Delft University of Technology

Document Version
Final published version

Citation (APA)
Morão Patrício, M. L. (2026). *Toward Socially Aware Robots: A systems-theoretic approach for behavioral control of social robots*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:e6abcd36-2651-41b2-a10a-b9ffbc86f7aa

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.

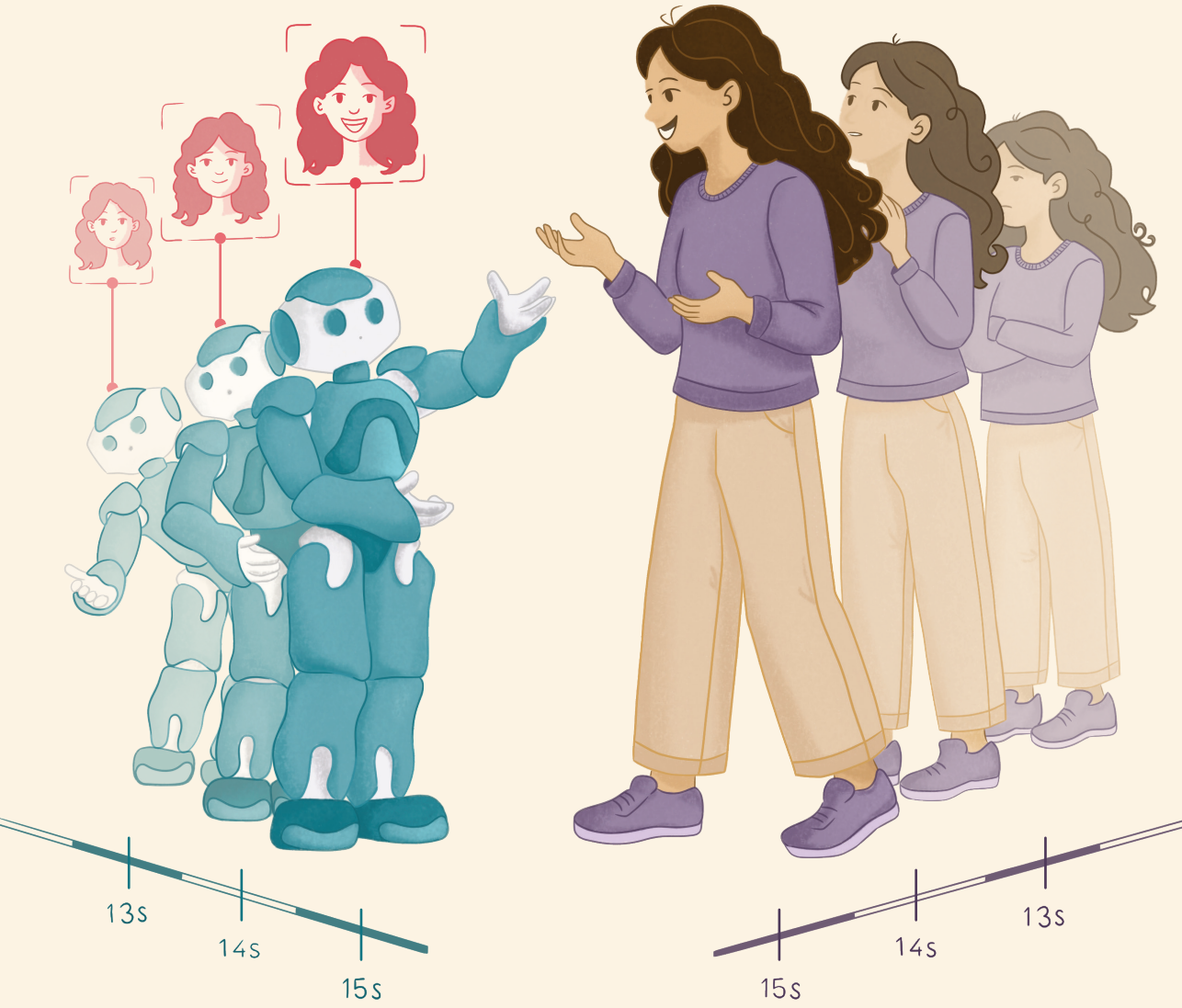
Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.

Toward Socially Aware Robots

A systems-theoretic approach for behavioral control of social robots

Maria L. Morão Patrício



Toward Socially Aware Robots

A systems-theoretic approach for behavioral control of social robots

Toward Socially Aware Robots

A systems-theoretic approach for behavioral control of social robots

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, September 18, 2026 at 12:30

by

Maria Luís MORÃO PATRÍCIO

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

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Prof. dr. ir. J. Hellendoorn	Delft University of Technology, *promotor*
Dr. A. Jamshidnejad	Delft University of Technology, *copromotor*
Independent members:	
Prof. dr. M. T. J. Spaan	Delft University of Technology
Prof. dr. M. M. Dastani	Utrecht University, the Netherlands
Prof. dr. R. Alami	Centre National de la Recherche Scientifique, France
Prof. dr. P. U. Lima	Instituto Superior Técnico, University of Lisbon, Portugal
Prof. dr. L. Sentis	University of Texas, Austin, USA
Prof. dr. ir. W. P. Brinkman	Delft University of Technology, *reserve member*



Keywords: Systems and control; Social robots; Theory of Mind; Human-social robot interaction.
Printed by: Ipskamp Printing
Cover by: Samantha Alves Martinho | The Daydreamer's Tale

Copyright © 2026 by M.L. Morão Patrício

ISBN 978-94-6518-423-4

An electronic copy of this dissertation is available at
https://repository.tudelft.nl/.

Summary

The growth and aging of the world population is causing an increase in the number of vulnerable individuals who require continuous support. However, health and social care systems do not possess enough human and material resources to address this growing demand. Social robots are a potential solution to help relieve this pressure by providing affordable, personalized, and continuous support to vulnerable individuals that can complement existing health and social care systems.

Nevertheless, state-of-the-art social robots struggle to sustain user engagement, especially in the long term, which ultimately limits their usability. Importantly, it is precisely through long-term interactions that social robots have the most potential to systematically tackle the overload in care systems. Therefore, enabling social robots to sustain user engagement is of utmost importance to facilitate their real-world deployment and societal impact.

The ability to sustain user engagement is strongly linked with the behaviors displayed by social robots and, consequently, with the approaches used to control said behavior. In state-of-the-art social robotics, most behavioral control approaches rely on either rule-based systems or machine-learning approaches, where both paradigms struggle to generate adaptive and personalized robot behavior. To sustain user engagement, social robots must replicate the responsiveness and adaptability that characterize human-to-human interactions, including being sensitive to others' mental states. However, behavioral controllers of social robots do not typically model the dynamic mental states of users, nor do they adapt robot behavior to those states.

Therefore, this thesis addresses this crucial limitation by presenting a behavioral control framework for social robots that models the dynamics of user mental states and systematically generates robot behavior according to these dynamics. To do so, the proposed framework is grounded in systems and control-theoretic methods, which enable transparent and systematic decision-making.

The first step toward such a behavioral control framework is presented in Chapter 2, where a dynamic mathematical model of human perception, cognition, and decision-making is developed based on Theory of Mind. This model represents the dynamic evolution of beliefs, goals, and emotions as a state-space formulation, enabling systematic integration of the model with systems and control-theoretic approaches that can be used to steer the behavior of social robots. The model's capability to estimate the mental states and predict the actions of users was evaluated in an experiment with human volunteers.

Building on the dynamic user model introduced in Chapter 2, Chapter 3 presents a predictive, model-based behavioral control framework for social robots. The model is incorporated into a closed-loop, constrained multi-objective optimization problem that systematically selects robot actions to steer the interaction toward desired user mental states. This chapter constitutes the first integration of systems and control-theoretic methods for

behavioral control of social robots, enabling transparent, adaptive, and cognition-aware decision-making. The proposed controller was validated in an experiment with human participants interacting with a humanoid robot, demonstrating improved engagement and sustained interaction quality.

Since the performance of model-based controllers depends on accurate estimation of the user's mental states, Chapter 4 presents three model-based state estimators for social robots that embed the dynamic model presented in Chapter 2. These estimators explicitly address the sparse and subjective nature of measurements in human-social robot interaction by personalizing process and measurement noise covariance matrices and proposing three tailored Extended Kalman Filter variants. The methods were validated using data from an experiment involving human volunteers interacting with a humanoid robot, demonstrating reduced cumulative prediction errors and improved robustness of mental state tracking over time.

To position the proposed behavioral control framework within the field of social robotics, Chapter 5 presents a systematic review of control approaches for behavior generation in social robots. The survey identifies dominant algorithmic paradigms and analyzes their underlying assumptions, strengths and limitations, and capacity to support adaptive and socially appropriate behavior in long-term human-social robot interactions. Building on this review, the chapter then synthesizes cross-category insights, evaluates control approaches by interaction length, system role, and application, and outlines emerging research directions toward more adaptive, interpretable, and cognitively grounded behavioral control frameworks. By clarifying existing strengths and gaps, this chapter motivates the need for transparent, model-based, systems-theoretic behavioral control architectures.

Together, these contributions establish a coherent, theoretically grounded behavioral control framework that bridges cognitive modeling, state estimation, and behavioral control for social robots. This thesis demonstrates that treating human-social robot interaction as a dynamic control problem, where mental states are estimated and steered over time, is both feasible and effective. The proposed framework, therefore, offers a concrete path to deploying social robots that can meaningfully support individuals across sustained, real-world interactions.

Samenvatting

De groei en vergrijzing van de wereldbevolking leiden tot een toename van het aantal kwetsbare personen dat continue ondersteuning nodig heeft. De gezondheidszorg en sociale zorg beschikken echter niet over voldoende menselijke en materiële middelen om aan deze groeiende vraag te voldoen. Sociale robots bieden een potentiële oplossing om deze druk te verlichten door betaalbare, gepersonaliseerde en continue ondersteuning te bieden aan kwetsbare personen, als aanvulling op de bestaande gezondheidszorg en sociale zorg.

Desalniettemin hebben de meest geavanceerde sociale robots moeite om de betrokkenheid van gebruikers te behouden, vooral op de lange termijn, wat uiteindelijk hun bruikbaarheid beperkt. Juist door langdurige interacties hebben sociale robots het grootste potentieel om de overbelasting van de zorgsystemen systematisch aan te pakken. Het is daarom van cruciaal belang dat sociale robots de betrokkenheid van gebruikers kunnen behouden om hun inzet in de praktijk en hun maatschappelijke impact te bevorderen.

Het vermogen om de betrokkenheid van gebruikers te behouden is sterk gekoppeld aan het gedrag dat sociale robots vertonen en dus aan de methoden die worden gebruikt om dat gedrag te sturen. In de meest geavanceerde sociale robotica zijn de meeste gedragsbesturingsmethoden gebaseerd op regelgebaseerde systemen of *machine learning*-methoden, waarbij beide paradigma's moeite hebben om adaptief en gepersonaliseerd robotgedrag te genereren. Om de betrokkenheid van gebruikers te behouden, moeten sociale robots de responsiviteit en het aanpassingsvermogen repliceren die kenmerkend zijn voor interacties tussen mensen, inclusief gevoeligheid voor de mentale toestand van anderen. Gedragsregelaars voor sociale robots modelleren echter doorgaans niet de dynamische mentale toestand van gebruikers, noch passen ze het robotgedrag aan die toestand aan.

Daarom pakt dit proefschrift deze cruciale beperking aan door een raamwerk voor gedragsbesturing van sociale robots te presenteren dat de dynamiek van de mentale toestand van gebruikers modelleert en systematisch robotgedrag genereert op basis van deze dynamiek. Het voorgestelde raamwerk is gebaseerd op systeem- en regeltheoretische methoden, die transparante en systematische besluitvorming mogelijk maken.

De eerste stap naar een dergelijk raamwerk voor gedragsbesturing wordt gepresenteerd in Hoofdstuk 2, waar een dynamisch wiskundig model van menselijke perceptie, cognitie en besluitvorming wordt ontwikkeld op basis van de *Theory of Mind*. Dit model beschrijft de dynamische evolutie van overtuigingen, doelen en emoties in de vorm van een toestandsruimteformulering, waardoor systematische integratie van het model met systeem- en regeltheoretische benaderingen die gebruikt kunnen worden om het gedrag van sociale robots te sturen, mogelijk gemaakt wordt. Het vermogen van het model om mentale toestanden te schatten en acties van gebruikers te voorspellen, werd geëvalueerd in een experiment met menselijke vrijwilligers.

Voortbouwend op het dynamische gebruikersmodel dat in Hoofdstuk 2 werd geïntroduceerd, presenteert Hoofdstuk 3 een voorspellend, modelgebaseerd gedragsbesturings-

kader voor sociale robots. Het model is geïntegreerd in een gesloten-lus, multi-objectief optimalisatieprobleem met randvoorwaarden dat systematisch robotacties selecteert om de interactie te sturen naar de gewenste mentale toestanden van de gebruiker. Dit hoofdstuk vormt de eerste integratie van systeem- en regeltheoretische methoden voor gedragsbesturing van sociale robots, waardoor transparante, adaptieve en cognitiebewuste besluitvorming mogelijk wordt. De voorgestelde regelaar werd gevalideerd in een experiment met menselijke deelnemers die interacteerden met een humanoïde robot, waarbij verhoogde betrokkenheid en een duurzame interactiekwaliteit werden aangetoond.

Omdat de prestaties van modelgebaseerde regelaars afhangen van een nauwkeurige schatting van de mentale toestand van de gebruiker, presenteert Hoofdstuk 4 drie modelgebaseerde toestandschatters voor sociale robots die het dynamische model uit Hoofdstuk 2 integreren. Deze schatters pakken expliciet het schaarse en subjectieve karakter van metingen in de interactie tussen mens en sociale robot aan door covariantiematrices voor proces- en meetruis te personaliseren en drie op maat gemaakte varianten van het uitgebreide Kalman-filter voor te stellen. De methoden werden gevalideerd met behulp van data van een experiment waarin menselijke vrijwilligers interacteerden met een humanoïde robot. De resultaten toonden een vermindering in de cumulatieve voorspellingsfouten en een verbeterde robuustheid van de mentale toestandsregistratie in de loop van de tijd aan.

Om het voorgestelde raamwerk voor gedragsbesturing te positioneren binnen het vakgebied van de sociale robotica, presenteert Hoofdstuk 5 een systematisch overzicht van besturingsbenaderingen voor het genereren van gedrag in sociale robots. Het overzicht identificeert dominante algoritmische paradigma's en analyseert hun onderliggende aannames, sterke en zwakke punten, en hun vermogen om adaptief en sociaal passend gedrag te ondersteunen in langdurige interacties tussen mens en sociale robot. Voortbouwend op dit overzicht synthetiseert het hoofdstuk vervolgens cross-categoriale inzichten, evalueert het besturingsbenaderingen op basis van interactieduur, systeemrol en toepassing, en schetst het opkomende onderzoeksrichtingen naar meer adaptieve, interpreteerbare en cognitief onderbouwde raamwerken voor gedragsbesturing. Door bestaande sterke punten en lacunes te verduidelijken, motiveert dit hoofdstuk de behoefte aan transparante, modelgebaseerde, systeemtheoretische architecturen voor gedragsbesturing.

Samen vormen deze bijdragen een coherent, theoretisch onderbouwd raamwerk voor gedragsbesturing dat een brug slaat tussen cognitieve modellering, toestandschatting en gedragsbesturing voor sociale robots. Dit proefschrift toont aan dat het behandelen van de interactie tussen mens en sociale robot als een dynamisch besturingsprobleem, waarbij mentale toestanden worden geschat en bijgestuurd in de loop van de tijd, zowel haalbaar als effectief is. Het voorgestelde raamwerk biedt daarom een concrete weg naar de inzet van sociale robots die individuen op zinvolle wijze kunnen ondersteunen tijdens langdurige interacties in de echte wereld.

Contents

Summary vii

Samenvatting ix

1. Introduction 1
   - 1.1. Socially Assistive Robots: From Promise to Practice . . . 2
     - 1.1.1. Challenges to Effective Social Assistance . . . 3
     - 1.1.2. Toward User Understanding . . . 4
   - 1.2. Research Objective and Research Questions . . . 6
     - 1.2.1. User Modeling . . . 7
     - 1.2.2. Model-based Control and Estimation . . . 8
     - 1.2.3. Validation and Evaluation of the Behavioral Control Framework . 9
   - 1.3. Methodologies . . . 10
     - 1.3.1. Systems and Control Theory as the Primary Approach . . . 10
     - 1.3.2. Development Process of the Behavioral Control Framework . . . 11
     - 1.3.3. Experimental Procedures . . . 12
     - 1.3.4. Evaluation Metrics . . . 12
   - 1.4. Thesis Outline . . . 13

2. Dynamic Mathematical Models of Theory of Mind for Social Robots 17
   - 2.1. Introduction . . . 18
     - 2.1.1. Related Work and Open Challenges . . . 20
   - 2.2. Methodologies . . . 23
     - 2.2.1. Main Motivations . . . 23
     - 2.2.2. Network Representation: Identifying Variables and Inter-dynamics 25
     - 2.2.3. Dynamic Mathematical Representation of Perception, Cognition, and Decision-making . . . 30
   - 2.3. Case study . . . 35
     - 2.3.1. Experiment Setup . . . 35
     - 2.3.2. Implementation Setup . . . 37
     - 2.3.3. Training Phase for the Models . . . 38
   - 2.4. Results and Discussions . . . 39
   - 2.5. Conclusions and Topics for Future Research . . . 45

3. A Model-Based Behavioral Control Approach for Human-Social Robot Interaction 47
   - 3.1. Introduction . . . 48
     - 3.1.1. Main Contributions . . . 50

3.1.2. Related Work . . . 51
3.2. Methodology . . . 55
3.2.1. Leveraging MMM for Model-based Control of SRs . . . 56
3.2.2. Identification of the Model . . . 63
3.2.3. Implementation of MMM . . . 67
3.2.4. Closed-loop Model-based Control of SRs . . . 69
3.2.5. Assumptions Underlying MMM and the Proposed Framework . . . 70
3.3. Case study . . . 72
3.3.1. Experimental Setup . . . 72
3.3.2. Model-based Controller for Nao . . . 77
3.4. Results and Discussions . . . 84
3.4.1. Results for Model Identification . . . 84
3.4.2. Results for Model-based Controller . . . 85
3.4.3. Computational Efficiency . . . 92
3.4.4. Limitations and Mitigation Strategies . . . 93
3.5. Conclusions and Topics for Future Research . . . 94

4. A Systems-Theoretic Approach to Mental State Estimation for Theory-of-Mind-Aware Social Robots 97
4.1. Introduction . . . 98
4.1.1. Main Contributions . . . 99
4.1.2. Related Work . . . 101
4.2. Methodology . . . 103
4.2.1. Construction of the Noise Covariance Matrices . . . 103
4.2.2. Implementation of the Extended Kalman Filter . . . 107
4.3. Case Study . . . 113
4.3.1. Experimental Setup . . . 113
4.3.2. Estimation Assessment . . . 115
4.4. Results and Discussions . . . 116
4.4.1. State Estimation Accuracy . . . 116
4.4.2. Computational Efficiency . . . 119
4.4.3. Convergence Analysis . . . 121
4.5. Conclusions and Topics for Future Research . . . 123

5. Behavioral Control of Social Robots: State-of-the-Art and Future Directions 125
5.1. Introduction . . . 126
5.1.1. Research Questions . . . 127
5.1.2. Positioning Among Existing Surveys . . . 128
5.2. Methods . . . 129
5.2.1. Literature Identification and Selection . . . 129
5.2.2. Review Procedure . . . 131
5.2.3. Relevant Concepts . . . 132
5.3. Control Approaches Used in the Literature . . . 133
5.3.1. Symbolic and Rule-Based Approaches . . . 134
5.3.2. Learning-Based and Data-Driven Approaches . . . 147
5.3.3. Modeling Approaches . . . 158

5.3.4. Probabilistic Approaches . . . 175
5.4. Discussion and Synthesis . . . 178
5.4.1. Suitability of Control Approaches by Interaction Length . . . 178
5.4.2. Suitability of Control Approaches by System Role . . . 179
5.4.3. Suitability of Control Approaches by Application . . . 186
5.5. Conclusions . . . 187

6. Discussions and Conclusions **189**
6.1. From Promise to Practice: Revisiting the Motivation and Research Gaps . . 190
6.2. Revisiting the Research Objective and Questions . . . 190
6.2.1. Research Questions . . . 191
6.3. Limitations and Recommendations for Future Work . . . 198
6.3.1. Theoretical and Mathematical Assumptions . . . 198
6.3.2. Modeling and System Design Choices . . . 199
6.3.3. Experimental and Validation Limitations . . . 201
6.4. Final Remarks . . . 202

A. Examples that illustrate the importance of the elements of the perception, cognition, and decision-making model **203**

B. Supplementary Material for the Nao Chess Interaction Case Study **209**
B.1. Difficulty levels of the chess puzzles . . . 209
B.2. Structure of the modules of the MMM study . . . 210
B.3. Reward movements performed by Nao . . . 212
B.4. Questions posed to the participants during the interactions with Nao . . . 213
B.5. Questionnaire . . . 214

C. Methodology and Overview of the Literature Review presented in Chapter 5 **215**
C.1. Literature Search Strategy and Source Overview . . . 215
C.2. Comparative Overview of Reviewed Literature . . . 216

References **227**

Acknowledgements **241**

Curriculum Vitæ **245**

List of Publications **247**

Propositions **249**

Acronyms **252**

1

Introduction

1

1.1. Socially Assistive Robots: From Promise to Practice

With the growth and aging of the global population, as well as improved diagnosis of developmental and cognitive conditions, the number of vulnerable individuals requiring continuous support, including older adults, children with autism spectrum disorder, and people with cognitive impairments, is steadily increasing (Bradwell et al., 2021; Macalupu et al., 2025; Matarić & Scassellati, 2016). However, the human and material resources available to provide adequate support to these populations are not keeping pace with demand, placing growing pressure on healthcare and social care systems (Bradwell et al., 2021; Macalupu et al., 2025; Matarić & Scassellati, 2016). Assistive robots, defined as robotic systems that provide support or aid to users (Feil-Seifer & Matarić, 2005), hold significant potential to alleviate such pressure by providing complementary forms of affordable support to vulnerable groups (Bradwell et al., 2021; Matarić & Scassellati, 2016).

