exp(r_i(t)-R_L)}\right)\tanh(5\alpha_i(t)) - R_L\alpha_i(t) \\
d_{O,i}(t) &= R_L\left(\frac{w}{R} - \alpha_i\right)\max\left\{0, 1 - \frac{1}{32}\exp(5h_i(t))\right\}^2 + h_i(t) - \ln(2) \\
h_i(t) &= \ln\left(1+\exp(r_i(t)-R)\right)\tanh\left(5\left(\frac{\pi}{4} - \alpha_i(t)\right)\right) \\
R_L &= R - \frac{1}{2}w,
\end{aligned} \tag{A.9}$$

as well as the escape acceleration $a_{\text{esc}} = 2\,\text{ms}^{-2}$ and escape time t_{esc}

$$t_{\text{esc}}(t) = \frac{\max\left\{\dot{D}_A(t) - \dot{D}_C(t), 0\right\}}{a_{\text{esc}}}. \tag{A.10}$$

A

For this dataset, the optimization of Equation (A.1) results in $\Delta t = 2.53\,\mathrm{s}$.

A.2.4 Intersection

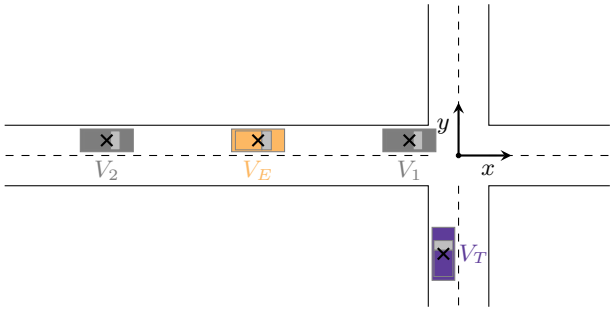


Figure A.4: Entering the intersection. The black crosses symbolize the position vector $\boldsymbol{x}_i$ belonging to agent V_i.

This dataset created in the Leeds driving simulator [363] cover the behavior of vehicles at an intersection, where both V_E and V_T want to go straight through the intersection. Like in the dataset above, further agents are included as well. This are V_1 and V_2, respectively driving in front of or behind V_E inside the roundabout (Figure A.4). This dataset was created under assumption of vehicles driving on the left side of the road, in opposition to the previous two datasets.

The following transformation is used to convert the two dimensional position data into the one dimensional form:

$$\begin{aligned} D_C(t) &= -\left(x_E(t) + w + \frac{1}{2}l + 1\,\mathrm{m}\right) \\ D_A(t) &= -\left(x_E(t) + w + \frac{1}{2}l + 1\,\mathrm{m}\right) \\ D_1(t) &= x_1(t) - x_E(t) - l \\ D_2(t) &= x_E(t) - x_2(t) - l \\ D_3(t) &= 1000\,\mathrm{m} \\ L_E(t) &= w + 1\,\mathrm{m} \\ L_T(t) &= w \end{aligned} \tag{A.11}$$

For this dataset, the optimization of Equation (A.1) results in $\Delta t = 6.22\,\mathrm{s}$.

A.3 Models

A.3.1 Trajectron++ (*T+*)

The *Trajectron++* model uses many different neural network, such as long-short-term memories (LSTM) and convolutional neural networks (CNN). It requires velocity and acceleration data as inputs, necessitating some preprocessing of the given position data $\boldsymbol{X}_{T_I}$ and $\boldsymbol{T}_I$. For the hyperparameters, the values used by the authors when applying the model to

the *nuScenes* dataset [142] are selected [129], as this dataset also mainly includes motor vehicles. Furthermore, an attention radius between vehicles of 150 m was chosen. No other changes have been made to the model.

A.3.2 AgentFormer (*AF*)

The *AgentFormer* model is a trajectory prediction model based on the concept of transformers (a technique proposed by Vaswani et al. [32]). Compared to other deep learning trajectory prediction models, which encode behavior over time and interactions between agents separately, this model processes all information simultaneously. For the hyperparameters, like for the previous model, the ones used by the authors when applying the model to the *nuScenses* dataset are selected [74].

A.3.3 Logistic Regression (*LR*)

While normally specifically selected properties are used as inputs of a logistic regression model, the positional data in $\boldsymbol{X}_I$ is given here. This results in an input with a dimensionality of $2n_I|\boldsymbol{V}|$. Although this is not done normally in the literature, where velocity data is common as well, this is no contradiction (as long as $n_I \geq 2$) since, during training, the model can extract the velocity data from the difference in positions at different time-steps:

$$\begin{aligned} &a_x x_1 + b_x x_2 = a x_1 + b v_1 \\ &v_1 = \frac{x_2 - x_1}{\delta t}\,,\; a = a_x + b_x\,,\; b = b_x \delta t\,. \end{aligned} \tag{A.12}$$

The same can be said for both relative distances and velocities and even accelerations (for the latter as long as $n_I > 2$). Due to the convexity of the optimized cost function during the model training, it will ultimately arrive at the same result, despite the linear transformation of the inputs.

A.3.4 Random forest (*RF*)

As defined by Nagalla et al. [113], random forests can have two hyperparameters, the number of bagged trees and the number of features considered for splits. Those are chosen in this implementation by a grid search with ten-fold cross-validation.

A.3.5 Deep Belief Network (*DB*)

For the deep belief network, the approach for determining the network structure (i.e., number of layers and nodes) is taken from Xie et al. [51]. Compared to that work, which relied on relative distances and velocities as inputs, trajectories are here used unaltered. Still, as the first step in the model is a multiplication of a weight matrix to the input, the lack of velocities should not be a problem either, as a linear transformation can be counteracted (as shown above). However, as the cost function $\mathscr{L}$ of a deep neural network is not convex, this cannot be guaranteed.

A.3.6 Meta-heuristic model (*MH*)

The meta-heuristic model is created by combining the predictions of multiple other models. Here, such a model is created by combing the predictions a_p of the other models on

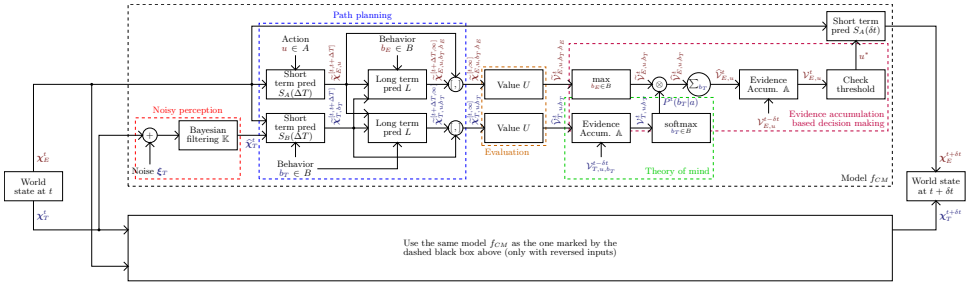


Figure A.5: The detailed version of the *Commotions* model. The theoretic parts are highlighted. For a full prediction, this process has to be repeated multiple times. The parameters $\boldsymbol{\theta}$ of the function f_{CM} are described in Table A.1.

a sample with the sample's original input $\boldsymbol{X}_{T_l}$. A logistic regression model—which according to Khelfa et al. [110] is the most promising approach—is then trained on these expanded inputs.

A.3.7 *Commotions* model

The *Commotions* model is a model consisting out of multiple parts (Figure A.5), trying to fully model the human decision making process, applying cognitive theory. While the logic behind this can be found in more detail in the original work [48], to understand the specific modifications made, a short introduction is nonetheless needed.

Each of the two agents in the *Commotions* model are assigned a state variable $\boldsymbol{\chi}$

$$\boldsymbol{\chi}_i^t = \left\{\chi_i(t), \dot{\chi}_i(t), \ddot{\chi}_i(t)\right\}, \tag{A.13}$$

where

$$\begin{aligned} \chi_E(t) &= D_C(t) + \frac{1}{2}\left(L_E(t) + l\right) \\ \chi_T(t) &= D_A(t) + \frac{1}{2}\left(L_T(t) + l\right). \end{aligned} \tag{A.14}$$

To get a simulated prediction, one has to apply the model transfer function f_{CM}, which is parameterized with $\boldsymbol{\theta}$:

$$\begin{aligned} \boldsymbol{\chi}_E^{t+\delta t} &= f_{CM}\left(\boldsymbol{\chi}_E^t, \boldsymbol{\chi}_T^t | \boldsymbol{\theta}\right) \\ \boldsymbol{\chi}_T^{t+\delta t} &= f_{CM}\left(\boldsymbol{\chi}_T^t, \boldsymbol{\chi}_E^t | \boldsymbol{\theta}\right). \end{aligned} \tag{A.15}$$

By using this model (Figure A.5 shows exemplary how to calculate $\boldsymbol{\chi}_E^{t+\delta t}$), one can simulate the future behavior over multiple time-steps. A binary prediction a as well as a prediction of t_A for a single sample can then be extracted (using linear interpolation between simulated time steps if necessary):

$$a_{\text{pred}} = \begin{cases} 1 & \chi_T(t_C) < \frac{1}{2}\left(L_T(t_C) + l\right) \\ 0 & \text{else} \end{cases} \tag{A.16}$$

$$t_{A,\text{pred}} = \min\left\{t \,|\, \chi_T(t) < \frac{1}{2}\left(L_T(t) + l\right)\right\} \tag{A.17}$$

The parameters $\boldsymbol{\theta}$ and their purpose and range can be found in Table A.1. If their range is given as a power of 10, $\ln \theta_i$ will be optimized instead of θ_i, which results in small values of the parameter being searched with higher likelihood during optimization.

The following two loss functions were then used to find the optimal values of those parameters $\boldsymbol{\theta}$:

$$\begin{aligned} \mathscr{L}_1 &= \sum_i \frac{1}{n_p} \sum_{p=1}^{n_p} 4 \left| a_i - a_{\text{pred},i,p} \right| + \begin{cases} \left(t_A - t_{A,\text{pred},i,p} \right)^2 & a_i = 1 \\ \left(t_C - \min\left\{ t_C, t_{A,\text{pred},i,p} \right\} \right)^2 & a_i = 0 \end{cases} \\ \mathscr{L}_2 &= \mathscr{L}_1 + \sum_i 100 \mathbb{V}\left(t_{A,\text{pred},i} \right) - 20 \sqrt{\mathbb{V}\left(t_{A,\text{pred},i} \right)} + 1 \,. \end{aligned} \tag{A.18}$$

Here, the second loss function additionally forces the model to produce more varied predictions.

As f_{CM} and thus those loss functions are non-differentiable in regard to $\boldsymbol{\theta}$, zeroth-order optimization methods have to be used to find the best model parametrization. We used Bayesian optimization [205] to this end, as this algorithm is capable to also deal with random loss functions, which is here the case due to the influence of $\boldsymbol{\xi}_T$ in f_{CM}. For this Bayesian optimization, we generated 210 initial samples using based on sobol sampling [364]. For the one step optimization (*1O*), then 150 additional parameter settings where samples (using expected improvement as an acquisition function). Meanwhile, for the two stage method (*2O*), only 100 such samples where created in each stage. In this method, the search space was restricted to one fifth of each original along each dimension, centered around the optimized parameter setting from the first stage.

A

Table A.1: The parameters of the *Commotions* model (inside f_{CM}). Equation numbers refer to the original work [48].

θ_i	Range	Description
β_V	$10^{[0,2.3]}$	Multiplier in the softmax function that calculates $P(b_T\|a)$ (Eq. (22)).
$\Delta V_{\text{th, rel}}$	$10^{[-3,-1]}$	Relative factor for calculating threshold ΔV_{th} for choosing a^* (Eq. (3)).
T	$[0.1, 0.5]$	Constant for calculating the evidence accumulation damping factor in $\mathbb{A}$ (Eq. (4)).
T_δ	$10^{[1,2]}$	Scaling factor for the time discount inside L (Eq. (16)).
σ_v	$10^{[-2.3,0.196]}$	Angular noise at the retina of a human, determining the added noise ξ_T.
$\sigma_{\dot{\chi}}$	$10^{[-2,0]}$	Standard deviation added to the velocity in the Bayesian filtering $\mathbb{K}$.
ΔT	$[1.01\delta t, 5\delta t]$	Time period that a given action a^* is applied over in S_A (Eq. (2)).
T_s	$[0, 2]$	Safety time margin (both for short and long term planning S_B, L) for a behavior b to be considered safe.
D_s	$[0, 5]$	Safety distance margin (used in short and long term planning S_B, L) for a behavior b to be considered safe.
$u_{0,\text{rel}}$	$10^{[0,2]}$	Relative factor for calculating the scale factor u_0 applied to the unrestricted value in U (Eq. (8)).
k_{da}	$10^{[-1,1]}$	Factor in U that is used to punish high acceleration in a motion plan (Eq. (10)).
$\sigma_{R_0\chi,\text{rel}}$	$10^{[-1,1]}$	Relative factor used to scale the initial observation noise $\sigma_{R_0\chi} = \|\chi(t_0)\|\sigma_{R_0\chi,\text{rel}}$ of the position in the Bayesian filtering $\mathbb{K}$.
$\sigma_{R_0\dot{\chi},\text{rel}}$	$10^{[-1,1]}$	Relative factor used to scale the initial observation noise $\sigma_{R_0\dot{\chi}} = \|\dot{\chi}(t_0)\|\sigma_{R_0\dot{\chi},\text{rel}}$ of the velocity in the Bayesian filtering $\mathbb{K}$.
$v_{\text{free,rel}}$	$10^{[-0.3,0.3]}$	Relative factor used to scale the desired free speed.
ΔT_{free}	10	Assumed acceleration time (in S_B and L) to reach free speed, if there is no chance of a dangerous interaction.
h_E	1.5	The eye height of an observer for determining ξ_T.

A

A.4 Metrics

A.4.1 Accuracy

$$F = \max_{\tau \in [0,1]} \frac{1}{N_A + N_{\neg A}} \left| \left\{ i \,|\, a_i = 0 \,\bar{\vee}\, a_{\text{pred},i} > \tau \right\} \right| \tag{A.19}$$

Here, with $\bar{\vee}$ symbolizing XOR, τ is the decision threshold chosen to maximize the final value of F. For a random binary predictor, this threshold would be either 0 or 1 (depending on the bias in the dataset), resulting in

$$F_r = \frac{\max\{N_A, N_{\neg A}\}}{N_A + N_{\neg A}}. \tag{A.20}$$

A.4.2 *AUC*

$$F = \frac{1}{N_A N_{\neg A}} \left(\left(\sum_i a_i r_i \right) - \frac{N_A (N_A + 1)}{2} \right) \tag{A.21}$$

Here, r_i is the position of sample i according to its value $a_{\text{pred},i}$ ($r_i = 1$ for the lowest value of a_{pred} and $r_i = N_A + N_{\neg A}$ for the highest value).

A.4.3 Average displacement error (*ADE*)

$$\begin{aligned} F &= \frac{1}{(N_A + N_{\neg A}) \left\lceil n_p \beta \right\rceil} \sum_i \sum_{p=1}^{\left\lceil n_p \beta \right\rceil} D_{i,p} \\ D_{i,p} &= \frac{1}{n_{O,i}} \sum_{t \in T_{O,i}} \left\| \boldsymbol{x}_{T,i,p}(t) - \boldsymbol{x}_{T,i}(t) \right\| \end{aligned} \tag{A.22}$$

with

$$D_{i,p} \leq D_{i,p+1} \; \forall \, p \in \left\{ 1, \ldots, n_p - 1 \right\}.$$

If */beta* is not explicitly given, $\beta = 1$ is assumed.

A.4.4 True negative rate for perfect recall (*TNR-PR*)

$$F = \frac{\left| \left\{ i \,|\, a_i = 0 \wedge a_{\text{pred},i} < \min\left\{ a_{\text{pred},i} \,|\, a_i = 1 \right\} \right\} \right|}{N_{\neg A}} \tag{A.23}$$

B

An Active Inference Model of Human Behavior

To model the behavior of human drivers in different scenarios, we combine Active Inference and Evidence Accumulation. For sake of either notation, we use the shorthand of $\boldsymbol{b}_{1:B} = \{b_i | i \in \{1, \dots, B\}\}$. Additionally, while the predicted beliefs about states $\tilde{q}_s(\boldsymbol{s}_\tau)$ and observations $\tilde{q}_o(\boldsymbol{o}_\tau)$ for $\tau > t$ depend on the current belief $q(\boldsymbol{s}_t)$ and the chosen policy $\boldsymbol{\pi}_t$ (sequence of planned actions), so that the correct notation would be $\tilde{q}_s(\boldsymbol{s}_\tau | \boldsymbol{\pi}_t, q(\boldsymbol{s}_t))$ and $\tilde{q}_o(\boldsymbol{o}_\tau | \boldsymbol{\pi}_t, q(\boldsymbol{s}_t))$ respectively, we will use the shorthand expressions $\tilde{q}_s(\boldsymbol{s}_\tau)$ and $\tilde{q}_o(\boldsymbol{o}_\tau)$ to improve readability.

B.1 The general active inference model

We start our model description with a general overview of the active inference framework, before describing the specific implementation for our driving scenarios.

B.1.1 Generative process and generative model

The basic concept of active inference is the idea that a human agent does not know the actual mechanism underlying their surrounding world (called *generative process*), but instead relies on an internal model (the so called *generative model*) approximating the true world. These describe how the true state of the world ($\boldsymbol{\eta}_t$), a state of the agents' belief about the world ($\boldsymbol{s}_t$)), the agent's actions ($\boldsymbol{a}_t$), and the observations of the true world $\boldsymbol{o}_t$ at a time t interact with each other. Here, the *generative process* consists out of two parts:

- The true transition probability $\hat{p}(\boldsymbol{\eta}_{t+1} | \boldsymbol{\eta}_t, \boldsymbol{a}_t)$. It describes how a certain action at time t influences the future world state.
- The true observation probability $\hat{p}(\boldsymbol{o}_t | \boldsymbol{\eta}_t)$. It describes how likely it is that a certain observation can be perceived given the current state of the world.

The *generative model* consists out of two parts as well.

- The internal state transition probability $p(\boldsymbol{s}_{t+1} | \boldsymbol{s}_t, \boldsymbol{a}_t)$.

- The internal observation probability $p(\boldsymbol{o}_t|\boldsymbol{s}_t)$.

The second main idea of active inference is that an agent is uncertain about its belief about the world $\boldsymbol{s}_t$, meaning that instead of a single values, we instead assume that the agent holds a probabilistic belief, denoted by $q(\boldsymbol{s}_t)$. In general, the agent could also have some uncertainty regarding the generative model itself (*i.e.*, there is a probabilistic belief $q(\boldsymbol{\theta})$ about the *general model*'s parameters $\boldsymbol{\theta}$), but we will not include this assumption in favor of a computationally more efficient model. We represent stochastic beliefs $q(\boldsymbol{s}_t)$ by $N = 75$ equally likely representative samples $\boldsymbol{S}_t = \boldsymbol{s}_{t,1:N}$.

B.1.2 Belief update

Given a world with the previous state $\boldsymbol{\eta}_{t-1}$, an belief $\boldsymbol{S}_{t-1}$, and a chosen action $\boldsymbol{a}_{t-1}$, we can update those in the following way:

1. We randomly sample our future world state $\boldsymbol{\eta}_t$ using the transition probability of the *generative process*:

$$\boldsymbol{\eta}_t \sim \hat{p}(\boldsymbol{\eta}'|\boldsymbol{\eta}_{t-1}, \boldsymbol{a}_{t-1}) \tag{B.1}$$

2. We generate the observations $\boldsymbol{o}_t$ that the agent makes. Under the assumption that the uncertainties in the *generative process* are negligible compared to the uncertainties in the *generative model*, we use the expected value:

$$\boldsymbol{o}_t = \mathbb{E}_{\hat{p}(\boldsymbol{o}'|\boldsymbol{\eta}_t)}\boldsymbol{o}' \tag{B.2}$$

3. We lastly have to update the internal belief of the agent. Here, using Bayesian inference, we get

$$\begin{aligned} q(\boldsymbol{s}_t) &\propto p(\boldsymbol{o}_t|\boldsymbol{s}_t)\mathbb{E}_{q(\boldsymbol{s}_{t-1})}p(\boldsymbol{s}_t|\boldsymbol{s}_{t-1}, \boldsymbol{a}_{t-1}) \\ &\propto p(\boldsymbol{o}_t|\boldsymbol{s}_t)\, q_A(\boldsymbol{s}_t), \end{aligned} \tag{B.3}$$

   where $q(\boldsymbol{s}_t) = q(\boldsymbol{s}_t|\boldsymbol{o}_t)$ is the posterior, $p(\boldsymbol{o}_t|\boldsymbol{s}_t)$ the likelihood, and $q_A(\boldsymbol{s}_t)$ the prior.

   To apply this to our sample based belief representation, we represent the prior $q_A(\boldsymbol{s}_t)$ with $\boldsymbol{S}_{A,t} = \boldsymbol{s}_{A,t,1:N}$,using the *generative model's* state transition probability $p(\boldsymbol{s}_t|\boldsymbol{s}_{t-1}, \boldsymbol{a}_{t-1})$ to generate $\boldsymbol{s}_{A,t,n} \sim p(\boldsymbol{s}'|\boldsymbol{s}_{t-1,n}, \boldsymbol{a}_{t-1})$.

   While we could use some form of combination of density estimation for $q_A(\boldsymbol{s}_t)$ based on $\boldsymbol{S}_{A,t}$ together with the metropolis hastings algortihm, we could sample the updated belief samples $\boldsymbol{S}_t$.

   However, this process can be done much faster if we can find a density estimation for $q_A(\boldsymbol{s}_t)$ that is a conjugate prior to the likelihood $p(\boldsymbol{o}_t|\boldsymbol{s}_t)$, which would allowing us to get an analytical expression for the posterior $q(\boldsymbol{s}_t)$ from which we can easily sample directly. Generally, for this to be the case, the probability density function would have to be of the form

$$p(\boldsymbol{o}_t|\boldsymbol{s}_t) = p_{\mathcal{N}}(\boldsymbol{o}_{\mathcal{N},t}|\boldsymbol{s}_{\mathcal{N},t})p_{\mathcal{B}}(\boldsymbol{o}_{\mathcal{B},t}|\boldsymbol{s}_{\mathcal{B},t}), \tag{B.4}$$

where it can be split up (with states $\boldsymbol{s} = \{s_{\mathcal{N}}, s_{\mathcal{B}}\}$ and observations $\boldsymbol{o} = \{o_{\mathcal{N}}, o_{\mathcal{B}}\}$) in a gaussian part

$$\begin{aligned} p(\boldsymbol{o}_{\mathcal{N},t}|\boldsymbol{s}_{\mathcal{N},t}) &= \mathcal{N}\left(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t})|\mu(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t})) + \mathcal{L}(\boldsymbol{A}\boldsymbol{s}_{\mathcal{N},t} + \boldsymbol{b}), \Sigma(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}))\right)\left|\det J_{\mathcal{L}}(\boldsymbol{o}_{\mathcal{N},t})\right| \\ &= \mathcal{N}\left(\mathcal{L}(\boldsymbol{A}\boldsymbol{s}_{\mathcal{N},t} + \boldsymbol{b})|\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}) - \mu(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t})), \Sigma(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}))\right)\left|\det J_{\mathcal{L}}(\boldsymbol{o}_{\mathcal{N},t})\right| \end{aligned} \tag{B.5}$$

and Bernoulli part

$$p_{\mathcal{B}}(\boldsymbol{o}_{\mathcal{B},t}|\boldsymbol{s}_{\mathcal{B},t}) = \prod_{d=1}^{D_{\mathcal{B}}} \mathcal{B}\left(o_{\mathcal{B},t,d}|s_{\mathcal{B},t,d}^{N_{\mathcal{B}}}\right). \tag{B.6}$$

Here, the mapping $\mathcal{L}$ must be bijective. We then can generate the posterior samples $\boldsymbol{S}_t = \{\boldsymbol{S}_{\mathcal{N},t}, \boldsymbol{S}_{\mathcal{B},t}\}$ analytically in the two separate parts:

- For the gaussian part, we do analytical Bayesian inference over $\boldsymbol{s}_{\mathcal{L},t} = \mathcal{L}(\boldsymbol{A}\boldsymbol{s}_{\mathcal{N},t} + \boldsymbol{b})$. Namely, we can use a Gaussian KDE to calculate the conjugate prior $q_{\mathcal{L},A}(\boldsymbol{s}_{\mathcal{L},t})$, where with $\boldsymbol{s}_{\mathcal{L},A,t,n} = \mathcal{L}(\boldsymbol{A}\boldsymbol{s}_{\mathcal{N},A,t,n} + \boldsymbol{b})$ we get

$$q_{\mathcal{L},A}(\boldsymbol{s}_{\mathcal{L},t}) \approx \frac{1}{N}\sum_{n=1}^{N} \mathcal{N}\left(\boldsymbol{s}_{\mathcal{L},t}|\boldsymbol{s}_{\mathcal{L},A,t,n}, \Sigma_{\mathcal{L},A,t}\right). \tag{B.7}$$

  Substituting (B.7) and (B.5) then allows us to express $q_{\mathcal{L}}(\boldsymbol{s}_{\mathcal{L},t})$ as a Gaussian multi mixture model, from which sampling $\boldsymbol{S}_{\mathcal{L},t}$ is trivial:

$$\begin{aligned} q_{\mathcal{L}}(\boldsymbol{s}_{\mathcal{L},t}) &\propto \mathcal{N}\left(\boldsymbol{s}_{\mathcal{L},t}|\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}) - \mu(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t})), \Sigma(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}))\right) q_{\mathcal{L},A}(\boldsymbol{s}_{\mathcal{L},t}) \\ &\propto \sum_{n=1}^{N} \mathcal{N}\left(\boldsymbol{s}_{\mathcal{L},t}|\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}) - \mu(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t})), \Sigma(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}))\right) \mathcal{N}\left(\boldsymbol{s}_{\mathcal{L},t}|\boldsymbol{s}_{\mathcal{L},A,t,n}, \Sigma_{\mathcal{L},A,t}\right) \\ &\propto \sum_{n=1}^{N} w_{t,n}\, \mathcal{N}\left(\boldsymbol{s}_{\mathcal{L},t}|\boldsymbol{\mu}_{t,n}, \boldsymbol{\Sigma}_t\right) \end{aligned} \tag{B.8}$$

  with $\boldsymbol{\mu}_o = \mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}) - \mu(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}))$ and

$$\begin{aligned} \boldsymbol{\Sigma}_t &= \left(\Sigma_{\mathcal{L},A,t}^{-1} + \Sigma(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}))^{-1}\right)^{-1} \\ \boldsymbol{\mu}_{t,n} &= \boldsymbol{\Sigma}_t\left(\Sigma_{\mathcal{L},A,t}^{-1}\boldsymbol{s}_{\mathcal{L},A,t,n} + \Sigma(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}))^{-1}\boldsymbol{\mu}_o\right) \\ w_{t,n} &\propto \exp\left(-\frac{1}{2}\left(\boldsymbol{s}_{\mathcal{L},A,t,n} - \boldsymbol{\mu}_o\right)^T\left(\Sigma_{\mathcal{L},A,t} + \Sigma(\mathcal{L}(\boldsymbol{o}_{\mathcal{N},t}))\right)^{-1}\left(\boldsymbol{s}_{\mathcal{L},A,t,n} - \boldsymbol{\mu}_o\right)\right). \end{aligned} \tag{B.9}$$

  After sampling $\boldsymbol{s}_{\mathcal{L},t,1:N}$ from the GMM, we can get our final sample $\boldsymbol{S}_{\mathcal{N},t} = \boldsymbol{s}_{\mathcal{N},t,1:N}$ with $\boldsymbol{s}_{\mathcal{N},t,n} = \boldsymbol{A}^{-1}\left(\mathcal{L}^{-1}(\boldsymbol{s}_{\mathcal{L},t,n}) - \boldsymbol{b}\right)$.
  If $\boldsymbol{A}$ is not full rank, we will have to simply assume that the distribution $q(\boldsymbol{s}_t)$ orthogonally to the image of $\boldsymbol{A}$ will be identical to the one in $q_A(\boldsymbol{s}_t)$. However, it must be noted that in our simulations $\boldsymbol{A}$ will be the identity matrix.

- Meanwhile, for the Bernoulli part, we can generate a conjugate prior $q_A(\boldsymbol{s}_{\mathscr{B},t})$ using a product of Beta distributions:

$$q_A(\boldsymbol{s}_{\mathscr{B},t}) = \prod_{d=1}^{D_{\mathscr{B}}} \frac{1}{N} \sum_{n=1}^{N} \text{Beta}\left(s_{\mathscr{B},t,d} | \alpha_{A,t,d,n}, \beta_{A,t,d,n}\right), \tag{B.10}$$

where we can calculate the parameters $\boldsymbol{\alpha}_{A,t}$ and $\boldsymbol{\beta}_{A,t}$ based on the samples $\boldsymbol{S}_{\mathscr{B},A,t} = \boldsymbol{s}_{\mathscr{B},A,t,1:N}$. Specifically, we we can estimate the parameters of the Beta distributions using a moment-matching approach based on the sample mean and variance, adjusted for numerical stability. First, we compute the sample standard deviation $\sigma_{\mathscr{B},t,d}$ along each particle dimension using $s_{\mathscr{B},A,t,1:N}$. We then scale the standard deviation by a bandwidth factor $h = 2\left(N \cdot 0.75\right)^{-0.2}$ (*e.g.*, based on Silverman's rule of thumb), and then bound it:

$$\sigma'_{\mathscr{B},t,d,n} = \max\left(\left\{\min\left(\left\{\sigma_{\mathscr{B},t,d} \cdot h,\, s_{\mathscr{B},A,t,d,n}(1 - s_{\mathscr{B},A,t,d,n}) - 10^{-6}\right\}\right), \epsilon\right\}\right), \tag{B.11}$$

where $\epsilon = 0.005$ is a lower bound to prevent degenerate variances, and the upper bound ensures that the standard deviation does not exceed the maximal allowable variance of a Beta distribution.

We then can get the parameters by inverting the formulas for the mean and variance of a Beta distribution:

$$\begin{aligned} \alpha_{A,t,d,n} &= \frac{(1 - s_{\mathscr{B},A,t,d,n}) s^2_{\mathscr{B},A,t,d,n}}{(\sigma'_{\mathscr{B},t,d,n})^2} - s_{\mathscr{B},A,t,d,n} \\ \beta_{A,t,d,n} &= \alpha_{A,t,d,n} \left(\frac{1}{s_{\mathscr{B},A,t,d,n}} - 1\right). \end{aligned} \tag{B.12}$$

Finally, we can get our posterior for a dimension d $q(\boldsymbol{s}_{\mathscr{B},t,d})$ as

$$\begin{aligned} q(\boldsymbol{s}_{\mathscr{B},t,d}) &\propto \mathscr{B}\left(o_{\mathscr{B},t,d} | s^{N_{\mathscr{B}}}_{\mathscr{B},t,d}\right) q_A(\boldsymbol{s}_{\mathscr{B},t,d}) \\ &\propto \frac{1}{N} \sum_{n=1}^{N} \mathscr{B}\left(o_{\mathscr{B},t,d} | s^{N_{\mathscr{B}}}_{\mathscr{B},t,d}\right) \text{Beta}\left(s_{\mathscr{B},t,d} | \alpha_{A,t,d,n}, \beta_{A,t,d,n}\right) \\ &\propto \sum_{n=1}^{N} w_{t,d,n} \, \text{Beta}\left(s_{\mathscr{B},t,d} | \alpha_{t,d,n}, \beta_{t,d,n}\right) \end{aligned} \tag{B.13}$$

with

$$\begin{aligned} \alpha_{t,d,n} &= \alpha_{A,t,d,n} + N_{\mathscr{B}} \, o_{\mathscr{B},t,d} \\ \beta_{t,d,n} &= \beta_{A,t,d,n} + M_{\mathscr{B}}(1 - o_{\mathscr{B},t,d}) \\ w_{t,d,n} &= \exp\left(\log \text{B}(\alpha_{t,d,n}, \beta_{t,d,n}) - \log \text{B}(\alpha_{A,t,d,n}, \beta_{A,t,d,n})\right), \end{aligned} \tag{B.14}$$

where B is the beta function. Meanwhile, $M_{\mathscr{B}}$ is chosen, as it it minimizes the error

$$\int_0^1 \left\| \left(1 - x^{N_{\mathscr{B}}}\right) - (1-x)^{M_{\mathscr{B}}} \right\| \mathrm{d}x \,. \tag{B.15}$$

Consequently, it is the closest way we can update the posterior with a simple addition.

Sampling from the Beta mixture model then gives us the final samples $\boldsymbol{s}_{\mathscr{B},t} = \boldsymbol{s}_{\mathscr{B},t,1:N}$.

In the update of our model, we use a time step size of $\Delta t = 0.2\,\mathrm{s}$

B.1.3 Expected free energy

After the agent updates its internal belief $q(\boldsymbol{s})$, it then has to generate a new policy $\boldsymbol{\pi}_t = \boldsymbol{a}_{t:(t+H-1)}$ (with $\boldsymbol{a}_t = [\pi_t]_1$) over a prediction horizon of H time steps (we use $H = 30$ in our implementation). In general active inference, it is postulated that such a plan is selected based on the minimization of the *expected free energy* G, here defined using the preference function $p(\boldsymbol{o})$:

$$\begin{aligned} G(\boldsymbol{\pi}_t | q(\boldsymbol{s}_t)) &= \sum_{\tau=t+1}^{t+H} g(\boldsymbol{\pi}_t, q(\boldsymbol{s}_t), \tau) \\ &= \sum_{\tau=t+1}^{t+H} \underbrace{-\,\mathbb{E}_{\tilde{q}_o(\boldsymbol{o}_\tau)} \ln p(\boldsymbol{o}_\tau)}_{\text{Pragmatic value } g_{\text{pragm}}} \underbrace{-\left(\mathscr{H}(\tilde{q}_o(\boldsymbol{o}_\tau)) - \mathbb{E}_{\tilde{q}_s(\boldsymbol{s}_\tau)} \mathscr{H}(p(\boldsymbol{o}_\tau|\boldsymbol{s}_\tau))\right)}_{\text{Epistemic value } g_{\text{epist}}} \end{aligned} \tag{B.16}$$

Meanwhile, $\tilde{q}_s(\boldsymbol{s}_\tau)$ and $\tilde{q}_o(\boldsymbol{o}_\tau)$ correspond to the beliefs that the agents predict for internal states and observations when following a certain policy, with

$$\begin{aligned} \tilde{q}_s(\boldsymbol{s}_\tau) &= \mathbb{E}_{\tilde{q}_s(\boldsymbol{s}_{\tau-1})} p(\boldsymbol{s}_\tau | \boldsymbol{s}_{\tau-1}, \boldsymbol{a}_{\tau-1}) \\ \tilde{q}_o(\boldsymbol{o}_\tau) &= \mathbb{E}_{\tilde{q}_s(\boldsymbol{s}_\tau)} p(\boldsymbol{o}_\tau | \boldsymbol{s}_\tau) \end{aligned} \tag{B.17}$$

We use again our sample based approach for belief representation, with $\tilde{\boldsymbol{S}}_\tau = \boldsymbol{s}_{\tau,1:N}$ representing $\tilde{q}_s(\boldsymbol{s}_t)$ (with $\boldsymbol{s}_{\tau,n} \sim p(\boldsymbol{s}'|\boldsymbol{s}_{\tau-1,n}, \boldsymbol{a}_{\tau-1})$) and $\widetilde{\boldsymbol{O}}_\tau = \boldsymbol{o}_{\tau,1:N}$ approximating $\tilde{q}_o(\boldsymbol{o}_\tau)$ (where $\boldsymbol{o}_{\tau,n} \sim p(\boldsymbol{o}'|\boldsymbol{s}_{\tau,n})$). For the initial step of $\tau = t+1$, we can assume that $\tilde{\boldsymbol{S}}_t = \boldsymbol{S}_t$. Based on this, one can then calculate the expected free energy g at one timestep with

$$g(\boldsymbol{\pi}_t, q(\boldsymbol{s}_t), \tau) \approx -\frac{1}{|\widetilde{\boldsymbol{O}}_\tau|} \sum_{\boldsymbol{o}_\tau \in \widetilde{\boldsymbol{O}}_\tau} \ln p(\boldsymbol{o}_\tau) - \left(\mathscr{H}\left(\tilde{q}_o(\boldsymbol{o}_\tau)\right) - \frac{1}{N} \sum_{\boldsymbol{s}_\tau \in \tilde{\boldsymbol{S}}_\tau} \mathscr{H}\left(p(\boldsymbol{o}'|\boldsymbol{s}_\tau)\right) \right), \tag{B.18}$$

where we use the approximation

$$\begin{aligned} \mathscr{H}\left(\tilde{q}_o(\boldsymbol{o}_\tau)\right) &= \mathscr{H}\left(\mathbb{E}_{\tilde{q}_s(\boldsymbol{s}_\tau)} p(\boldsymbol{o}_\tau|\boldsymbol{s}_\tau)\right) \\ &= -\frac{1}{N} \sum_{\boldsymbol{o}_\tau \in \widetilde{\boldsymbol{O}}_\tau} \ln \left(\frac{1}{N} \sum_{\boldsymbol{s}_\tau \in \tilde{\boldsymbol{S}}_\tau} p(\boldsymbol{o}_\tau|\boldsymbol{s}_\tau) \right) \end{aligned} \tag{B.19}$$

To maximize G, we use the Cross Entropy Method (CEM) for model predictive control, which is an iterative method over $k \in 1, \dots, K$ with $K = 20$:

1. We define a distribution $p_{\pi,k}(\boldsymbol{\pi}_t) = \mathcal{N}(\boldsymbol{\pi}_t | \boldsymbol{\mu}_{\pi,t,k-1}, \mathrm{diag}(\boldsymbol{\sigma}^2_{\pi,t,k-1}))$ over the policy space, from which we sample the $M = 100$ policies $\boldsymbol{\pi}_{t,k,1:M}$ from $p_{\pi,k}(\boldsymbol{\pi}_t)$ and calculate the respective expected free energy (after adjusting for pedals with f_{real}) $G_{t,k,m} = G(f_{\text{real}}(\boldsymbol{\pi}_{t,k,m}), q(\boldsymbol{s}_t))$ (Equation (B.18)).

2. We select the $\beta = 0.1 \in [0, 1]$ percent samples $\boldsymbol{\pi}_{t,k,1:M}$ with the lowest expected free energy.

3. We update our distribution, where $\boldsymbol{\mu}_{\pi,t,k}$ and $\boldsymbol{\sigma}_{\pi,t,k}$ are the mean and standard deviation of the aforementioned βM selected best plans.

For the first iteration, we choose $\boldsymbol{\mu}_{\pi,t,0} = 0$, while we choose as standard deviations in $\boldsymbol{\sigma}_{\pi,t,0}$ value of $5\,\mathrm{ms}^{-2}$ for accelerations a and $0.1\,\mathrm{s}^{-1}$ for steering rates ω. The final policy is then selected as

$$\boldsymbol{\pi}_t = f_{\text{real}}\left(\underset{m \in \{1,\dots,M\}}{\operatorname{argmin}} \; G(f_{\text{real}}(\boldsymbol{\pi}_{t,K,m}), q(\boldsymbol{s}_t)) \right). \tag{B.20}$$

Here, f_{real} is used to prevent unrealistic control inputs, as further described below.

B.1.4 Evidence accumulation

Commonly, the policy $\boldsymbol{\pi}_t$ is re-chosen at every timestep. However, research has shown that humans tend to make decisions (such a changing preselected policies) only if there is enough evidence supporting such a decision, in a process called *evidence accumulation* [44, 45, 228, 229]. Here, we implement this concept by having the agent accumulate evidence E_t towards the need for selecting a new policy. The agent then updates its policy in the following way:

1. We update our previous policy $\boldsymbol{\pi}_{t-1}$, resulting in $\widetilde{\boldsymbol{\pi}}_t$, with $[\widetilde{\pi}_t]_{1:H-1} = [\pi_{t-1}]_{2:H}$, and only optimize the last needed time step $[\widetilde{\pi}_t]_H$ with the method described in Section B.1.3.

2. We calculate the evidence for choosing a new plan based on the normalized, negative pragmatic value (*i.e.*, the *surprise*) or residual information [243], with

$$\epsilon_t = \epsilon(\widetilde{\boldsymbol{\pi}}_t | q(\boldsymbol{s}_t)) = H \max_{\boldsymbol{o}'} \ln p(\boldsymbol{o}') - \sum_{\tau=t+1}^{t+H} \mathbb{E}_{\tilde{q}_o(\boldsymbol{o}_\tau)} \ln p(\boldsymbol{o}_\tau, \boldsymbol{a}_{\tau-1}). \tag{B.21}$$

   We then update our accumulated surprise with $E_t = E_{t-1} + \lambda \epsilon_t$, where we use a drift rate of $\lambda = 10^{-5.95}$.

3. If we see that $E_t \geq 1$, then we optimize the full policy $\boldsymbol{\pi}_t$ using the method described in Section B.1.3, and set $E_t = 0$. Otherwise, we use the continued policy $\widetilde{\boldsymbol{\pi}}_t$ as our current policy $\boldsymbol{\pi}_t$

B.2 Model implementation details

While the previous section described our general framework for using active inference, this section will detail the exact *generative process* and *generative model* we used in our scenarios.

B.2.1 State transition probability

When implementing the model, we use for the state transition function of both the *generative process* and the *generative model* a common bicycle model *B*. In this, model each vehicle can be defined by three parameters, its width d, its front length l_f and its rear length l_r. Additionally, the kinematic state $\boldsymbol{x}$ of each agent v then consists of its position markers x and y, its longitudinal speed v, its current heading angle θ and steering angle δ:

$$\boldsymbol{x}_v = \{x_v, y_v, v_v, \theta_v, \delta_v\}. \tag{B.22}$$

Each agent v is then controlled by the acceleration $a_{\text{long}} \in [-a_{\max}, a_{\max}]$ and steering rate $\omega \in [-\omega_{\max}, \omega_{\max}]$ with $a_{\max} = 8\,\text{ms}^{-2}$ and $\omega_{\max} = 1.22\,\text{s}^{-1}$:

$$\boldsymbol{u}_v = \{a_{\text{long},i}, \omega_v\}. \tag{B.23}$$

One then can get the differential equation $\dot{\boldsymbol{x}} = B(\boldsymbol{x}, \boldsymbol{u})$:

$$\begin{aligned}
\dot{x} &= v\cos\left(\theta + \gamma\right) \\
\dot{y} &= v\sin\left(\theta + \gamma\right) \\
\dot{v} &= k_{\text{tire}}\, a_{\text{long}} \\
\dot{\theta} &= \frac{v}{l_f + l_r}\tan\left(k_{\text{tire}}\,\delta\right)\cos\left(\gamma\right) \\
\dot{\delta} &= \begin{cases} 0 & \frac{\text{sgn}(\omega)\text{sgn}(\delta)}{k_{\text{tire}}} > 1 \\ \omega & \text{Otherwise} \end{cases} \\
\text{with } \gamma &= \arctan\left(\frac{l_r}{l_f + l_r}\tan\left(k_{\text{tire}}\,\delta\right)\right)
\end{aligned} \tag{B.24}$$

Here, γ is the angle between the vehicles orientation θ and its actual heading, while we use k_{tire} and $\hat{\delta}$ to represent the limitations imposed by the tire friction:

$$k_{\text{tire}} = \frac{a_{\max}}{\max\left\{a_{\max}, \sqrt{a[\text{long}]^2 + \left(\frac{v^2}{l_f + l_r}\delta\right)^2}\right\}} \tag{B.25}$$

In each scenario, where we model the ego agent in interaction with the other agents $V = \{V_1, V_2, \ldots\}$, we then can generally find the control actions $\boldsymbol{a} = \{\boldsymbol{\gamma}_{\text{ego}}, \boldsymbol{u}_{\text{ego}}\}$ and the states

$$\boldsymbol{\eta} = \boldsymbol{o} = \boldsymbol{s} = \{\boldsymbol{x}_v, \boldsymbol{\gamma}_v, \boldsymbol{u}_v | v \in \{\text{ego}\} \cup V\} \tag{B.26}$$

Here, $\boldsymbol{\gamma} = \{\gamma_A, \gamma_Y\}$ are the signalling states defined in Chapter 8 (*i.e.*, for Chapter 7, it can be assumed that $\boldsymbol{\gamma} = \emptyset$ for all agents).

We assume that the *generative process* is deterministic, which allows us to get the following, where $f_B(\boldsymbol{x}, \boldsymbol{u})$ describes the usage of Heun's methods to propagate the state forward according to Equation (B.24):

$$\begin{aligned}\hat{p}(\boldsymbol{\eta}'|\boldsymbol{\eta}, \boldsymbol{a}) = \delta\left(\boldsymbol{x}'_{\text{ego}} - f_B\left(\boldsymbol{x}_{\text{ego}}, \boldsymbol{a}\right)\right)\delta\left(\left\{\boldsymbol{\gamma}'_{\text{ego}}, \boldsymbol{u}'_{\text{ego}}\right\} - \boldsymbol{a}\right) \\ \prod_{v\in V}\delta\left(\boldsymbol{x}'_v - f_B\left(\boldsymbol{x}_v, \boldsymbol{u}_v\right)\right)\delta\left(\boldsymbol{\gamma}'_v - \boldsymbol{\gamma}_{v,\text{preset}}\right)\delta\left(\boldsymbol{u}'_v - \boldsymbol{u}_{v,\text{preset}}\right)\end{aligned} \tag{B.27}$$

Here, the next control inputs $\boldsymbol{u}_{v,\text{preset}}$ (and similar signals, if included) are predefined to allow the other vehicle to follow a prescribed trajectory, which depends on the scenario (Sections B.3.1 and B.3.2). Meanwhile, some uncertainty is involved in the *generative model*:

$$\begin{aligned}p_0(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a}) = \delta\left(\boldsymbol{x}'_{\text{ego}} - f_B\left(\boldsymbol{x}_{\text{ego}}, \boldsymbol{a}\right)\right)\delta\left(\left\{\boldsymbol{\gamma}'_{\text{ego}}, \boldsymbol{u}'_{\text{ego}}\right\} - \boldsymbol{a}\right) \\ \prod_{v\in V}\delta\left(\boldsymbol{x}'_v - f_B\left(\boldsymbol{x}_v, \boldsymbol{u}'_v\right)\right)p_\gamma\left(\boldsymbol{\gamma}'_v|\boldsymbol{\gamma}_v, \boldsymbol{a}\right)\mathcal{N}\left(\boldsymbol{u}'_v|\boldsymbol{u}_v, \text{diag}(\boldsymbol{\sigma}_{\boldsymbol{u}}^2)\right),\end{aligned} \tag{B.28}$$

where in Chapter 8, we update the belief about another agents signaling state with the function p_γ, which is defined as

$$p_\gamma\left(\boldsymbol{\gamma}'_v|\boldsymbol{\gamma}_v, \boldsymbol{a}\right) = \mathcal{N}_T(\gamma'_{A,v} \mid \alpha\gamma_{A,v}, \sigma_\gamma)\begin{cases}\mathcal{B}(\gamma'_{Y,v} \mid \gamma_{Y,v}) & \gamma_{A,\text{ego}} = 1 \\ \mathcal{N}_T(\gamma'_{Y,v} \mid \gamma_{Y,v}, \sigma_\gamma) & \gamma_{A,\text{ego}} = 0\end{cases}. \tag{B.29}$$

Here, $\mathcal{B}$ is the Bernoulli distribution and $\mathcal{N}_T$ is a truncated normal distribution with bounds in $(0, 1)$. We assume for the belief update (Equation (B.3)) that $\boldsymbol{\sigma}_{\boldsymbol{u}} = \boldsymbol{\sigma}_{\boldsymbol{u},0} = [3\,\text{ms}^{-2}, 0.4575\,\text{s}^{-1}]$ and, when using communication in Chapter 8, $\sigma_\gamma = \sigma_{\gamma,0} = 0.005$. Meanwhile,

$$\begin{aligned}\boldsymbol{\sigma}_{\boldsymbol{u}} &= 0.2\boldsymbol{\sigma}_{\boldsymbol{u},0} \\ \sigma_\gamma &= 0.2\sigma_{\gamma,0}\end{aligned} \tag{B.30}$$

is used when predicting future states during model evaluation (Equation (B.17)). Here, the factor of 0.2 was hand tuned.