While early developments of assistive robots focused primarily on physical assistance, such as guiding limb movements during rehabilitation, enabling user mobility, and manipulating objects on behalf of users, it has long been acknowledged that assistance often entails a social dimension (Feil-Seifer & Matarić, 2005; Matarić & Scassellati, 2016). As social interaction is an essential component of human assistance, effective assistive robots should exhibit social behaviors, such as motivating patients or offering emotional support (Matarić & Scassellati, 2016). Consequently, the relevance of social conduct in assistive roles led to the combination of Assistive Robotics with Social Robotics, a field dedicated to enabling robots to interact with humans in a socially meaningful way. This integration resulted in Socially Assistive Robotics, which focuses on robots that provide support to users via social, rather than physical, interaction (Feil-Seifer & Matarić, 2005; Matarić & Scassellati, 2016).

The potential of Socially Assistive Robots (SARs) to enhance the quality of life for vulnerable populations is evidenced by their applications across diverse domains, including health care, education, and companionship. In healthcare, they have been applied in post-stroke rehabilitation (Bettosi et al., 2024; M. H. Lee et al., 2023; Tapus & Mataric, 2008), gait rehabilitation (Céspedes et al., 2020; Céspedes et al., 2021), rehabilitation for individuals with cognitive impairments (Benedictis et al., 2023; Kubota et al., 2020), and facilitation of pediatric vaccination (Rossi et al., 2020). In education, SARs have been used for tutoring (Sequeira et al., 2016), playing educational games with students (Cao, Beir et al., 2017; Páez et al., 2020; Schicchi & Pilato, 2018), and supporting skill development in children with autism spectrum disorder (Clabaugh et al., 2019; Scassellati et al., 2018). Beyond healthcare and education, SARs can also serve as companions (Cooper et al., 2020; Luperto et al., 2024; Mitchinson & Prescott, 2016) or as general assistive tools to support users in their daily life (Dziergwa et al., 2018; Rodríguez-Lera et al., 2018). Overall, SARs represent a promising approach to address the growing societal need for assistive technologies (Bradwell et al., 2021; Matarić & Scassellati, 2016). Nevertheless, several challenges continue to limit their widespread real-world implementation (Matheus et al., 2025).

1.1.1. Challenges to Effective Social Assistance

Although SARs have shown promising outcomes across a range of applications in studies involving short-term interactions, real-world applications capable of meaningfully addressing shortages in health and social care typically require sustained, long-term user support. These applications involve guiding users through extended processes of learning, rehabilitation, or behavior change, such as teaching skills to users with developmental impairments, supporting post-stroke rehabilitation plans, or facilitating adherence to healthy lifestyles. For SARs to be effective in such applications, the interaction between the user and the robot should be sustained over long periods of time, a concept known as long-term interaction (Maroto-Gómez, Alonso-Martín et al., 2023; Matheus et al., 2025; Tahir et al., 2020).

Despite the recognized importance of long-term interactions with SARs for assistive applications, studies involving such interactions remain relatively scarce in the literature (Maroto-Gómez, Alonso-Martín et al., 2023; Matarić & Scassellati, 2016). This scarcity can be attributed to various factors, including the need for technically robust and autonomous systems capable of operating reliably over extended periods, difficulties in recruiting participants who are willing to engage in long-term studies, and the higher costs associated with conducting extended experiments (Clabaugh et al., 2019; Matheus et al., 2025). Beyond these practical obstacles, a more fundamental barrier limits long-term interactions: the difficulty of sustaining user engagement over time. In contrast to practical limitations that affect the feasibility of long-term studies, user engagement directly determines whether long-term socially assistive interactions can be sustained (Clabaugh et al., 2019; Matheus et al., 2025).

Sustaining user engagement, therefore, emerges as a central challenge for enabling effective interactions between humans and SARs. Low engagement can lead to decreased interaction frequency or eventual abandonment, affecting the ability of SARs to achieve their intended assistive purposes and, ultimately, hindering their acceptability and usability in real-world settings (Matarić & Scassellati, 2016; Matheus et al., 2025). However, unlike technical or logistical barriers, this challenge can be directly linked to the behavior of SARs. In fact, one of the most widely recognized factors that affects user engagement is the novelty effect, which refers to users being temporarily more interested or attentive when first encountering a SAR. However, as users become acquainted with the actions and behavioral patterns of the robot, the novelty effect fades, leading to a decrease in engagement (Leite et al., 2013; Matheus et al., 2025). Sustaining user engagement, therefore, requires understanding and systematically improving the behaviors exhibited by SARs.

Although the need to improve the ability of SARs to sustain user engagement has been acknowledged in the field for decades, it remains an unresolved challenge (Leite et al., 2013; Maroto-Gómez, Alonso-Martín et al., 2023; Matarić & Scassellati, 2016; Tapus et al., 2007). This highlights the importance of critically examining the approaches and paradigms used to control the behavior of SARs, in order to understand how these may contribute to the difficulties in sustaining engagement and displaying human-like social behaviors. By doing so, it becomes possible to explore and identify novel strategies with the potential to overcome these limitations.

1.1.2. Toward User Understanding

State-of-the-art SARs are predominantly designed for specific tasks or application domains, limiting their ability to adapt to novel or unexpected interaction scenarios (Cao et al., 2019; Maroto-Gómez, Alonso-Martín et al., 2023; Matarić & Scassellati, 2016). This design paradigm is linked to the prevalent approaches used for behavioral control of SARs. The majority of existing approaches rely on rule-based or Machine Learning (ML) strategies to generate behavior (Chapter 5). Rule-based systems, while interpretable, require manual design and inherently struggle to handle situations that fall outside predefined rules. ML approaches, despite capturing complex behavioral patterns, face critical limitations when applied in socially assistive robotics. The scarcity of large, high-quality datasets constrains the training of behavioral control systems, which leads deployments to simplify interaction scenarios or constrain the robot's behavioral repertoire, further reducing adaptability (Chapter 5). More recently, cognitive architectures that emulate the internal processes of SARs for generating behavior have received increasing attention in the literature, providing a more structured approach to behavior generation (Chapter 5). Despite these advances, SARs still largely lack adaptability and personalization capabilities, which are critical for effective long-term interactions with users (Cao et al., 2019; Leite et al., 2013; Matheus et al., 2025). Addressing these limitations requires examining the fundamental aspects of human social interaction that current SARs fail to replicate.

While humans intuitively infer and respond to the mental states of others, SARs still struggle to mimic this capability. During social interactions, humans naturally model each other's mental states, including beliefs, desires, and emotions (Benedictis et al., 2023). This ability is closely related to the concept of Theory of Mind (ToM) (Premack & Woodruff, 1978), i.e., an individual's capacity to attribute mental states to others while recognizing that those mental states may differ from their own. Furthermore, humans adjust their own behaviors in accordance with their estimations of others' mental states and of the expected evolution of those mental states (Baker et al., 2007; Baker et al., 2008; Benedictis et al., 2023). In contrast, most SARs still lack these capabilities. Although the literature acknowledges the potential benefits of SARs modeling user mental states and manifesting ToM, few existing SARs (or even Social Robots (SRs), in general) implement these capabilities comprehensively (Benedictis et al., 2023; Matarić & Scassellati, 2016; Patacchiola & Cangelosi, 2020; Tahir et al., 2020; Tapus et al., 2007; Zhu et al., 2024). Importantly, the challenges related to mental-state modeling and socially-aware behavior generation go beyond assistive contexts and are general to social robotics (Patacchiola & Cangelosi, 2020; Tahir et al., 2020; Zhu et al., 2024). As SARs are a subset of SRs, developments in behavioral control for SRs that address mental state modeling and socially-aware behavior are equally relevant to SARs.

A few initial attempts have been made to embed ToM principles into the behavioral control systems of SRs (including SARs), illustrating their potential for improving social behaviors while also highlighting current limitations. Gray and Breazeal (2014) designed a ToM-inspired belief manipulation approach, which simulates the user perspective to model user beliefs and then selects actions to influence those beliefs. Although the framework broadly refers to "mental states", its implementation is limited

to beliefs concerning the physical world and does not readily extend to other types of mental states, such as goals, intentions, or emotions. The SR proposed by Vossen et al. (2018) handles uncertainty and conflicting information in social interactions by implementing ToM principles. Any information acquired by the SR is stored with source attribution, which enables the robot to model different perspectives for various people who may hold conflicting beliefs. J. J. Lee et al. (2019) developed two ToM-inspired behavioral controllers for SRs interacting in storytelling contexts: a storyteller controller models a listener's attentional state to select cues that foster their engagement; the listener controller infers the storyteller's belief about the listener's engagement to create the impression that the listener is paying attention. Importantly, the framework supports instantiating either role in a SR. Both behavioral controllers showed modest improvements over baseline controllers that did not account for mental state modeling, demonstrating the importance of modeling the beliefs of interaction partners. Finally, a key ToM concept recently explored in social robotics is the ability to pass the false-belief test, which involves recognizing that the beliefs of others can differ from one's own. Tani et al. (2020) and Dissing and Bolander (2020) both proposed computational approaches that enable SRs to pass variants of the false-belief test, demonstrating the feasibility of explicit perspective-taking mechanisms. However, both approaches focus primarily on the cognitive reasoning required for specific test scenarios and do not constitute general-purpose, integrated social behavior control architectures. While the approaches reviewed in this paragraph represent important first steps in ToM incorporation in SRs controllers, they are narrowly focused on specific ToM components and fail to provide the comprehensive mental state modeling that characterizes human social cognition. Furthermore, these approaches primarily focus on static representations and do not model the dynamic and predictive aspects of human social cognition, i.e., the evolution of users' mental states over time and the impact of the robot's actions on that evolution.

In addition to ToM-inspired approaches, some recent studies have more broadly incorporated mental states of users in behavioral control systems of SRs, including SARs. Rossi et al. (2020) proposed a SAR behavioral control system to facilitate pediatric vaccination, in which a small rule set maps the manually-assessed anxiety level and age of the child to predefined conversational content and positive or negative emotional displays. Filippini et al. (2021) developed an infrared-based approach for emotion detection in SRs. In their case study, the SR detected a two-dimensional emotional state of its user and adjusted its behavior based on a simple rule set. Bagheri et al. (2021) proposed a framework enabling SRs to display empathy by detecting one of four user emotions (anger, sadness, happiness, or surprise) and by accordingly selecting one of four predefined verbal empathic behaviors using Reinforcement Learning (RL). Cao, Beir et al. (2017) introduced a behavior generation architecture for SARs in social therapy with children, in which the detected levels of user attention, performance, and emotion directly influence the internal drive dynamics that determine action selection. Páez et al. (2020) proposed a behavioral control framework enabling SARs to provide educational support, selecting robot behavior based on a set of goals that depend on user knowledge, skills, and emotional state. These three elements are estimated by fusing multi-sensor audio-visual inputs using computational models informed by

psychological theory. Benedictis et al. (2023) proposed a cognitive control architecture for SARs providing general health-care assistance. Using rule sets and symbolic reasoning, the control system adjusts the behavior of the robot based on long-term medical needs of the user and short-term variables. Two of the short-term variables, user intention and sentiment, relate to the user's mental states and are inferred from their utterances. All in all, most state-of-the-art approaches that incorporate mental states in SRs are either overly simplistic and/or highly specialized, and have only been tested in short-term settings. Many consider only a limited set of mental states in their controllers and rely on simple mechanisms, such as rule sets or ML algorithms with low-dimensional state and action spaces, to define robot behavior. The approaches that do estimate a broader range of mental states and/or implement more complex behavioral control typically require domain-specific customization, such as designing tailored logic based on context-specific knowledge. Additionally, these approaches do not model the temporal evolution of users' mental states or explicitly anticipate how current robot actions may influence future mental state dynamics of users. In contrast, human social cognition involves estimating and updating a broad set of mental states across situations and over time. This enables behavior to adapt fluidly as interactions unfold. Consequently, although current SR behavioral control approaches have demonstrated effectiveness in short-term interaction settings, they provide little support for sustained user engagement over extended periods.

Therefore, SARs still lack a general framework capable of systematically representing a comprehensive set of relevant user mental states in a way that can be directly leveraged for behavior generation across applications. Developing such a framework is essential for enabling SARs, and other SRs, to interact adaptively and appropriately with users and, as such, is a core objective of this thesis.

1.2. Research Objective and Research Questions

The previous section described the longstanding challenge of sustaining user engagement in long-term Human-Social-Robot Interactions (HSRIs) and highlighted the urgent need for SRs to understand and act accordingly to the mental states and behaviors of users.

Although this thesis is motivated by challenges in Socially Assistive Robotics, the proposed behavioral control framework is not inherently limited to assistive contexts. Rather, it addresses fundamental requirements for socially-aware behavior generation in robots interacting with humans — that is, SRs.

Therefore, the present thesis focuses on developing a behavioral control framework for SRs that aims to address this persisting, crucial research gap. Accordingly, the main research objective of this thesis is framed as follows:

Research Objective

Develop a context-independent behavioral control framework for Social Robots that models and leverages key mental states and behavioral elements of users to enable autonomous generation of meaningful, socially-aware, and interpretable robot behavior that sustains user engagement over the long term.

In the research objective, the expression "key mental states and behavioral elements" refers to the set of mental states and elements that are necessary to capture the behavior of humans in social interaction contexts. This contrasts with state-of-the-art approaches that focus on one or a few mental states for a specific application.

In order to achieve the proposed research objective, six research questions have been defined. For clarity of presentation, the research questions are grouped into three categories: the development of a user model, the design and implementation of the behavioral control framework, and the validation and evaluation of the framework in HSRIs.

1.2.1. User Modeling

The first step toward achieving the research objective is to design a model of the user that comprehensively represents human behavior in social interactions with robots. Such a model is the cornerstone of the behavioral control framework and, as such, should be carefully and rigorously developed. Therefore, the first two research questions of this thesis concern developing a user model that adequately captures the characteristics and requirements of HSRIs.

To develop the user model, it is necessary to draw inspiration from prior work, including frameworks and theories from psychology and cognitive neuroscience, which support an appropriate representation of user behavior for SRs. However, existing frameworks for user behavior and cognitive modeling are generally developed for broad purposes rather than for SR behavioral control, and, as a result, they may lack elements that are particularly important in HSRIs. Therefore, it is necessary to identify and incorporate additional behavioral or cognitive elements that are critical for developing a context-appropriate and comprehensive model. Thus, the first research question is defined as:

Research Question 1

Which theoretical frameworks and behavioral or cognitive elements can provide a foundation for developing a comprehensive and context-appropriate model of human behavior in social interactions?

Subsequently, the modeling frameworks, together with the behavioral and cognitive elements identified in the first research question, should be meaningfully integrated into a model of human behavior. Furthermore, this model should be mathematically formulated to systematically represent the evolution of human behavior in a way that

enables its integration within a behavioral control framework for SRs. Accordingly, a research question emerges regarding the formalization and formulation of the model:

Research Question 2

How should the identified modeling frameworks, along with the behavioral and cognitive elements, be integrated and mathematically formulated to develop a model that captures the dynamic evolution of human behavior in social interactions?

Together, the first and second research questions establish the foundations of the behavioral control framework proposed in this thesis by defining a theoretically grounded and operationalizable model of the user for SR control.

1.2.2. Model-based Control and Estimation

After developing a rigorous, mathematical model of human behavior, the next step is to introduce a behavioral control framework based on this model that steers the behavior of SRs in interactions with users. This is the focus of the third, fourth, and fifth research questions of this thesis.

One essential step of this process is to identify the control approaches that can leverage the user model to achieve the research objective. Determining these approaches requires reviewing not only SR literature to identify existing control approaches, but also general control theory approaches to identify potential methods that have not yet been applied in social robotics, but hold significant promise. This raises the question of which control approaches are the most suitable for leveraging the developed user model to generate socially-aware behaviors in SRs, i.e., behaviors that adapt to varying contexts and personalize to diverse users:

Research Question 3

Which control approaches are most suitable for enabling autonomous, socially-aware, and interpretable behavior generation in SRs, based on the developed user model?

On the one hand, the user model has a structure that draws on cognitive neuroscience and psychology literature, and models human mental states rather than well-defined physical quantities (e.g., position and speed), as is common in engineering and physical systems. On the other hand, control approaches are typically designed for such engineering and physical systems with quantifiable variables.

Furthermore, while the proposed behavioral control framework is designed to control the behavior of SRs in theory, real-world HSRIs present additional challenges. For example, variability in human behavior and limited data available for identifying the user model may negatively impact the reliability of the model and, consequently, the performance of the controller.

These considerations suggest that further adaptations are required both to integrate

the user model with the chosen control approach and to enable the effective operation of the controller in real-world HSRI settings. This leads to the following research question:

Research Question 4

What adaptations are required when integrating the user model into the control framework to ensure the effective operation of the controller in real HSRIs, given the requirements and limitations of these interactions?

As with most mathematical models of real-world phenomena, there is no guarantee that the quantified mental states will remain accurate throughout the interaction. The developed model predicts the evolution of these states based on previously estimated values. Because real systems rarely match their mathematical description exactly, small prediction errors can accumulate over time, even when the model is rigorous and theoretically sound.

Since the controller relies on these inferred mental states to generate the robot's actions, such drift can negatively impact the appropriateness and effectiveness of the robot's behavior during HSRIs. Therefore, it becomes necessary to maintain an accurate estimation of the user's mental states throughout the interaction. This motivates the following research question:

Research Question 5

How can the accuracy of model-based estimates of user mental states be maintained throughout interaction with the SR, in order to mitigate state drift and preserve controller performance?

All in all, the third, fourth, and fifth research questions focus on leveraging the user model for functional behavioral control of SRs, ensuring that the model can be implemented effectively while remaining reliable during real-world HSRI.

1.2.3. Validation and Evaluation of the Behavioral Control Framework

Finally, given that the proposed behavioral control framework is designed for user interactions with SRs, it is essential to evaluate its impact and benefits in real HSRIs. In particular, it is important to assess whether the framework enhances the awareness of SRs about user mental states and improves the quality of HSRIs in realistic contexts.

In addition, evaluating the contribution of the behavioral control framework to the landscape of behavioral control systems for SRs requires considering how its underlying principles and mechanisms relate to existing control approaches in social robotics, and how its application may inform future developments in the field.

Accordingly, the final research question of this thesis addresses both the impact of the proposed behavioral control framework in real HSRIs, as well as its role in informing future approaches to behavioral control of SRs:

1

Research Question 6

To what extent does the proposed control framework enhance the awareness of SRs regarding user mental states and improve the quality of HSRIs, and what implications does this have for future behavioral control approaches of SRs?

Thus, the sixth research question evaluates the practical impact and conceptual contribution of the behavioral control framework in shaping socially-aware SR behavior, based on the evaluation metrics described in Section 1.3.4.

1.3. Methodologies

This section describes the main methods adopted in this thesis to address its central research objective. First, the choice of systems and control theory as the primary approach is discussed. Next, the overall research workflow is outlined, highlighting the main steps and their relation to the research questions. Finally, the experimental procedures used to test and validate the framework are briefly summarized, with detailed descriptions provided in later chapters.

1.3.1. Systems and Control Theory as the Primary Approach

Throughout this thesis, we adopt systems and control theory as the primary methodological foundation for formulating both the user model and the SR behavioral control strategies. Systems and control-theoretic methods remain largely unexplored in social robotics, where behavioral control of robots is typically based on ML or rule-based approaches. Compared to ML approaches, systems and control theory is inherently transparent: the relationship between the user model and the resulting SR actions or decisions is explicit and interpretable, whereas most ML approaches function as black boxes. Additionally, systems and control-theoretic methods are more data-efficient, as they can leverage system models rather than requiring extensive training datasets, directly addressing the data scarcity challenge that limits ML deployment in social robotics. Compared to rule-based systems, formulating SR behavior as a systems and control problem leads to systematic decision-making, where formal mechanisms determine SR actions rather than manually predefined rules. Importantly, control-theoretic approaches support online adaptation, enabling systematic behavior adjustment to individual users and evolving circumstances, thereby mitigating the rigidity typical of manually specified rules in long-term HSRIs. Moreover, control-theoretic formulations allow establishing formal guarantees on constraint satisfaction that can support generating robot behaviors that ensure user well-being and safety, which are generally not available in rule-based systems.

Furthermore, systems and control theory offers specific advantages for modeling and guiding HSRIs. Feedback mechanisms enable the robot to adjust its actions in real time to changes in the user's behavior or mental states, ensuring adaptability during the interactions. At the same time, the use of an explicit user model provides a structured method to generate SR actions that are interpretable, as they are computed based on the model. Finally, the dynamic nature of systems and control-theoretic models

enables leveraging the user model in a predictive way. This allows estimating future user states and selecting robot actions that optimize these predicted states with respect to predefined objectives, such as maintaining engagement or minimizing discomfort.

1.3.2. Development Process of the Behavioral Control Framework

The methodological steps followed in the development and evaluation of the behavioral control framework proposed in this thesis, along with the research questions addressed at each stage, are summarized in Table 1.1.

Table 1.1: Research workflow of the development process of the proposed behavioral control framework.

Description of steps and substeps		Research Question
1	**Development of the user model**	
	Inter-disciplinary literature review of theoretical frameworks and key behavioral and mental state elements	RQ1
	Formalization of the user model	RQ2
	Mathematical formulation of the user model	RQ2
	Case-study: Validation of the user model for mental state and action estimation	RQ6
2	**Incorporation of the user model in a SR controller**	
	Selection of suitable control approaches and system architecture	RQ3
	Mathematical reformulation of the model and control formulation for their integration	RQ4
	Case-study: Validation of the proposed behavioral control framework with a three-session HSRIs	RQ6
3	**Incorporation of the user model in a state estimator**	
	Selection of a suitable state estimation algorithm	RQ5
	Adjustment of the state estimation algorithm to fit the human-centered context of HSRI	RQ5
	Experimental assessment of mental state estimation improvement	RQ6

In order to develop a behavioral control framework for SRs based on a model of user behavior, three main research steps are followed. The first step was to design, formalize, and mathematically formulate the user model. Subsequently, the user model was integrated into a predictive model-based control loop, which was then evaluated through real-life experiments with volunteer participants and a Nao robot as the SR. Finally, the user model was incorporated into a model-based state estimator to improve the estimations of mental states performed by the SR in scenarios with limited or

unavailable user feedback.

Importantly, these steps required mathematical adaptations to ensure that the leveraged systems and control-theoretic concepts, which are originally developed for engineering and physical systems, are appropriately tailored to the human-centered context of HSRI.

The order of these steps reflects the dependency structure of the proposed behavioral control framework: the user model constitutes the foundation for both the controller and the state estimator, and therefore should be established first; the controller is subsequently developed around this model to generate behavior informed by the user mental states predicted by the model; the state estimator is introduced as an additional component built upon the model, with the purpose of maintaining accurate inference of user mental states during long-term HSRIs, thereby supporting consistent controller performance over time.

These three research steps correspond to Chapters 2, 3, and 4, respectively, where each step is presented and evaluated in detail. A detailed description of how these steps are presented in the thesis is provided in Section 1.4.

1.3.3. Experimental Procedures

Two experiments with human volunteers were conducted to test and validate the proposed behavioral control framework. The first experiment involved twenty-one participants interacting with a computer simulation and was designed to evaluate the user model. The second experiment involved ten participants interacting with a Nao robot over three sessions to evaluate the performance and effectiveness of the model-based behavioral controller and the state estimator.

Both experiments were prepared and conducted in accordance with institutional ethical standards, with approval obtained from the Human Research Ethics Committee of Delft University of Technology (HREC application numbers 2398 and 3780).

1.3.4. Evaluation Metrics

The following metrics were selected to rigorously evaluate the user model, model-based controller, and state estimator proposed in this thesis across experimental studies.

The proposed user model was evaluated using intention estimation accuracy and action prediction accuracy across simulated emergency decision-making scenarios. These metrics were selected to assess the model's capability to infer latent mental states and to translate them into observable behavioral predictions. Additionally, comparisons with a reduced baseline model were used to assess the contribution of emotions and biases to mental-state representation.

The model-based controller was evaluated using three complementary measurement methods: user self-reports of mental states during interactions, post-interaction questionnaires, and automated video-based engagement analysis. Self-reported ratings of beliefs, goals, emotions, and intentions were used to evaluate interaction quality and provided ground truth for assessing the model's accuracy in predicting the mental states of users within HSRIs. Post-interaction Likert-scale questionnaires assessed perceived engagement, robot awareness, adaptability, and personalization, capturing subjective

interaction quality. Additionally, a computer-vision-based engagement detection tool provided a non-intrusive behavioral measure of engagement. The combination of these metrics enabled cross-validation of controller performance and ensured a multi-perspective evaluation of interaction quality and mental-state awareness.

Finally, the state estimator was evaluated by comparing mental-state prediction error against open-loop model-based estimation. Cumulative error reduction served as the primary metric to assess mitigation of estimation drift, while covariance evolution was analyzed to evaluate convergence and bounded uncertainty of the estimator.

Together, these metrics provide a comprehensive, multi-level validation of the social awareness of the robot, the user engagement, and the overall interaction quality in HSRIs. Additionally, they provide a quantifiable basis for addressing the sixth research question.

1.4. Thesis Outline

This thesis consists of six chapters, which aim to achieve the research objective and answer the research questions posed in Section 1.2, while following the methodology described in Section 1.3. Figure 1.1 presents a visual roadmap of this thesis.

Chapter 2, Chapter 3, and Chapter 4 correspond to the development of the framework, with each chapter addressing one of the research steps described in Section 1.3.2.

Chapter 2 describes the development of the user model, including a review of the literature to identify relevant theories and elements to inform model design, followed by formalization and mathematical formulation of the model. This chapter concludes with a case study, in which human participants interacted with a simulated environment to validate the capability of the model to estimate their mental states and predict their actions. The work presented in this chapter directly addresses the first and second research questions and provides an initial contribution toward answering the sixth research question. This chapter is based on the journal article *Dynamic Mathematical Models of Theory of Mind for Socially Assistive Robots*, published in IEEE Access (Morão Patrício & Jamshidnejad, 2023a).

Chapter 3 details the formulation of the SR controller that integrates the model proposed in Chapter 2. This includes the selection of a suitable control approach and extensive mathematical reformulations to both the model and the selected control approach to enable their integration. Furthermore, this chapter includes a case study that assesses the performance of the control framework in real-world HSRIs. The developments described in this chapter fully address the fourth research question and partially contribute to the third and sixth research questions. This chapter is based on the journal article *Leveraging Systems and Control Theory for Social Robotics: A Model-Based Behavioral Control Approach to Human-Robot Interaction*, published in Applied Intelligence (Morão Patrício & Jamshidnejad, 2026b).

Chapter 4 describes the incorporation of the user model into a state estimator. This includes the selection of a suitable state estimation algorithm and proposed adaptations to the classical state estimator to handle the abstract, non-physical variables represented in the user model. Additionally, the accuracy and performance of the proposed state

1

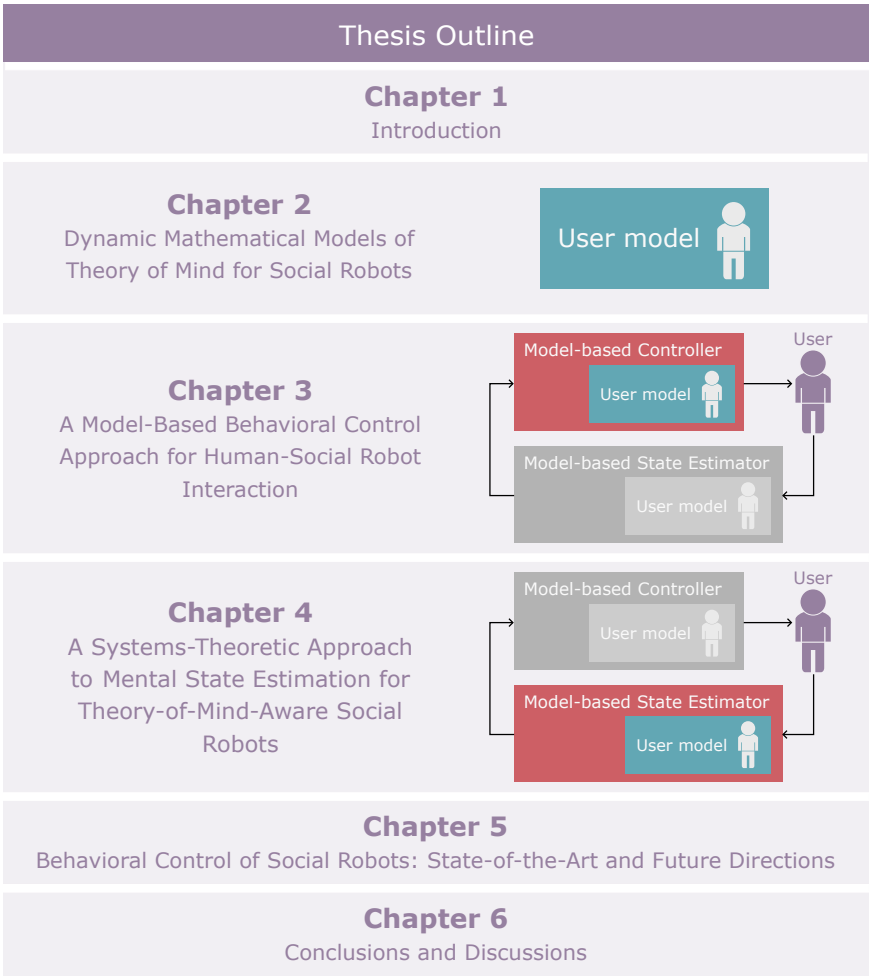


Figure 1.1: Thesis outline

estimator are assessed and compared to the case without a state estimator using the experimental data from the case study described in Chapter 3. The results reported in this chapter inform the fifth research question and provide additional evidence to the sixth research question. This chapter is based on the journal article *A Systems-Theoretic Approach to Mental State Estimation for Theory-of-Mind-Aware Social Robots*, published in IEEE Access (Morão Patrício & Jamshidnejad, 2025b).

Chapter 5 critically examines the current landscape of behavior control in social robotics, identifying and grouping the dominant frameworks, analyzing the assumptions that underlie them, and assessing their effectiveness in real-world HSRIs. Building on this analysis and the findings of Chapters 2, 3, and 4, which present the first efforts to apply classical systems-and-control-theoretic methods to human-centered

social robotics, Chapter 5 evaluates the limitations of existing methods and investigates emerging and underexplored directions. It further discusses the expected contributions of these methods and how strategic combinations of the reviewed approaches can potentially drive future advancements in the field. Finally, the analysis in this chapter offers the main contribution toward answering the third research question and establishes broader context supporting the sixth research question.

Finally, Chapter 6 synthesizes the key results of the previous chapters, revisits the research questions posed in Section 1.2, and discusses their broader implications. Chapter 6 concludes by outlining limitations of the work presented in this thesis and proposing directions for future research.

2

Dynamic Mathematical Models of Theory of Mind for Social Robots

Social robots hold significant potential to address the increasing demands on healthcare and social care systems by providing socially mediated support to vulnerable populations. However, the limited richness and adaptability of their social behaviors hinder sustained engagement in long-term interactions, ultimately constraining their applicability in real-world contexts. Overcoming these limitations requires control approaches that systematically represent and leverage the mental states of users, enabling social robots to interact more naturally. This chapter presents the foundational step toward such a framework by developing and validating the first dynamic model of user perception, cognition, and decision-making for social robots. Derived from established psychological research, the model is mathematically formalized as a state-space representation to facilitate integration with systematic and transparent control theoretic approaches.

This chapter is based on the following publication:

Morão Patrício, M. L., & Jamshidnejad, A. (2023a). Dynamic mathematical models of theory of mind for socially assistive robots. *IEEE Access*, *11*, 103956–103975. https://doi.org/10.1109/ACCESS.2023.3316603

Abstract

Interactive machines should establish and maintain meaningful social interactions with humans. Thus, they need to understand and predict the mental states and actions of humans. Based on Theory of Mind (ToM), in order to understand and interact with each other, humans develop cognitive models of one another. Our main goal is to provide a mathematical framework based on ToM to improve the understanding of interactive machines regarding the perception, cognition, and decision-making of humans. Most state-of-the-art models of behavioral theories based on machine learning are focused on input-output black-box representations. Thus, they lack transparency and generalizability, and exhaustive training procedures are needed to personalize them for various humans. Moreover, these models lack dynamics, i.e., they do not mathematically describe the evolution of the mental states and actions of humans in time. Following a systems-and-control-theoretic point of view, we represent for the first time the perception, cognition, and decision-making of humans via a dynamic, mathematical framework by introducing a novel formalization and an extension to Fuzzy Cognitive Maps (FCMs). The resulting models are given in a general state-space representation, which can be used by interactive machines within known model-based state estimation and control methods. In a case study, the resulting models were identified and validated for 21 participants, in scenarios where predicting the intentions and behavior of the participants required understanding the dynamics of their mental procedures. The results of these experiments show that our model is capable of incorporating the dynamics to estimate the intentions and predict the behavior of the participants, with an accuracy of, respectively, 81.55% and 66.06%. Moreover, we compared our model with a state-of-the-art formalization of human cognition, which was made dynamic using our introduced FCM framework. Our model, which in addition to the elements of the state-of-the-art model included emotions, personality traits, and biases (thus providing a more transparent insight about the mental procedures of the participants) showed 6.25% and 2.45% more accuracy in, respectively, estimating the intentions and predicting the behavior of the participants.

2.1. Introduction

Social Robots (SRs) assist humans mentally and physically by socially interacting with them. SRs have proven successful in boosting the outcomes of therapeutic and educational assistance for humans (Clabaugh et al., 2019; Feil-Seifer & Matarić, 2005; Kidd & Breazeal, 2008; Martinez-Martin & Cazorla, 2019; Scassellati et al., 2018; Tapus & Mataric, 2008). Most applications of SRs in literature involve short-term interactions (e.g., a few interactive sessions lasting no more than a month) with humans (Leite et al., 2013; Matarić & Scassellati, 2016) that exclude in-depth analysis of the perception, cognition, and decision-making of the individuals (Matarić & Scassellati, 2016). Since intelligent machines, including SRs, are becoming more prominent, it is essential for them to maintain long-term, meaningful and engaging interactions with humans (Matarić & Scassellati, 2016), whereas state-of-the-art SRs face serious challenges regarding this (Clabaugh et al., 2019; Kidd & Breazeal, 2008; Leite et al., 2013; Scassellati et al., 2018). Particularly, rudimentary social skills displayed by SRs

in interaction with humans negatively impact the human engagement, and thus the effectiveness of SRs (Al-Qaderi & Rad, 2018; Kidd & Breazeal, 2008; Leite et al., 2013; Scassellati et al., 2018). Although non-verbal cues, e.g., joint attention (i.e., drawing the attention of others to an object or person by looking or pointing at it) (Scassellati et al., 2018), eye contact (Kidd & Breazeal, 2008), and facial expressions (Clabaugh et al., 2019), have been implemented for SRs, these robots often fail to recognize the best time to display these cues in social interactions, due to a lack of (deep) understanding of the human cognition (Leite et al., 2013; Tapus et al., 2007). Moreover, personalizing the behavior of SRs to every human is crucial to maintain meaningful interactions (Clabaugh et al., 2019; Scassellati et al., 2018; Tapus & Mataric, 2008; Tapus et al., 2007). However, without a white-box approach (e.g., transparent mathematical models), personalization of SRs has been task-specific (Martinez-Martin & Cazorla, 2019; Tapus & Mataric, 2008), case-based, and ungeneralizable (Matarić & Scassellati, 2016). In fact, personalization for SRs has been limited to learning interactive behaviors that improve the performance of humans in the given tasks (Clabaugh et al., 2019; Scassellati et al., 2018; Tapus & Mataric, 2008; Tapus et al., 2007). Such task-specific, black-box methods used to control SRs result in interactions that are perceived as less natural and engaging for humans, especially in the long term.