Importantly, in Equation (B.28), we only define p_0, not p. This is because this equation only reflects the kinematic likelihood of a state. Meanwhile, our actual transition function is defined as

$$p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a}) \propto \check{p}_{\text{n}}(\boldsymbol{s}')p_0(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a}) \tag{B.31}$$

Here, the *projected normative probability* $\check{p}_{\text{n}}$ implements our norm conditioning. Specifically, we assign lower likelihood to states which in the near or medium future will result in norm violating behavior, with

$$\check{p}_n(\boldsymbol{s}_t) \propto \min\left\{p_{\text{n}}(\boldsymbol{s}_t), \left(\frac{1}{|\check{T}|}\sum_{\tau\in\check{T}}p_n(\check{\boldsymbol{s}}_{t+\tau})^{-1}\right)^{-1}\right\}. \tag{B.32}$$

In the collision avoidance model in Chapter 7, we use $\check{T} = \{1, H_{\text{n}}\}$ for the calculation of $\check{p}_{\text{n}}$, while in interactive model in Chapter 8, we instead use $\check{T} = \{1, \frac{1}{2}(1 + H_{\text{n}}), H_{\text{n}}\}$. Here, the

normative probability p_n is used to weigh a state's current norm compliance (*i.e.*, while some states $\boldsymbol{s}$ might be kinematically equally likely to others, it might be perceived as less likely by a human driver because it violates some norms, such as driving on the wrong side of the road). Meanwhile, we use $H_\mathrm{n} = 20$ as our medium future prediction horizon. The upper bound of $p_\mathrm{n}(\boldsymbol{s}_t)$ ensures that if the current state is already norm-violating, we increase uncertainty (as our bias towards norm-compliant behavior is no longer supported by our observations). Specifically, we use our base state transition function p_0 (Equation (B.28)) to get the states in the near future:

$$\check{\boldsymbol{s}}_\tau \sim \hat{p}(\check{\boldsymbol{s}}_\tau \mid \check{\boldsymbol{s}}_{\tau-1}, \boldsymbol{a}_{v,t-1}). \tag{B.33}$$

Importantly, during this update, we use constant control actions for the other agents, by setting $\boldsymbol{\sigma}_{\boldsymbol{u}} = 0$ in Equation (B.28).

However, unless $\check{p}_\mathrm{n}$ turns out to be Gaussian (which is highly unlikely), we cannot sample from $p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$ directly. Therefore, under the assumption that we can sample from $p_0(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$, we generate $M_\mathrm{n} = 32$ candidate samples $\boldsymbol{s}'_m$ and randomly select one based on their normative likelihood:

$$\boldsymbol{s}' \sim \frac{\sum_{m=1}^{M_\mathrm{n}} \check{p}_\mathrm{n}(\boldsymbol{s}'_m)\delta(\boldsymbol{s}' - \boldsymbol{s}'_m)}{\sum_{m=1}^{M_\mathrm{n}} \check{p}_\mathrm{n}(\boldsymbol{s}'_m)}. \tag{B.34}$$

Additionally, by assuming that normative expectations p_n are attributed to individual agents independently, and that, therefore, we have

$$p_\mathrm{n}(\check{\boldsymbol{s}}) = \prod_{v \in V} p_{\mathrm{n},V}(\check{\boldsymbol{x}}_v, \check{\boldsymbol{y}}_v, \check{\boldsymbol{u}}_v \mid \check{\boldsymbol{x}}_\mathrm{ego}, \check{\boldsymbol{y}}_\mathrm{ego}, \check{\boldsymbol{u}}_\mathrm{ego}), \tag{B.35}$$

where $p_{\mathrm{n},V}$ is the single-agent normative weight function assigning the likelihood of norm compliant interactions between a surrounding other agent v and the ego agent making the predictions.

We can then calculate an agent wise $\check{p}_{\mathrm{n},V}$ (similar to Equation (B.32)), enabling us to randomly compose the final $\boldsymbol{s}'$ out of components from the M_n candidate states $\boldsymbol{s}'_m$ (*i.e.*, we choose the final $\boldsymbol{x}'_v$ and $\boldsymbol{u}'_v$ for each surrounding other agent v from a potentially different $\boldsymbol{s}'_m$). This allows us to preserve the independence between agents with a significantly smaller number M_n.

However, it must be acknowledged that if $p_{\mathrm{n},V}$ actually depends on the states of the ego vehicle, the behavior prediction of the surrounding agents is no longer independent from the policy selection, and can therefore no longer be preformed as a separate step before. Instead, the behavior of the other vehicles now needs to be rolled out for each considered policy separately, which will increase computation times significantly.

B.2.2 Observation probability

For the *generative process*, we assume that observations are exact.

$$\hat{p}(\boldsymbol{o}'|\boldsymbol{\eta}) = \delta\left(\boldsymbol{o}' - \boldsymbol{\eta}\right) \tag{B.36}$$

For the *generative model* meanwhile, we use a observation probability that follows the style laid out in Equations (B.4), (B.5), and (B.6). This requires use to define

1. the bijective mapping $\mathscr{L}$.
2. the linear transformation from state to observation space defined by $\boldsymbol{A}$ and $\boldsymbol{b}$.
3. the offset function μ.
4. the variance function $\boldsymbol{\Sigma}$.

An important part of our observation function is looming-based perception. For this, we can introduce the looming state $\boldsymbol{\varphi}$ for an other agent $v \in V$, including looming angle φ, looming $\dot{\varphi}$, and looming rate $\ddot{\varphi}$:

$$\boldsymbol{\varphi}_v = \{\varphi_v, \dot{\varphi}_v, \ddot{\varphi}_v, y_v, \theta_v, \delta_v, \omega_v\} = f_{\text{loom}}(\boldsymbol{x}_{\text{ego}}, \boldsymbol{x}_v, \boldsymbol{u}_v). \tag{B.37}$$

Those looming states can be calculated in the following way:

$$\begin{aligned}
\varphi_v &\approx 2\arctan\left(\frac{d}{2(x_v - x_{\text{ego}})}\right) \\
\dot{\varphi}_v &\approx -\frac{d\left(v_v\cos(\theta_v) - v_{\text{ego}}\right)}{(x_v - x_{\text{ego}})^2 + \frac{1}{4}d^2} \\
\ddot{\varphi}_v &\approx \frac{d}{(x_v - x_{\text{ego}})^2 + \frac{1}{4}d^2}\left(a_{\text{long,ego}} - a_v\cos(\theta_v) + \frac{2(x_v - x_{\text{ego}})\left(v_v\cos(\theta_v) - v_{\text{ego}}\right)^2}{(x_v - x_{\text{ego}})^2 + \frac{1}{4}d^2}\right).
\end{aligned} \tag{B.38}$$

It must be noted that this mapping is a rough one-dimensional estimate assuming that $\theta_{\text{ego}} \approx 0$, which is only the case for the scenarios discussed in Chapter 7. Additionally, in our model, we make the simplifying assumption that the looming based perception update is only used if the other agent is roughly in front of the agent. This is expressed as condition

$$\mathscr{C}_{\text{loom}}(\boldsymbol{o}, v) = x_v - x_{\text{ego}} > l_r + l_f \tag{B.39}$$

This definition of $\mathscr{C}_{\text{loom}}$ again assumes an ego vehicle moving along the x-axis, and is therefore only applicable to the scenarios discussed in Chapter 7. However, in Chapter 8, we did not use looming-based perception at all, so it can be considered there that $\mathscr{C}_{\text{loom}}(\boldsymbol{o}, v)$ is always false.

The four functions outlined above that are needed to define the observation probability $p(\boldsymbol{o}|\boldsymbol{s})$ based on Equation (B.5), are defined as follows:

1. The mapping $\mathscr{L}$ is implemented as (Equations (B.26), (B.37), and (B.39)):

$$\mathscr{L}(\boldsymbol{o}) = \left\{\boldsymbol{x}_{\text{ego}}, \boldsymbol{u}_{\text{ego}}\right\} \cup \left\{ \begin{cases} \boldsymbol{\varphi}_v & \mathscr{C}_{\text{loom}}(\boldsymbol{o}, v) \\ \boldsymbol{x}_v, \boldsymbol{u}_v & \text{Otherwise} \end{cases} \mid v \in V \right\} \tag{B.40}$$

This mapping is indeed bijective, as the functions defined in Equation (B.38) are.

2. As our state and observation states $\boldsymbol{s}$ and $\boldsymbol{o}$ overlap (Equation (B.26)), we use $\boldsymbol{A} = \boldsymbol{I}$ and $\boldsymbol{b} = 0$.

3. The offset function $\mu(\boldsymbol{o}_{\mathscr{L}})$ to implement the looming threshold. Given a looming threshold of $\dot{\varphi}_0 = 0.002\,15\,\mathrm{s}^{-1}$, we can define here:

$$\mu(\boldsymbol{o}_{\mathscr{L}}) = \{0_7\} \cup \left\{ \begin{cases} \left\{0, \dot{\varphi}_v, \ddot{\varphi}_v - \frac{d}{(x_v - x_{\text{ego}})^2 + \frac{1}{4}d^2}\, a_{\text{ego}}, 0, 0, 0, 0\right\} & \mathscr{C}_{\text{loom}}(\boldsymbol{o}, v) \wedge \left(|\dot{\varphi}_v| \le \dot{\varphi}_0\right) \\ 0_7 & \text{Otherwise} \end{cases} \mid v \in V \right\} \tag{B.41}$$

Here, the offsets defined would correspond to the states of $\boldsymbol{\varphi}$ defined in Equation (B.37).

4. Meanwhile, for the variance $\boldsymbol{\Sigma}$, we define

$$\boldsymbol{\Sigma}(\boldsymbol{o}_{\mathscr{L}}) = \operatorname{diag}\left(\boldsymbol{\sigma}\left(\boldsymbol{o}_{\mathscr{L}}\right)^2\right), \tag{B.42}$$

where

$$\boldsymbol{\sigma}\left(\boldsymbol{o}_{\mathscr{L}}\right) = \{\boldsymbol{\sigma}_{\text{ego}}\} \cup \left\{ \begin{cases} \boldsymbol{\sigma}_{\text{loom,thres}} & |\dot{\varphi}_v| \le \dot{\varphi}_0 \\ \boldsymbol{\sigma}_{\text{loom}} & \text{Otherwise} \end{cases} \quad \begin{matrix} \mathscr{C}_{\text{loom}}(\boldsymbol{o}, v) \\ \text{Otherwise} \end{matrix} \mid v \in V \right\}. \tag{B.43}$$

Here, the states in

$$\boldsymbol{\sigma}_{\text{loom,thres}} = \{0.000\,01, 0.0043\,\mathrm{s}^{-1}, 0.000\,43\,\mathrm{s}^{-2}, 0.000\,02\,\mathrm{m}, 0.0002, 0.002, 0.002\,\mathrm{s}^{-1}\} \tag{B.44}$$

and

$$\boldsymbol{\sigma}_{\text{loom}} = \{0.000\,01, 0.000\,01\,\mathrm{s}^{-1}, 0.000\,001\,\mathrm{s}^{-2}, 0.000\,02\,\mathrm{m}, 0.0002, 0.002, 0.002\,\mathrm{s}^{-1}\} \tag{B.45}$$

correspond to the states of $\boldsymbol{\varphi}$ defined in Equation (B.37), while those in

$$\boldsymbol{\sigma}_{\text{ego}} = \{0.0002\,\mathrm{m}, 0.000\,001\,\mathrm{m}, 0.0002\,\mathrm{ms}^{-1}, 0.000\,001, 0.000\,001, 0\,\mathrm{ms}^{-2}, 0\,\mathrm{s}^{-1}\} \tag{B.46}$$

and

$$\boldsymbol{\sigma}_{\text{kart}} = \{0.000\,01, 0.000\,01\,\mathrm{s}^{-1}, 0.000\,001\,\mathrm{s}^{-2}, 0.000\,02\,\mathrm{m}, 0.0002, 0.002, 0.002\,\mathrm{s}^{-1}\} \tag{B.47}$$

correspond to the states of $\{\boldsymbol{x}, \boldsymbol{u}\}$ (Equations (B.22) and (B.23)).

Regarding the binary variables in Equation (B.6), we use them in Chapter 8 with $\boldsymbol{o}_{\mathscr{B}} = \boldsymbol{s}_{\mathscr{B}} = \{\gamma_v \mid v \in \{\text{ego}\} \cup V\}$ (*i.e.*, for the communication signals of all agents). Here, we generally assume $N_{\mathscr{B}} = 1$ (which would result in $M_{\mathscr{B}} = 1$) for the ego agent, and $N_{\mathscr{B}} = 10$ for the other agents V (with, respectively, $M_{\mathscr{B}} = 0.15$).

B

B.2.3 Control limits

We also want to limit the acceleration and deceleration patterns not achievable by actual human input. To this end, we use f_{real} (Equation (B.20)), which in our case will use two functions, f_{pedal} and f_{jerk}. f_{pedal} prevents unrealistically fast switching between gas and brake pedals, by setting for an acceleration $a_{\text{long},\tau}$:

$$f_{\text{pedal}}(a_{\text{long},\tau}) = \begin{cases} a_0 & (a_{\text{long},\tau-1} - a_0)(a_{\text{long},\tau} - a_0) < 0 \\ a_\tau & \text{otherwise} \end{cases} \tag{B.48}$$

In our current model we assume that the acceleration observed when releasing both pedals is $a_0 = -0.1\,\text{ms}^{-2}$, to approximate the fact that with neutral pedals, wind and roll resistances will lead to some decelerations.

Meanwhile, f_{jerk} tries to implement realistic rates at which the pedals can be pressed and released, by limiting the jerks applied:

$$f_{\text{jerk}}(a_{\text{long},\tau}) = \min\left\{a_{\text{long},\tau-1} + j_{\min}\Delta t, \max\left\{a_{\text{long},\tau}, a_{\text{long},\tau-1} + j_{\max}\Delta t\right\}\right\} \tag{B.49}$$

For the jerk limits, we use:

$$j_{\min} = \begin{cases} -5\,\text{ms}^{-3} & (a_{\text{long},\tau-1} - a_{\text{long},\tau-2}) < 0 \wedge (a_{\text{long},\tau} - a_{\text{long},\tau-1}) > 0 \\ -30\,\text{ms}^{-3} & \text{otherwise} \end{cases}$$

$$j_{\max} = \begin{cases} 0\,\text{ms}^{-3} & (a_{\text{long},\tau-1} - a_{\text{long},\tau-2}) > 0 \wedge (a_{\text{long},\tau} - a_{\text{long},\tau-1}) < 0 \\ \begin{cases} 5\,\text{ms}^{-3} & a_{\text{long},\tau} \geq 0 \\ 15\,\text{ms}^{-3} & a_{\text{long},\tau} < 0 \end{cases} & \text{otherwise} \end{cases} \tag{B.50}$$

Both those functions are applied recursively, with

$$\left[f_{\text{real}}(\boldsymbol{\pi})\right]_\tau = f_{\text{pedal}}\left(f_{\text{jerk}}\left(f_{\text{pedal}}\left(\left[f_{\text{real}}(\boldsymbol{\pi})\right]_{\tau-1}\right)\right)\right) \tag{B.51}$$

B.2.4 Preference function

We use the following preference function p when minimizing the expected free energy G (Equation (B.18)):

$$p(\boldsymbol{o}) = \mathcal{N}(v_{\text{ego}}|v_0, \sigma_v)\mathcal{N}(a_{\text{long,ego}}|0, \sigma_a)\mathcal{N}(\omega_{\text{ego}}|0, \sigma_\omega)p_{\text{lat}}(y_{\text{ego}})p_{\text{coll}}(\boldsymbol{o})p_{\text{safe}}(\boldsymbol{o}) \tag{B.52}$$

Here,

$$p_{\text{lat}}(y_{\text{ego}}) = \mathcal{T}\left(y_{\text{rel}}(\boldsymbol{o}) \mid \frac{w-d}{2}, g_{LC}, g_{LL}\right) \tag{B.53}$$

with the triangular function $\mathcal{T}$:

$$\mathcal{T}(x|x_0, p_1, p_2) \propto \begin{cases} \exp\left(\frac{|x|}{x_0}p_1\right) & |x| \leq x_0 \\ \exp(p_2) & \text{otherwise} \end{cases} \tag{B.54}$$

We also need to define the collision preference p_{coll} with the surrounding vehicles V:

$$p_{\text{coll}}(\boldsymbol{o}_\tau) = \min\{p_{\text{coll}}(\boldsymbol{o}_{\tau-1}), \min_{v \in V} f_{\text{coll}}(\boldsymbol{o}_\tau, v)\} \tag{B.55}$$

The role of the minimum here is to ensure that all timesteps following the collision are still punished, as the model itself has no collision mechanics, allowing vehicles to phase through each other. We than get the collision preference at a single timestep f_{coll}, where we have collision condition $\mathscr{C}(\boldsymbol{o}, v)$, which based on the kinematic states $\boldsymbol{x}_{\text{ego}}$ and $\boldsymbol{x}_v$ of the ego vehicle and the other vehicle v and their length $l = l_f + l_r$ and width d defines whether a collision is occurring by checking for overlapping bounding boxes using the separating axis theorem [365]:

$$f_{\text{coll}}(\boldsymbol{o}, v) = \begin{cases} \exp\left(g_C\left(0.2 + 0.8\frac{|v_{\text{ego}} - v_v \cos(\theta_{\text{ego}} - \theta_v)|}{10\,\text{m}\,\text{s}^{-1}}\right)\right) & \mathscr{C}(\boldsymbol{o}, v) \\ \mathcal{N}\left(\max\left\{\frac{\dot{\varphi}_v}{\varphi_v}, 0.2\,\text{s}^{-1}\right\} \middle| 0.2\,\text{s}^{-1}, \sigma_{\tau^{-1}}\right) & \mathscr{C}_{\text{loom}}(\boldsymbol{o}, v) \\ 1 & \text{Otherwise} \end{cases} \quad \begin{matrix} \\ \\ \text{Otherwise} \end{matrix} . \tag{B.56}$$

Here, the mean for the normal distribution over $\tau_v^{-1} = \frac{\dot{\varphi}_v}{\varphi_v}$ is taken from Markkula *et al.* [213]. In the collision cases ($\mathscr{C}(\boldsymbol{o}, v)$), we adjust the collision cost based on the relative collision speed, based on the assumption that human agents prefer to collide with lower impact velocities, if a collision cannot be avoided.

Lastly, we define the preference p_{safe}, which represents a driver's desire to keep a safe distance when following another vehicle, based on counterfactual assumptions about the lead vehicle's possible braking profile and the driver's own response capability:

$$p_{\text{safe}}(\boldsymbol{o}_\tau) = \min_{v \in V} \begin{cases} \exp\left(\frac{1}{2} g_C\left(0.2 + 0.8\frac{v_{\text{ego}} - v_v \cos(\theta_{\text{ego}} - \theta_v)}{10\,\text{m}\,\text{s}^{-1}}\right)\right) & \mathscr{C}_{\text{brake}}(\boldsymbol{o}, v) \wedge a_{\text{ego, req}} < -a_{\max} \\ 1 & \text{Otherwise} \end{cases} \tag{B.57}$$

The condition $\mathscr{C}_{\text{brake}}$ limits the application of this preference function to car following scenarios (*i.e.*, vehicles are in the same lane, the other vehicle is in front of the model ego agent, and they are driving in the same direction):

$$\mathscr{C}_{\text{brake}}(\boldsymbol{o}, v) = \left(|y_v - y_{\text{ego}}| \leq 1.15\,d\right) \wedge \mathscr{C}_{\text{loom}}(\boldsymbol{o}, v) \wedge \left(\text{sgn}(v_{\text{ego}})\,\text{sgn}\left(v_v \cos(\theta_v)\right) \geq 0\right) \tag{B.58}$$

Meanwhile, $a_{\text{ego, req}}$ is the required deceleration applied after a reaction time of t_{react} needed to avoid a collision if the other vehicle v suddenly started to accelerate toward-s/brake in front of the ego vehicle with $a_{v,\text{test}} = \min\{a_{\text{long},v}, a_{OV,\min}\}$.

$$\begin{aligned} a_{\text{ego, req}} &= -\frac{1}{2}\frac{\max\{v_{\text{ego,react}}, 0\}^2}{\max\{d_{\text{ego,react}} - 1.15\,(l_f + l_r), 0\}} \\ v_{\text{ego,react}} &= v_{\text{ego}} + \min\{a_{\text{long,ego}}, 0\}\,t_{\text{react}} \\ d_{\text{ego,react}} &= \left(x_v - \frac{1}{2}\frac{v_v^2}{a_{\text{nu,test}}}\right) - \left(x_{\text{ego}} + v_{\text{ego}}\,t_{\text{react}} + \frac{1}{2}\min\{a_{\text{long,ego}}, 0\}\,t_{\text{react}}^2\right) \end{aligned} \tag{B.59}$$

Again this function assumes that the ego vehicle is moving along the x-axis, and is therefore only applicable to the scenarios discussed in Chapter 7, however, this is not problematic, as $\mathscr{C}_{\text{loom}}$ and therefore $\mathscr{C}_{\text{brake}}$ would be false in other scenarios, disabling this part of the preference function.

In the interactive scenario in Chapter 8, we expand this preference function with additional terms to also include preferences for explicit communications or deontic cues such as stop signs. For those, we define the relative positions:

$$\begin{aligned} d_{\text{lat}} &= x\cos(\theta_{\text{road}}) + y\sin(\theta_{\text{road}}) \\ d_{\text{lat}} &= y\cos(\theta_{\text{road}}) - x\sin(\theta_{\text{road}}) \end{aligned} \tag{B.60}$$

A further generalization for cases where the lane centerline does not cross the origin could be considered, but was not required in the scenario from Figure 2.1. Those added components will then be multiplied to the preference prior:

- **Speed limit:** The speed limit preference prior discourages agents from driving to fast. It is implemented as

$$p_{\text{SL}}(v_{\text{ego}}) = \begin{cases} \exp\left(g_S \frac{v_{\text{ego}} - 10\,\text{ms}^{-1}}{4.2\,\text{ms}^{-1}}\right) & v_{\text{ego}} > 10.278\,\text{ms}^{-1} \\ 1 & \text{Otherwise} \end{cases} \tag{B.61}$$

- **Stop sign:** For implementing the stop sign rules, for each agent, we implement an additional boolean h_v, which indicate if the ego vehicle has stopped already or not. In the generative model's state transition function, it is updated for an agent v with

$$p(h'_v \mid \boldsymbol{s}, \boldsymbol{a}) = \delta\left(h'_v - f_h(\boldsymbol{s}' \sim p(\boldsymbol{s}' \mid \boldsymbol{s}, \boldsymbol{a}), h_v, v)\right) \tag{B.62}$$

  with the state transition function defined in Equation (B.31) and

$$f_h(\boldsymbol{s}, h_v, v) = h_v \vee \left(\left(-19.425\,\text{m} \leq d_{\text{long},v} \leq -3.925\,\text{m}\right) \wedge \left(v_v < 0.278\,\text{ms}^{-1}\right)\right). \tag{B.63}$$

  Additionally, in the observation function, we assume that $p(h_{o,v}|h_{s,v}) = \delta(h_{o,v} - h_{s,v})$. We can then define the corresponding preference term

$$p_{\text{stop}}(\boldsymbol{o}) = \begin{cases} \exp(g_S) & (-3.925\,\text{m} \leq d_{\text{long,ego}}) \wedge \neg h_{\text{ego}} \\ 1 & \text{Otherwise} \end{cases} \tag{B.64}$$

- **Priority rule:** The preference prior implementing the "first come, first go" priority rule is interactive, *i.e.*, similar to the collision loss, it depends on both the state of the ego agent and the other agent v. For that, we add an additional boolean state p_{ego} which indicates if an agent v assumes that it has priority. Its state transition function is defined as

$$p(p'_{\text{ego}} \mid \boldsymbol{s}, \boldsymbol{a}) = \delta\left(p'_{\text{ego}} - f_{\text{lead}}(\boldsymbol{s}' \sim p(\boldsymbol{s}' \mid \boldsymbol{s}, \boldsymbol{a}), \boldsymbol{s}, p_{\text{ego}}, v)\right). \tag{B.65}$$

Here, the function $f_{\text{lead}}(\boldsymbol{s}', \boldsymbol{s}, p_{\text{ego}}, v)$ assigns priority to the ego agent over agent v (if so, return 1, otherwise, return 0), depending on the existence of stop signs. If those exist, then we define

$$f_{\text{lead}}(\boldsymbol{s}', \boldsymbol{s}, p_{\text{ego}}, v) = p_{\text{ego}} \vee \left(h'_{\text{ego}} \wedge \neg h_{\text{ego}} \wedge \left(d'_{\text{long,ego}} > d'_{\text{long},v} \right) \right). \tag{B.66}$$

Without stop signs, this becomes

$$f_{\text{lead}}(\boldsymbol{s}', \boldsymbol{s}, p_{\text{ego}}, v) = p_{\text{ego}} \vee \left(h_{\text{closed}} \wedge \left(\frac{d'_{\text{long,ego}}}{\max\{0, v_{\text{ego}}\}} > \frac{d'_{\text{long},v}}{\max\{0, v_v\}} \right) \right). \tag{B.67}$$
$$h_{\text{closed}} = d_{\text{long,ego}} < -20\,\text{m} \leq d'_{\text{long,ego}}$$