The key to successful social interactions by humans, based on ToM (Premack & Woodruff, 1978), is to build cognitive models of each other and to interact based on these models. In order to interact as humanly as possible, SRs should exhibit similar understanding of rational agents (Al-Qaderi & Rad, 2018; Leite et al., 2013; Matarić & Scassellati, 2016; Scassellati, 2002; Tapus et al., 2007). Thus, we focus on developing dynamic mathematical models for perception, cognition, and decision-making of humans that can be used by intelligent machines in human-machine interactions to estimate the mental states of humans and to predict their behavior. In order to provide generalizable, white-box models with proper mathematical representation and axiomatization that can effectively be used in model-based state estimation and control methods by SRs, we provide the following main contributions:

- We follow systems-and-control-theoretic methods in order to model the perception, cognition, and decision-making of humans via dynamic equation-based models for the first time. We identify and represent the corresponding (controllable and uncontrollable) input, output, auxiliary, and state variables, as well as the identification parameters. To provide accuracy and transparency, we introduce additional state variables (e.g., emotions, intentions) and auxiliary variables (e.g., bias, perceived knowledge) into the perception, cognition, and decision-making models based on the existing behavioral theories from cognitive science literature.

- We propose a novel framework based on an extended version of FCMs. This framework transforms static representations of ToM into dynamic models with a general state-space representation. A main advantage of representing perception, cognition, and decision-making of humans via equation-based state-space models is that they can directly be embedded within established model-based state estimation and control methods to steer intelligent machines.

- The resulting models are trained and personalized for 21 human participants, and are validated via experiments that simulate scenarios of emergency evacuation in 2-dimensional environments. The participants complete each scenario by controlling a virtual agent in order to select its trajectory and sequence of tasks. The developed models are used to estimate the intentions and actions of each participant. These estimated intentions and actions are then compared to the intentions and actions selected by the participants to assess the accuracy of the models.

Including the auxiliary variable "biases" and the identifiable parameters, and defining new concepts of *rational intention selection* and *rational action selection*, which are agent-specific, instead of using a universal principle of rational action for all agents, provide our models with the unique potential to be personalized per human.

The rest of the chapter is structured as follows. Section 2.1.1 gives an overview of previous related work and identifies the open challenges. Section 2.2 presents the proposed methodologies, including the main motivations and mathematical approaches. In Section 2.3, the proposed models are implemented and assessed via real-life experiments with 21 volunteer human participants. Finally, Section 2.5 concludes the chapter and presents topics for future research.

2.1.1. Related Work and Open Challenges

Understanding the (individual) cognitive procedures of humans in their social interactions is important for both cognitive psychology and cognitive computing fields (Farrell & Lewandowsky, 2011; Farrell & Lewandowsky, 2018; Guest & Martin, 2021; Kidd & Breazeal, 2008; Mareschal & Thomas, 2007; Scassellati et al., 2018).

ToM was first proposed in the neuroscience domain (Premack & Woodruff, 1978), and since then has been used to develop computational frameworks for human cognition (Baker et al., 2008; Hansen & Bolander, 2020; J. J. Lee et al., 2019; Patacchiola & Cangelosi, 2020; Rabinowitz et al., 2018; Tani et al., 2020). ToM relies on the principle of rationality (Dennett, 1987; Popper, 1994), which implies that humans take actions that, according to their beliefs and goals, maximize their desired outcomes, and/or minimize their losses. This principle is the cornerstone of most ToM-based computational frameworks. While ToM involves three main mental procedures of humans, i.e., perception, cognition, and decision-making, Belief-Desire-Intention (BDI) (Bratman, 1987) is a framework that particularly models the decision-making (Bosse et al., 2011; Broersen et al., 2001; Rao, 1995). Based on BDI, the actions of rational agents are a consequence of their beliefs, desires, and intentions (without imposing specific conditions, e.g., rationality) (Bratman, 1987). ToM and BDI can complement each other (Bosse et al., 2011; Vossen et al., 2018): in fact, the concept of intentions introduced in BDI can be included in a dynamic, mathematical framework of ToM, in order to account for the interlinks within a sequence of actions of the agent, for achieving a particular intention (Rao, 1995).

Baker (2012) has proposed a Bayesian ToM model based on the network representation in Figure 2.1. Using Partially Observable Markov Decision Processes (POMDPs) (Åström, 1965) and Bayesian inference (Stigler, 1982), this model makes

forward and inverse inferences about, respectively, the actions and the beliefs and goals of rational agents that follow the principle of rationality. A simplified version of the model, excluding general world knowledge and general preferences, was implemented in Baker et al. (2011), where virtual rational agents were moving in two-dimensional spaces. The predictions that were made by the model and by human participants about the beliefs and goals of the virtual rational agent were compared, where the results showed close similarities, and acknowledged that goals and beliefs need to be inferred simultaneously. The model and experimental scenarios in Baker et al. (2011) did not account for the differences of humans and represented them via idealized rational agents that do not deal with biases, which is not true in real-life scenarios (Block & Funder, 1986; Witkin, 1949). Additionally, the stochastic nature of the model formulation makes its use limited for model-based predictive methods, since decision trees, which can exponentially grow with the size of the prediction horizon, need to be generated, and using these trees in optimization-based decision-making frameworks is not suited for real-time implementations. Finally, similarly to most computational frameworks of ToM, the model in Baker et al. (2011) does not include inferences about the emotions of humans. In fact, based on a recently published review paper (Ong et al., 2020), the amount of literature that models the understanding of emotional states from the perspective of an observer is very limited.

To summarize, the majority of the research on computational frameworks for ToM is focused on representing human behavior via POMDPs (Baker et al., 2008; Hansen & Bolander, 2020; J. J. Lee et al., 2019) or approximating it with Neural Networks (NNs) (Hansen & Bolander, 2020; J. J. Lee et al., 2019; Rabinowitz et al., 2018; Tani et al., 2020). The research aiming at using computational models of ToM to perform inverse inferences is even more scarce (Jara-Ettinger, 2019; Saxe & Houlihan, 2017). Furthermore, most state-of-the-art models are static representations that do not include dynamics (Hansen & Bolander, 2020; Patacchiola & Cangelosi, 2020; Tani et al., 2020), i.e., do not mathematically describe the evolution of the mental states of humans in time. The articles that do present a dynamic model are scarce (Battini Sönmez et al., 2022; J. J. Lee et al., 2019), and none of them proposes a general framework that covers the main aspects of ToM for a general context. For example, Battini Sönmez et al. (2022) dynamically models the emotions but does not focus on any other mental states, or on perception or decision-making, and J. J. Lee et al. (2019) presents a dynamic model that is tailored to one specific context. To the best of our knowledge, there has been no follow-up research on potential extensions for the model given in Baker (2012) in order to address its challenges and to exploit its potential for representing a ToM framework. The core focus of this chapter is on how to develop an equation-based model of and mathematically represent the mentioned theories (ToM and BDI), which have before been represented as verbal models, agent-based models, or computer simulations (Smaldino, 2023). Thus, the model shown in Figure 2.1 is our main inspiration for proposing the first dynamic, mathematical framework for ToM.

In order to complete this discussion, next we briefly refer to three other behavioral theories that have been proposed to understand human behavior. Although we do not use these theories in this chapter, since the approaches that are proposed, especially the dynamic framework, are not tailored to a specific behavioral theory, they may be

adopted for the following theories as well.

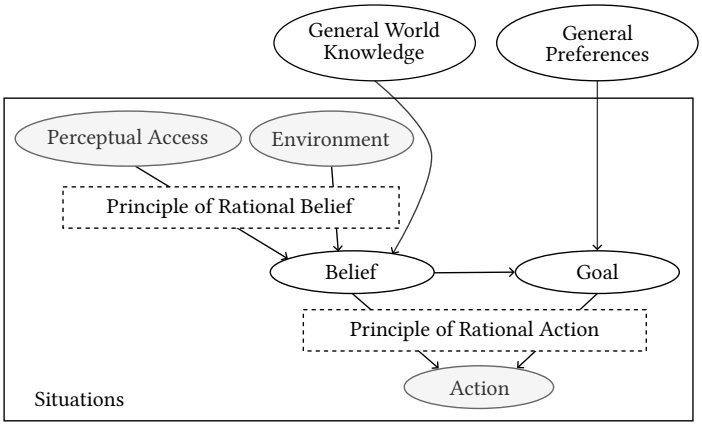


Figure 2.1: Network representation of human cognition from Baker (2012), explaining the actions taken by humans based on the principles of rational action and rational belief.

The Theory of Planned Behavior (TPB) describes the behavior of rational agents as a consequence of their intentions and perceived behavioral control (Ajzen, 1991), i.e., the perception of the agent regarding the feasibility of the intended action (Navarro-Barrientos et al., 2011). The intention is affected by the perceived behavioral control, the attitude (the value of the action according to the agent), and the subjective norm (the social value of the action). Related to TPB is Social Cognitive Theory (SCT), which describes behavior as emerging from a triadic reciprocity between personal factors (e.g., cognition, affect), environment, and behavior. Notably, similarly to ToM, TPB and SCT have been argued to require dynamic, rather than static, formulations to adequately capture behavior. Riley et al. (2011) reviewed applications of TPB and SCT in mobile health interventions and argued that, being largely static and between-person in nature, such theories are ill-suited to modeling the time-intensive, adaptive interventions enabled by modern sensing technologies. Importantly, they proposed control systems engineering as a means of addressing this limitation. Following this direction, Martín et al. (2014) modeled SCT as a dynamical system using a fluid analogy, representing constructs such as self-efficacy and behavior as interacting inventories that evolve over time. This use of systems and control theory to render a behavioral theory dynamic parallels the approach proposed in this chapter. Nevertheless, since these theories are mainly used in medical field applications, we have not focused on them in this chapter.

Common Model of Mind (CMM) (Laird et al., 2017) is a framework proposed with the aim of integrating the biological, cognitive, and motor procedures for modeling the behavior of rational agents. Due to its novelty when compared with other frameworks that describe human behavior, implementations of CMM are scarce (Chella & Pipitone, 2020), and do not provide significant contributions to the cognitive domain.

2.2. Methodologies

Next, we explain our methodologies for developing a dynamic mathematical framework for ToM, based on an extended version of BDI that includes emotions and biases. In the rest of the chapter, a rational[1] agent that makes inferences about the mental states and actions of another rational agent is called an *observer agent*. The other agent is called an *observed agent*. First, we briefly motivate our research.

2.2.1. Main Motivations

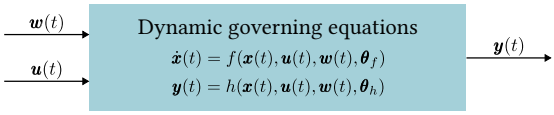


Figure 2.2: Mathematical modeling of dynamical systems.

Mathematical models are used in systems and control theory in order to represent the governing dynamics of systems as a set of differential (or difference) equations (Ogata, 1990). Figure 2.2 shows a system that, in general, receives controlled input vector $\boldsymbol{u}$ and uncontrolled input (external disturbances) $\boldsymbol{w}$ and generates output vector $\boldsymbol{y}$ (Matz et al., 2020). A mathematical dynamic model in continuous time may be represented by the following state space representation[2]

$$\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{w}(t), \boldsymbol{\theta}_f) \tag{2.1}$$
$$\boldsymbol{y}(t) = h(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{w}(t), \boldsymbol{\theta}_h) \tag{2.2}$$

where $\boldsymbol{x}(t)$ is a vector of dimension n_x that includes all the state variables of the system, $\boldsymbol{u}(t)$ is a vector of dimension n_u that includes all the inputs to the system, $\boldsymbol{w}(t)$ is a vector of dimension n_w that includes all the uncontrollable inputs to the system, and $\boldsymbol{y}(t)$ is a vector of dimension n_y that includes all the outputs of the system. Moreover, $f(\cdot)$ represents the dynamic model, $h(\cdot)$ is a function that generates the output. Finally, $\boldsymbol{\theta}_f$ and $\boldsymbol{\theta}_h$ are defined as vectors that include all the identification parameters of the corresponding mathematical model. In fact, when an observed system is being analyzed, in addition to determining $f(\cdot)$ and $h(\cdot)$, these parameter vectors should be identified. For instance, for linear time-invariant systems without disturbances, where (2.1) and (2.2) are transformed into $\dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + B\boldsymbol{u}(t)$ and $\boldsymbol{y}(t) = C\boldsymbol{x}(t) + D\boldsymbol{u}(t)$, with A, B, C, and D matrices of dimensions, respectively, $n_x \times n_x$, $n_x \times n_u$, $n_y \times n_x$, and $n_y \times n_u$, (Ogata, 1990; Zak et al., 2003), the vector $\boldsymbol{\theta}_f$ includes all elements of A and B, and similarly the vector $\boldsymbol{\theta}_h$ includes all elements of C and D. Assuming that the system is time-invariant, $\boldsymbol{\theta}_f$ and $\boldsymbol{\theta}_h$ do not depend on time, and thus the argument time t is not shown for these two vectors (Ogata, 1990).

[1] The approaches that are proposed are not limited to the principle of rationality. In fact, by proposing a generalized, white-box model of ToM, it will be possible to model different policies, objectives, and rationality levels.

[2] Throughout the chapter, for the mathematical notations, italic fonts are used for scalar variables and to distinguish vectors, italic bold fonts are used.

In most real-life applications, it is common to discretize the model given by (2.1) and (2.2) in time, with the corresponding discrete-time model given by

$$\boldsymbol{x}(k+1) = f_d(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{w}(k), \boldsymbol{\theta}_f) \tag{2.3}$$

$$\boldsymbol{y}(k+1) = h_d(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{w}(k), \boldsymbol{\theta}_h) \tag{2.4}$$

with k the discrete time step, and $f_d(\cdot)$ and $h_d(\cdot)$ the discrete version of $f(\cdot)$ and $h(\cdot)$.

A model given by (2.1) and (2.2) (or similarly by (2.3) and (2.4) in the discrete time) can be used via model-based state estimation methods in order to provide information about internal mental states of the human (e.g., in a human-machine interaction context) that are not measured directly via the output. Figure 2.3 illustrates such a model-based state estimator for estimation of the internal mental states of a human (which, according to ToM, manifest themselves in the output, i.e., action of the human, and are thus observable according to the systems theory definition (Ogata, 1990)). A more advanced version of such a state estimator may also be adaptive in the course of the human-machine interactions. More specifically, re-identification of the parameters of the cognitive and decision-making models (i.e., parameters that are represented via θ_f and θ_h) in (2.1)-(2.4) may occur with a frequency lower than the frequency of the operation of the state estimator, i.e., per simulation step. In fact, the adaptive algorithm that performs the re-identification of the parameters may only be activated when necessary (e.g., whenever the estimation error surpasses a certain threshold). Moreover, such models can effectively be used by predictive control methods, e.g., in a loop with an optimizer, to propose a sequence of controlled inputs within a prediction window for the dynamical system that guarantees given requirements on desired performance criteria, and that satisfies imposed and desired constraints.

These two applications, i.e., estimating internal states (see Section 2.2.2) and making controlled inputs (i.e., decisions) that account for their long-term effect on the performance, are highly relevant for SRs that should understand and interact with humans, and will directly contribute to closing the gaps that have been identified in Section 2.1.1. Figures 2.3 and 2.4 illustrate our main idea for using such mathematical dynamic models in estimating the mental states of humans and in enabling SRs to make future-aware decisions in interactions with humans that guarantee the success of these interactions in long terms.

In order to facilitate the formulation, the proposed ToM network representation has been divided into four submodules (see Figure 2.5): (1) *cognition module*, including the internal variables that play a role in the dynamic evolution of the fast-dynamics state variables, (2) *perception module*, including the variables and functions that contribute to the perceived knowledge of the rational agent, (3) *decision-making module*, including rational intention selection and rational action selection, and (4) *world model*, which formulates the influence of the actions of the rational agent on real-life data.

The mental states and thus actions and behaviors of an observed agent are influenced by its fixed (i.e., invariant in short terms) characteristics including the general world knowledge, general preferences, and personality traits (Baker, 2012; Snyder & Ickes, 1985), as well as by its dynamic state variables, including beliefs, goals, and emotions (Dennett, 1987; Saxe & Houlihan, 2017). Moreover, the environmental data of an observed agent is perceived by the agent in a personalized way (Block & Funder, 1986;

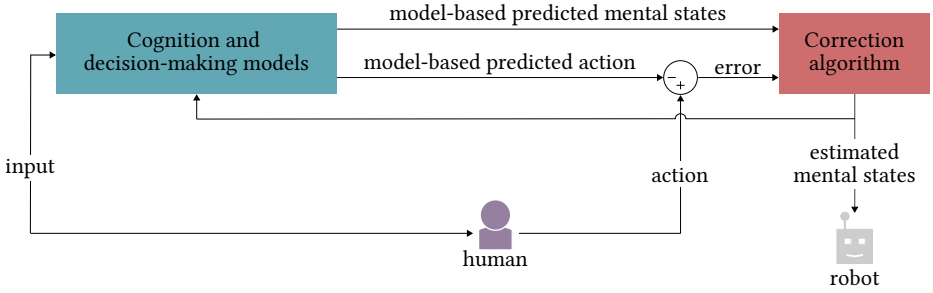


Figure 2.3: Model-based state estimator for making inferences about internal mental states of humans. In this figure, we use the cognitive and decision-making modules of the proposed model to predict internal mental states and action of a human, which correspond to, respectively, $\boldsymbol{x}(t)$ and $\boldsymbol{y}(t)$ in (2.1) and (2.2). the predicted output and the real output, i.e., the action of the human. Finally, the estimated mental states of the human determined by this model-based state estimator are fed back to the model (which closes the loop) and are sent to the observer agent (e.g., a social robot).

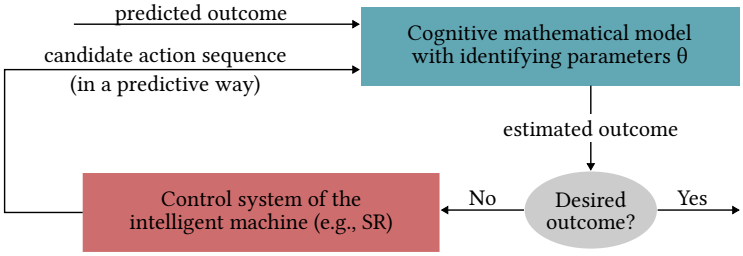


Figure 2.4: Model-based predictive control of intelligent machines that interact with humans.

Witkin, 1949). Next, we explain how the proposed cognitive models incorporate these characteristics.

2.2.2. Network Representation: Identifying Variables and Inter-dynamics

We propose a network representation (see Figure 2.5) of the perception, cognition, and decision-making procedures of humans that is composed of various elements connected via directed links. These links represent the interdependencies and influences of the elements. The oval-shaped elements are used for variables, and the rectangular elements show mathematical functions.

The cognition module in this representation is an extension of the model proposed by Baker (2012) (see section II for details), where in addition to beliefs, goals, general world knowledge, and general preferences, we have included emotions and personality traits, which according to the literature (see the discussions in section II as well as Bono and Vey (2007); Heredia et al. (2022); Kwon et al. (2008); J. J. Lee et al. (2019); Leite et al. (2013); Saxe and Houlihan (2017); Tapus and Mataric (2008)) are highly relevant for the cognition and social interactions of humans. Moreover, in order to

represent the inter-dynamics of all these variables with more transparency, we have introduced two auxiliary variables, called 'bias' and 'perceived knowledge'.

Remark 1 *Goals are immediate desires and needs of rational agents, such as finding food or reaching a location. General preferences of rational agents build up in long terms and remain invariant for long, and in order to identify them several interactions with a rational agent are needed. Examples of general preferences include favorite tastes, friends, and hobbies of a rational agent. Beliefs correspond to* temporary *knowledge or interpretations of rational agents from their world, while general world knowledge consists in* persistent *rationally perceived knowledge, which remains unchanged or is rarely updated. For example, a rational agent believes that a friend who has left an hour ago to fetch a medicine from the drugstore is now in the city center, whereas the exact location of the drugstore is the agent's general world knowledge.*

The perception module, which represents the procedure of transitioning real-life data (i.e., the input of the module) into rationally perceived knowledge (i.e., the output of the module), is composed of perceptual access (i.e., a function that models the observation of the agent), perceived data (i.e., the current representation of the real-life data according to the observation of the agent), and rational reasoning, which is a function that receives perceived data and generates rationally perceived knowledge.

Finally, the decision-making module includes the function "rational intention selection" that outputs the intention(s) of the observed agent, as well as the function "rational action selection", which generates the action of the agent based on the intention(s).

In summary, the main elements represented in Figure 2.5 correspond to:

1. External inputs (e.g., the weather conditions, the actions of a SR) that may influence the mental states and thus actions and behaviors of observed agents.
2. State variables, including the mental states (e.g., beliefs, goals, emotions) and auxiliary variables (e.g., biases) of the observed agent.
3. Fixed parameters, including general world knowledge, general preferences, and personality traits of observed agents.
4. Dynamic processes, which are functions that receive the fixed parameters, current external inputs, and state variables, and update the next step state variables of the observed agent or predict its behaviors.

Note that when a SR interacts with the observed agents, the actions of the SR are the controlled inputs to the model, whereas factors such as the weather conditions are the uncontrollable inputs.

Elements internal and external to an observed agent

While Baker (2012) distinguishes the elements in the cognitive network representation of Figure 2.1 based on whether or not they depend on the current situation, we differentiate the elements of our proposed network representation (Figure 2.5)

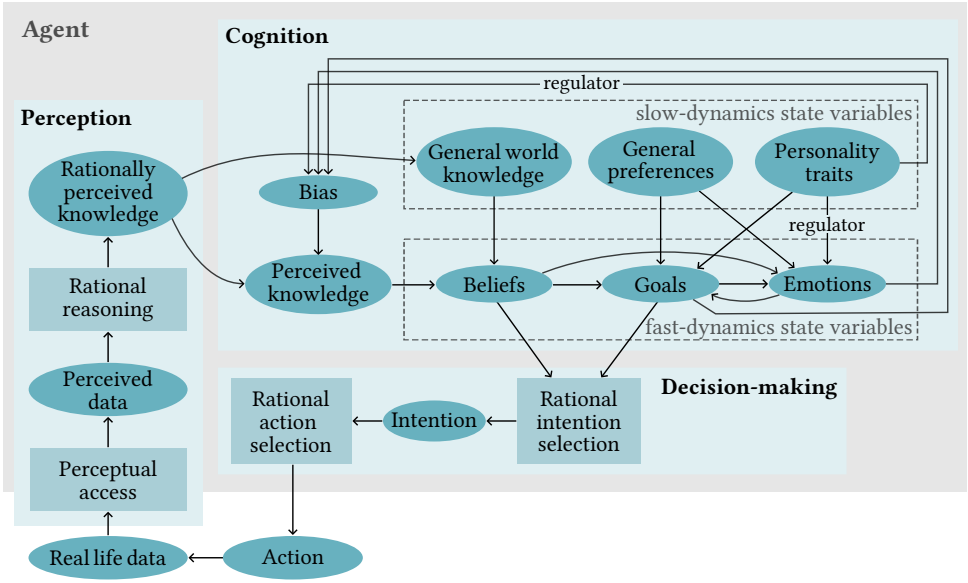


Figure 2.5: Proposed network representation of human perception, cognition, and decision-making procedures of humans including emotions, personality traits, and biases. Oval-shaped elements show input, output, state, and auxiliary variables, and rectangular elements correspond to processes or functions.

considering whether or not they are external to the observed agent. This explicit categorization of internal and external elements provides a model that can be personalized more efficiently than black-box approaches. Moreover, since the internal elements of an observed agent are not visible to the observer agent, their inference is, in general, personalized to an observer agent. Although out of the scope of this chapter, our framework is easily expandable for cases where second-order inferences, i.e., inference about the inference of an observer agent about the observed agent (Baker et al., 2008), are of interest. In addition to fully internal and fully external elements, our representation incorporates partially external elements. In particular, the perceptual access of an observed agent is partially external, given that this process is influenced by external inputs, and is partially internal, since it is shaped by internal characteristics of the individual. This dual influence on perceptual access is explained in detail in Section 2.2.2.

Fast-dynamics and slow-dynamics state variables

The state variables of the proposed model are distinguished according to their relevance for the duration (i.e., short-term or long-term) of interactions between two rational agents and to the frequency of their dynamics. Consequently, two categories of state variables are defined: (i) Fast-dynamics state variables, which may constantly vary (with a timescale in the range of seconds or minutes) as a response to specific situations the observed agent faces. Goals, beliefs, and emotions in Figure 2.5 are fast-dynamics state variables. (ii) Slow-dynamics state variables, which vary according

to large time scales (months or years). General world knowledge, general preferences, and personality traits in Figure 2.5 are slow-dynamics state variables. Fast-dynamics state variables are more relevant for short-term interactions, while slow-dynamics state variables become more relevant throughout long-term interactions, when the fixed or repetitive patterns of cognitive procedures resulting from these slow-dynamics state variables provide extra information for the observer agent to make more precise estimates and predictions (Snyder & Ickes, 1985).

Remark 2 *Slow-dynamics state variables may influence the evolution of fast-dynamics state variables (see Example 1-Example 4 in Appendix A), while the opposite is not necessarily true (especially in short terms). The main aim of this chapter is to formalize and formulate the evolution of fast-dynamics state variables. Modeling the evolution of slow-dynamics state variables is out of the scope of this chapter. Thus, slow-dynamics state variables are mainly considered as fixed parameters.*

In the next sections, we explain how the inter-dynamics of the cognition, perception, and decision-making modules of the proposed model, shown in Figure 2.5, have been developed based on literature and real-life examples given in Appendix A.

Perception module

In the model proposed by Baker et al. (see Baker et al. (2011) and Baker (2012)), the perception procedure that generates the beliefs is represented by a single element called the principle of rational belief (see Figure 2.1). This simplification was shown to be sufficient to explain the relationship between the environmental inputs and the inferred beliefs in the simple environments and scenarios considered in Baker et al. (2011). In real-life scenarios, however, a more complicated procedure occurs before a belief is developed based on the raw real-life data. More specifically, rational agents may (deliberately or indeliberately) access and perceive only a portion of the real-life data. For instance, in the same environment, different rational agents may notice different types of data (Witkin, 1949) (e.g., one may notice a sound, while another agent filters it out). Moreover, rational agents may receive partial data due to external constraints (e.g., missing visual data due to occlusion). Therefore, observed agents may hold false or inaccurate beliefs (cf. the Sally-Anne experiment (Baron-Cohen et al., 1985)), which is essential for ToM-based observer agents to recognize (Fumero & Ahmetoglu, 2008; Rabinowitz et al., 2018; Wellman et al., 2001). Furthermore, the same perceived data can be differently interpreted by different rational agents (Block & Funder, 1986; Zelenski, 2007).

To address these aspects, in our proposed model, the perception process that transforms real-life data into beliefs is decomposed into smaller, well-defined subprocesses i.e., *perceptual access* and *rational reasoning* (see Figure 2.5 and Examples 9 and 10 in Appendix A). Real-life data from the environment is perceived via *perceptual access*, which, as discussed above, is partially personalized and partially depends on the environment. Thus, the rectangular element corresponding to the perceptual access function in Figure 2.5 is located partially inside the agent box and partially in the environment. The perceived data is then processed via *rational*

reasoning, which, as opposed to the universal principle of rational reasoning applied by Baker et al. (2011) and Baker (2012), is specific to a rational agent. Accordingly, the rational agent makes a judgment called the *rationally perceived knowledge*, which is then transformed into a belief or a piece of general world knowledge in the cognitive module.

Cognitive module

The cognitive module receives the rationally perceived knowledge as input. Since, compared to the model of Baker shown in Figure 2.1, the cognitive module of our model includes emotions and personality traits as well, we first discuss the inter-dynamics of the emotions with the other fast-dynamic state variables. Next, we explain the influence of personality traits on the fast-dynamics state variables.

Emotions of a rational agent are stimulated by its beliefs (Ekman & Cordaro, 2011; Scherer, 2009), where goals and general preferences, alongside beliefs, can also stimulate emotions (Scherer, 2009): On the one hand, when a rational agent develops a belief that is in line with the fulfillment of its goal, positive emotions may be stimulated (see Example 6 in Appendix A). On the other hand, when the agent develops a belief that hinders the chances of fulfilling its goal, negative emotions may be stimulated. Similarly, when a general preference is supported by a developed belief, positive emotions can be generated, while beliefs that conflict with the general preferences may result in negative emotions (see Example 7 in Appendix A). We assume that direct influences from general world knowledge on emotions are negligible, since when a piece of general world knowledge is relevant in the current context, it generates a belief (see Figure 2.5), and this belief can then influence emotions.

Studies show that emotions can affect the goals of rational agents (Andrade & Ariely, 2009; George & Dane, 2016; Heredia et al., 2022; Lerner et al., 2015; Raghunathan & Pham, 1999). For instance, gratitude can galvanize rational agents into helping others (Lerner et al., 2015), or anxiety may trigger rational agents to avoid stressful situations (Raghunathan & Pham, 1999). More specifically, emotions may result in the development of a goal that contradicts with general preferences of the rational agent, or in the change of a goal that was previously made by the agent. The influence of emotions on goals is shown in Figure 2.5 via a directed link (also see Example 8 in Appendix A). While emotions do not directly influence beliefs, they can affect the processes that result in judgements or beliefs of rational agents (Andrade & Ariely, 2009; Raghunathan & Pham, 1999; Zelenski, 2007). Thus, we have introduced the auxiliary variables "bias" and "perceived knowledge" in order to illustrate this influence. For example, positive emotions may introduce optimistic biases into the process of generation of new beliefs, whereas negative emotions may lead to the formation of overly pessimistic beliefs (Lerner et al., 2015). Therefore, the influence of emotions on the development of beliefs will be introduced into the proposed cognitive model via bias. Similarly to emotions, goals of rational agents may introduce some bias into their rational reasoning processes (Mele, 1997), where the intensity of the bias may depend on the personality traits of rational agents (Zelenski, 2007) (see Example 12 in Appendix A). For the purpose of describing the cognition of an observed agent, the bias is introduced into the belief after the rational reasoning process is executed (see

Figure 2.5). This results in the *perceived knowledge* (as opposed to *rationally* perceived knowledge), which generates the biased beliefs.

Regarding the influences of personality traits on fast-dynamics state variables, there is no evidence that beliefs are directly affected by personality traits, whereas goals may directly be affected by personality traits (Snyder & Ickes, 1985) (see Figure 2.5): For instance, while the goal of an introverted rational agent is to avoid strangers, the goal of an extroverted rational agent is to make new friends. Moreover, while emotions are not generated by personality traits (Bono & Vey, 2007; Zelenski, 2007), they are regulated by personality traits. In fact, personality traits of rational agents may determine the extent to which certain beliefs affect their emotions (Bono & Vey, 2007; Canli et al., 2001; Scherer, 2009). For instance, extroverted individuals are more likely to experience positive and intense emotions when under the same stimuli (Zelenski, 2007).

Remark 3 *In summary, beliefs, either alone or in combination with generated goals or general preferences, and boosted or hindered by personality traits, trigger the emotions. For the sake of brevity, we use the following terminology:* ***emotion trigger*** *1,* ***emotion trigger*** *2, and* ***emotion trigger*** *3 for, respectively, a combination of beliefs and general preferences, solely beliefs, and a combination of beliefs and goals, triggering emotions.*

Remark 4 *Note that, as it is demonstrated in Figure 2.5, the rationally perceived knowledge that is the input of the cognitive module can be transformed into a belief, or if this piece of knowledge does not change in the short term, into a general world knowledge (Chella & Pipitone, 2020; Laird et al., 2017).*

Decision-making module

Based on the BDI theory, we introduce the rational intention selection procedure (see the rectangular element in Figure 2.5), which is an intermediate process in the generation of actions and behaviors of rational agents. This process generates or selects an intended high-level action that maximizes the chance of fulfillment of the goals of the rational agent, given its beliefs. In other words, mathematically, rational intention selection is a function, with beliefs and goals as inputs and intentions as outputs. The generated intention can then be translated into lower-level actions by the rational action selection function. For example, the intention of the agent may be to drink a glass of water, where one action is to open the cabinet where the glasses are kept.

2.2.3. Dynamic Mathematical Representation of Perception, Cognition, and Decision-making

The main aim of this chapter is to model the dynamic evolution of the fast-dynamics state variables for the network representation given in Figure 2.5, and to predict the resulting action. Thus, we need to develop mathematical formulations for the perception, cognition, and decision-making modules given in Figure 2.5. Note that a mathematical model that represents perception receives real-life data as input and generates rationally perceived knowledge as output. This output is the input

to the mathematical model of cognition, which generates two outputs, beliefs and goals. Finally, these two outputs enter the decision-making module as input, and the corresponding model generates action as output.

Previous works (Baker et al., 2011; Baker et al., 2017; Jara-Ettinger et al., 2016; J. J. Lee et al., 2019; Saxe & Houlihan, 2017) mainly use Bayes' theorem to describe human cognition. Modeling cognition under this assumption corresponds to representing the cognitive network representation as a Bayesian network (Heckerman, 2008). However, this requires knowing beforehand the probability functions of all the variables that are part of the network. Furthermore, the integration of a probabilistic-based formulation of the model with model-based predictive approaches would require the use of decision trees, which tend to grow exponentially with the size of the prediction horizon. Therefore, this formulation cannot be integrated with model-based predictive methods in real-life applications.

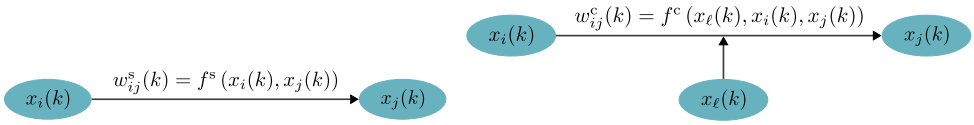
Alternatively, FCMs (Kosko, 1986) represent concepts and variables that correspond to complex and/or uncertain systems and their interlinks and interactions. Contrarily to Bayesian networks, FCMs support cyclic connections (Stylios & Groumpos, 2004), which is very relevant for modeling cognitive procedures of humans (cf. Figure 2.5). Moreover, concepts or variables in an FCM can be represented mathematically by fuzzy variables, which excellently fit the concepts that are involved in human cognitive procedures (e.g., beliefs, goals, emotions, etc.). Therefore, in this chapter, based on the idea of FCMs, we propose an extended FCM representation and use it to mathematically formulate the cognitive module of our proposed ToM model.

Novel FCM-based framework for transforming static network representations into dynamic mathematical models

In an FCM, we denote the vector of all the n state variables that describe the system at time k by $\boldsymbol{x}(k)$, and the i^{th} state variable by $x_i(k)$ with $i = \{1, ..., n\}$. This vector for the cognitive module of our proposed model includes the fast-dynamics and slow-dynamics state variables, and the auxiliary variables. Mathematically, x_i (e.g., emotion) can be represented as a fuzzy variable in an FCM, with $\tilde{X}_{i,j}$ a possible realization that is mathematically given by a fuzzy set[3]. Fuzzy variables (Zadeh, 1975) are used to mathematically represent concepts that are imprecise, vague, and mainly given by human linguistic terms. Fuzzy sets are a generalization of regular (crisp) sets; while an element either belongs to or does not belong to a crisp set (i.e., its membership degree to the crisp set is either 1 or 0), it may partially belong to a fuzzy set, with its membership degree varying within the interval $[0, 1]$ (Zadeh, 1988). In order to represent fuzzy sets and perform mathematical operations on them, their corresponding membership functions are used (Zadeh, 1988). The fuzzy sets $\tilde{X}_{i,j}$ should cover the domain $\mathbb{X}_i$ of x_i.

The influence of variable x_i over variable x_j in an FCM is represented by a directed line called a *linkage* (see Figure 2.6a). In the classic FCM formulation, every linkage between x_i and x_j is characterized by a weight $w_{ij} \in [-1, 1]$ that reflects the level of influence of variable x_i over variable x_j. Whenever w_{ij} is positive (negative), an

[3]In this chapter, to distinguish fuzzy sets from crisp sets, we use a tilde symbol.



(a) Simple Linkage. The horizontal arrow is a simple linkage, shown by (i, j).

(b) Complex Linkage. The vertical arrow is a side linkage, shown by (ℓ, i, j).

Figure 2.6: Simple and complex linkage.

increase of x_i implies an increase (a decrease) of x_j, and the larger the absolute value of w_{ij}, the larger the influence of x_i over x_j (whenever w_{ij} is null, changes in variable x_i do not influence variable x_j).

In FCMs, the weights w_{ij} are considered to be constant. However, in order to accurately model most real-world systems with FCMs, variable weights may be required (Carvalho & Tomé, 2001; Mourhir et al., 2016). For instance, in rule-based FCM (Carvalho & Tomé, 2001) the values of the weights depend on the realized values of the causing variable x_i. In our proposed cognitive network representation, due to the mutual influences of the fast-dynamics state variables on one another, the value of weight w_{ij} may vary in time and, in general, depend on the realized values of the causing variable $x_i(k)$, affected variable $x_j(k)$, or other intermediate variables $x_\ell(k)$. To address this, we consider weights as functions of the causing, affected, or intermediate variables and accordingly define *simple linkages*, *side linkages*, and *complex linkages*, which are explained next.

The linkage (i, j) that directly connects two variables $x_i(k)$ and $x_j(k)$ and is not influenced by any intermediate variables (see Figure 2.6a) is called *simple linkage*. A *side linkage* (ℓ, i, j) (see the vertical arrow in Figure 2.6b) corresponds to the directed influence of an intermediate variable $x_\ell(k)$ over a linkage (i, j) that connects variables $x_i(k)$ and $x_j(k)$. The group of a linkage that is influenced by one or several side linkages and all its side linkages is called a *complex linkage* (see Figure 2.6b). The set of all pairs (i, j) corresponding to simple linkages is given by $\mathbb{L}$ and the set of all trios[4] (ℓ, i, j) that correspond to complex linkages is given by $\overline{\mathbb{L}}$. Note that $\mathbb{L}$ and $\overline{\mathbb{L}}$ are complementary sets, and have no common element. The weight $w_{ij}^{s}(k)$ of a simple linkage and $w_{ij}^{c}(k)$ of a complex linkage for time step k are computed via, respectively, function $f^{s} : \mathbb{X}_i \times \mathbb{X}_j \to [-1, 1]$ and function $f^{c} : \mathbb{X}_i \times \mathbb{X}_j \times \mathbb{X}_\ell \to [-1, 1]$. We have

$$w_{ij}^{s}(k) = f^{s}(x_i(k), x_j(k)), \quad \forall i, j \text{ for which } (i, j) \in \mathbb{L} \tag{2.5}$$

$$w_{ij}^{c}(k) = f^{c}(x_\ell(k), x_i(k), x_j(k)), \quad \forall i, j \text{ for which } \exists \ell \text{ such that } (\ell, i, j) \in \overline{\mathbb{L}} \tag{2.6}$$

for all $k \in \{1, 2, \ldots\}$, and $x_i(k) \in \mathbb{X}_i$, $x_j(k) \in \mathbb{X}_j$, and $x_\ell(k) \in \mathbb{X}_\ell$ are elements of the state vector $\boldsymbol{x}(k)$.