Similar to the stop rule, in the observation function, we assume that $p(p_{o,\text{ego}}|p_{s,\text{ego}}) = \delta(p_{o,\text{ego}} - p_{s,\text{ego}})$. We can then define the corresponding preference prior:

$$p_{\text{prior}}(\boldsymbol{o}) = \begin{cases} \exp\left(\frac{1}{2} g_S\right) & \left(d_{\text{long,ego}} > \max\{-3.925\,\text{m}, d_{\text{long},v} - 4.425\,\text{m}\}\right) \wedge \neg p_{\text{ego}} \\ 1 & \text{Otherwise} \end{cases} \tag{B.68}$$

Assuming that communication is enabled (see the next item), this function is slightly adjusted so that yielding by the other agent can override priority rules. The notation for the signal states γ is introduced in Section 8.2.3. The adjusted preference prior then becomes:

$$p_{\text{prior}}(\boldsymbol{o}) = \begin{cases} \exp\left(\frac{1}{2} g_S\right) & \left(d_{\text{long,ego}} > \max\{-3.925\,\text{m}, d_{\text{long},v} - 4.425\,\text{m}\}\right) \wedge \neg p_{\text{ego}} \wedge \neg \gamma_{Y,v} \\ 1 & \text{Otherwise} \end{cases} \tag{B.69}$$

- **Signaling:** For signaling there are two different aspects to consider. The first one is the general cost of signaling

$$p_{comm}(\boldsymbol{o}) = \exp\left(g_\gamma \left(\gamma_{A,\text{ego}} + \gamma_{Y,\text{ego}}\right)\right). \tag{B.70}$$

If there are stop signs, it is instead adjusted to prevent signaling before stopping:

$$p_{comm}(\boldsymbol{o}) = \exp\left(g_\gamma \left(\gamma_{A,\text{ego}} + \gamma_{Y,\text{ego}}\right) \begin{cases} 1 & h_{\text{ego}} \\ 10 & \text{Otherwise} \end{cases} \right). \tag{B.71}$$

Meanwhile, there is also a preference prior against uncooperative agents, punish both signaling of yielding without the intention to yield and the lack of a yield signal if intending to yield and prompting by the other agent w. This would be expressed as

$$p_{\text{coop}}(\boldsymbol{o}) = \begin{cases} \exp(g_W) & \left(\gamma_{Y,\text{ego}} \wedge p_{\text{ego}}\right) \vee \left(\gamma_{A,v} \wedge \neg p_{\text{ego}} \wedge \neg \gamma_{Y,\text{ego}}\right) \\ 1 & \text{Otherwise} \end{cases} \tag{B.72}$$

In cases without priority rules, using the sigmoid function σ, we assume that

$$p_{\text{ego}} \sim \mathcal{B}\left(p \mid \sigma\left(3\left(\frac{d_{\text{long,ego}}}{\max\{0, v_{\text{ego}}\}} - \frac{d_{\text{long},v}}{\max\{0, v_v\}}\right)\right)\right) \tag{B.73}$$

The preference function can then be parameterized by the eleven parameters $\sigma_v = 0.5\,\text{ms}^{-1}$, $\sigma_a = 0.1\,\text{ms}^{-2}$, $\sigma_\omega = 0.02\,\text{s}^{-1}$, $\sigma_{\tau^{-1}} = 0.125\,\text{s}^{-1}$, $g_{LC} = -1000$, $g_{LL} = -15000$, $g_C = -10000$, $g_S = -10000$, $g_\gamma = -0.125$, $g_W = -15000$, and $t_{\text{react}} = 1\,\text{s}$. Meanwhile, depending on the simulation, we choose $a_{\text{OV,min}}$ so that the given initial distance and speed would result in stable car following, with lower bound of $a_{\min} = -a_{\max} = -8\,\text{ms}^{-2}$.

Regarding the latter, to achieve this, we run a separate front-to-rear scenario with a leading other vehicle at constant velocity v_0 for different values of $a_{\text{OV,min}}$. For each simulation, we then extract the steady-state following distance that the agent adapts, and the corresponding time gap (*i.e.*, we have a dataset where we have triplets of values for velocity v_0, $a_{\text{OV,min}}$, and the following distance). Using that dataset, we then can use interpolation to get the specific value for $a_{\text{OV,min}}$ associated with a given a velocity v_0 and desired following distance (or desired time gap) in our presented scenarios.

While aimed to be as generalizable as possible, the preference function might require adjustments between scenarios. Namely, when calculating the y_{rel} from Equation (B.59), we have to represent that, for instance, in the front-to-rear scenario both lanes go in one direction, while in the lateral incursion scenario, the left lane is designed for oncoming traffic.

B.3 Implementations of simulation scenarios

Lastly, we present the scenario specific implementations used to generate the results presented in the main text in Chapter 7 and 8.

B.3.1 Front-to-rear scenario

The scenario in Section 7.3.1, which models the response of a driver to the leading other vehicle suddenly braking, contains two vehicles (with the list of other vehicles $V = \{\text{OV}\}$).

Initial state In this scenario, there are 12 initial conditions, with $\boldsymbol{x}_{\text{ego},0} = \{0\,\text{m}, 0\,\text{m}, v_0, 0, 0\}$ and $\boldsymbol{x}_{\text{OV},0} = \{v_0 \Delta t_{\text{tgp},0} + l_f + l_r, 0\,\text{m}, v_0, 0, 0\}$, with $\Delta t_{\text{tgp},0} \in \{0.5\,\text{s}, 1.0\,\text{s}, 1.5\,\text{s}, 2.0\,\text{s}, 2.5\,\text{s}, 3.0\,\text{s}, 3.5\,\text{s}\}$ and $v_0 \in \{10\,\text{ms}^{-1}, 15\,\text{ms}^{-1}, 20\,\text{ms}^{-1}, 25\,\text{ms}^{-1}\}$. We assume a lane width $w = 3.65\,\text{m}$, and vehicle sizes of $d = 1.72\,\text{m}$, $l_f = 2.1\,\text{m}$, and $l_r = 2.1\,\text{m}$.

Behavior of the other vehicle In this scenario, we set $\boldsymbol{u}_{\text{OV,preset}}$ so that the other vehicle will drive straight on for exactly 0.6 s, after which it will start to decelerate, applying a jerk of $-10\,\text{ms}^{-3}$ until reaching an acceleration value of $-6\,\text{ms}^{-2}$. It will keep this acceleration until it comes to a standstill.

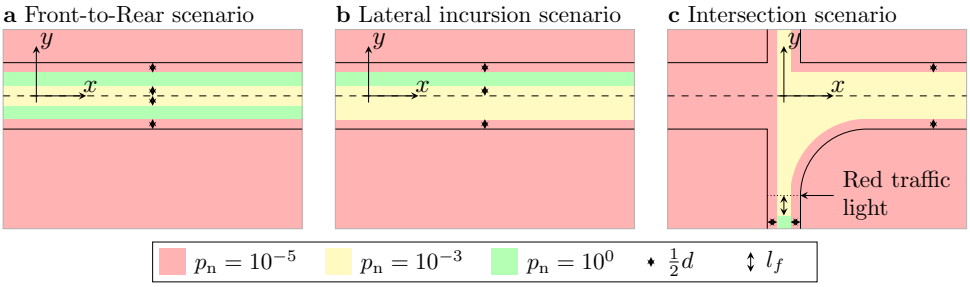


Figure B.1: A symbolic representation of the normative probability $p_n(s)$ in our considered scenarios. As this only depends on x_{OV} and y_{OV} in our specific implementation, we depict is a birds-eye view map. Importantly, those sketches are not to scale.

Lateral preference Here, we calculate the underlying lateral relative position y_{rel} from Equation (B.59) as:

$$y_{rel}(\boldsymbol{o}) = \begin{cases} y_{ego} & y_{ego} \leq \frac{w-d}{2} \\ \frac{w-d}{2} & \frac{w-d}{2} < y_{ego} \leq \frac{w+d}{2} \\ y_{ego} - w & \frac{w+d}{2} < y_{ego} \end{cases} \tag{B.74}$$

Norm conditioning Our definition of the the normative probability $p_n(s)$ (Equation (B.18)) can be seen in Figure B.1**a**. Given the usage of $p_n(s)$ as a weighing function between different samples, we did not normalize it here (*i.e.*, ensuring $\int p_n(s)\mathrm{d}s = 1$ is not necessary).

B.3.2 Opposite-direction lateral incursion scenario

The scenario in Section 7.3.2 contains two vehicles (with the list of other vehicles $V = \{OV\}$) and was implemented to replicate the scenario in [244] as closely as possible.

Initial state In this scenario, there is a single initial condition, with $\boldsymbol{x}_{ego,0} = \{0\,m, 0\,m, v_0, 0, 0\}$ and $\boldsymbol{x}_{OV,0} = \{202.688\,m, 0\,m, v_0, \pi, 0\}$ (the initial velocities $v_0 = 17.88\,ms^{-1}$ correspond to 40 mph). Lane width and vehicle size are identical to the front-to-rear scenario (Section B.3.1)

Other vehicle's behavior The other vehicle's path is preprogrammed in a manner that it start turning to the left when the time to collision $((x_{OV} - x_{ego})/(v_{OV} + v_{ego}))$ falls below 5.15 s. This turn will last for 3.3 s, at which point the other vehicle's front left corner should start crossing the central lane marker, after which the other vehicle will follow a straight path. Following [244], we run this scenario in 3 different variants, where after 5.15 s seconds, we perceive $y_{OV} = -0.4w$ (Steep incursion), $y_{OV} = 0\,m$ (Medium incursion), or $y_{OV} = 0.45w$ (Shallow incursion) (Figure B.2).

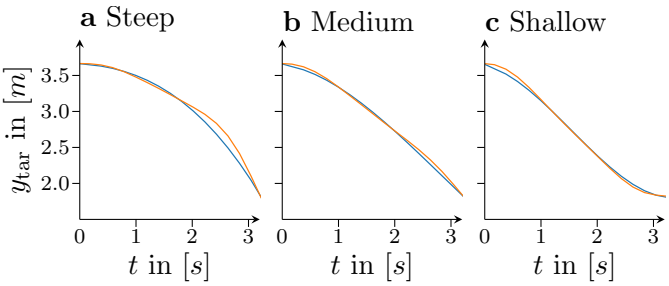


Figure B.2: The different maneuvers by the other vehicle. The orange line represents the trajectory used in our simulation, while the blue line corresponds to the original experiment [244]. $t = 0$ corresponds to the start of the maneuver, while the horizontal velocity stays constant. Afterwards, the other vehicle continues along a straight line with constant velocity.

.

Lateral preference Here, we calculate the underlying lateral relative position y_{rel} from Equation (B.59) as:

$$y_{\text{rel}}(\boldsymbol{o}) = \begin{cases} y_{\text{ego}} & y_{\text{ego}} \leq \frac{w-d}{2} \\ \frac{w-d}{2} & \frac{w-d}{2} < y_{\text{ego}} \leq \frac{3w-d}{2} \\ y_{\text{ego}} - w & \frac{3w-d}{2} < y_{\text{ego}} \end{cases} \tag{B.75}$$

Norm conditioning Our definition of the the normative probability $p_{\text{n}}(\boldsymbol{s})$ (Equation (B.18)) can be seen in Figure B.1**b**.

B.3.3 Intersection scenario

The scenario from Section 7.3.3 contains two vehicles (with the list of other vehicles $V = \{\text{OV}\}$) and was implemented to replicate the scenario in [250] as closely as possible.

Initial state In this scenario, there are two different conditions. In the first condition (RS), the other vehicle entering the intersection is initially stopped in front of the red light, waiting to perform a right turn on red (which is legal in US and Canada), while in the second condition (RNS), the other vehicle drives with a constant velocity. The red light and its associated stopping line is located 10 m in front of the entrance to the intersection. With the coordinate system centered on the intersection of the two orthogonal lanes, we therefore get for RS $x_{\text{OV},0} = 0$, $y_{\text{OV},0} = -13.925\,\text{m}$, and $v_{\text{OV},0} = 0$. In the RNS scenario, we got $x_{\text{OV},0} = 0$, $y_{\text{OV},0} = -18.71\,\text{m}$, and $v_{\text{OV},0} = 6.802\,\text{ms}^{-1}$. Given that starting location, we also have in both scenarios $\theta_{\text{OV},0} = \frac{1}{2}\pi$. In general, while participants in [250] were told to follow the speed limit of 60 kph, actual velocities were in the range of 50 to 60 kph (no concrete distribution given). We assumed therefore that with equal likelihood the ego vehicle had one of three initial velocities $v_{\text{ego},0} \in \{14.67\,\text{ms}^{-1}, 15.5\,\text{ms}^{-1}, 16.33\,\text{ms}^{-1}\}$. As the time to collision (assuming constant velocities by the ego agent) was fixed, we got six different starting locations for the ego agent, each associated with a different scenario type and

starting velocity. For RS, we got $\boldsymbol{x}_{\text{ego},0} \in \{-62.67\,\text{m}, -66.49\,\text{m}, -70.27\,\text{m}\}$, and for RNS, it was $\boldsymbol{x}_{\text{ego},0} \in \{-35.60\,\text{m}, -37.87\,\text{m}, -40.11\,\text{m}\}$, respectively. Meanwhile, we had $\boldsymbol{y}_{\text{ego},0} = \theta_{\text{ego},0} = 0$. Lane width and vehicle size are identical to the front-to-rear scenario (Section B.3.1)

Other vehicle's behavior The other vehicle is preprogrammed to follow a specific path, where once it reaches $\boldsymbol{y}_{\text{OV}} = -13.925\,\text{m}$, it will try to follow a quarter circle toward the right with a radius of 13.925 m. Once it levels out (correspondingly at $\boldsymbol{y}_{\text{OV}} = 0\,\text{m}$, it will follow its lane up to a straight line. Important to note, this circle path is an assumption, as it was not directly specified in [250].

In RS scenario, the other vehicle will start to accelerate at $t = 0.6\,\text{s}$ with $\boldsymbol{a}_{\text{OV}} = 1.4715\,\text{ms}^{-2}$ until reaching $\boldsymbol{v}_{\text{OV}} = 6.802\,\text{ms}^{-1}$. By contrast, in the RNS scenario, the other vehicle will keep a constant velocity.

Lateral preference Here, we calculate the underlying lateral relative position y_{rel} from Equation (B.59) as:

$$y_{\text{rel}}(\boldsymbol{o}) = \begin{cases} y_{\text{ego}} & y_{\text{ego}} \leq \frac{w-d}{2} \\ \frac{w-d}{2} & \frac{w-d}{2} < y_{\text{ego}} \leq \frac{w+d}{2} \\ y_{\text{ego}} - w & \frac{w+d}{2} < y_{\text{ego}} \end{cases} \tag{B.76}$$

Norm conditioning Our definition of the the normative probability $p_{\text{n}}(\boldsymbol{s})$ (Equation (B.18)) can be seen in Figure B.1**c**.

B.3.4 Interactive scenario

The scenario Chapter 8 contains two modeled vehicles. However, as each agent has their own *generative model*, the list of other vehicles is $V = \{\text{OV}\}$.

Initial state This is described in detail in Chapter 8 itself.

Lateral preference Here, we calculate the underlying lateral relative position y_{rel} from Equation (B.59) as:

$$y_{\text{rel}}(\boldsymbol{o}) = d_{\text{lat,ego}} \tag{B.77}$$

Norm conditioning For the normative probability $p_{\text{n}}(\boldsymbol{s})$ (Equation (B.18)), we use the following parts, which are multiplied if applicable:

- **Lane following:** We assume that the other vehicle $v \in V$ will try to follow their lane:

$$p_{n,\text{LF}}(\boldsymbol{s}) = \begin{cases} 10^{-5} & |d_{\text{lat},v}| > \frac{w-d}{2} \\ 1 & \text{Otherwise} \end{cases}. \tag{B.78}$$

- **Speed limit:** In regard to speeding of the other vehicle v, we assume that p_n is multiplied with

$$p_{n,\mathrm{SL}}(\boldsymbol{s}) = \begin{cases} 0.02 & v_v > 11.5\,\mathrm{ms}^{-1} \\ 1 & \text{Otherwise} \end{cases}. \tag{B.79}$$

- **Stop sign:** If stop signs exist, we use the further multiplier

$$p_{n,\mathrm{stop}}(\boldsymbol{s}) = \begin{cases} 0.02 & (-3.925\,\mathrm{m} \leq d_{\mathrm{long},v}) \wedge \neg h_v \\ 1 & \text{Otherwise} \end{cases}. \tag{B.80}$$

  with the boolean h_v defined in Equation (B.62).

- **Priority rule:** Meanwhile, if the priority rule is enabled, the ego agent assigns the other agent v the following normative probability multiplier, depending on its priority state p_{ego} (Equation (B.65)):

$$p_{n,\mathrm{priority}}(\boldsymbol{s}) = \begin{cases} 0.02 & \left(d_{\mathrm{long},v} > \max\{-3.925\,\mathrm{m}, d_{\mathrm{long,ego}} - 4.425\,\mathrm{m}\}\right) \wedge p_{\mathrm{ego}} \\ 1 & \text{Otherwise} \end{cases}. \tag{B.81}$$

- **Signaling:** If communication is enabled, the model is biased to believe that the other agent will follow up on their signaled yielding (the signal states γ are defined in Section 8.2.3). However, to prevent random fluctuation in this belief influencing predictions, we assume that this will only impact the belief in the other agent's yielding intention needs to be sufficiently strong to have an effect ($\gamma_{Y,v} > 0.2$):

$$p_{n,\mathrm{coop}}(\boldsymbol{s}) = \begin{cases} 1 - 1.4 \max\{\gamma_{Y,v} - 0.3, 0\} & d_{\mathrm{long},v} > \max\{-3.925\,\mathrm{m}, d_{\mathrm{long,ego}} - 4.425\,\mathrm{m}\} \\ 1 & \text{Otherwise} \end{cases}. \tag{B.82}$$

C

Additional Experiments for the Active Inference Model

In this chapter, further experiments that extend Chapter 7 are presented. Specifically, Section C.1 illustrates the effect of the norm-conditioned particle filter in a non-incursion scenario, while Section C.2 shows the impact of the epistemic value in a limited visibility scenario. Finally, Section C.3 presents an evaluation of a deep learning model in the opposite-direction lateral incursion scenario.

C.1 Norm conditioning in a non-incursion scenario

To further illustrate the effect of norm conditioning, we implemented a variation of the lateral incursion scenario, where the other vehicle just drives along its lane without any change in direction or speed. Specifically, we start the vehicles at an initial distance of 150 m, both driving with $15\,\mathrm{ms}^{-1}$. Those simulations are repeated 20 times for both the full proposed model as well as the one without the norm conditioned particle filter. As seen in Figure C.1, the full model does not deviate from its desired state in response to the other agent. By contrast, when using the model without norm conditioning, the agent either moves to the right, brakes, or does both. As this type of over-reactive behavior not representative of everyday driving behavior, these results highlight the need for the norm-conditioned particle filter. Of course, it must be noted that similar results without the norm-conditioned particle filter could be achieved by simply removing the prediction noise, but our analyses in the main text show that it is essential for realistic collision avoidance behavior.

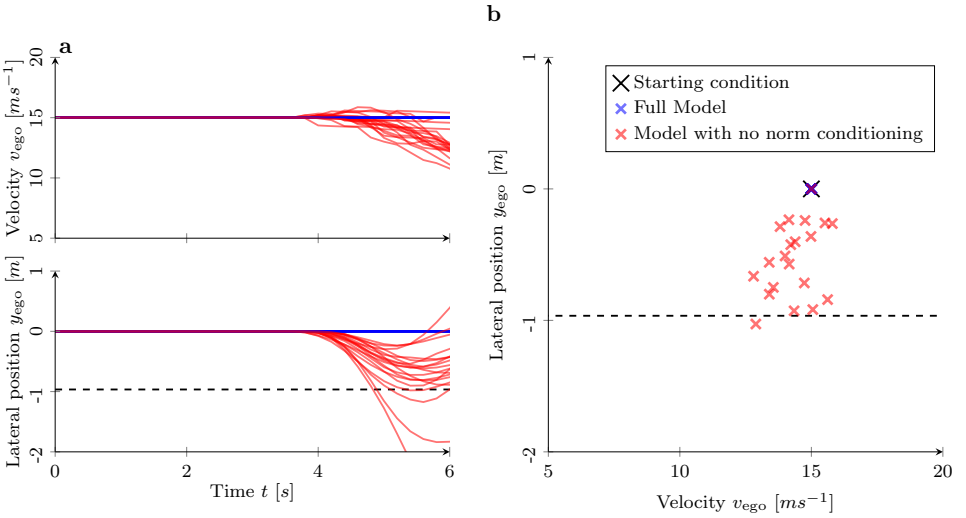


Figure C.1: Results in the 20 repetitions of the non-incursion scenario, for both the model with norm conditioned particle filter (blue) and without (red). **a)** Change of longitudinal velocity v_{ego} and lateral position y_{ego} of simulated agents over time. **b)** The kinematic state of the ego vehicle at the point in time when passing the oncoming other vehicle.

C.2 Epistemic value in a limited visibility scenario

In this scenario, we have a variation of the front-to-rear scenario, with an initial velocity of $v = 10\,\mathrm{ms}^{-1}$, and $x_{ego} = 0\,\mathrm{m}$ $x_{OV} = 20\,\mathrm{m}$ (i.e., there is an initial time gap of 1.58 s). However, instead of the leading other vehicle braking, we instead consider a scenario where there is a sudden onset of reduced visibility (e.g., due to fog) after the first timestep which severely limits the ego's sight distance. This is implemented in a way that for the condition $x_{ego} - x_{OV} < -17\,\mathrm{m}$, the standard deviation for the longitudinal states (either x_{OV}, v_{OV}, and a_{OV} without looming (see B.2.2) or φ, $\dot{\varphi}$, and $\ddot{\varphi}$ otherwise) in Equations (B.44), (B.45), and (B.46) are multiplied by 1000.

In this scenario, we then compared our full model (with epistemic value considered

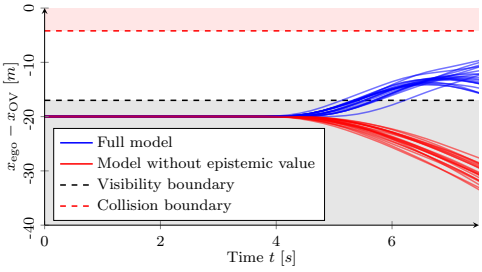


Figure C.2: Results in the 20 repetitions of the scenario with the reduced visibility, for both the model full model (blue) and the one without epistemic value (red). Change of the longitudinal distance $x_{ego} - x_{OV}$.

during the policy selection) against a version of the model without epistemic value (which could be seen as a baseline MPC model with the pragmatic value as the cost function). We can see that in this scenario the epistemic value has a significant impact, encouraging the model to accelerate and regain visibility, while in the baseline model without epistemic value, the vehicle slows down to very low speeds to reduce the likelihood of a collision. This effect is the result of the selection of an (accelerating) policy that improves observability, driven by epistemic value. Specifically, as described above, by decreasing the distance to the leading vehicle, the modeled agent can decrease the standard deviation σ in Equations (B.44), (B.45), and (B.46) and therefore the observation variances in Σ (see Equation (B.42)). Consequently, the observation probability $p(\boldsymbol{o}|\boldsymbol{s}, \boldsymbol{\theta}_o)$ as defined in Equation (B.5) will have a reduced entropy $\mathcal{H}(p(\boldsymbol{o}|\boldsymbol{s}, \boldsymbol{\theta}_o))$ (i.e., lower expected ambiguity in (B.16)). Based on Equation (B.18) and (B.16), this reduction in entropy will increase the epistemic value of such a policy, and thereby reduce the expected free energy and make the policy more likely to be selected.