[4] In our cognitive network representation given in Figure 2.5, complex linkages include no more than one side linkage. Thus, for the brevity of notations, we talk about pairs and trios only.

Then, the dynamic equation for updating variable $x_j(k)$ that evolves per time step k within the proposed extended FCM is formulated by

$$\begin{aligned} x_j(k+1) = &\sum_{(i,j)\in\mathbb{L}} f^{\mathrm{s}}\Big(x_i(k), x_j(k)\Big)x_i(k) + \\ &\sum_{(i,j,\ell)\in\overline{\mathbb{L}}} f^{\mathrm{c}}\Big(x_\ell(k), x_i(k), x_j(k)\Big)x_i(k) + f_j\Big(x_j(k)\Big) \end{aligned} \tag{2.7}$$

where function $f_j(\cdot)$ determines the influence of variable $x_j(k)$ on its evolved value $x_j(k+1)$.

Next, we describe the mathematical representations of the perception, cognition, and decision-making modules. Since, for the sake of compactness of the mathematical notations, we use a number of abbreviations in the mathematical notations of the following sections, Table 2.1 represents the meaning of these mathematical notations.

Table 2.1: Mathematical notations used in (2.8)-(2.13).

Mathematical notation	Process or variable
f_{PA}	perceptual access
f_{RR}	rational reasoning
d_{RL}	real-life data
d_{p}	perceived data
κ_{RP}	rationally perceived knowledge
f_{RIS}	rational intention selection
f_{RAS}	rational action selection
x_{B}	belief
x_{G}	goal
i	intention
a	action

Mathematical representation of perception

The perception procedure includes two functions (see Figure 2.5), i.e., perceptual access (shown by $f_{\mathrm{PA}}(\cdot)$) and rational reasoning (shown by $f_{\mathrm{RR}}(\cdot)$). Perceptual access receives real-life data (which we show by d_{RL}) and generates perceived data (which we show by d_{P}). Thus, at time step k we have

$$d_{\mathrm{p}}(k) = f_{\mathrm{PA}}(d_{\mathrm{RL}}(k)) \tag{2.8}$$

Similarly, rational reasoning receives perceived data and generates rationally perceived knowledge (shown by κ_{RP}). Thus, at time step k we have

$$\kappa_{RP}(k) = f_{\mathrm{RR}}(d_{\mathrm{p}}(k)) \tag{2.9}$$

Hence, from (2.8) and (2.9), the mathematical representation for the perception module is given by

$$\kappa_{RP}(k) = f_{\text{RR}}(f_{\text{PA}}(d_{\text{RL}}(k))) \tag{2.10}$$

Mathematical representation of cognition

For the cognitive module of our proposed model, (2.7) may be used with $x_j(k) \in \{\text{current beliefs, current goals, current emotions, current biases}\}$, $x_i(k) \in \{\text{current beliefs, current goals, current emotions, current biases, general world knowledge, general preferences, personality traits, current perceived knowledge}\}$, and $x_\ell(k) \in \{\text{current goals, general preferences, personality traits}\}$. In fact, using (2.7) will provide us with the desired state space representation (2.3), where the current "perceived knowledge" acts as input $\boldsymbol{u}(k)$, and the weights computed via $f^s(\cdot)$ and $f^c(\cdot)$, as well as the "general world knowledge", the "general preferences", and the "personality traits", act as the parameters in $\boldsymbol{\theta_f}$. Note that in this case there are no particular external disturbances $\boldsymbol{w}(k)$ and $f_d(\cdot)$ in (2.3) has been represented as additive functions in (2.7).

Remark 5 *In order to define functions $f^s(\cdot)$, $f^c(\cdot)$, and $f_j(\cdot)$ for the proposed cognitive models, different approaches may be used, such as using explicit mathematical representations or describing these functions via Fuzzy Inference Systems (FISs) (see, e.g., Carvalho and Tomé (2001)). When we use explicit expressions for $f^s(\cdot)$, $f^c(\cdot)$, and $f_j(\cdot)$, in order to use fuzzy variables for the extended FCM, we should in general use Zadeh's extension principle (Zadeh, 1975). In case this is (computationally or analytically) not possible, these functions can be modeled via FISs (Carvalho & Tomé, 2001).*

Mathematical representation of decision-making

The decision-making procedure is composed of two functions, rational intention selection $f_{\text{RIS}}(\cdot)$, and rational action selection $f_{\text{RAS}}(\cdot)$. The intention selection receives the belief $x_{\text{B}}(k+1)$ and goal $x_{\text{G}}(k+1)$ predicted for the next step and generates the intention of the rational agent $i(k+1)$ that is predicted also for the next step. Thus, for time step $k+1$ we have

$$i(k+1) = f_{\text{RIS}}(x_{\text{B}}(k+1), x_{\text{G}}(k+1)) \tag{2.11}$$

Similarly, the rational action selection receives the intention that is predicted for the next time step, and generates the predicted action for the next time step $k+1$. We have

$$a(k+1) = f_{\text{RAS}}(i(k+1)) \tag{2.12}$$

Thus, the decision-making is mathematically given by

$$a(k+1) = f_{\text{RAS}}(f_{\text{RIS}}(x_{\text{B}}(k+1), x_{\text{G}}(k+1))) \tag{2.13}$$

where $a(k+1)$ is the predicted action of the rational agent for time step $k+1$ based on the observations, measurements, and estimations at time step k. Moreover, $x_{\text{B}}(k+1)$ and $x_{\text{G}}(k+1)$ are the predicted next belief and next goal.

Remark 6 *For predictive decision-making by, e.g., a SR, with a prediction horizon bigger than 1, the influence of the actions on real-life data should also be modeled (see Figure 2.5). This, however, is out of the scope of this chapter, since it concerns a modeling of the environment of the rational agent.*

2.3. Case study

The ToM model proposed in this chapter (see Figure 2.5) was implemented via Python and was used to estimate the expected perception, cognition, and decision-making procedures of human participants in virtual scenarios of emergency evacuation. The experiments were designed to assess the following research questions:

1. How well does the proposed model predict the behavior and estimate the mental states of rational agents, particularly humans, in a 2-dimensional environment (similarly to the work in Baker et al. (2011) and Rabinowitz et al. (2018))?

2. How much do the cognitive elements (emotions, biases, and personality traits) that were proposed in this chapter contribute to improving such predictions?

3. How much do the models that were identified for each participant provide a relevant insight about the general preferences of the participants?

In particular, we answer these research questions for our model (which, in the rest of the case study, we call ToM-I) in comparison with a model that did not include the elements that were proposed in this chapter (which we call ToM-II and is illustrated in Figure 2.7). Since the personality trait is an adaptable parameter, ToM-II has one less parameter than ToM-I. To assess the contribution of the emotions and biases independently of the number of adaptable parameters, a third model, ToM-III, that is similar to ToM-I but does not include the personality traits, was also considered.

2.3.1. Experiment Setup

While Baker et al. (2011) and Rabinowitz et al. (2018) artificially generated the behavior of the observed agents to use as ground truth, we used the observed behavior of human participants in order to generate the ground truth behavior of the observed agents. This is because our main objective for answering research question (1) is to assess how well the proposed model estimates the mental states and behavior of real humans. Hence, the performance of our ToM model, ToM-I, was assessed based on how accurately it estimated the goals and predicted the behavior of the participants. Moreover, in order to answer research question (2), i.e., to assess the particular contribution of the elements emotions, biases, and personality traits that have been added to our model, ToM-I was compared with the model proposed by Baker (2012). For a fair comparison, the three models were formulated as described in section 2.2.3.

The experiments were performed via a computer simulation, with 21 human participants. The participants were aged between 14 and 32 years old. Every participant received an executable of a set of simulations that they could run on their own computer. In this set of simulations, participants were able to control the

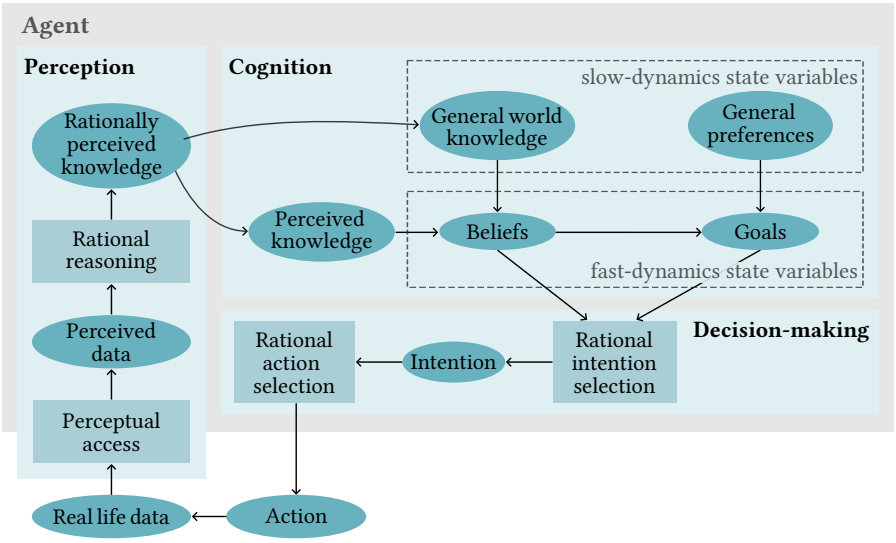
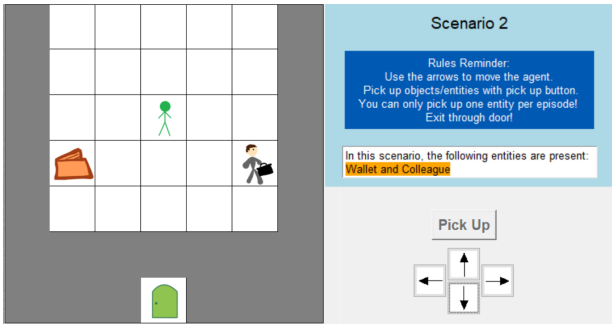
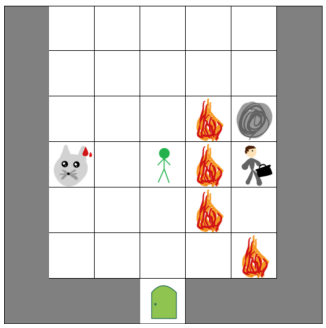


Figure 2.7: Baseline model used for comparison in the experiments, ToM-II. This baseline model is similar to the model represented in Figure 2.5, but it does not include the emotions, biases, and personality traits.

movements, actions, and decisions of a virtual rational agent in 16 simulated scenarios. The decisions taken by each participant were recorded into an Excel file that was sent back to the researchers. The aforementioned scenarios consisted of emergency evacuation scenarios that included a closed space that was populated by living beings (e.g., pets, humans) and objects (e.g., personal belongings). The participants were able to move the rational agent around and decide which entity they wanted to save in each scenario (see Figure 2.8). To increase the diversity in the general preferences, the scenarios included three types of entities: a dependent being (e.g., a child or a pet), a human colleague, and a personal object (e.g., a wallet or a set of keys). In order to make the scenarios more realistic and to increase the chance that the general preferences had a broad span of values, at the start of the experiment, each participant chose which entity they related to most in each category from a predefined set of options. During the simulation, the participants were able to save an entity by picking it up (see Figure 2.8a) and going to the exit (shown as a green gate in Figure 2.8a). Alternatively, the participants could go to the exit without saving any objects or beings. Once the rational agent reached the green door, the scenario terminated. In some scenarios, the entities were injured or surrounded by fire (see Figure 2.8b). These variations were added to induce emotions of fear in the participants for the entities that were at risk. Note that at every simulation step (i.e., after each action of the rational agent), the participants could only observe the closest positions to the rational agent and the previously visited positions. In order to decrease the exploring movements by the participants, the position of the exit was always visible.



(a) A screenshot of a simulated scenario that a participant can perform: On the left-hand side of Figure 2.8a, the participant sees the simulated emergency evacuation scenario, including various entities (e.g., a colleague and a wallet). On the top right-hand side of Figure 2.8a, the participant sees reminders about the main rules of the simulation. At the bottom right-hand side of Figure 2.8a, the participant can use the buttons to move the rational agent and to pick up objects or beings.



(b) An example of a scenario where one of the entities (the colleague) is surrounded by fire and another entity (the pet) is injured.

Figure 2.8: Examples of emergency evacuation scenarios used in our case study.

2.3.2. Implementation Setup

For the perception module, the real-life data contained the inputs from the two-dimensional environment, including the positions of all entities, the rational agent, and the obstacles, as well as the characteristics of the entities (e.g., if they are injured, or if they are a living being or an object). The perceptual access was modeled based on the field of view of the rational agent per simulation step. The field of view of the rational agent was assumed to cover an area of five by five cells. The cells that fell within the current field of view of the rational agent, as well as the cells that were previously in the field of view of the rational agent, were visible to both the participant and the corresponding ToM model (as it is also shown in Figure 2.8). The perceived data was based on the real-life data corresponding to the visible positions, but did not include any data from the unseen positions. The rationally perceived knowledge was deduced according to logical propositions and based on the perceived data. For example, the location of the fire and the current position of the rational agent were used to deduce a rationally perceived knowledge about whether or not the rational agent was near the fire.

Regarding the cognitive module, since humans express their mental states mainly via quantified expressions (e.g., relatively happy), the fast-dynamics state variables of the cognitive module (e.g., beliefs, goals, emotions) were represented by fuzzy variables, i.e., verbal terms were associated with the realizations of each variable. These variables in the experiments corresponded, for example, to the belief of the rational agent about an entity being injured, or to the goal of the rational agent to save one entity, or to the emotion of fear. As for the slow-dynamics state variables, we considered the preference toward each of the entities (i.e., the general preferences), and the altruism of the agent (i.e., a personality trait). The full list of all the fuzzy variables, and the corresponding linguistic terms and range values, can be found in Morão Patrício

and Jamshidnejad (2023b). The linkages that connected the elements of the cognitive module were mathematically represented by FISs (see Remark 5 for details).

As for the decision-making module, the intention of the rational agent was computed based on the goal fired with the highest value at each simulation step. The action of the rational agent was computed as the movement of the rational agent along the shortest path (according to the Dijkstra's Algorithm) to the entity that was associated with the current intention of the rational agent, except for when the position of the rational agent and the position of the entity were the same: in that case, the action would be to pick up the object or the being.

2.3.3. Training Phase for the Models

A proportion of the data that was collected from each participant was used to personalize the model to that participant. In fact, the data that was collected in 75% of the scenarios was used to train the ToM model, and the remaining was used to test the accuracy of the model in predicting the behavior of the agent in unseen scenarios. The personalization consisted in identifying the values of the slow-dynamic state variables (i.e., the general preferences toward each of the entities for ToM-I, ToM-II, and ToM-III, as well as the personality trait altruism for ToM-I) for each participant. The slow-dynamic state variables are also called the model parameters due to their very low update frequency. These parameters $\boldsymbol{\theta}$ were optimized using a grid search algorithm with a step of 0.1. The number of scenarios per participant was only 16, where this number was selected to provide sufficient data for training the models, while preventing the participants from getting distracted or tired, resulting in inaccurate answers. Hence, in order to increase the reliability of the results, the combination of scenarios that were used for training and for testing were shuffled in order to create 4 different combinations. Running one iteration of the model using an Intel CPU (i7-1185G7 3.0GHz, single thread implementation) took 0.20554 s on average. Furthermore, around 8 hours per participant were required to train the ToM-I model on a super computer with 48 cores (2x Intel XEON E5-6248R 24C 3.0GHz).

The action of the participants per simulation step included moving the rational agent to one of its (allowed) neighboring cells or picking up/rescuing an object/being. This action depended on two decision levels: (1) whether or not the rational agent decided to rescue a particular being or pick up a particular object; (2) which path the rational agent followed to approach the corresponding being/object or the exit. Therefore, for identifying the models per participant, for the training sets, two independent optimizations were performed in order to train and assess the models for two types of predictions: the prediction about the entity that the rational agent would save in each scenario (i.e., predicting the overall intention of the rational agent per scenario), as well as the prediction of the action that would be taken by the rational agent per simulation step. In order to train and assess the models regarding the first prediction, per simulation step the models were run, and the predicted actions of the rational agent were executed consecutively, until the exit was reached. The saved entity, as predicted by each model, was compared with the entity that was really saved by the participant who steered the rational agent in that scenario (see Figure 2.9a). In order to train and assess the models regarding the second prediction, the actions that were

chosen by the participant in each simulation step were compared with the actions that were predicted via each model for that simulation step (see Figure 2.9b).

Equation (2.14a) shows the cost function used to train and assess the models with respect to the prediction of the overall intention for participant p throughout a set $\mathcal{S}$ of training or test scenarios, where $i_{p,s}$ is the intention of participant p for scenario s that is predicted via the model, and $i^*_{p,s}$ is the intention of participant p in scenario s. Both $i_{p,s}$ and $i^*_{p,s}$ take one out of the four following realizations: save one of the three entities (i.e., the dependent being, the colleague, or the personal belonging), or directly move to the exit. The cost function used to train and assess the prediction of the step-wise actions of the rational agent for participant p is given by (2.14b), where $c_{k,p,s}(\boldsymbol{\theta})$ is 0 if the action taken by the participant p in scenario s at simulation step k is one of the predicted actions by the model for the same simulation step, and is 1 otherwise. Moreover, $n_{\text{steps},p,s}$ is the total number of the simulation steps taken by participant p in scenario s. For $p = 1, \ldots, n_\text{p}$ (with the number of participants $n_\text{p} = 21$ in our case study), we have

$$J_{\text{int},p}(\boldsymbol{\theta}) = \sum_{s\in\mathcal{S}} J_{\text{int},p,s} \quad \text{with } J_{\text{int},p,s}(\boldsymbol{\theta}) = \begin{cases} 0, & \text{if } i_{p,s} = i^*_{p,s} \\ 1, & \text{otherwise} \end{cases} \tag{2.14a}$$

$$J_{\text{act},p}(\boldsymbol{\theta}) = \sum_{s\in\mathcal{S}} J_{\text{act},p,s} \quad \text{with } J_{\text{act},p,s}(\boldsymbol{\theta}) = \frac{1}{n_{\text{steps},p,s}} \sum_{k=1}^{n_{\text{steps},p,s}} c_{k,p,s}(\boldsymbol{\theta}) \tag{2.14b}$$

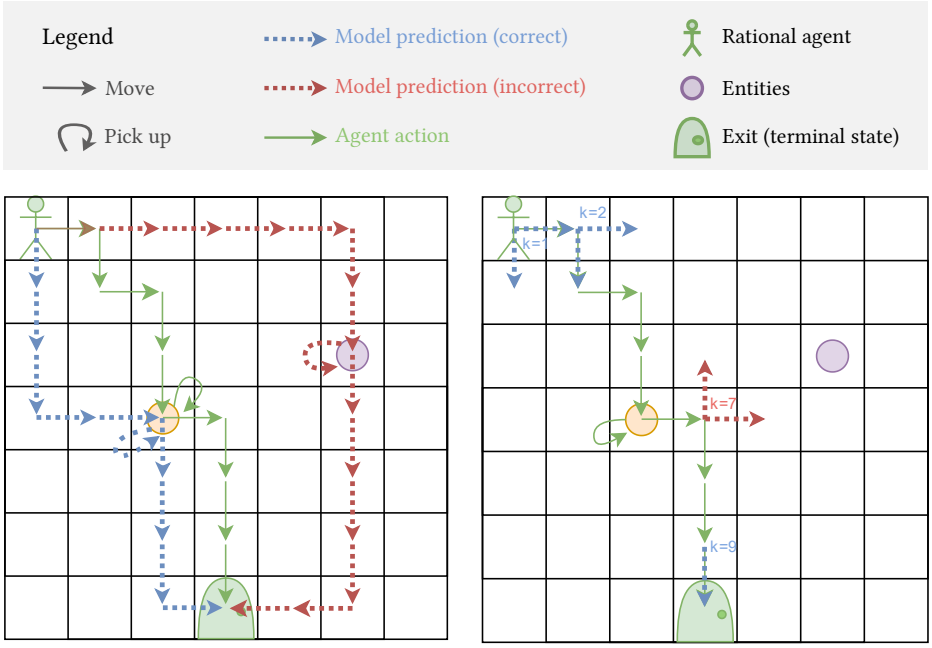
Note that for both optimization procedures, $\boldsymbol{\theta}$ is the optimization variable, which includes all the identification parameters of the corresponding model.