C.3 Evaluating deep learning models in collision avoidance

We evaluated a deep-learning–based human behavior model in the opposite-direction lateral incursion scenario. While such data-driven models usually perform well in non-conflict driving scenarios, their generalizability to collision-avoidance situations (typically sparse in the training data [67]) is less explored. Specifically, we used *ADAPT* [76], a model previously shown to exhibit state-of-the-art robustness and generalizability [124], trained on *Argoverse2*, a standard benchmark dataset for human trajectory prediction [72]. To make the model reactive, we executed the scenario in closed loop, updating the 64 rollouts every 0.1 s. As shown in Figure C.3, *ADAPT* failed to reproduce the behavior of the human participants. The model almost exclusively selected avoidance by swerving to the right, which is likely explained by the fact that encounters involving a vehicle entering oncoming traffic are exceedingly rare in its training data. It also reacted substantially faster than humans, which is expected for models trained with position-based losses that do not constrain reaction times. Although alternative loss formulations might improve the real-

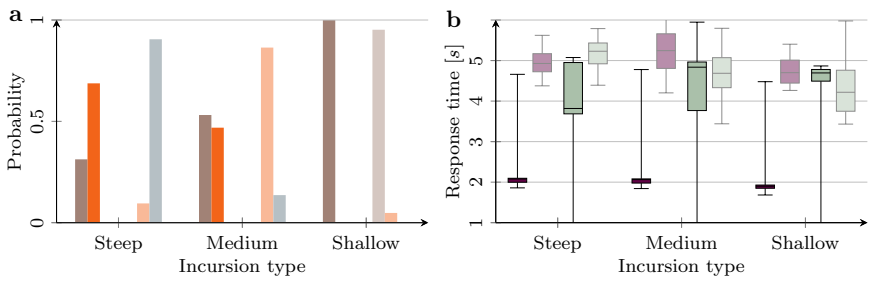


Figure C.3: Evaluation of the deep learning model *ADAPT* [76] in the opposite-direction lateral incursion scenario. **a**) The likelihood of the ego vehicle successfully avoiding the other vehicle by steering left (toward the opposite lane) or right (toward the shoulder), or failing to avoid a collision. **b**) Braking and steering response times. For a legend, see Figure 7.4.

ism of the model, it will likely be difficult to accurately model human collision avoidance behavior with these types of data driven models unless training datasets include substantial amounts of collision and near-collision behavior.

D

The Epistemic Value of Prompting in the Intersection Interaction

In the active inference model introduced in Chapter 8, the ego agent chooses to prompt the other agent v for their *yielding intent* ($\gamma_{A,\text{ego}} = 1$), if the epistemic value advantage of that action is bigger than the pragmatic cost g_γ associated with signaling. We can show that this overall advantage (here defined as g_{prompt}) mostly depends on the current predictive posterior entropy of this action.

$$g_{\text{prompt}}(q(\gamma_{Y,v})) = g_{\text{epist}}(\tilde{q}(\gamma_{Y,v}|\gamma_{A,\text{ego}} = 1)) - g_{\text{epist}}(\tilde{q}(\gamma_{Y,v}|\gamma_{A,\text{ego}} = 0)) - g_\gamma \tag{D.1}$$

Here, assuming that $q(\gamma_{Y,v}) \approx \delta(\gamma_{Y,v} - p_{\text{yield}})$, we can find the following using the state transition function (Equation (B.29)):

$$\begin{aligned}
\tilde{q}(\gamma'_{Y,v}|\gamma_{A,\text{ego}} = 1) &= \mathbb{E}_{q(\gamma_{Y,v})} p(\gamma'_{Y,v}|\gamma_{Y,v}, \gamma_{A,\text{ego}}) \\
&= \mathbb{E}_{\delta(\gamma_{Y,v} - p_{\text{yield}})} \mathcal{B}\left(\gamma'_{Y,v}|\gamma_{Y,v}\right) \\
&= \mathcal{B}\left(\gamma'_{Y,v}|p_{\text{yield}}\right) \\
\tilde{q}(\gamma'_{Y,v}|\gamma_{A,\text{ego}} = 0) &= \mathbb{E}_{q(\gamma_{Y,v})} p(\gamma'_{Y,v}|\gamma_{Y,v}, \gamma_{A,\text{ego}}) \\
&= \mathbb{E}_{\delta(\gamma_{Y,v} - p_{\text{yield}})} \mathcal{N}_T(\gamma'_{Y,v}|\gamma_{Y,v}, \sigma_y) \\
&\approx \delta(\gamma'_{Y,v} - p_{\text{yield}})
\end{aligned} \tag{D.2}$$

Here, we ignore the influence of the noise σ_y, as for one timestep, it is negligible. Using the definition of the epistemic value in Equation (7.7) (where, using Equation (B.6) with $N_{\mathcal{B}} = 10$ we assume that $p(\boldsymbol{o}|\boldsymbol{s}) = p(\gamma_{Y,v}|\gamma_{Y,v}) = \mathcal{B}\left(\gamma_{Y,v}|\gamma^{10}_{Y,v}\right)$), we can find that

$$\begin{aligned}
g_{\text{epist}}(\tilde{q}(\gamma_{Y,v}|\gamma_{A,\text{ego}} = 1)) &\approx g_{\text{epist}}\left(\mathcal{B}\left(\gamma_{Y,v}|p_{\text{yield}}\right)\right) \\
&\approx \mathcal{H}\left(\mathbb{E}_{\mathcal{B}\left(\gamma_{Y,v}|p_{\text{yield}}\right)} \mathcal{B}\left(\gamma'_{Y,v}|\gamma^{10}_{Y,v}\right)\right) \\
&\quad - \mathbb{E}_{\mathcal{B}\left(\gamma_{Y,v}|p_{\text{yield}}\right)} \mathcal{H}\left(\mathcal{B}\left(\gamma'_{Y,v}|\gamma^{10}_{Y,v}\right)\right) \\
&\approx \mathcal{H}\left(\mathcal{B}\left(\gamma'_{Y,v}|p_{\text{yield}}\right)\right),
\end{aligned} \tag{D.3}$$

where the second term goes to zero, as – being sampled from a Bernoulli distribution – $\gamma_{Y,v} \in \{0,1\}$ (and therefore $\gamma^{10}_{Y,v} \in \{0,1\}$) results in $\mathscr{H}\left(\mathscr{B}\left(\gamma'_{Y,v}|0\right)\right) = \mathscr{H}\left(\mathscr{B}\left(\gamma'_{Y,v}|1\right)\right) = 0$. We can also find that

$$\begin{aligned} g_{\text{epist}}(\tilde{q}(\gamma_{Y,v}|\gamma_{A,\text{ego}}=0)) &\approx g_{\text{epist}}(\delta(\gamma_{Y,v}-p_{\text{yield}})) \\ &\approx \mathscr{H}\left(\mathbb{E}_{\delta(\gamma_{Y,v}-p_{\text{yield}})}\mathscr{B}\left(\gamma'_{Y,v}|\gamma^{10}_{Y,v}\right)\right) \\ &\quad - \mathbb{E}_{\delta(\gamma_{Y,v}-p_{\text{yield}})}\mathscr{H}\left(\mathscr{B}\left(\gamma'_{Y,v}|\gamma^{10}_{Y,v}\right)\right) \\ &\approx \mathscr{H}\left(\mathscr{B}\left(\gamma'_{Y,v}|p^{10}_{\text{yield}}\right)\right) - \mathscr{H}\left(\mathscr{B}\left(\gamma'_{Y,v}|p^{10}_{\text{yield}}\right)\right) \\ &\approx 0, \end{aligned} \tag{D.4}$$

and therefore Equation (D.1) simplifies to:

$$\begin{aligned} g_{\text{prompt}}(q(\gamma_{Y,v})) &= g_{\text{epist}}(\tilde{q}(\gamma_{Y,v}|\gamma_{A,\text{ego}}=1)) - g_{\text{epist}}(\tilde{q}(\gamma_{Y,v}|\gamma_{A,\text{ego}}=0)) - g_\gamma \\ &\approx \mathscr{H}\left(\mathscr{B}\left(\gamma'_{Y,v}|p_{\text{yield}}\right)\right) - g_\gamma\,. \end{aligned} \tag{D.5}$$

Consequently, it can be easily seen that $g_{\text{prompt}}(q(\gamma_{Y,v}) \approx \delta(\gamma_{Y,v} - p_{\text{yield}}))$ is maximized for $p_{\text{yield}} \approx 0.5$.

Bibliography

References

[1] Waymo LLC. Safety at waymo: Self-driving cars and other road users, May 2019. Accessed: 2026-01-28.

[2] Daniel Holland-Letz, Matthias Kässer, Benedikt Kloss, and Thibaut Müller. Mobility ' s future: An investment reality check | McKinsey, April 2021.

[3] Ana Norman-López, Matthias Weitzel, Marie Tamba, Louison Duboz, Jette Krause, and Biagio Ciuffo. Economy-wide impacts from different speeds of deployment of automated vehicles on european union roads. *Technology in Society*, page 103066, 2025.

[4] Piotr Szymanski, Biagio Ciuffo, Georgios Fontaras, Giorgio Martini, and Ferenc Pekar. The future of road transport in europe. environmental implications of automated, connected and low-carbon mobility. *Combustion Engines*, 60, 2021.

[5] World Health Organization. *Global status report on road safety 2023: country and territory profiles*. World Health Organization, 2024.

[6] Santokh Singh. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. Technical report, Transportation Research Board, 2015.

[7] Safe Autonomy: A Reality Check on the 94 Percent Human Error Statistic for Automated Cars, June 2018.

[8] Jaspreet Singh Brar and Brian Caulfield. Impact of autonomous vehicles on pedestrians' safety. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pages 714–719, October 2017.

[9] Jonas Meyer, Henrik Becker, Patrick M. Bösch, and Kay W. Axhausen. Autonomous vehicles: The next jump in accessibilities? *Research in Transportation Economics*, 62:80–91, June 2017.

[10] Jelena Pisarov and Gyula Mester. The future of autonomous vehicles. *FME Transactions*, 49(1):29–35, 2021.

[11] Mohamed Abdel-Aty and Shengxuan Ding. A matched case-control analysis of autonomous vs human-driven vehicle accidents. *Nature communications*, 15(1):4931, 2024.

[12] Michael Milford, Sam Anthony, and Walter Scheirer. Self-Driving Vehicles: Key Technical Challenges and Progress Off the Road. *IEEE Potentials*, 39(1):37–45, January 2020.

[13] Jun Wang, Li Zhang, Yanjun Huang, and Jian Zhao. Safety of Autonomous Vehicles. *Journal of Advanced Transportation*, 2020:e8867757, October 2020.

[14] Amolika Sinha, Sai Chand, Vincent Vu, Huang Chen, and Vinayak Dixit. Crash and disengagement data of autonomous vehicles on public roads in California. *Scientific Data*, 8(1):298, December 2021.

[15] Sajjad Mozaffari, Omar Y. Al-Jarrah, Mehrdad Dianati, Paul Jennings, and Alexandros Mouzakitis. Deep Learning-Based Vehicle Behavior Prediction for Autonomous Driving Applications: A Review. *IEEE Transactions on Intelligent Transportation Systems*, 23(1):33–47, August 2020.

[16] Julian F. Schumann, Jens Kober, and Arkady Zgonnikov. Benchmarking Behavior Prediction Models in Gap Acceptance Scenarios. *IEEE Transactions on Intelligent Vehicles*, 8(3):2580–2591, March 2023.

[17] Daniel J Fremont, Edward Kim, Yash Vardhan Pant, Sanjit A Seshia, Atul Acharya, Xantha Bruso, Paul Wells, Steve Lemke, Qiang Lu, and Shalin Mehta. Formal scenario-based testing of autonomous vehicles: From simulation to the real world. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–8. IEEE, 2020.

[18] Nico Montali, John Lambert, Paul Mougin, Alex Kuefler, Nicholas Rhinehart, Michelle Li, Cole Gulino, Tristan Emrich, Zoey Yang, Shimon Whiteson, Brandyn White, and Dragomir Anguelov. The Waymo Open Sim Agents Challenge. *Advances in Neural Information Processing Systems*, 36:59151–59171, December 2023.

[19] Waymo Team. Benchmarking av safety: Demonstrating how the waymo driver outperforms the collision responses of always attentive human drivers via industry-leading assessment methods. `https://waymo.com/blog/2022/09/benchmarking-av-safety/`, September 2022. Accessed: 2026-03-11.

[20] Pierluigi Olleja, Gustav Markkula, and Jonas Bärgman. Validation of human benchmark models for automated driving system approval: How competent and careful are they really? *Accident Analysis & Prevention*, 213:107922, 2025.

[21] Christian P Janssen, Martin Baumann, and Antti Oulasvirta. Computational models of cognition for human-automated vehicle interaction: State-of-the-art and future directions. *International Journal of Human-Computer Studies*, 185, 2024.

[22] Jiming Xie, Yaqin Qin, Yan Zhang, Tianshun Chen, Bijun Wang, Qiyue Zhang, and Yulan Xia. Towards human-like automated vehicles: review and perspectives on behavioural decision making and intelligent motion planning. *Transportation Safety and Environment*, 7(1), 2025.

[23] Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. Explainable ai: A review of machine learning interpretability methods. *Entropy*, 23(1):18, 2020.

[24] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

[25] Reza Olfati-Saber, J Alex Fax, and Richard M Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[26] Xuan Di and Rongye Shi. A survey on autonomous vehicle control in the era of mixed-autonomy: From physics-based to ai-guided driving policy learning. *Transportation research part C: emerging technologies*, 125:103008, 2021.

[27] Pamul Yadav, Ashutosh Mishra, and Shiho Kim. A comprehensive survey on multi-agent reinforcement learning for connected and automated vehicles. *Sensors*, 23(10):4710, 2023.

[28] Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, 28(10):2222–2232, 2016.

[29] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[30] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-Feasible Trajectory Forecasting with Heterogeneous Data. In *Computer Vision –ECCV 2020*, pages 683–700, Cham, 2020.

[31] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social LSTM: Human Trajectory Prediction in Crowded Spaces. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 961–971, Las Vegas, NV, USA, June 2016.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[33] Francesco Giuliari, Irtiza Hasan, Marco Cristani, and Fabio Galasso. Transformer Networks for Trajectory Forecasting. In *International Conference on Pattern Recognition (ICPR)*, pages 10335–10342, January 2021.

[34] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M. Kitani. AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting. In *IEEE/CVF International Conference on Computer Vision*, pages 9813–9823, 2021.

[35] Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S. Refaat, and Benjamin Sapp. Wayformer: Motion Forecasting via Simple & Efficient Attention Networks. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2980–2987, May 2023.

[36] Weicheng Zhang, Hao Cheng, Fatema T Johora, and Monika Sester. Forceformer: Exploring social force and transformer for pedestrian trajectory prediction. In *2023 IEEE Intelligent Vehicles Symposium (IV)*, pages 1–7. IEEE, 2023.

[37] Christoph Schöller and Alois Knoll. FloMo: Tractable Motion Prediction with Normalizing Flows. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7977–7984, September 2021.

[38] Anna Mészáros, Julian F. Schumann, Javier Alonso-Mora, Arkady Zgonnikov, and Jens Kober. TrajFlow: Learning Distributions over Trajectories for Human Behavior Prediction. In *2024 IEEE Intelligent Vehicles Symposium (IV)*, Jeju, June 2024.

[39] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[40] Tianpei Gu, Guangyi Chen, Junlong Li, Chunze Lin, Yongming Rao, Jie Zhou, and Jiwen Lu. Stochastic Trajectory Prediction via Motion Indeterminacy Diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17113–17122, 2022.

[41] Weibo Mao, Chenxin Xu, Qi Zhu, Siheng Chen, and Yanfeng Wang. Leapfrog diffusion model for stochastic trajectory prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5517–5526, 2023.

[42] Inhwan Bae, Young-Jae Park, and Hae-Gon Jeon. Singulartrajectory: Universal trajectory predictor using diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17890–17901, 2024.

[43] Dorsa Sadigh, Shankar Sastry, Sanjit A. Seshia, and Anca D. Dragan. Planning for Autonomous Cars that Leverage Effects on Human Actions. In *Robotics: Science and Systems XII*, 2016.

[44] R. Duncan Luce. *Response times: Their role in inferring elementary mental organization.* Oxford University Press on Demand, 1986. Issue: 8.

[45] Roger Ratcliff and Hans PA Van Dongen. Diffusion model for one-choice reaction-time tasks and the cognitive effects of sleep deprivation. *Proceedings of the National Academy of Sciences*, 108(27):11285–11290, 2011.

[46] Arkady Zgonnikov, David Abbink, and Gustav Markkula. Should I Stay or Should I Go? Cognitive Modeling of Left-Turn Gap Acceptance Decisions in Human Drivers. *Human Factors*, page 00187208221144561, December 2024. Publisher: SAGE Publications Inc.

[47] Jami Pekkanen, Oscar Terence Giles, Yee Mun Lee, Ruth Madigan, Tatsuru Daimon, Natasha Merat, and Gustav Markkula. Variable-Drift Diffusion Models of Pedestrian Road-Crossing Decisions. *Computational Brain & Behavior*, 5(1):60–80, March 2022.

[48] Gustav Markkula, Yi-Shin Lin, Aravinda Ramakrishnan Srinivasan, Jac Billington, Matteo Leonetti, Amir Hossein Kalantari, Yue Yang, Yee Mun Lee, Ruth Madigan, and Natasha Merat. Explaining human interactions on the road by large-scale integration of computational psychological theory. *PNAS Nexus*, 2(6):pgad163, June 2023.

[49] Athanasios Theofilatos, Apostolos Ziakopoulos, Oscar Oviedo-Trespalacios, and Andrew Timmis. To cross or not to cross? Review and meta-analysis of pedestrian gap acceptance decisions at midblock street crossings. *Journal of Transport & Health*, 22:101108, September 2021.

[50] Dongfang Yang, Haolin Zhang, Ekim Yurtsever, Keith A. Redmill, and Umit Özgüner. Predicting Pedestrian Crossing Intention With Feature Fusion and Spatio-Temporal Attention. *IEEE Transactions on Intelligent Vehicles*, 7(2):221–230, June 2022. Conference Name: IEEE Transactions on Intelligent Vehicles.

[51] Dong-Fan Xie, Zhe-Zhe Fang, Bin Jia, and Zhengbing He. A data-driven lane-changing model based on deep learning. *Transportation Research Part C: Emerging Technologies*, 106:41–60, September 2019.

[52] Sajjad Mozaffari, Eduardo Arnold, Mehrdad Dianati, and Saber Fallah. Early Lane Change Prediction for Automated Driving Systems Using Multi-Task Attention-Based Convolutional Neural Networks. *IEEE Transactions on Intelligent Vehicles*, 7(3):758–770, September 2022. Conference Name: IEEE Transactions on Intelligent Vehicles.

[53] Malin Svärd, Gustav Markkula, Jonas Bärgman, and Trent Victor. Computational modeling of driver pre-crash brake response, with and without off-road glances: Parameterization using real-world crashes and near-crashes. *Accident Analysis & Prevention*, 163:106433, 2021.

[54] Giulio Bianchi Piccinini, Esko Lehtonen, Fabio Forcolin, Johan Engström, Deike Albers, Gustav Markkula, Johan Lodin, and Jesper Sandin. How Do Drivers Respond to Silent Automation Failures? Driving Simulator Study and Comparison of Computational Driver Braking Models. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 62(7):1212–1229, November 2020.

[55] Ran Wei, Anthony D. McDonald, Alfredo Garcia, and Hananeh Alambeigi. Modeling Driver Responses to Automation Failures With Active Inference. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):18064–18075, October 2022. Conference Name: IEEE Transactions on Intelligent Transportation Systems.

[56] Miao Xu, Zhi Liu, Bingyi Wang, and Shengyan Li. A survey of autonomous driving trajectory prediction: Methodologies, challenges, and future prospects. *Machines*, 13(9), 2025.

[57] Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, and Giovanni Pezzulo. Active Inference: A Process Theory. *Neural Computation*, 29(1):1–49, January 2017.

[58] Andrey Rudenko, Luigi Palmieri, Michael Herman, Kris M Kitani, Dariu M Gavrila, and Kai O Arras. Human motion trajectory prediction: a survey. *The International Journal of Robotics Research*, 39(8):895–935, July 2020.

[59] Jiaming Jiang, Kai Yan, Xindong Xia, and Biao Yang. A survey of deep learning-based pedestrian trajectory prediction: Challenges and solutions. *Sensors*, 25(3):957, 2025.

[60] Tingjing Wang, Daiquan Xiao, Xuecai Xu, and Quan Yuan. A survey on vehicle trajectory prediction procedures for intelligent driving. *Sensors*, 25(16), 2025.

[61] Boris Ivanovic and Marco Pavone. Injecting Planning-Awareness into Prediction and Detection Evaluation. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 821–828, June 2022.

[62] Alec Farid, Sushant Veer, Boris Ivanovic, Karen Leung, and Marco Pavone. Task-relevant failure detection for trajectory predictors in autonomous vehicles. In *Conference on Robot Learning*, pages 1959–1969. PMLR, 2023.

[63] Feifei Liu, Haozhe Wang, Zejun Wei, Qirong Lu, Yiyang Wen, Xiaoyu Tang, Jingyan Jiang, and Zhijian He. Beyond ade and fde: A comprehensive evaluation framework for safety-critical prediction in multi-agent autonomous driving scenarios. *IFAC-PapersOnLine*, 59(35):442–447, 2025.

[64] Erica Weng, Hana Hoshino, Deva Ramanan, and Kris Kitani. Joint metrics matter: A better standard for trajectory forecasting. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20258–20269. IEEE, 2023.

[65] Aravinda Ramakrishnan Srinivasan, Mohamed Hasan, Yi-Shin Lin, Matteo Leonetti, Jac Billington, Richard Romano, and Gustav Markkula. Comparing merging behaviors observed in naturalistic data with behaviors generated by a machine learned model. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pages 3787–3792, September 2021.

[66] Aravinda Ramakrishnan Srinivasan, Yi-Shin Lin, Morris Antonello, Anthony Knittel, Mohamed Hasan, Majd Hawasly, John Redford, Subramanian Ramamoorthy, Matteo Leonetti, Jac Billington, et al. Beyond rmse: Do machine-learned models of road user interaction produce human-like behavior? *IEEE Transactions on Intelligent Transportation Systems*, 24(7):7166–7177, 2023.

[67] Boris Ivanovic, Guanyu Song, Igor Gilitschenski, and Marco Pavone. trajdata: A Unified Interface to Multiple Human Trajectory Datasets. *Advances in Neural Information Processing Systems*, 36:27582–27593, December 2023.

[68] Saeed Rahmani, Zhenlin Xu, Simeon C Calvert, and Bart van Arem. Automated vehicles at unsignalized intersections: Safety and efficiency implications of mixed human and automated traffic. *Transportation Research Record*, page 03611981251370343, 2025.

[69] Changhe Chen, Mozhgan Pourkeshavarz, and Amir Rasouli. Criteria: a new benchmarking paradigm for evaluating trajectory prediction models for autonomous driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8265–8271. IEEE, 2024.

[70] Samyeul Noh. Probabilistic Collision Threat Assessment for Autonomous Driving at Road Intersections Inclusive of Vehicles in Violation of Traffic Rules. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4499–4506, October 2018.

[71] Qingzhao Zhang, Shengtuo Hu, Jiachen Sun, Qi Alfred Chen, and Z. Morley Mao. On Adversarial Robustness of Trajectory Prediction for Autonomous Vehicles. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15159–15168, 2022.

[72] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*, 2021.

[73] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R Qi, Yin Zhou, et al. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9710–9719, 2021.

[74] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Marco Kitani. AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting, August 2022.

[75] Roger Girgis, Florian Golemo, Felipe Codevilla, Martin Weiss, Jim Aldon D'Souza, Samira E. Kahou, Felix Heide, and Christopher Pal. LATENT VARIABLE SEQUENTIAL SET TRANSFORMERS FOR JOINT MULTI-AGENT MOTION PREDICTION. In *10th International Conference on Learning Representations, ICLR 2022*, 2022.

[76] Görkay Aydemir, Adil Kaan Akan, and Fatma Güney. ADAPT: Efficient Multi-Agent Trajectory Prediction with Adaptation. In *2023 IEEE/CVF International Conference on Computer Vision*, pages 8295–8305, 2023.

[77] Luke Rowe, Martin Ethier, Eli-Henry Dykhne, and Krzysztof Czarnecki. FJMP: Factorized Joint Multi-Agent Motion Prediction Over Learned Directed Acyclic Interaction Graphs. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13745–13755, 2023.

[78] Zikang Zhou, Jianping Wang, Yung-Hui Li, and Yu-Kai Huang. Query-centric trajectory prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17863–17873, 2023.

[79] Chen Feng, Hangning Zhou, Huadong Lin, Zhigang Zhang, Ziyao Xu, Chi Zhang, Boyu Zhou, and Shaojie Shen. Macformer: Map-agent coupled transformer for real-time and robust trajectory prediction. *IEEE Robotics and Automation Letters*, 8(10):6795–6802, 2023.

[80] Xishun Wang, Tong Su, Fang Da, and Xiaodong Yang. Prophnet: Efficient agent-centric motion forecasting with anchor-informed proposals. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21995–22003, 2023.

[81] Xing Gao, Xiaogang Jia, Yikang Li, and Hongkai Xiong. Dynamic scenario representation learning for motion forecasting with heterogeneous graph convolutional recurrent networks. *IEEE Robotics and Automation Letters*, 8(5):2946–2953, 2023.

[82] Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. MTR++: Multi-Agent Motion Prediction With Symmetric Scene Modeling and Guided Intention Querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3955–3971, May 2024.

[83] Mingkun Wang, Xiaoguang Ren, Ruochun Jin, Minglong Li, Xiaochuan Zhang, Changqian Yu, Mingxu Wang, and Wenjing Yang. Futurenet-lof: Joint trajectory prediction and lane occupancy field prediction with future context encoding. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8841–8848. IEEE, 2025.

[84] Jeroen Hagenus, Frederik Baymler Mathiesen, Julian F Schumann, and Arkady Zgonnikov. Robustness in trajectory prediction for autonomous vehicles: a survey. In *2024 IEEE Intelligent Vehicles Symposium (IV)*, pages 969–976. IEEE, 2024.

[85] David Fernández Llorca, Ronan Hamon, Henrik Junklewitz, Kathrin Grosse, Lars Kunze, Patrick Seiniger, Robert Swaim, Nick Reed, Alexandre Alahi, Emilia Gómez, et al. Testing autonomous vehicles and ai: perspectives and challenges from cybersecurity, transparency, robustness and fairness. *arXiv preprint arXiv:2403.14641*, 2024.

[86] Yulong Cao, Chaowei Xiao, Anima Anandkumar, Danfei Xu, and Marco Pavone. Advdo: Realistic adversarial attacks for trajectory prediction. In *European Conference on Computer Vision*, pages 36–52. Springer, 2022.

[87] Saeed Saadatnejad, Mohammadhossein Bahari, Pedram Khorsandi, Mohammad Saneian, Seyed-Mohsen Moosavi-Dezfooli, and Alexandre Alahi. Are socially-aware trajectory prediction models really socially-aware? In *Transportation Research Part C: Emerging Technologies*, volume 141, page 103705, August 2022.

[88] Yulong Cao, Danfei Xu, Xinshuo Weng, Zhuoqing Mao, Anima Anandkumar, Chaowei Xiao, and Marco Pavone. Robust Trajectory Prediction against Adversarial Attacks. In *Proceedings of The 6th Conference on Robot Learning*, pages 128–137. PMLR, March 2023. ISSN: 2640-3498.

[89] Gustav Markkula, Ruth Madigan, Dimitris Nathanael, Evangelia Portouli, Yee Mun Lee, Andre Dietrich, J Billington, Anna Schieben, and Natasha Merat. Defining interactions: a conceptual framework for understanding interactive behaviour in human and automated road traffic. *Theoretical Issues in Ergonomics Science*, 2020.

[90] Boris Ivanovic, James Harrison, and Marco Pavone. Expanding the deployment envelope of behavior prediction via adaptive meta-learning. *arXiv preprint arXiv:2209.11820*, 2022.

[91] Frederik Diehl, Thomas Brunner, Michael Truong Le, and Alois Knoll. Graph Neural Networks for Modelling Traffic Participant Interaction. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 695–701, June 2019.

[92] Sarvesh Kolekar, Joost de Winter, and David Abbink. Human-like driving behaviour emerges from a risk-based driver model. *Nature Communications*, 11(1):4850, December 2020.

[93] Xin Huang, Guy Rosman, Igor Gilitschenski, Ashkan Jasour, Stephen G. McGill, John J. Leonard, and Brian C. Williams. HYPER: Learned Hybrid Trajectory Prediction via Factored Inference and Adaptive Sampling. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2906–2912, May 2022.

[94] Alexander Cui, Sergio Casas, Abbas Sadat, Renjie Liao, and Raquel Urtasun. LookOut: Diverse Multi-Future Prediction and Planning for Self-Driving. In *IEEE/CVF International Conference on Computer Vision*, pages 16107–16116, 2021.

[95] Rohan Chandra, Aniket Bera, and Dinesh Manocha. Using Graph-Theoretic Machine Learning to Predict Human Driver Behavior. *IEEE Transactions on Intelligent Transportation Systems*, 23(3):2572–2585, March 2022.

[96] L. Sun, X. Jia, and A. Dragan. On Complementing End-To-End Human Behavior Predictors with Planning. *Robotics science and systems*, January 2021.

[97] Zhangjie Cao, Erdem Biyik, Guy Rosman, and Dorsa Sadigh. Leveraging Smooth Attention Prior for Multi-Agent Trajectory Prediction. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 10723–10730, May 2022.

[98] A. Bighashdel, P. Meletis, and G. Dubbelman. Towards Equilibrium-based Interaction Modeling for Pedestrian Path Prediction. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–8, September 2020.

[99] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. SoPhie: An Attentive GAN for Predicting Paths Compliant to Social and Physical Constraints. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1349–1358, Long Beach, CA, USA, June 2019.

[100] Parth Kothari, Brian Sifringer, and Alexandre Alahi. Interpretable Social Anchors for Human Trajectory Forecasting in Crowds. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15551–15561, June 2021.

[101] Fanta Camara, Nicola Bellotto, Serhan Cosar, Florian Weber, Dimitris Nathanael, Matthias Althoff, Jingyuan Wu, Johannes Ruenz, André Dietrich, Gustav Markkula,

Anna Schieben, Fabio Tango, Natasha Merat, and Charles Fox. Pedestrian Models for Autonomous Driving Part II: High-Level Models of Human Behavior. *IEEE Transactions on Intelligent Transportation Systems*, 22(9):5453–5472, September 2021.

[102] J. Yue, D. Manocha, and H. Wang. Human Trajectory Prediction via Neural Social Physics. In *Lecture Notes in Computer Science*, Tel Aviv, July 2022.

[103] Suresh Kumaar Jayaraman, Lionel P. Robert, X. Jessie Yang, and Dawn M. Tilbury. Multimodal Hybrid Pedestrian: A Hybrid Automaton Model of Urban Pedestrian Behavior for Automated Driving Applications. *IEEE Access*, 9:27708–27722, 2021.

[104] Somayeh Mafi, Yassir Abdelrazig, and Ryan Doczy. Analysis of Gap Acceptance Behavior for Unprotected Right and Left Turning Maneuvers at Signalized Intersections using Data Mining Methods: A Driving Simulation Approach. *Transportation Research Record*, 2672(38):160–170, December 2018.

[105] A. Abhishek, Marko A. A. Boon, and Michel Mandjes. Generalized gap acceptance models for unsignalized intersections. *Mathematical Methods of Operations Research*, 89(3):385–409, June 2019.

[106] Mahmoud Arafat, Mohammed Hadi, Thodsapon Hunsanon, and Kamar Amine. Stop Sign Gap Assist Application in a Connected Vehicle Simulation Environment. *Transportation Research Record*, 2675(9):1127–1135, September 2021.

[107] Yeping Hu, Xiaogang Jia, Masayoshi Tomizuka, and Wei Zhan. Causal-based Time Series Domain Generalization for Vehicle Intention Prediction. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 7806–7813, May 2022.

[108] E. Balal and R. L. Cheu. Comparative Evaluation of Fuzzy Inference System, Support Vector Machine and Multilayer Feed-Forward Neural Network in Making Discretionary Lane Changing Decisions. *Neural Network World*, 28(4):361–378, 2018.

[109] Anik Das, Md Nasim Khan, and Mohamed M. Ahmed. Nonparametric Multivariate Adaptive Regression Splines Models for Investigating Lane-Changing Gap Acceptance Behavior Utilizing Strategic Highway Research Program 2 Naturalistic Driving Data. *Transportation Research Record*, 2674(5):223–238, May 2020.

[110] Basma Khelfa, Ibrahima Ba, and Antoine Tordeux. Predicting highway lane-changing maneuvers: A benchmark analysis of machine and ensemble learning algorithms. *Physica A: Statistical Mechanics and its Applications*, 612:128471, February 2023.

[111] B Raghuram Kadali, P. Vedagiri, and Nivedan Rathi. Models for pedestrian gap acceptance behaviour analysis at unprotected mid-block crosswalks under mixed traffic conditions. *Transportation Research Part F: Traffic Psychology and Behaviour*, 32:114–126, July 2015.

[112] Ruoyu Yao, Weiliang Zeng, Yihao Chen, and Zhaoshui He. A deep learning framework for modelling left-turning vehicle behaviour considering diagonal-crossing

motorcycle conflicts at mixed-flow intersections. *Transportation Research Part C: Emerging Technologies*, 132:103415, November 2021.

[113] Rakshita Nagalla, Prasanna Pothuganti, and Digvijay S. Pawar. Analyzing Gap Acceptance Behavior at Unsignalized Intersections Using Support Vector Machines, Decision Tree and Random Forests. *Procedia Computer Science*, 109:474–481, January 2017.

[114] Minming Yang, Xuesong Wang, and Mohammed Quddus. Examining lane change gap acceptance, duration and impact using naturalistic driving data. *Transportation Research Part C: Emerging Technologies*, 104:317–331, July 2019.

[115] Ewoud A. I. Pool, Julian F. P. Kooij, and Dariu M. Gavrila. Crafted vs Learned Representations in Predictive Models—A Case Study on Cyclist Path Prediction. *IEEE Transactions on Intelligent Vehicles*, 6(4):747–759, December 2021.

[116] Henning Müller, Wolfgang Müller, Stéphane Marchand-Maillet, Thierry Pun, and David McG. Squire. A Framework for Benchmarking in CBIR. *Multimedia Tools and Applications*, 21(1):55–73, September 2003.

[117] Mubariz Zaffar, Sourav Garg, Michael Milford, Julian Kooij, David Flynn, Klaus McDonald-Maier, and Shoaib Ehsan. VPR-Bench: An Open-Source Visual Place Recognition Evaluation Framework with Quantifiable Viewpoint and Appearance Change. *International Journal of Computer Vision*, 129(7):2136–2174, July 2021.

[118] Robert Krajewski, Julian Bock, Laurent Kloeker, and Lutz Eckstein. The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pages 2118–2125, November 2018.

[119] Robert Krajewski, Tobias Moers, Julian Bock, Lennart Vater, and Lutz Eckstein. The rounD Dataset: A Drone Dataset of Road User Trajectories at Roundabouts in Germany. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, September 2020.

[120] E. Barnard and L.F.A. Wessels. Extrapolation and interpolation in neural network classifiers. *IEEE Control Systems Magazine*, 12(5):50–53, October 1992.

[121] Jin Huang and C.X. Ling. Using AUC and accuracy in evaluating learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 17(3):299–310, March 2005.

[122] Gary Marcus. The Next Decade in AI: Four Steps Towards Robust Artificial Intelligence, February 2020. arXiv: 2002.06177[cs].

[123] Michael van Bekkum, Maaike de Boer, Frank van Harmelen, André Meyer-Vitali, and Annette ten Teije. Modular design patterns for hybrid learning and reasoning systems. *Applied Intelligence*, 51(9):6528–6546, September 2021.

[124] Julian F Schumann, Anna Mészáros, Jens Kober, and Arkady Zgonnikov. Step: Structured training and evaluation platform for benchmarking trajectory prediction models, 2025.

[125] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020.

[126] Parth Kothari, Sven Kreiss, and Alexandre Alahi. Human trajectory forecasting in crowds: A deep learning perspective. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):7386–7400, 2021.

[127] Andrey Rudenko, Luigi Palmieri, Wanting Huang, Achim J Lilienthal, and Kai O Arras. The atlas benchmark: An automated evaluation framework for human motion prediction. In *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 636–643. IEEE, 2022.

[128] Lan Feng, Mohammadhossein Bahari, Kaouther Messaoud Ben Amor, Éloi Zablocki, Matthieu Cord, and Alexandre Alahi. Unitraj: A unified framework for scalable vehicle trajectory prediction. In *European Conference on Computer Vision*, pages 106–123. Springer, 2024.

[129] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-Feasible Trajectory Forecasting With Heterogeneous Data, August 2022.

[130] Song Wen, Hao Wang, and Dimitris Metaxas. Social ode: Multi-agent trajectory forecasting with neural ordinary differential equations. In *European Conference on Computer Vision*, pages 217–233. Springer, 2022.

[131] Sajjad Mozaffari, MReza Alipour Sormoli, Konstantinos Koufos, Graham Lee, and Mehrdad Dianati. Trajectory prediction with observations of variable-length for motion planning in highway merging scenarios. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, pages 5633–5640. IEEE, 2023.

[132] Julian F. Schumann, Aravinda R. Srinivasan, Jens Kober, Gustav Markkula, and Arkady Zgonnikov. Using Models Based on Cognitive Theory to Predict Human Behavior in Traffic: A Case Study. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, pages 5870–5875, September 2023. ISSN: 2153-0017.

[133] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2255–2264, Salt Lake City, UT, June 2018. IEEE.

[134] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*, 2009.

[135] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by Example. *Computer Graphics Forum*, 26(3):655–664, September 2007.

[136] Alessio Monti, Angelo Porrello, Simone Calderara, Pasquale Coscia, Lamberto Ballan, and Rita Cucchiara. How many observations are enough? knowledge distillation for trajectory forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6553–6562, 2022.

[137] Yi Xu and Yun Fu. Adapting to length shift: Flexilength network for trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15226–15237, 2024.

[138] Kaiyuan Tan, Jun Wang, and Yiannis Kantaros. Targeted adversarial attacks against neural network trajectory predictors. In *Learning for Dynamics and Control Conference*, pages 431–444. PMLR, 2023.

[139] Yingkai Dong, Li Wang, Zheng Li, Hao Li, Peng Tang, Chengyu Hu, and Shanqing Guo. Safe driving adversarial trajectory can mislead: Toward more stealthy adversarial attack against autonomous driving prediction module. *ACM Transactions on Privacy and Security*, 28(2):1–28, 2025.

[140] Julian F. Schumann, Jeroen Hagenus, Frederik Baymler Mathiesen, and Arkady Zgonnikov. Realistic Adversarial Attacks for Robustness Evaluation of Trajectory Prediction Models via Future State Perturbation. *ACM Journal on Autonomous Transportation Systems*, 2026.

[141] John Houston, Guido Zuidhof, Luca Bergamini, Yawei Ye, Long Chen, Ashesh Jain, Sammy Omari, Vladimir Iglovikov, and Peter Ondruska. One thousand and one hours: Self-driving motion prediction dataset. In *Conference on Robot Learning*, pages 409–418. PMLR, 2021.

[142] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A Multimodal Dataset for Autonomous Driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11621–11631, 2020.

[143] Wei Zhan, Liting Sun, Di Wang, Haojie Shi, Aubrey Clausse, Maximilian Naumann, Julius Kümmerle, Hendrik Königshof, Christoph Stiller, Arnaud de La Fortelle, and Masayoshi Tomizuka. INTERACTION Dataset: An INTERnational, Adversarial and Cooperative moTION Dataset in Interactive Driving Scenarios with Semantic Maps. *arXiv:1910.03088 [cs, eess]*, sep 2019.

[144] Julian Bock, Robert Krajewski, Tobias Moers, Steffen Runde, Lennart Vater, and Lutz Eckstein. The ind dataset: A drone dataset of naturalistic road user trajectories at german intersections. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1929–1934. IEEE, 2020.

[145] Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Motion Transformer with Global Intention Localization and Local Movement Refinement. *Advances in Neural Information Processing Systems*, 35:6531–6543, December 2022.

[146] A Mészáros, JF Schumann, J Alonso-Mora, A Zgonnikov, and J Kober. Rome: Robust multi-modal density estimator. In *33rd International Joint Conference on Artificial Intelligence*, pages 4751–4759, 2024.

[147] Luca Anthony Thiede and Pratik Prabhanjan Brahma. Analyzing the variety loss in the context of probabilistic trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9954–9963, 2019.

[148] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[149] Ruoqi Wei, Cesar Garcia, Ahmed El-Sayed, Viyaleta Peterson, and Ausif Mahmood. Variations in variational autoencoders-a comparative evaluation. *IEEE Access*, 8:153651–153670, 2020.

[150] Ivan Kobyzev, Simon JD Prince, and Marcus A Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3964–3979, 2020.

[151] Donna Xu, Yaxin Shi, Ivor W Tsang, Yew-Soon Ong, Chen Gong, and Xiaobo Shen. Survey on multi-output learning. *IEEE Transactions on Neural Networks and Learning Systems*, 31(7):2409–2429, 2019.

[152] Amir Rasouli. Deep learning for vision-based prediction: A survey. *arXiv preprint arXiv:2007.00095*, 2020.

[153] Marc Peter Deisenroth, A. Aldo Faisal, and Cheng Soon Ong. *Mathematics for Machine Learning*. Cambridge University Press, April 2020. Google-Books-ID: pFjPDwAAQBAJ.

[154] Don O Loftsgaarden and Charles P Quesenberry. A nonparametric estimate of a multivariate density function. *The Annals of Mathematical Statistics*, 36(3):1049–1051, 1965.

[155] Pascal Vincent and Yoshua Bengio. Manifold parzen windows. *Advances in Neural Information Processing Systems*, 15, 2002.

[156] Thomas Nagler and Claudia Czado. Evading the curse of dimensionality in nonparametric density estimation with simplified vine copulas. *Journal of Multivariate Analysis*, 151:69–89, 2016.

[157] Junwei Liang, Lu Jiang, Kevin Murphy, Ting Yu, and Alexander Hauptmann. The garden of forking paths: Towards multi-future trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10508–10518, 2020.

[158] Emanuel Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

[159] Bernard W Silverman. *Density estimation for statistics and data analysis*. Routledge, 1998.

[160] Xiaoxia Wang, Peter Tino, Mark A Fardal, Somak Raychaudhury, and Arif Babul. Fast parzen window density estimator. In *2009 International Joint Conference on Neural Networks*, pages 3267–3274. IEEE, 2009.

[161] Jia-Xing Gao, Da-Quan Jiang, and Min-Ping Qian. Adaptive manifold density estimation. *Journal of Statistical Computation and Simulation*, 92(11):2317–2331, 2022.

[162] Geoffrey J McLachlan and Suren Rathnayake. On the number of components in a gaussian mixture model. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 4(5):341–355, 2014.

[163] Harry Joe. *Dependence modeling with copulas*. CRC press, 2014.

[164] Yves I Ngounou Bakam and Denys Pommeret. Nonparametric estimation of copulas and copula densities by orthogonal projections. *Econometrics and Statistics*, 2023.

[165] Håkon Otneim and Dag Tjøstheim. The locally gaussian density estimator for multivariate data. *Statistics and Computing*, 27:1595–1616, 2017.

[166] Mihael Ankerst, Markus M Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics: Ordering points to identify the clustering structure. *ACM Sigmod Record*, 28(2):49–60, 1999.

[167] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, volume 96, pages 226–231, 1996.

[168] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[169] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3):37–52, 1987.

[170] Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

[171] Cédric Villani. *Optimal transport: Old and new*, volume 338. Springer, 2009.

[172] Tarald O Kvalseth. Generalized divergence and gibbs' inequality. In *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, volume 2, pages 1797–1801. IEEE, 1997.

[173] Davoud Moulavi, Pablo A Jaskowiak, Ricardo JGB Campello, Arthur Zimek, and Jörg Sander. Density-based clustering validation. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 839–847. SIAM, 2014.

[174] Nils-Bastian Heidenreich, Anja Schindler, and Stefan Sperlich. Bandwidth selection for kernel density estimation: a review of fully automatic selectors. *AStA Advances in Statistical Analysis*, 97:403–433, 2013.

[175] Ruochen Jiao, Xiangguo Liu, Takami Sato, Qi Alfred Chen, and Qi Zhu. Semi-supervised Semantics-guided Adversarial Training for Robust Trajectory Prediction. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8173–8183, October 2023. ISSN: 2380-7504.

[176] Andrea Tocchetti, Lorenzo Corti, Agathe Balayn, Mireia Yurrita, Philip Lippmann, Marco Brambilla, and Jie Yang. Ai robustness: a human-centered perspective on technological challenges and opportunities. *ACM Computing Surveys*, 57(6):1–38, 2025.

[177] Huilin Yin, Jiaxiang Li, Pengju Zhen, and Jun Yan. Sa-attack: Speed-adaptive stealthy adversarial attack on trajectory prediction. In *2024 IEEE Intelligent Vehicles Symposium (IV)*, pages 1772–1778. IEEE, 2024.

[178] Jiping Fan, Zhenpo Wang, and Guoqiang Li. Adversarial attack on trajectory prediction for autonomous vehicles with generative adversarial networks. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1026–1031. IEEE, 2024.

[179] Jingkang Wang, Ava Pun, James Tu, Sivabalan Manivasagam, Abbas Sadat, Sergio Casas, Mengye Ren, and Raquel Urtasun. AdvSim: Generating Safety-Critical Scenarios for Self-Driving Vehicles. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9904–9913, Nashville, TN, USA, June 2021. IEEE.

[180] Davis Rempe, Jonah Philion, Leonidas J Guibas, Sanja Fidler, and Or Litany. Generating useful accident-prone driving scenarios via a learned traffic prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17305–17315, 2022.

[181] Mozhgan Pourkeshavarz, Mohammad Sabokrou, and Amir Rasouli. Adversarial backdoor attack by naturalistic data poisoning on trajectory prediction in autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14885–14894, 2024.

[182] Yunxiao Qin, Yuanhao Xiong, Jinfeng Yi, and Cho-Jui Hsieh. Training meta-surrogate model for transferable adversarial attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

[183] Ang Duan, Ruyan Wang, Yaping Cui, Peng He, and Luo Chen. Causal Robust Trajectory Prediction Against Adversarial Attacks for Autonomous Vehicles. *IEEE Internet of Things Journal*, pages 1–1, 2023.

[184] R. Craig Coulter. Implementation of the pure pursuit path tracking algorithm. Technical Report CMU-RI-TR-92-01, Carnegie-Mellon University Pittsburgh PA Robotics Institute, 1992.

[185] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

[186] Abraham Savitzky and Marcel JE Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8):1627–1639, 1964.

[187] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International conference on machine learning*, pages 2137–2146. PMLR, 2018.

[188] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pages 1310–1320. PMLR, 2019.

[189] Jiping Fan, Zhenpo Wang, and Guoqiang Li. A novel unsupervised anomaly detection method on adversarial attacks for autonomous vehicles trajectory prediction. In *2024 IEEE 22nd International Conference on Industrial Informatics (INDIN)*, pages 1–18. IEEE, 2024.

[190] Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.

[191] Kai Tian, Gustav Markkula, Chongfeng Wei, Yee Mun Lee, Ruth Madigan, Natasha Merat, and Richard Romano. Explaining unsafe pedestrian road crossing behaviours using a psychophysics-based gap acceptance model. *Safety science*, 154:105837, 2022.

[192] Malin Svärd, Gustav Markkula, Johan Engström, Fredrik Granum, and Jonas Bärgman. A quantitative driver model of pre-crash brake onset and control. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 61, pages 339–343. SAGE Publications, September 2017.

[193] Johan Engström, Gustav Markkula, Qingwan Xue, and Natasha Merat. Simulating the effect of cognitive load on braking responses in lead vehicle braking scenarios. *IET Intelligent Transport Systems*, 12(6):427–433, 2018. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1049/iet-its.2017.0233.

[194] Alexandra Fries, Felix Fahrenkrog, Katharina Donauer, Marcus Mai, and Florian Raisch. Driver behavior model for the safety assessment of automated driving. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, pages 1669–1674. IEEE, 2022.

[195] Alexandra Fries, Ludwig Lemberg, Felix Fahrenkrog, Marcus Mai, and Arun Das. Modeling driver behavior in critical traffic scenarios for the safety assessment of automated driving. *Traffic Injury Prevention*, 24(sup1):S105–S110, April 2023.

[196] Lucas Fonseca Alexandre de Oliveira, Lars Schories, Lukas Brostek, and Martin Meywerk. Simulation-Based Evaluation of a Generic Autonomous Emergency Braking System Using a Cognitive Pedestrian Behavior Model. In *27th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, 2023. Number: 23-0217.

[197] Christian Rössert, Johannes Drever, and Lukas Brostek. Cognitive behavior model replicates road user response timing in naturalistic rear-end traffic conflicts, November 2024.

[198] Ran Wei, Alfredo Garcia, Anthony McDonald, Gustav Markkula, Johan Engstrom, and Matthew O' Kelly. Learning an active inference model of driver perception and control: Application to vehicle car-following. *IEEE Transactions on Intelligent Transportation Systems*, 2025.