2.4. Results and Discussions

In order to assess the performance of our model and the influence of our contributions, the average optimal costs achieved by each one of the three models for the test phase were compared. The optimal costs $J^*_{\text{int},p}$ and $J^*_{\text{act},p}$ that are achieved by each model for participant p are obtained by evaluating equations (2.14a) and (2.14b) using the parameters $\boldsymbol{\theta}^*$ that have been obtained by minimizing these cost functions in the training phase. The average optimal costs $\bar{J}^*_{\text{int}}$ and $\bar{J}^*_{\text{act}}$ for each trained model are an average of the aforementioned costs $J^*_{\text{int},p}$ and $J^*_{\text{act},p}$ across all the n_p participants and are divided by the number of the scenarios, as it is given by (2.15) for the average intention cost $\bar{J}^*_{\text{int}}$. Given that the costs computed via (2.14) are normalized, the costs computed via (2.15) are also normalized, and we report them in percentage.

$$\bar{J}^*_{\text{int}} = \frac{1}{n_\text{s}} \cdot \frac{1}{n_\text{p}} \sum_{p=1}^{n_\text{p}} J^*_{\text{int},p}(\boldsymbol{\theta}^*) \tag{2.15}$$

The average optimal cost $\bar{J}^*_{\text{act}}$ for the action prediction is calculated similarly to $\bar{J}^*_{\text{int}}$. The average optimal costs obtained both in training and in test conditions when the models estimated the intention of the participants and predicted their actions are shown, respectively, in Tables 2.2 and 2.3 for the four combinations of training and test scenarios.



(a) Predictions by the models regarding the overall intention of the rational agent, where these predictions are considered correct if the model predicted a trajectory in which the agent chose the same intentions as the participants (i.e., either to save a specific entity or to go directly to the exit). The dashed arrows in blue represent an accurate prediction of the overall intention (i.e., the rational agent rescued the orange entity, which was also rescued by the participant in this scenario, and then went to the exit), while the dashed red arrows represent a wrong prediction about the intention of the participant via the models (i.e., the rational agent rescued the purple entity, which was different from the orange entity that was in reality rescued by the participant in this scenario, and then went to the exit).

(b) Predictions by the models regarding the action(s) of the rational agent for different simulation steps. In the first two simulation steps (specified by $k = 1$ and $k = 2$), correct predictions are shown by the dashed blue arrows. Since the model estimates that the rational agent wants to save the orange entity, any of the two predicted actions shown in the figure for the first simulation step (i.e., moving downwards or moving to the right) are equally likely to be chosen by the agent. An incorrect prediction is shown via dashed red arrows for the simulation step $k = 7$, where the model estimates the optimal actions to be moving toward the purple entity, while the participant wants to go to the exit.

Figure 2.9: Graphic representation of the two types of predictions made by the models.

Regarding the estimations of the intentions of the participants, we first focus on the comparison between ToM-I and ToM-II, and we subsequently focus on comparing ToM-I with ToM-III. ToM-I obtained the best performance, i.e., the lowest estimation error compared to ToM-II in both the training and the test phases for all the four combinations of training and test scenarios (compare the average costs given in the last row of Table 2.2 for ToM-I (see column 2 and column 5) with those for ToM-II). On average, ToM-I achieved an error of 6.94% on the training scenarios and of 18.45% on the test scenarios, representing an improvement of 8.63% in training conditions and of 6.25% in test conditions when compared to the performance of ToM-II. Although ToM-I had one more parameter to be optimized compared to ToM-II, it was possible to establish that the improvement of the performance of ToM-I was not uniquely

Table 2.2: Average minimum cost achieved by each model when predicting the intentions of each participant. The smaller this minimum cost value, the more satisfactory the performance of the corresponding model.

Model	Training phase			Test phase		
	ToM-I	ToM-II	ToM-III	ToM-I	ToM-II	ToM-III
Combination 1	7.14	15.48	10.32	20.24	34.52	22.62
Combination 2	6.75	13.49	10.71	16.67	19.05	17.86
Combination 3	5.56	15.08	7.94	21.43	25.00	23.81
Combination 4	8.33	18.25	11.11	15.48	20.24	17.86
Average	6.94	15.57	10.02	18.45	24.70	20.54

Table 2.3: Average minimum cost achieved by each model when predicting the actions of each participant.

Model	Training phase			Test phase		
	ToM-I	ToM-II	ToM-III	ToM-I	ToM-II	ToM-III
Combination 1	31.81	34.97	33.44	34.56	36.52	35.13
Combination 2	32.90	34.70	34.34	32.15	37.26	32.95
Combination 3	31.84	35.38	33.32	33.90	36.16	34.50
Combination 4	32.06	35.31	33.37	35.14	35.61	35.68
Average	32.15	35.09	33.62	33.94	36.39	34.57

prompted by the extra parameter by comparing the performances of ToM-II and ToM-III: In fact, ToM-III outperformed ToM-II for all combinations of training and test sets in both the training and the test phase, where on average, ToM-III had an error of 5.55% less in estimating the intentions of the participants than ToM-II in the training phase and of 4.16% less in the test phase. Finally, compared to ToM-III, ToM-I showed a better accuracy in estimating the intentions of the participants both in the training and in the test phases, which confirms that including and personalizing the parameter that corresponds to the personality trait of altruism for each participant improves the performance of the model.

Regarding the predictions of the actions that each rational agent took per simulation step, ToM-I made more accurate predictions than ToM-II for all combinations of the training and test sets in both training and test phases. On average, compared to ToM-II, ToM-I was 2.94% more accurate in the training phase and 2.45% more accurate in the test phase. In order to show that the improved performance of ToM-I with respect to ToM-II was not only due to the extra parameter that was identified in ToM-I, but also that the additional cognitive elements that have been added to ToM-I have influenced its performance, the performances of ToM-II and ToM-III were also compared. ToM-III

was more accurate than ToM-II in the training phase for all the combinations of training and test scenarios. ToM-III also outperformed ToM-II for three out of the four combinations of training and test scenarios in the test phase. On average, ToM-III was 1.47% more accurate than ToM-II in the training phase and 1.82% more accurate in the test phase. Nonetheless, including and identifying the altruism parameter allows ToM-I to improve its prediction capabilities, which is shown by the fact that the performance of ToM-I was better than the performance of ToM-III in all cases.

Furthermore, the parameter vector $\boldsymbol{\theta}$ that was identified per participant has the advantage of providing transparency and an opportunity for more detailed analysis, due to assigning real (rather than abstract) meaning to the elements of this vector, compared to the existing black-box approaches, e.g., machine learning methods. More specifically, for example, the three parameters that are used in all the three models ToM-I, ToM-II, and ToM-III, correspond to the preferences of each participant toward each one of the three entities in the scenarios. Thus, in order to assess how much the identified parameters provided a correct insight about the preferences of the participants, at the end of each experiment, the participants were asked to evaluate on a scale from 0 to 10 how much they cared for each one of the entities. The self-assessed preferences $\boldsymbol{\theta}_p^{\text{s}}$ of each participant p were compared to the parameter vector $\boldsymbol{\theta}_p^*$ that was identified by each model for that participant[5]. However, the scale of the parameters in the parameter vector $\boldsymbol{\theta}_p^*$ can be different from the scale used by the participants to self-assess their preferences, since the former depends on the fuzzy systems that were defined to represent the cognitive module. This occurs because the choice of the membership functions and the ranges of the variables, as well as the definition of the rules, was done manually and intuitively by the researchers as to represent the relationships between the variables in Fig. 5 in a way that is general (i.e., not personalized) to all rational agents. These design choices can introduce a shift (i.e., an offset) in the range of the parameters, where this offset was naturally corrected during the identification procedure. For example, the set of parameters identified for a participant p (e.g., $\boldsymbol{\theta}_p = [0.9, 0.2, 0.5]$) when using a certain fuzzy system to represent the cognitive module, could be slightly different (e.g., $\boldsymbol{\theta}_p = [0.8, 0.1, 0.4]$) if the fuzzy system had been projected following different design choices (e.g., using different shapes of the membership functions, or a different number of linguistic terms per variable). Nevertheless, when we compare the values of one parameter amongst participants (or of the parameters identified for one participant p), it is still possible to obtain relevant insights about the preferences of the participants. For this reason, there may be an offset between the subjective scales used by the participants and the scale of the parameters, which should be corrected. Similarly, the ranges of ToM-I and ToM-II can have different offsets, since the fuzzy rules that are defined for these models are different. Furthermore, the range of the parameters that corresponded to the preferences of the participants for the living entities (i.e., the dependent being and the colleague) was different from the range of the parameter that corresponded to the preference of the participants for the personal belonging. This is because the fuzzy rules that defined the influence of the parameters corresponding to the preference for

[5]For this purpose, the answers that were given by the participants were divided by a factor of 10, as to be in the same scale as the parameters identified by the models.

the living beings over the fast-dynamics state variables were different from the fuzzy rules that described the influence of the parameter corresponding to the preference for the personal belonging over these state variables (the entire rule base of these fuzzy systems has been made available via Morão Patrício and Jamshidnejad (2023b)). For example, the variable "fear for an entity" (emotion) is only considered when the entity is a living being. Thus, there are fuzzy rules that describe the influence of the parameters corresponding to general preferences toward the living beings over emotions, while there are no fuzzy rules describing the influence of the general preference toward the objects over emotions. In practice, for each model, we computed two offsets: the offset of the parameters $\theta_{j=1}$ and $\theta_{j=2}$ (for the preferences toward the living beings), and the offset of the parameter $\theta_{j=3}$ (for the preference toward the objects). The offset of each group of parameters was computed as the average error between the identified parameter ${\theta^*}_{j,p}$, and the self-assessed preference ${\theta^s}_{j,p}$ across all the participants and all the parameters of the group. The computation of the offsets is shown in (2.16).

$$\delta_{1,2} = \frac{1}{2} \cdot \frac{1}{n_\mathrm{p}} \sum_{j=1}^{2} \sum_{p=1}^{n_\mathrm{p}} ({\theta^*}_{j,p} - {\theta^s}_{j,p}) \tag{2.16a}$$

$$\delta_3 = \frac{1}{n_\mathrm{p}} \sum_{p=1}^{n_\mathrm{p}} ({\theta^*}_{3,p} - {\theta^s}_{3,p}) \tag{2.16b}$$

Afterwards, each of these parameters can be corrected per participant using the offsets, where the corrected parameter is represented by $\hat{\theta}_{p,j}$ for $j = 1, 2, 3$, as it is shown in (2.17).

$$\hat{\theta}_{j,p} = {\theta^*}_{j,p} - \delta_{1,2}, \quad j = 1, 2 \tag{2.17a}$$

$$\hat{\theta}_{3,p} = {\theta^*}_{3,p} - \delta_3 \tag{2.17b}$$

The computed offsets for ToM-I and for ToM-II in the contexts of intention estimation and action prediction can be found in Table 2.4 and Table 2.5, respectively. Note that the optimal parameters of ToM-I and ToM-III are the same, and consequently the optimal parameters of ToM-III are not represented. Finally, to assess the degree to which the identified parameters corresponded to the self-assessed preferences of each participant p, an average of the absolute errors between the corrected parameters $\hat{\theta}_{j,p}$ identified by the model and the self-assessed preferences ${\theta^s}_{j,p}$ across all the participants and across the three parameters was computed as it is shown in (2.18). The errors obtained by ToM-I and ToM-II are also given in Table 2.4 and Table 2.5.

$$e_\theta = \frac{1}{3} \cdot \frac{1}{n_\mathrm{p}} \sum_{j=1}^{3} \sum_{p=1}^{n_\mathrm{p}} |\hat{\theta}_{j,p} - {\theta^s}_{j,p}| \tag{2.18}$$

The average errors e_θ in the parameters that were identified by ToM-I and by ToM-II when estimating the intentions of the participants were 18.4% and 23.4%, respectively. When predicting the actions of the participants, the average errors e_θ were 21.4% for ToM-I and 24.2% for ToM-II. In both cases, the parameters identified by

Table 2.4: Parameter offsets and errors between parameters and self-assessed preferences achieved by ToM-I and ToM-II when estimating the intentions of the participants.

Model	ToM-I			ToM-II		
Offsets/error	$\delta_{\theta_{1,2}}$	δ_{θ_3}	$e_\theta(\%)$	$\delta_{\theta_{1,2}}$	δ_{θ_3}	$e_\theta(\%)$
Combination 1	-0.569	-0.281	17.8	-0.313	-0.270	26.2
Combination 2	-0.462	-0.276	21.3	-0.233	-0.285	17.8
Combination 3	-0.576	-0.300	17.3	-0.370	-0.265	25.2
Combination 4	-0.602	-0.305	17.0	-0.335	-0.350	24.3
Average	-	-	18.4	-	-	23.4

Table 2.5: Parameter offsets and errors between parameters and self-assessed preferences achieved by ToM-I and ToM-II when predicting the actions of the participants.

Model	ToM-I			ToM-II		
Offsets/error	$\delta_{\theta_{1,2}}$	δ_{θ_3}	$e_\theta(\%)$	$\delta_{\theta_{1,2}}$	δ_{θ_3}	$e_\theta(\%)$
Combination 1	-0.602	-0.271	24.1	-0.350	-0.315	26.0
Combination 2	-0.669	-0.310	20.3	-0.328	-0.330	24.1
Combination 3	-0.631	-0.319	20.0	-0.443	-0.300	21.9
Combination 4	-0.657	-0.333	21.0	-0.303	-0.365	24.8
Average	-	-	21.4	-	-	24.2

ToM-I are closer to the self-identified preferences of the participants, compared to the parameters that were identified by ToM-II. It is important to mention that an error of 0 would be impossible to reach in practice, since the self-assessment of the participants might not correspond to the ground truth, as they can be biased in the cognitive procedure of assessing their own preferences. Moreover, the interpretation of the scale is subjective and can diverge per participant. Nonetheless, the achieved results show a strong relationship between the corrected identified parameters and the opinions of the participants about their own preferences. We can then conclude that these parameters give a relevant practical insight about the preferences of the participants.

All in all, by assessing the performance of the three ToM-based models when predicting the actions and estimating the intentions of the participants, the experiments acknowledged the benefits of including the emotions, biases, and slow-dynamics state-variables such as personality traits and general preferences in the ToM-based models. They also showed that the model can make dynamic predictions and that the identified parameters provide a realistic insight into the meaning and contribution of each parameter, contrarily to the parameters that are often identified in black-box approaches.

2.5. Conclusions and Topics for Future Research

The main aim of this chapter is to provide interactive machines, e.g., social robots, with the capability of exhibiting ToM in their interactions with humans. ToM states that in their interactions, humans create mental models of each other's cognitive procedures, in order to estimate the mental states of one another. Therefore, we proposed a novel formalization for the perception, cognition, and decision-making procedures of humans using a network representation. We transformed this network representation into a dynamic formulation by introducing an extended version of FCMs, and generated a state-space representation for the mental procedures of humans, in order to model the ToM for interactive machines.

Compared to the state-of-the-art representations of human cognition, which mainly include beliefs, goals, and intentions of humans, we also included emotions, personality traits, and biases in our ToM-based models, which have been shown to be important elements in cognition and decision-making procedures of humans (see, e.g., Bono and Vey (2007); Kwon et al. (2008); J. J. Lee et al. (2019); Leite et al. (2013); Saxe and Houlihan (2017); Tapus and Mataric (2008)).

The resulting ToM models were identified and validated in experiments that included computer-based simulations with 21 human participants. We performed several analyses about the performance and accuracy of the resulting models, where the results showed both the success of the model in predicting the intentions and actions of the participants, which are not possible unless the model is dynamic, as well as the accuracy of the model compared to a model that does not include emotions, personality traits, and biases. In the future, a more extensive validation with a larger number of participants and longer-term interaction per participant can be carried out for various contexts of human-machine interaction, so that the model can be validated and used for different real-life applications. Furthermore, the linkages of our mathematical representation can be represented by parameterized polynomial functions rather than by FISs in case the training needs to be accomplished faster. In fact, this change of the mathematical representation leads to an improvement of more than 400 times in the computation time. Moreover, for online applications, the identification of the parameters can be done using a gradient descent algorithm instead of a grid search, which can significantly decrease the computational time.

In the future, the proposed cognitive models will be used for interactive robots for two purposes: (1) to estimate the internal mental states of humans, using model-based state estimation methods, and (2) to predict the evolution of the mental states and actions of humans, and let the robot behave accordingly, in order to improve the quality of the human-robot interactions.

3

A Model-Based Behavioral Control Approach for Human-Social Robot Interaction

The previous chapter established the necessary theoretical foundation for achieving the research objective of this thesis by introducing a dynamic, white-box model of user perception, cognition, and decision-making. However, such a model only becomes impactful when it is systematically leveraged to steer the actions of social robots. This chapter addresses the critical transition from modeling to execution by embedding the proposed model within a model-based controller for social robots. The presented control framework enables social robots to adapt their actions in real-time to users' dynamic mental states, including beliefs, goals, and emotions. Additionally, this chapter includes an experimental evaluation of the proposed control framework through multi-session interactions with a Nao robot involving ten volunteer participants.

This chapter is based on the following publication:

Morão Patrício, M. L., & Jamshidnejad, A. (2026b). Leveraging systems and control theory for social robotics: A model-based behavioral control approach to human-robot interaction. *Applied Intelligence, 56,* 141. https://doi.org/10.1007/s10489-026-07100-9

Abstract

Social Robots (SRs) are increasingly expected to assist in healthcare, education, and companionship, thereby addressing the growing need for personalized and affordable health and social care. However, sustaining long-term user engagement remains a major challenge for SRs, largely due to their limited understanding of human mental states. Accordingly, we leverage a recently introduced mathematical dynamic model of human perception, cognition, and decision-making for behavioral control of SRs. By identifying the parameters of this model and deploying it within a model-based behavioral steering system, SRs can autonomously adapt their actions to evolving user mental states, enhancing long-term engagement and personalization. To achieve this, we introduce the first integration of a systems-theoretic cognitive model into a closed-loop predictive behavioral control framework for SRs, formulated as a constrained multi-objective optimization problem that enables transparent, cognition-aware adaptation. In experiments with 10 participants interacting with a Nao robot across three chess puzzle sessions (45 to 90 minutes each), the identified model achieved a Mean Squared Error (MSE) of 0.067 (i.e., 1.675% of the maximum possible MSE) in tracking beliefs, goals, and emotions of participants, and increased engagement by 16% ($p = 0.009$) compared to a model-free baseline. Post-interaction participant questionnaires further confirmed the perceived engagement and awareness of the model-based controller. Overall, the framework provides a practical pathway toward SRs that autonomously adapt to users in real time, sustain long-term engagement, and ultimately deliver more effective and personalized assistance in domains such as healthcare, education, and companionship.