[199] Johan Engström, Ran Wei, Anthony D. McDonald, Alfredo Garcia, Matthew O'Kelly, and Leif Johnson. Resolving uncertainty on the fly: modeling adaptive driving behavior as active inference. *Frontiers in Neurorobotics*, 18, March 2024. Publisher: Frontiers.

[200] Pawel J. Matusz, Suzanne Dikker, Alexander G. Huth, and Catherine Perrodin. Are We Ready for Real-world Neuroscience? *Journal of Cognitive Neuroscience*, 31(3):327–338, March 2019.

[201] Simone G. Shamay-Tsoory and Avi Mendelsohn. Real-Life Neuroscience: An Ecological Approach to Brain and Behavior Research. *Perspectives on Psychological Science*, 14(5):841–859, September 2019. Publisher: SAGE Publications Inc.

[202] Wilka Carvalho and Andrew Lampinen. Naturalistic Computational Cognitive Science: Towards generalizable models and theories that capture the full range of natural behavior, February 2025. arXiv:2502.20349 [q-bio].

[203] Oh-Sang Kwon, Duje Tadin, and David C. Knill. Unifying account of visual motion and position perception. *Proceedings of the National Academy of Sciences of the United States of America*, 112(26):8142–8147, June 2015.

[204] Wenxiang Chen, Xiangling Zhuang, Zixin Cui, and Guojie Ma. Drivers ' recognition of pedestrian road-crossing intentions: Performance and process. *Transportation Research Part F: Traffic Psychology and Behaviour*, 64:552–564, July 2019.

[205] Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization*, 13(4):455–492, December 1998.

[206] Aravinda Ramakrishnan Srinivasan, Julian Schumann, Yueyang Wang, Yi-Shin Lin, Michael Daly, Albert Solernou, Arkady Zgonnikov, Matteo Leonetti, Jac Billington, and Gustav Markkula. The COMMOTIONS Urban Interactions Driving Simulator Study Dataset, May 2023. arXiv:2305.11909 [cs].

[207] Julian F. Schumann, Johan Engström, Leif Johnson, Matthew O'Kelly, Joao Messias, Jens Kober, and Arkady Zgonnikov. Active inference as a model of collision avoidance behavior in human drivers. *Nature Communication*, 17, June 2026.

[208] Jonas Bärgman, Vera Lisovskaja, Trent Victor, Carol Flannagan, and Marco Dozza. How does glance behavior influence crash and injury risk? A 'what-if' counterfactual simulation using crashes and near-crashes from SHRP2. *Transportation Research Part F: Traffic Psychology and Behaviour*, 35:152–169, November 2015.

[209] Johan Engström, Shu-Yuan Liu, Azadeh Dinparastdjadid, and Camelia Simoiu. Modeling road user response timing in naturalistic traffic conflicts: A surprise-based framework. *Accident Analysis & Prevention*, 198:107460, April 2024.

[210] Qingwan Xue, Gustav Markkula, Xuedong Yan, and Natasha Merat. Using perceptual cues for brake response to a lead vehicle: Comparing threshold and accumulator models of visual looming. *Accident Analysis & Prevention*, 118:114–124, September 2018.

[211] Chen Guo, Xiaolan Wang, Lili Su, and Yansong Wang. Safety distance model for longitudinal collision avoidance of logistics vehicles considering slope and road adhesion coefficient. *Proceedings of the Institution of Mechanical Engineers*, 2021.

[212] Olger Siebinga, Arkady Zgonnikov, and David A Abbink. A model of dyadic merging interactions explains human drivers' behavior from control inputs to decisions. *PNAS Nexus*, 3(10):pgae420, October 2024.

[213] Gustav Markkula, Johan Engström, Johan Lodin, Jonas Bärgman, and Trent Victor. A farewell to brake reaction times? Kinematics-dependent brake response in naturalistic rear-end emergencies. *Accident Analysis & Prevention*, 95:209–226, October 2016.

[214] Tianyou Li, Jordanka Kovaceva, and Marco Dozza. Modeling collision avoidance maneuvers for micromobility vehicles. *Journal of Safety Research*, 87:232–243, December 2023.

[215] Simon Suo, Sebastian Regalado, Sergio Casas, and Raquel Urtasun. Trafficsim: Learning to simulate realistic multi-agent behaviors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10400–10409, 2021.

[216] Maximilian Igl, Daewoo Kim, Alex Kuefler, Paul Mougin, Punit Shah, Kyriacos Shiarlis, Dragomir Anguelov, Mark Palatucci, Brandyn White, and Shimon Whiteson. Symphony: Learning realistic and diverse agents for autonomous driving simulation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2445–2451. IEEE, 2022.

[217] Lancelot Da Costa, Thomas Parr, Noor Sajid, Sebastijan Veselic, Victorita Neacsu, and Karl Friston. Active inference on discrete state-spaces: a synthesis. *Journal of Mathematical Psychology*, 99:102447, December 2020. arXiv:2001.07203 [q-bio].

[218] Thomas Parr, Giovanni Pezzulo, and Karl J. Friston. *Active Inference: The Free Energy Principle in Mind, Brain, and Behavior*. MIT Press, March 2022. Google-Books-ID: UrZNEAAAQBAJ.

[219] Ryan Smith, Paul Badcock, and Karl J. Friston. Recent advances in the application of predictive coding and active inference models within clinical neuroscience. *Psychiatry and Clinical Neurosciences*, 75(1):3–13, 2021. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/pcn.13138.

[220] Jared Vasil, Paul B. Badcock, Axel Constant, Karl Friston, and Maxwell J. D. Ramstead. A World Unto Itself: Human Communication as Active Inference. *Frontiers in Psychology*, 11, March 2020. Publisher: Frontiers.

[221] Hugo Bottemanne and Karl J. Friston. An active inference account of protective behaviours during the COVID-19 pandemic. *Cognitive, Affective, & Behavioral Neuroscience*, 21(6):1117–1129, December 2021.

[222] David J. Harris, Tom Arthur, David P. Broadbent, Mark R. Wilson, Samuel J. Vine, and Oliver R. Runswick. An Active Inference Account of Skilled Anticipation in Sport: Using Computational Models to Formalise Theory and Generate New Hypotheses. *Sports Medicine*, 52(9):2023–2038, September 2022.

[223] Filip Novický, Ajith Anil Meera, Fleur Zeldenrust, and Pablo Lanillos. Precision not prediction: Body-ownership illusion as a consequence of online precision adaptation under Bayesian inference, September 2024.

[224] Johan Engström, Jonas Bärgman, Daniel Nilsson, Bobbie Seppelt, Gustav Markkula, Giulio Bianchi Piccinini, and Trent Victor. Great expectations: a predictive processing account of automobile driving. *Theoretical Issues in Ergonomics Science*, 19(2):156–194, March 2018. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/1463922X.2017.1306148.

[225] Ran Wei, Alfredo Garcia, Anthony McDonald, Gustav Markkula, Johan Engström, Isaac Supeene, and Matthew O' Kelly. World Model Learning from Demonstrations with Active Inference: Application to Driving Behavior. In Christopher L. Buckley, Daniela Cialfi, Pablo Lanillos, Maxwell Ramstead, Noor Sajid, Hideaki Shimazaki, and Tim Verbelen, editors, *Active Inference*, Communications in Computer and Information Science, pages 130–142, Cham, 2023. Springer Nature Switzerland.

[226] D. N. Lee. A theory of visual control of braking based on information about time-to-collision. *Perception*, 5(4):437–459, 1976.

[227] James J. Gibson. *The Ecological Approach to Visual Perception: Classic Edition*. Psychology Press, New York, December 2014.

[228] Joshua I. Gold and Michael N. Shadlen. The Neural Basis of Decision Making. *Annual Review of Neuroscience*, 30(Volume 30, 2007):535–574, July 2007. Publisher: Annual Reviews.

[229] Gustav Markkula. Modeling driver control behavior in both routine and near-accident driving. In *Proceedings of the human factors and ergonomics society annual meeting*, volume 58, pages 879–883. SAGE Publications Sage CA: Los Angeles, CA, 2014.

[230] Johannes Fischer and Omer Sahin Tas. Information Particle Filter Tree: An Online Algorithm for POMDPs with Belief-Based Rewards on Continuous Domains. In *Proceedings of the 37th International Conference on Machine Learning*, pages 3177–3187. PMLR, November 2020. ISSN: 2640-3498.

[231] Jami Pekkanen, Otto Lappi, Paavo Rinkkala, Samuel Tuhkanen, Roosa Frantsi, and Heikki Summala. A computational model for driver's cognitive state, visual perception and intermittent attention in a distracted car following task. *Royal Society Open Science*, 5(9):180194, September 2018. Publisher: Royal Society.

[232] Errol R Hoffmann. Estimation of Time to Vehicle Arrival—Effects of Age on Use of Available Visual Information. *Perception*, 23(8):947–955, August 1994. Publisher: SAGE Publications Ltd STM.

[233] Dave Lamble, Matti Laakso, and Heikki Summala. Detection thresholds in car following situations and peripheral vision: implications for positioning of visually demanding in-car displays. *Ergonomics*, 42(6):807–815, June 1999. Publisher: Taylor & Francis.

[234] Philip Polack, Florent Altché, Brigitte d'Andréa Novel, and Arnaud de La Fortelle. The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles? In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 812–818, June 2017.

[235] Karl Friston, Lancelot Da Costa, Danijar Hafner, Casper Hesp, and Thomas Parr. Sophisticated inference. *Neural Computation*, 33(3):713–763, 2021.

[236] Hélène Laurent, Marc Sangnier, and Carole Treibich. Traffic safety and norms of compliance with rules: An exploratory study. *Economics Bulletin*, 41(4):2464–2483, 2021.

[237] Sheer Abbas, Sidra Fatima, Azhar Sharif, and Shiekh Muhammad Adnan. Drivers' knowledge, attitude and practices towards traffic rules and regulations. *Journal of Road Safety*, 35(3):24–31, 2024.

[238] Pieter-Tjerk de Boer, Dirk P. Kroese, Shie Mannor, and Reuven Y. Rubinstein. A Tutorial on the Cross-Entropy Method. *Annals of Operations Research*, 134(1):19–67, February 2005.

[239] Hendrik Weber, Lutz Eckstein, Felix Beringhoff, and Johanna Josten. Towards Modeling Driver Performance Within Crash-Relevant Scenarios As Virtual Reference for the Safety of Automated Vehicles. Technical report, Transportation Research Board, 2023. Number: 23-0126.

[240] Roger Ratcliff, Philip L Smith, Scott D Brown, and Gail McKoon. Diffusion decision model: Current issues and history. *Trends in cognitive sciences*, 20(4):260–281, 2016.

[241] Baris Metin, Herbert Roeyers, Jan R Wiersema, Jaap J van der Meere, Margaret Thompson, and Edmund Sonuga-Barke. Adhd performance reflects inefficient but not impulsive information processing: a diffusion model analysis. *Neuropsychology*, 27(2):193, 2013.

[242] Alice Reinhartz, Tilo Strobach, Thomas Jacobsen, and Claudia C von Bastian. Mechanisms of training-related change in processing speed: a drift-diffusion model approach. *Journal of Cognition*, 6(1):46, 2023.

[243] Azadeh Dinparastdjadid, Isaac Supeene, and Johan Engstrom. Measuring Surprise in the Wild, May 2023. arXiv:2305.07733 [cs].

[244] Leif Johnson, Johan Engström, Aravinda Srinivasan, Ibrahim Özturk, and Gustav Markkula. Looking for an out: Affordances, uncertainty and collision avoidance behavior of human drivers, 2025.

[245] J. Engstroem. Scenario criticality determines the effect of working memory load on brake response time. In *Proceedings of the European Conference on Human Centred Design for Intelligent Transport Systems*, pages 25–36, April 2010.

[246] Karel A. Brookhuis, Gerbrand de Vries, and Dick de Waard. The effects of mobile telephoning on driving performance. *Accident Analysis & Prevention*, 23(4):309–316, August 1991.

[247] J. D. Lee, B. Caven, S. Haake, and T. L. Brown. Speech-based interaction with in-vehicle computers: the effect of speech-based e-mail on drivers' attention to the roadway. *Human Factors*, 43(4):631–640, 2001.

[248] Trent Victor, Marco Dozza, Jonas Bärgman, Christian-Nils Boda, Johan Engström, Carol Flannagan, John D Lee, and Gustav Markkula. Analysis of naturalistic driving study data: Safer glances, driver inattention, and crash risk. Technical report, Transportation Research Board, 2015.

[249] Johan Engström, Julia Werneke, Jonas Bärgman, Noel Nguyen, and Bryon Cook. Analysis of the role of inattention in road crashes based on naturalistic on-board safety monitoring data. In *3rd International Conference on Driver Distraction and Inattention*, 2013.

[250] Erika Ziraldo, Shady Attalla, Sam Kodsi, and Michele Oliver. Driver Response to Right Turning Path Intrusions at Signalized Intersections. *SAE International Journal of Advances and Current Practices in Mobility*, 2(3):1623–1632, April 2020. Number: 2020-01-0884.

[251] Herbert A Simon. A behavioral model of rational choice. *The quarterly journal of economics*, pages 99–118, 1955.

[252] Heikki Summala. Towards Understanding Motivational and Emotional Factors in Driver Behaviour: Comfort Through Satisficing. In P. Carlo Cacciabue, editor, *Modelling Driver Behaviour in Automotive Environments: Critical Issues in Driver Interactions with Intelligent Transport Systems*, pages 189–207. Springer, London, 2007.

[253] Hanna Oh, Jeffrey M. Beck, Pingping Zhu, Marc A. Sommer, Silvia Ferrari, and Tobias Egner. Satisficing in split-second decision making is characterized by strategic cue discounting. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 42(12):1937–1956, 2016. Place: US Publisher: American Psychological Association.

[254] Frederick Callaway, Bas van Opheusden, Sayan Gul, Priyam Das, Paul M. Krueger, Thomas L. Griffiths, and Falk Lieder. Rational use of cognitive resources in human planning. *Nature Human Behaviour*, 6(8):1112–1125, August 2022.

[255] Mattias Brännström, Erik Coelingh, and Jonas Sjöberg. Decision-making on when to brake and when to steer to avoid a collision. *International Journal of Vehicle Safety*, 7(1):87–106, January 2014. Publisher: Inderscience Publishers.

[256] Vindhya Venkatraman, John D. Lee, and Chris W. Schwarz. Steer or Brake?: Modeling Drivers' Collision-Avoidance Behavior by Using Perceptual Cues. *Transportation Research Record*, 2602(1):97–103, January 2016. Publisher: SAGE Publications Inc.

[257] Manjiang Hu, Yuan Liao, Wenjun Wang, Guofa Li, Bo Cheng, and Fang Chen. Decision Tree-Based Maneuver Prediction for Driver Rear-End Risk-Avoidance Behaviors in Cut-In Scenarios. *Journal of Advanced Transportation*, 2017:e7170358, February 2017. Publisher: Hindawi.

[258] Abhijit Sarkar, Jeffrey S. Hickman, Anthony D. McDonald, Wenyan Huang, Tobias Vogelpohl, and Gustav Markkula. Steering or braking avoidance response in SHRP2 rear-end crashes and near-crashes: A decision tree approach. *Accident Analysis & Prevention*, 154:106055, May 2021.

[259] Mikael Ljung Aust, Johan Engström, and Matias Viström. Effects of forward collision warning and repeated event exposure on emergency braking. *Transportation Research Part F: Traffic Psychology and Behaviour*, 18:34–46, May 2013.

[260] Anthony D. McDonald, Hananeh Alambeigi, Johan Engström, Gustav Markkula, Tobias Vogelpohl, Jarrett Dunne, and Norbert Yuma. Toward Computational Simulations of Behavior During Automated Driving Takeovers: A Review of the Empirical and Modeling Literatures. *Human Factors*, 61(4):642–688, June 2019. Publisher: SAGE Publications Inc.

[261] George EP Box and George C Tiao. *Bayesian inference in statistical analysis*. John Wiley & Sons, 2011.

[262] Robert J Tibshirani and Bradley Efron. An introduction to the bootstrap. *Monographs on statistics and applied probability*, 57(1):1–436, 1993.

[263] George Casella and Roger Berger. *Statistical inference*. Chapman and Hall/CRC, 2024.

[264] Kristofer D. Kusano and Hampton C. Gabler. Safety Benefits of Forward Collision Warning, Brake Assist, and Autonomous Braking Systems in Rear-End Collisions. *IEEE Transactions on Intelligent Transportation Systems*, 13(4):1546–1555, December 2012. Conference Name: IEEE Transactions on Intelligent Transportation Systems.

[265] Constantin Hubmann, Marvin Becker, Daniel Althoff, David Lenz, and Christoph Stiller. Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles. In *2017 IEEE intelligent vehicles symposium (IV)*, pages 1671–1678. IEEE, 2017.

[266] Sebastian Brechtel, Tobias Gindele, and Rüdiger Dillmann. Probabilistic decision-making under uncertainty for autonomous driving using continuous pomdps. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 392–399, 2014.

[267] Maxime Bouton, Akansel Cosgun, and Mykel J Kochenderfer. Belief state planning for autonomously navigating urban intersections. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 825–830. IEEE, 2017.

[268] Thomas H. B. FitzGerald, Philipp Schwartenbeck, Michael Moutoussis, Raymond J. Dolan, and Karl Friston. Active Inference, Evidence Accumulation, and the Urn Task. *Neural Computation*, 27(2):306–328, February 2015.

[269] Francisco J Varela, Evan Thompson, and Eleanor Rosch. *The embodied mind, revised edition: Cognitive science and human experience*. MIT press, 2017.

[270] Evan Thompson. *Mind in life: Biology, phenomenology, and the sciences of mind*. Harvard University Press, 2010.

[271] Karl Friston. Life as we know it. *Journal of The Royal Society Interface*, 10(86):20130475, September 2013. Publisher: Royal Society.

[272] Julian Kiverstein, Michael D. Kirchhoff, and Tom Froese. The Problem of Meaning: The Free Energy Principle and Artificial Agency. *Frontiers in Neurorobotics*, 16, June 2022. Publisher: Frontiers.

[273] Karl Friston. Affordance and Active Inference. In Zakaria Djebbara, editor, *Affordances in Everyday Life: A Multidisciplinary Collection of Essays*, pages 211–219. Springer International Publishing, Cham, 2022.

[274] Maxwell J. D. Ramstead. Easy as 1, 2, 3: On the Short History of the Use of Affordance in Active Inference. In Zakaria Djebbara, editor, *Affordances in Everyday Life: A Multidisciplinary Collection of Essays*, pages 193–202. Springer International Publishing, Cham, 2022.

[275] Paco Calvo and Karl Friston. Predicting green: really radical (plant) predictive processing. *Journal of The Royal Society Interface*, 14(131):20170096, June 2017. Publisher: Royal Society.

[276] Toon Van de Maele, Bart Dhoedt, Tim Verbelen, and Giovanni Pezzulo. A hierarchical active inference model of spatial alternation tasks and the hippocampal-prefrontal circuit. *Nature Communications*, 15(1):9892, November 2024. Publisher: Nature Publishing Group.

[277] Natalie Kastel, Casper Hesp, K. Richard Ridderinkhof, and Karl J. Friston. Small steps for mankind: Modeling the emergence of cumulative culture from joint active inference communication. *Frontiers in Neurorobotics*, 16, January 2023. Publisher: Frontiers.

[278] Ambreen Chohan, Geert JP Savelsbergh, Paulien Van Kampen, Marline Wind, and Martine HG Verheul. Postural adjustments and bearing angle use in interceptive actions. *Experimental brain research*, 171:47–55, 2006.

[279] John F Larish and John M Flach. Sources of optical information useful for perception of speed of rectilinear self-motion. *Journal of experimental psychology: human perception and performance*, 16(2):295, 1990.

[280] Brett R Fajen. Calibration, information, and control strategies for braking to avoid a collision. *Journal of Experimental Psychology: Human Perception and Performance*, 31(3):480, 2005.

[281] Ran Wei, Joseph Lee, Shohei Wakayama, Alexander Tschantz, Conor Heins, Christopher Buckley, John Carenbauer, Hari Thiruvengada, Mahault Albarracin, Miguel de Prado, et al. Navigation under uncertainty: Trajectory prediction and occlusion reasoning with switching dynamical systems. In *2025 IEEE International Automated Vehicle Validation Conference (IAVVC)*, pages 1–6. IEEE, 2025.

[282] Marie Pelé, Caroline Bellut, Elise Debergue, Charlotte Gauvin, Anne Jeanneret, Thibault Leclere, Lucie Nicolas, Florence Pontier, Diorne Zausa, and Cédric Sueur. Cultural influence of social information use in pedestrian road-crossing behaviours. *Royal Society open science*, 4(2):160739, 2017.

[283] Türker Özkan, Timo Lajunen, Joannes El Chliaoutakis, Dianne Parker, and Heikki Summala. Cross-cultural differences in driving behaviours: A comparison of six countries. *Transportation research part F: traffic psychology and behaviour*, 9(3):227–242, 2006.

[284] Karttikeya Mangalam, Yang An, Harshayu Girase, and Jitendra Malik. From Goals, Waypoints & Paths to Long Term Human Trajectory Forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15233–15242, 2021.

[285] Ran Wei, Anthony D. McDonald, Ranjana K. Mehta, and Alfredo Garcia. Active Inference Models of AV Takeovers: Relating Model Parameters to Trust, Situation Awareness, and Fatigue. *Human Factors*, page 187208241295932, November 2024.

[286] Pablo Lanillos, Cristian Meo, Corrado Pezzato, Ajith Anil Meera, Mohamed Baioumy, Wataru Ohata, Alexander Tschantz, Beren Millidge, Martijn Wisse, Christopher L. Buckley, and Jun Tani. Active Inference in Robotics and Artificial Agents: Survey and Challenges, December 2021. arXiv:2112.01871 [cs].

[287] Pietro Mazzaglia, Tim Verbelen, Ozan Çatal, and Bart Dhoedt. The Free Energy Principle for Perception and Action: A Deep Learning Perspective. *Entropy*, 24(2):301, February 2022. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[288] Asif Uzzaman, Monirul Islam, and Md Shimul Hossain. Testing autonomous vehicles in virtual environments: A review of simulation tools and techniques. *Control Syst. Optim. Lett*, 3:151–157, 2025.

[289] Julian F. Schumann, Johan Engström, Ran Wei, Shu-Yuan Liu, Jens Kober, and Arkady Zgonnikov. Resolving space-sharing conflicts in road user interactions through uncertainty reduction: An active inference-based computational model, 2026.

[290] Mikael Ljung Aust, Helen Fagerlind, and Fridulv Sagberg. Fatal intersection crashes in Norway: Patterns in contributing factors and data collection challenges. *Accident; analysis and prevention*, 45:782–91, March 2012.

[291] Azra Habibovic, Emma Tivesten, Nobuyuki Uchida, Jonas Bärgman, and Mikael Ljung Aust. Driver behavior in car-to-pedestrian incidents: An application of the driving reliability and error analysis method (dream). *Accident Analysis & Prevention*, 50:554–565, 2013.

[292] Joshua E Domeyer, John D Lee, and Heishiro Toyoda. Vehicle automation–other road user communication and coordination: Theory and mechanisms. *IEEE Access*, 8:19860–19872, 2020.

[293] Yee Mun Lee and Joshua Domeyer. Editorial to the special issue on road user interactions in the age of vehicle automation, 2026.

[294] FH Amundsen and C Hyden. Proceedings of first workshop on traffic conflicts. *Oslo, TTI, Oslo, Norway and LTH Lund, Sweden*, 78, 1977.

[295] Evangelia Portouli, Dimitris Nathanael, and Nicolas Marmaras. Drivers' communicative interactions: on-road observations and modelling for integration in future automation systems. *Ergonomics*, 57(12):1795–1805, 2014.

[296] Herbert H Clark and Susan E Brennan. Grounding in communication. *Perspectives on socially shared cognition*, 1991.

[297] Dorsa Sadigh, Nick Landolfi, Shankar S. Sastry, Sanjit A. Seshia, and Anca D. Dragan. Planning for cars that coordinate with people: leveraging effects on human actions for planning and active information gathering over human internal state. *Autonomous Robots*, 42(7):1405–1426, October 2018.

[298] Joshua Domeyer, Azadeh Dinparastdjadid, John D Lee, Grace Douglas, Areen Alsaid, and Morgan Price. Proxemics and kinesics in automated vehicle–pedestrian communication: Representing ethnographic observations. *Transportation research record*, 2673(10):70–81, 2019.

[299] Koen De Clercq, Andre Dietrich, Juan Pablo Núñez Velasco, Joost De Winter, and Riender Happee. External human-machine interfaces on automated vehicles: Effects on pedestrian crossing decisions. *Human factors*, 61(8):1353–1370, 2019.

[300] Yee Mun Lee, Ruth Madigan, Oscar Giles, Laura Garach-Morcillo, Gustav Markkula, Charles Fox, Fanta Camara, Markus Rothmueller, Signe Alexandra Vendelbo-Larsen, Pernille Holm Rasmussen, Andre Dietrich, Dimitris Nathanael, Villy Portouli, Anna Schieben, and Natasha Merat. Road users rarely use explicit communication when interacting in today ' s traffic: implications for automated vehicles. *Cognition, Technology & Work*, 23(2):367–380, May 2021.

[301] Joshua E Domeyer, John D Lee, Heishiro Toyoda, Bruce Mehler, and Bryan Reimer. Interdependence in vehicle-pedestrian encounters and its implications for vehicle automation. *IEEE transactions on intelligent transportation systems*, 23(5):4122–4134, 2020.

[302] T Zach Noonan, Pnina Gershon, Josh Domeyer, Bruce Mehler, and Bryan Reimer. Interdependence of driver and pedestrian behavior in naturalistic roadway negotiations. *Traffic injury prevention*, 23(sup1):S62–S67, 2022.

[303] Joshua E Domeyer, John D Lee, Heishiro Toyoda, Bruce Mehler, and Bryan Reimer. Driver-pedestrian perceptual models demonstrate coupling: Implications for vehicle automation. *IEEE Transactions on Human-Machine Systems*, 52(4):557–566, 2022.

[304] Fanta Camara, Serhan Cosar, Nicola Bellotto, Natasha Merat, and Charles W Fox. Towards pedestrian-av interaction: method for elucidating pedestrian preferences. In *IEEE International Workshop on Intelligent Robots and Systems (IROS)*, 2018.

[305] Charles Fox, Fanta Camara, Gustav Markkula, Richard Romano, Ruth Madigan, Natasha Merat, et al. When should the chicken cross the road. *Game theory for autonomous vehicle-human interactions*, 2018.

[306] Jaime F Fisac, Eli Bronstein, Elis Stefansson, Dorsa Sadigh, S Shankar Sastry, and Anca D Dragan. Hierarchical game-theoretic planning for autonomous vehicles. In *2019 International conference on robotics and automation (ICRA)*, pages 9590–9596. IEEE, 2019.

[307] Kyungwon Kang and Hesham A Rakha. Game theoretical approach to model decision making for merging maneuvers at freeway on-ramps. *Transportation Research Record*, 2623(1):19–28, 2017.

[308] Wilko Schwarting, Alyssa Pierson, Javier Alonso-Mora, Sertac Karaman, and Daniela Rus. Social behavior for autonomous vehicles. *Proceedings of the National Academy of Sciences*, page 201820676, November 2019.

[309] Prateek Thalya, Jordanka Kovaceva, Alessia Knauss, Nils Lubbe, and Marco Dozza. Modeling driver behavior in interactions with other road users. In *Proceedings of 8th Transport Research Arena*, 2020.

[310] Christian-Nils Boda, Esko Lehtonen, and Marco Dozza. A computational driver model to predict driver control at unsignalised intersections. *IEEE Access*, 8:104619–104631, 2020.

[311] Oscar Giles, Gustav Markkula, Jami Pekkanen, Naoki Yokota, Naoto Matsunaga, Natasha Merat, and Tatsuru Daimon. At the zebra crossing: Modelling complex decision processes with variable-drift diffusion models. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 41, 2019.

[312] Arkady Zgonnikov and Gustav Markkula. Evidence Accumulation Account of Human Operators' Decisions in Intermittent Control During Inverted Pendulum Balancing. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 716–721, October 2018. ISSN: 2577-1655.

[313] Fatema T Johora and Jörg P Müller. Zone-specific interaction modeling of pedestrians and cars in shared spaces. *Transportation research procedia*, 47:251–258, 2020.

[314] Olger Siebinga, Arkady Zgonnikov, and David A Abbink. Modelling communication-enabled traffic interactions. *Royal Society open science*, 10(5), 2023.

[315] Karl J Friston and Christopher D Frith. Active inference, communication and hermeneutics. *cortex*, 68:129–143, 2015.

[316] Karl Friston and Christopher Frith. A duet for one. *Consciousness and cognition*, 36:390–405, 2015.

[317] Giovanni Pezzulo, Günther Knoblich, Domenico Maisto, Francesco Donnarumma, Elisabeth Pacherie, and Uri Hasson. A Predictive Processing Framework for Joint Action and Communication, February 2026.

[318] Cristina Bicchieri. *Norms in the wild: How to diagnose, measure, and change social norms*. Oxford University Press, 2016.

[319] Laura Fraade-Blanar, Francesca Favarò, Johan Engström, Melissa Cefkin, Ryan Best, John Lee, and Trent Victor. Being good (at driving): Characterizing behavioral expectations on automated and human driven vehicles. *arXiv preprint arXiv:2502.08121*, 2025.

[320] Axel Constant, Maxwell JD Ramstead, Samuel PL Veissiere, John O Campbell, and Karl J Friston. A variational approach to niche construction. *Journal of the Royal Society Interface*, 15(141), 2018.

[321] Jelle Bruineberg, Erik Rietveld, Thomas Parr, Leendert van Maanen, and Karl J Friston. Free-energy minimization in joint agent-environment systems: A niche construction perspective. *Journal of theoretical biology*, 455:161–178, 2018.

[322] Axel Constant, Maxwell JD Ramstead, Samuel PL Veissière, and Karl Friston. Regimes of expectations: An active inference model of social conformity and human decision making. *Frontiers in psychology*, 10:679, 2019.

[323] Kai Tian, Athanasios Tzigieras, Chongfeng Wei, Yee Mun Lee, Christopher Holmes, Matteo Leonetti, Natasha Merat, Richard Romano, and Gustav Markkula. Deceleration parameters as implicit communication signals for pedestrians' crossing decisions and estimations of automated vehicle behaviour. *Accident Analysis & Prevention*, 190:107173, 2023.

[324] Debargha Dey and Jacques Terken. Pedestrian interaction with vehicles: roles of explicit and implicit communication. In *Proceedings of the 9th international conference on automotive user interfaces and interactive vehicular applications*, pages 109–113, 2017.

[325] Linda Miller, Jasmin Leitner, Johannes Kraus, and Martin Baumann. Implicit intention communication as a design opportunity for automated vehicles: Understanding drivers' interpretation of vehicle trajectory at narrow passages. *Accident Analysis & Prevention*, 173:106691, 2022.

[326] Arkady Zgonnikov, Niek Beckers, Ashwin George, David Abbink, and Catholijn Jonker. Nudging human drivers via implicit communication by automated vehicles: Empirical evidence and computational cognitive modeling. *International Journal of Human-Computer Studies*, 185:103224, 2024.

[327] Erving Goffman. *Interaction ritual: Essays in face-to-face behavior*. Routledge, 2017.

[328] Edwin B Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927.

[329] Matus Sucha, Daniel Dostal, and Ralf Risser. Pedestrian-driver communication and decision strategies at marked crossings. *Accident Analysis & Prevention*, 102:41–50, 2017.

[330] Domenico Maisto, Francesco Donnarumma, and Giovanni Pezzulo. Interactive inference: A multi-agent model of cooperative joint actions. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 54(2):704–715, 2023.

[331] Olger Siebinga. A model of the sidewalk salsa. In *2025 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 7298–7305. IEEE, 2025.

[332] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

[333] Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in neural information processing systems*, 35:4328–4343, 2022.

[334] Chiyu Jiang, Andre Cornman, Cheolho Park, Benjamin Sapp, Yin Zhou, Dragomir Anguelov, et al. Motiondiffuser: Controllable multi-agent motion prediction using diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9644–9653, 2023.

[335] Anna Mészáros, Javier Alonso-Mora, and Jens Kober. Studying the effect of explicit interaction representations on learning scene-level distributions of human trajectories. In *2026 IEEE Intelligent Vehicles Symposium (IV)*, pages 75–82. IEEE, 2026.

[336] Boris Ivanovic and Marco Pavone. Rethinking trajectory forecasting evaluation. *arXiv preprint arXiv:2107.10297*, 2021.

[337] Nathan Drenkow, Numair Sani, Ilya Shpitser, and Mathias Unberath. A systematic review of robustness in deep learning for computer vision: Mind the gap? *arXiv preprint arXiv:2112.00639*, 2021.

[338] Arsham Gholamzadeh Khoee, Yinan Yu, and Robert Feldt. Domain generalization through meta-learning: a survey. *Artificial Intelligence Review*, 57(10):285, 2024.

[339] Pedro RAS Bassi, Sergio SJ Dertkigil, and Andrea Cavalli. Improving deep neural network generalization and robustness to background bias via layer-wise relevance propagation optimization. *Nature Communications*, 15(1):291, 2024.

[340] Samir H.A. Mohammad, Haneen Farah, and Arkady Zgonnikov. Modeling Gap Acceptance in Overtaking: A Cognitive Process Approach. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, pages 5925–5931, September 2023. ISSN: 2153-0017.

[341] Patricia R DeLucia. Effects of size on collision perception and implications for perceptual theory and transportation safety. *Current directions in psychological science*, 22(3):199–204, 2013.

[342] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

[343] Karl Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2):127–138, 2010.

[344] Chenhui Yin, Marco Cecotti, Daniel J Auger, Abbas Fotouhi, and Haobin Jiang. Deep-learning-based vehicle trajectory prediction: A review. *IET Intelligent Transport Systems*, 19(1):e70001, 2025.

[345] Renhao Huang, Hao Xue, Maurice Pagnucco, Flora D Salim, and Yang Song. Vision-based multi-future trajectory prediction: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

[346] Mahmoud Taha, Ahmed Zaky, Hirozumi Yamaguchi, and Ahmed Fares. Pedestrian trajectory and intention prediction: a comprehensive review of models, datasets, and challenges. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–33, 2026.

[347] Yongjun Yan, Peng Zhang, Chao Du, Hongliang Wang, Hao Wang, Dawei Pi, and Ye-Hwa Chen. Vehicle trajectory prediction for autonomous driving applications: State-of-the-art review, research challenges, and future directions. *Automotive Innovation*, pages 1–33, 2026.

[348] Sotiris Pelekis, Thanos Koutroubas, Afroditi Blika, Anastasis Berdelis, Evangelos Karakolis, Christos Ntanos, Evangelos Spiliotis, and Dimitris Askounis. Adversarial machine learning: a review of methods, tools, and critical industry sectors: S. pelekis et al. *Artificial Intelligence Review*, 58(8):226, 2025.

[349] Sungyong Seo, Sercan Arik, Jinsung Yoon, Xiang Zhang, Kihyuk Sohn, and Tomas Pfister. Controlling Neural Networks with Rule Representations. In *Advances in Neural Information Processing Systems*, volume 34, pages 11196–11207, 2021.

[350] Karl Friston, Conor Heins, Tim Verbelen, Lancelot Da Costa, Tommaso Salvatori, Dimitrije Markovic, Alexander Tschantz, Magnus Koudahl, Christopher Buckley, and Thomas Parr. From pixels to planning: Scale-free active inference. *Frontiers in Network Physiology*, 5:1521963, 2025.

[351] United Nations Economic Commission for Europe. Guidelines and recommendations for automated driving system safety requirements, assessments and test methods to inform regulatory development. Technical Report ECE/TRANS/WP.29/2024/39, UNECE, Geneva, Switzerland, 2025.

[352] European Commission. Ai act - regulatory framework for artificial intelligence. `https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai`, 2026. Accessed: 2026-03-11.

[353] Alan Ohnsman. Waymo raises $16 billion to speed up its global robotaxi push. `https://www.forbes.com/sites/alanohnsman/2026/02/02/waymo-raises-16-billion-to-speed-up-its-global-robotaxi-push/`, 2026. Accessed: 2026-03-11.

[354] Kristofer D Kusano, John M Scanlon, Yin-Hsiu Chen, Timothy L McMurry, Tilia Gode, and Trent Victor. Comparison of waymo rider-only crash rates by crash type to human benchmarks at 56.7 million miles. *Traffic Injury Prevention*, 26(sup1):S8–S20, 2025.

[355] John M Scanlon, Kristofer D Kusano, Johan Engstrom, and Trent Victor. Collision avoidance effectiveness of an automated driving system using a human driver behavior reference model in reconstructed fatal collisions. In *WCX SAE World Congress Experience*. SAE Technical Paper, 2026.

[356] Kristofer Kusano, Kurt Beatty, Scott Schnelle, Francesca Favaro, Cam Crary, and Trent Victor. Collision avoidance testing of the waymo automated driving system. In *WCX SAE World Congress Experience*. SAE Technical Paper, 2026.

[357] Guy Hoffman, Tapomayukh Bhattacharjee, and Stefanos Nikolaidis. Inferring human intent and predicting human action in human–robot collaboration. *Annual Review of Control, Robotics, and Autonomous Systems*, 7(1):73–95, 2024.

[358] Javier Laplaza, Francesc Moreno, and Alberto Sanfeliu. Enhancing robotic collaborative tasks through contextual human motion prediction and intention inference. *International Journal of Social Robotics*, 17(10):2077–2096, 2025.

[359] Séverin Lemaignan, Mathieu Warnier, E Akin Sisbot, Aurélie Clodic, and Rachid Alami. Artificial cognition for social human–robot interaction: An implementation. *Artificial Intelligence*, 247:45–69, 2017.

[360] Luzia Knoedler, Chadi Salmi, Hai Zhu, Bruno Brito, and Javier Alonso-Mora. Improving pedestrian prediction models with self-supervised continual learning. *IEEE Robotics and Automation Letters*, 7(2):4781–4788, 2022.

[361] Thibault Kruse, Amit Kumar Pandey, Rachid Alami, and Alexandra Kirsch. Human-aware robot navigation: A survey. *Robotics and Autonomous Systems*, 61(12):1726–1743, 2013.

[362] Juan Fasola and Maja J Matarić. A socially assistive robot exercise coach for the elderly. *Journal of Human-Robot Interaction*, 2(2):3–32, 2013.

[363] Car Simulator (UoLDS).

[364] I.M Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4):86–112, January 1967.

[365] Franco P Preparata and Michael I Shamos. *Computational geometry: an introduction*. Springer Science & Business Media, 2012.

Acknowledgments

Looking back at the last five years I spent on my PhD, it must be admitted that this would not have been as enjoyable and fruitful without the help and involvement of many special people who. No project is truly a one-person effort, and I was blessed to have the opportunity to work and interact with so many talented and inspiring people. Consequently, I would like to take this opportunity to express my sincere thanks to all of them.

First and foremost, this thesis would not have been possible without the guidance, support on constant availability of my supervisors and promoters. Most of all, I would like to thank Arkady and Jens for placing this much trust in me and my capabilities, starting from their initial decision to hire me. Furthermore, being always able to count on their feedback and advice during all the stages of each of my project – from the conception to the writing of a paper and its submission – was greatly reassuring. Furthermore, Arkady deserves special appreciation for his effective networking on my behalf, which allowed me to interact, work with, and learn from many other talented researchers, especially at the University of Leeds and at Waymo. I would also like to thank Jens, Arkady, and Johan for their support in compiling and writing this thesis. I am also greatly indebted to my committee members (Prof. Martijn Wisse, Prof. Joost de Winter, Prof. Karl Friston, and Dr. Kaouther Messaoud) for taking time to read my thesis, provide feedback, and participate in my defense. I want to express my gratitude to Saray Bakker for her help with the Dutch translation of the summary, too.

As mentioned above, I was fortunate enough to be able to spend two months at the University of Leeds in the fall of 2022. There, I had the privilege to run experiments on a large scale driving simulator, and collaborate with many talented researchers. I would like to thank Aravinda, Yueyang, Gustav and Natasha for being such welcoming hosts and for their help and collaboration during my stay. Furthermore, it was nice to be included in many social events as well, such as the Christmas party at Natasha's house. I would also like to thank the Taylor Connectivity Fund for their financial support of this experience financially.

Since the beginning of 2024, I then had the opportunity to work with the human behavior modeling team under Dr. Johan Engström at Waymo, and I am pleased to see him participate in this defense as well. This collaboration allowed me to develop my PhD in a new direction, introducing me to Active Inference and the many talented people working thereon. I would like to thank Johan, as well as the rest of his team (Ran, Leif, Matthew, and Shu-Yuan) for their constant support and the pleasant working environment pushing me constantly toward excellence. I also enjoyed my time meeting you in person throughout the years at various conferences and workshops, and I am looking forward to continuing working with you as a part of your team at the end of the month. I would also like to thank Waymo for their funding of those conference visits and their general support for my PhD.

A highlight of this collaboration was my internship at Waymo in London in the summer of 2025. This stay gave me many valuable experience, helping me to improve my coding skills (especially in collaborative projects) and allowing me to work with many talented researchers. I would like to thank Yawei and Joao especially for supervising me during this time and all their effort for making this experience as seamless as possible. I would also like to thank Rongbing, Luke, Daphne, Paul Caroline, and Angad for constantly including me in different activities (such as the climbing, the musical visits, or the punting trip to Oxford) and making me feel so welcome. It really was a pleasure and made time fly by. I am looking forward to seeing you all again shortly, and continuing these relationships. I would also like to thank Arkady, Martijn, and Laura for their help in getting this internship approved by the faculty after my premature acceptance, I am in your debt.

I would also like to thank my all my co-authors for their time and effort in collaborating with me on various projects. A special thank you goes out to Anna for your help and fruitful collaboration across the multiple common projects. Most importantly, I am deeply grateful for your patience with my overly optimistic approach to our common projects and for suffering the multiple rounds of changes and revisions this forced onto you. I would I would also like to thank the members of the Herald Lab (Ashwin, Frederik, Steven, Eduardo, and Luca) for all their valuable feedback during my presentations in our meetings and the food they provided for them. I am happy that in the end I managed to do some collaborative work with you as well, and I wish you all the best for the end of your own PhD journeys and everything that comes after.

While work and projects are an important aspect of a PhD, the social part is a crucial aspects of this experience as well. Consequently, I need to thank all the nice people in the department that made my time much more enjoyable both in- and outside the office. First, I would like to thank my original office mates from F-1-480 for their warm welcome and for spending their time together. Specifically, I would like to thank Micah for his wisdom and Mentorship, but also Gustavo, Anna, Leandro (thank you again for inviting me to your wedding, that was really special), Irene, and Lorenzo (thank you for introducing me to 40k combat patrol, I miss beating you) for their friendship and the many fun moments we shared. Seeing most of you leave during the latter part of my PhD was difficult, but I am grateful for the memories we created. I also want to thank the newcomers to the department as well, especially Kate, Amy, and Beatrice for all the fun conversations.

I would also like to thank the people in F-1-240 for allowing me to become an unofficial member and allow me to join in their outings such as the climbing trips, going skiing, and the frequent tea brakes (I miss your enthusiasm for my cakes as well). Thanks again to Anna, Luzia, Saray, Yujie, Maxi, Elia, Domenico, and Andreo, you all made my time much more enjoyable. I would also like to thank many of you (and Corrado, Alvero, Natalia, and Patricia) for introducing me to bouldering and spending so many enjoyable hours at different climbing gyms. I also would like to thank Maxi, Lea, and Ahmat for humoring me on the tennis court as well.

Furthermore, I was also lucky enough to be admitted to the local student orchestra, Krashna Musika. Besides the weekly rehearsals Monday evening, I also had many wonderful and musical experiences during all the concerts, as well as the tours to France as well as Czechia and Poland. Having this constant presence of music in my life was on of the main cornerstones of my time in Delft.

I also need to thank my girlfriend Yika for her support during the later parts of my PhD. Having you at my side was a great source of encouragement and validation that made the final phase of my PhD much more enjoyable and much less stressful. I would also like to thank you for your feedback on this thesis and your help with proofreading. I am really happy to have you with me on this important day, and I am looking forward to our common future with great anticipation. 逸佳, 我爱你和非常感激你的陪伴.

Lastly, I would like to thank my parents for their lifelong support and encouragement leading me to this moment. Not only did you encourage my interest in mathematics and science from a very young age, but also did everything to facilitate my education and personal development. I would not be in this place without you. I would also like to thank my brother Florian, most of all for being a constant companion and support throughout my life, but also for providing me with a competition that always challenged me to be the best version of myself. I am also thankful for your help with proofreading this thesis. Danke für alles, was ihr für mich getan habt.

Julian
Delft, September 2026

List of Publications

Journal Publications

J. F. Schumann, J. Kober, A. Zgonnikov. "Benchmarking behavior prediction models in gap acceptance scenarios". IEEE Transactions on Intelligent Vehicles 8 (2023): 2580–2591.

J. F. Schumann*, J. Hagenus*, F. B. Mathiesen, A. Zgonnikov. "Realistic Adversarial Attacks for Robustness Evaluation of Trajectory Prediction Models via Future State Perturbation". ACM Journal on Autonomous Transportation Systems (2026).

J. F. Schumann, J. Engström, L. Johnson, M. O'Kelly, J. Messias, J. Kober, A. Zgonnikov. "Active inference as a model of collision avoidance behavior in human drivers". Nature Communications (2026).

J. F. Schumann*, A. Mészáros*, J. Kober, A. Zgonnikov. "STEP: Structured Training and Evaluation Platform for benchmarking trajectory prediction models". IEEE Open Journal of Intelligent Transportation Systems (2026, *under review*).

J. F. Schumann, J. Engström, R. Wei, S.-Y. Liu, J. Kober, A. Zgonnikov. "Resolving space-sharing conflicts in road user interactions through uncertainty reduction: An active inference-based computational model". Collective Intelligence (2026, *under review*).

Conference Publications

J. F. Schumann, A. R. Srinivasan, J. Kober, G. Markkula, A. Zgonnikov. "Using models based on cognitive theory to predict human behavior in traffic: A case study". 2023 IEEE International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2023.

F. S. B. Westerhout, **J. F. Schumann**, A. Zgonnikov. "Smooth-Trajectron++: Augmenting the Trajectron++ behaviour prediction model with smooth attention". 2023 IEEE International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2023.

A. Mészáros, **J. F. Schumann**, J. Alonso-Mora, A. Zgonnikov, J. Kober. "TrajFlow: Learning Distributions over Trajectories for Human Behavior Prediction". 2024 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2024.

J. Hagenus, F. B. Mathiesen, **J. F. Schumann**, A. Zgonnikov. "Robustness in trajectory prediction for autonomous vehicles: a survey". 2024 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2024.

J. F. Schumann*, A. Mészáros*, J. Alonso-Mora, A. Zgonnikov, J. Kober. "Rome: Robust multi-modal density estimator" 2024 International Joint Conference on Artificial Intelligence (IJCAI). IJCAI, 2024.

J. F. Schumann, E. Figueiredo, F. B. Mathiesen, L. Laurenti, J. Kober, A. Zgonnikov. "Randomized smoothing as a defense against adversarial attacks in trajectory prediction". 2026 IEEE International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2026.

J. F. Schumann, J. Engström, R. Wei, S.-Y. Liu, J. Kober, M. Wisse, A. Zgonnikov. "Emotion in an active inference model of human driving". International Workshop on Active Inference. IWAI, 2026.

E. Figueiredo, F. B. Mathiesen, **J. F. Schumann**, A. Zgonnikov, L. Laurenti. "Clustered Randomized Smoothing for Stochastic Prediction Functions". Neurips 2026 (under review).

Included in this thesis.
* Authors contributed equally.

Curriculum Vitæ

Julian Frederik Schumann

10/02/1999 Date of birth in Münster, Germany

Education

09/2021–09/2026 **Ph.D. Robotics**
Delft University of Technology, The Netherlands

08/2019–08/2021 **M.Sc. Mechanical Engineering** (cum laude)
Delft University of Technology, The Netherlands

09/2016–07/2019 **B.Sc. Mechanical Engineering**
Technische Universität Braunschweig, Germany

Work Experience

06/2025–09/2025 **Research Intern**, Waymo LLC, London, UK
Working on implementing active inference models for autonomous driving.

10/2022–12/2022 **Research Visit**, Institute for Transport Studies, University of Leeds, UK
Working on implementing a cognitive-science based model for human gap acceptance behavior.

06/2019–08/2019 **Research Intern**, Volkswagen AG, Braunschweig, Germany
Worked on the development of battery cooling systems for electric vehicles.

2017–2019 **Teaching Assistant**, Technische Universität Braunschweig, Germany
Holding seminars in the courses for mathematics, mechanics, and thermodynamics.