3.1. Introduction

With the growth and aging of the world population, an increasing number of individuals require targeted assistance. However, health and social care systems often struggle to meet this demand, due to limited resources and financial inaccessibility (Bradwell et al., 2021; Macalupu et al., 2025). Social Robots (SRs), i.e., robots designed to interact with humans in social contexts, offer a promising solution by providing complementary, personalized, and cost-effective assistance to vulnerable users, thereby alleviating the burden on caregivers (Bradwell et al., 2021; Macalupu et al., 2025; Matarić & Scassellati, 2016).

Over the past decades, SRs have been deployed in healthcare, for example, in post-stroke and gait rehabilitation (Bettosi et al., 2024; Céspedes et al., 2021; M. H. Lee et al., 2023), and to assist individuals with cognitive impairment (Benedictis et al., 2023; Kubota et al., 2020). They have also been deployed in education, both for tutoring and guiding educational games (Cao, Beir et al., 2017; Páez et al., 2020; Schicchi & Pilato, 2018; Sequeira et al., 2016), and for promoting skill development in children with autism spectrum disorder (Ascensão & Jamshidnejad, 2022; Clabaugh et al., 2019; Scassellati et al., 2018). Furthermore, SRs serve as companions and assistive tools for older adults (Cooper et al., 2020; Khosla et al., 2019; Luperto et al., 2024).

In assistive contexts, in order to be effective and beneficial, SRs must interact with their users for a prolonged period of time (Khosla et al., 2019; M. H. Lee et al., 2023;

Leite et al., 2013; Luperto et al., 2024; Matarić & Scassellati, 2016; Matheus et al., 2025). However, most SRs struggle to keep their users engaged in the long term. This prompts users to abandon using the robots over time (Clabaugh et al., 2019; Khosla et al., 2019; Matheus et al., 2025). A major challenge in sustaining long-term user engagement lies in the limited social and natural interaction capabilities of current SRs (Maroto-Gómez, Alonso-Martín et al., 2023; Matheus et al., 2025). To address this issue, state-of-the-art SRs typically display human-like behaviors, such as empathy (Tapus & Mataric, 2008), eye contact (Scassellati et al., 2018), and creativity (Dell'Anna & Jamshidnejad, 2022), with the goal of maintaining user interest over time. While these behaviors can successfully engage users initially, they are often undermined by the novelty effect, i.e., users are temporarily entertained by unfamiliar behaviors but gradually lose interest as these behaviors become familiar through repetition (Khosla et al., 2019; Leite et al., 2013). Consequently, existing approaches fail to sustain long-term engagement. This persistent challenge, recognized for decades yet remaining unresolved, is closely linked to the behavior of SRs, since user engagement over time depends on how varied, responsive, and adaptable those behaviors are (Leite et al., 2013; Maroto-Gómez, Alonso-Martín et al., 2023; Matheus et al., 2025). This emphasizes the necessity for developing behavioral control approaches for SRs that overcome current limitations in sustaining long-term user engagement.

State-of-the-art controllers of SRs are typically designed for specific applications or tasks. This task-specific design, which stems from underlying behavioral control approaches of SRs, limits their generalizability across contexts (Maroto-Gómez, Alonso-Martín et al., 2023). Current methods predominantly rely on rule-based systems, which struggle to handle unforeseen situations, or on machine learning approaches, which, in the SR domain, face data scarcity that forces simplified interactions and constrained behavioral repertoires (Bagheri et al., 2021; Clabaugh et al., 2019; M. H. Lee et al., 2023). As a result, robot behavior often becomes repetitive, predictable, and insufficiently responsive to user input, which can reduce the perceived social presence of the robot and ultimately hinder long-term user engagement (Matarić & Scassellati, 2016; Tapus et al., 2007).

Consequently, this leads to the fundamental question of which aspects of human social interaction SRs fail to replicate. During social interactions, humans naturally model each other's mental states, such as beliefs, desires, and emotions (Benedictis et al., 2023). Furthermore, they dynamically leverage this ability to predict one another's actions and to adjust their own behavior accordingly, a manifestation of Theory of Mind (ToM) (Premack & Woodruff, 1978). Nevertheless, despite prior work recognizing the potential benefits of such capabilities for SRs, few existing systems comprehensively implement these capabilities (Benedictis et al., 2023; Matarić & Scassellati, 2016; Patacchiola & Cangelosi, 2020).

Accordingly, we propose a behavioral control framework for SRs grounded in explicit mathematical modeling of user mental states and their dynamic evolution over time. These models are used to adapt the robot's behavior in real time to optimize interaction-related objectives (e.g., maximize engagement) while respecting constraints (e.g., ensuring a positive emotional state). This approach aims to enable SRs to behave in a more human-like, engaging, and personalized manner, thereby promoting

sustainable long-term interactions in Human-Social-Robot Interactions (HSRIs).

The remainder of this chapter is organized as follows: Section 3.1.1 describes the main contributions of this chapter, and Section 3.1.2 reviews relevant work. Section 3.2 of this article describes the approaches and mathematical changes applied to the model proposed in Chapter 2 (which we refer to as Mathematical Model of Mind (MMM) in this chapter) to make it suitable for HSRIs. A case-study where the framework is applied is given in Section 3.3, and the results in Section 3.4. Finally, Section 3.5 summarizes the conclusions taken in this study and the recommendations for the future.

3.1.1. Main Contributions

In order to advance social robotics and improve the interactions between Social Robots (SRs) and humans, this chapter offers the following contributions:

- **First deployment of systems-and-control-theoretic methods for cognitive Human-Social-Robot Interactions (HSRIs)**

  We leverage, for the first time, a *dynamic, control-theoretic* model of Theory of Mind (ToM) — the Mathematical Model of Mind (MMM) introduced in Chapter 2 — and integrate it into the behavioral control system of SRs. This enables personalization and real-time adaptation of SR behaviors to the evolving mental states of humans, such as beliefs, goals, and emotions. Leveraging MMM for SRs required adapting systems-and-control-theoretic methods to a controlled process that is inherently non-physical and unconventional: human cognition. Unlike physical systems and engineered processes, mental states are not directly measurable, evolve on multiple timescales, and are not governed by deterministic physical laws. Addressing these intrinsic characteristics makes the integration and adaptation of control principles non-trivial and establishes a bridge between traditional systems and control theory and human-centered domains, including social robotics.

- **Development of a model-based behavioral steering system for SRs**

  We introduce a model-based predictive control framework that optimizes robot actions over anticipated trajectories of user mental states, while respecting constraints that impact the HSRI, such as safety and social norms. The control objective is to maximize key criteria of effective HSRIs, including the engagement and emotional well-being of users, as well as the sustainability of HSRIs. To achieve this, we introduced several technical adaptations bridging classical control theory and social robotics, including: (i) specifying and parameterizing the mathematical formulations of perception, cognition, and decision-making processes of MMM to enable practical deployment; (ii) developing procedures to identify user-specific parameters under limited data; and (iii) embedding the identified model in a model-based behavioral controller. Collectively, these developments address central limitations of rule-based and model-free controllers by enabling more transparent, adaptive, and socially sustainable robot behaviors in real-world HSRIs.

- **Empirical validation through real-life long-term HSRIs**

  We conducted experiments with 10 participants in extended chess puzzle sessions with a Nao robot. The results demonstrated a 16% increase in the engagement level of users (based on self-reported boredom reduction, $p = 0.009$), as well as reduced intent to quit (i.e., -7.7%, $p = 0.032$) and skip puzzles (i.e., -8.9%, $p = 0.029$). The model also achieved a high estimation accuracy (i.e., Mean Squared Error (MSE) of 0.067). Post-interaction questionnaires completed by the participants confirmed that perceived personalization and awareness have increased with the robot that leveraged MMM. These results validate the feasibility and effectiveness of cognition-aware closed-loop control for improving the social awareness and responsiveness of SRs in real-world settings.

3.1.2. Related Work

Most state-of-the-art research on steering the behavior of SRs relies on model-free approaches, including learning-based and rule-based systems. Among learning-based methods, Reinforcement Learning (RL) (Bagheri et al., 2021; Clabaugh et al., 2019; Gao, Barendregt et al., 2018; Hussain et al., 2022; Patacchiola & Cangelosi, 2020) Neural Network (NN) (Benedictis et al., 2023; Filippini et al., 2021; M. H. Lee et al., 2023; Yu & Pei, 2021), and data-driven approaches (Bettosi et al., 2024) have been applied to steer the behavior of SRs.

In social robotics, RL is typically used to learn high-level interaction policies from experience. However, most applications rely on simple algorithms that operate in low-dimensional, discrete state/action spaces and that often produce static or minimally adaptive policies. For example, Gao, Barendregt et al. (2018) used RL to personalize the supportive behaviors of a tutor SR, but since the RL policy converged to a single fixed behavior per user, the adaptability of the SR remained very limited. Similarly, Clabaugh et al. (2019) applied RL to adjust the difficulty of educational math games and the feedback that was provided by the robot during HSRIs using performance-based rewards. The state space, however, was restricted to the ten predefined games, which in practice led the RL agent to learn one optimal action per game, again limiting its adaptability. Bagheri et al. (2021) used RL to select among four empathetic utterance categories based on the user's emotion and personality. However, the state space was limited to four discrete emotion labels combined with a fixed personality type per participant, resulting in limited adaptation during interaction. More recently, more sophisticated RL frameworks were introduced into social robotics. Patacchiola and Cangelosi (2020) proposed an actor-critic RL algorithm for trust-based learning, where the inputs consisted of audio and visual features. Meanwhile, Hussain et al. (2022) developed a batch deep RL algorithm that optimized user engagement by controlling smiling and nodding behaviors of a SR, with the state space consisting of speech features (such as pitch and intensity) extracted from the user's audio. Despite using more advanced RL methods, both approaches relied on measurable cues for state representation and reward design, rather than on explicit models of users' mental states.

The application of NNs in social robotics has been largely restricted to perception

tasks rather than behavior control (Benedictis et al., 2023; Filippini et al., 2021; M. H. Lee et al., 2023; Yu & Pei, 2021). Filippini et al. (2021) used a multilayer perceptron to estimate affective states of users in real time from visible and thermal images, with the NN output feeding into a simple rule set for behavioral control. Similarly, Bagheri et al. (2021) applied an NN-based emotion detection system to generate the state input to the low-dimensional RL agent described in the previous paragraph. M. H. Lee et al. (2023) combined a feedforward NN with a rule-based module to assess the movement quality of patients. The output of these modules was used by a SR acting as a rehabilitation coach to provide corrective feedback. Finally, Benedictis et al. (2023) relied on off-the-shelf NN modules for speech-to-text processing and sentiment and intention recognition. Overall, these studies show that NNs are primarily used to infer task performance and affective states of users. Nonetheless, the adaptation of SR behavior to these states is typically handled by simple rule-based systems or by machine learning agents, and is rarely grounded in multi-dimensional, dynamic models of user mental state. More recently, researchers have applied NNs to the generation of robot behavior itself. For example, Merino-Fidalgo et al. (2025) proposed a natural language interface for home assistance SRs, where a large language model translates user commands into behavior trees that control robot actions. While the system adapts the behavior trees in response to execution failures and environment constraints, it does not consider or adapt to user mental states. Similarly, Mascaro and Lee (2025) developed a transformer-based system that learns social interaction behaviors, such as handshakes, from human-human motion data. In particular, the system generates SR movements by refining its future motions based on the predicted motions of the human partner, enabling natural and coordinated physical interactions. While this system demonstrates the importance of adapting to the physical behavior of users, it does not adapt to their mental states, which is essential in HSRIs. Additionally, Bettosi et al. (2024) proposed a non-neural data-driven approach that clusters behaviors of human caregivers to generate SR interaction strategies for coaching and therapeutic contexts. In general, most machine learning controllers of SRs neither model the (dynamics of) user mental states nor act in accordance with their evolution over time.

Rule-based approaches are another widely applied control strategy in social robotics (Benedictis et al., 2023; Filippini et al., 2021; M. H. Lee et al., 2023; Rossi et al., 2020; Scassellati et al., 2018; Schrum et al., 2019). These approaches typically adapt SR behavior based on rules depending on interaction-specific metrics (e.g., task performance (Benedictis et al., 2023; M. H. Lee et al., 2023; Scassellati et al., 2018), user physical state (Benedictis et al., 2023; Schrum et al., 2019), and directly observable user affective states (Filippini et al., 2021; Rossi et al., 2020)), rather than modeling the mental states of the users or their dynamics. Several studies implemented highly constrained rule sets that specify only a few discrete conditions and corresponding robot behaviors. For example, Scassellati et al. (2018) adjusted game difficulty for children with autism based on fixed performance thresholds, increasing the level when accuracy exceeded 75% and decreasing it when below 25%; Schrum et al. (2019) regulated the intensity of dance movements proposed to the user according to whether the user's heart rate was above or below a target range; Rossi et al. (2020) activated one of the four pre-programmed behaviors for a SR based on the age and anxiety level

of the user; Filippini et al. (2021) selected between two actions (i.e., telling a story or singing a song) for a SR depending on the user's engagement. More structured designs include combining a finite state machine with parameterized if–then rules for rehabilitation exercises M. H. Lee et al. (2023), and adapting the communication style of the SR using a rule-based system informed by variables such as user sentiment, intent, and extraversion Benedictis et al. (2023). Essentially, rule-based systems remain dominant in practice, but their reliance on fixed or context-specific variables hinders adaptability and provides limited modeling of user mental states.

Several of these studies, including Bagheri et al. (2021), Filippini et al. (2021), Benedictis et al. (2023), and Rossi et al. (2020), employ affective computing methods, as they rely on emotion or sentiment recognition to guide robot behavior. However, these affective cues are typically used reactively and fail to capture the temporal evolution or inter-dynamics of affective and cognitive states of users.

Beyond affective computing, cognitive modeling approaches are typically used to generate the robot's behavior through models or architectures inspired by cognitive processes. Within these approaches, one line of research develops biological models of the brain (Patacchiola & Cangelosi, 2020; Tani et al., 2020), where the SR behavior is generated through a brain-inspired architecture. This line of work has primarily been proposed for enabling SRs to learn trust (Patacchiola & Cangelosi, 2020) or to distinguish false beliefs (Tani et al., 2020). Similarly, cognitive architectures have been used to structure the behavior of SRs, typically through hierarchical or modular systems that integrate perception, internal states, and task or context knowledge to generate coherent and context-aware actions (Benedictis et al., 2023; Cao et al., 2019; Dell'Anna & Jamshidnejad, 2024; Maroto-Gómez, Malfaz, Castro-González & Salichs, 2023; Páez et al., 2020). Nevertheless, these architectures model the internal processes and generate the behavior of the SR, not that of the user.

Some cognitive architectures, such as Benedictis et al. (2023), Páez et al. (2020), and Cao et al. (2019), have recently incorporated behavioral adaptation to user states, including mental states. For example, Benedictis et al. (2023) proposed a cognitive architecture, intended for SRs assisting older adults, that primarily models robot cognition but includes a reactive component that adjusts robot behavior according to user sentiment, intent, and extraversion. However, these user variables are treated as inputs to the architecture, rather than being explicitly modeled. Similarly, Páez et al. (2020) proposed a cognitive architecture for SRs that supported children's learning in problem-solving tasks. The architecture modeled the beliefs, desires, and intentions of the SR while adapting its behavior according to indicators such as the learner's performance and emotions. These indicators were inferred from inputs (e.g., speech, gestures, facial expressions) and incorporated into the robot's beliefs through rule-based processing, rather than by representing the learner as an autonomous agent within the architecture. Cao et al. (2019) presented a general-purpose therapeutic architecture that models the robot's affect and behavior, adapting them to the personality, preferences, and mood of users — which were predefined, detected via sensors, or set by a therapist, rather than explicitly modeled within the architecture. Overall, these architectures extensively model the internal cognitive processes of the robot, but treat user-related variables as static inputs. As a result, behavioral adaptation remains reactive to the

observed user states, whereas modeling the dynamics of the internal processes and mental states of users enables predictive, context-aware adaptation.

To address this limitation, some researchers have proposed that SRs should implement Theory of Mind (ToM), the capability of rational agents to be aware of and to reason about the perception, beliefs, desires, and decision-making of other rational agents (Premack & Woodruff, 1978; Scassellati, 2002). Scassellati (2002) poses that humanoid robots should exhibit ToM, and since then, some studies have applied the concept of ToM for SRs (Dissing & Bolander, 2020; J. J. Lee et al., 2019; Patacchiola & Cangelosi, 2020; Tani et al., 2020; Vossen et al., 2018). For example, Vossen et al. (2018) enabled SRs to represent that different individuals may hold distinct and even conflicting perspectives and beliefs. Additionally, J. J. Lee et al. (2019) developed two computational frameworks for storytelling interactions, in which the storyteller and listener agents each model and respond to the other's mental states: the storyteller infers and seeks to sustain the listener's engagement, while the listener estimates and influences the storyteller's belief about its engagement. More recently, Tani et al. (2020) and Dissing and Bolander (2020) explored approaches for equipping SRs with the ability to model and respond to others' false beliefs. However, these works focus mainly on one particular aspect of ToM, such as belief or engagement, rather than providing a comprehensive model of user cognition or decision-making, leaving a gap in fully understanding and predicting user behavior.

To overcome these limitations, in Chapter 2, we have proposed a mathematical framework that models the perception, cognition, and decision-making of humans, following the core ToM principles. In this chapter, we refer to this model as MMM. In particular, we leverage the MMM to be used by SRs to track and act on the mental states of their human users, by integrating the model within a control system that is suitable for the purposes of HSRIs.

In contrast to existing behavior control methods for SRs, our framework directly addresses the main limitations of these methods and introduces key advantages: It ensures adaptability and personalization by explicitly modeling the user's mental states, including their evolution and inter-dynamics, and by continuously adjusting the behavior of the SR based on this model. It achieves predictivity by optimizing robot actions over anticipated trajectories of user cognition, moving beyond the short-term reactivity typical of affective computing approaches. The generalizability of the framework stems from the domain-independent design of MMM, whereas rule-based and learning-based methods are often tailored to specific setups and are trained on limited datasets. Grounding MMM in psychology and cognitive science enhances data efficiency by embedding prior knowledge about human cognition, which is typically missing in learning-based approaches. Moreover, explicitly modeling user mental states provides transparency and explainability that are often lacking in machine learning approaches, while the control-theoretic foundation ensures robustness by incorporating dynamic constraints, such as safety and social considerations.

3.2. Methodology

In this section, we describe our proposed model-based control framework for Social Robots (SRs). In particular, we explain how the dynamic mathematical model of perception, cognition, and decision-making introduced in Chapter 2, i.e., the Mathematical Model of Mind (MMM), is leveraged to be used in a model-based control framework for SRs.

In Chapter 2, a path diagram, which illustrates variables that are connected via processes and weighted linkages, has been proposed to provide a white-box representation for the inter-dynamics of the (invisible) mental states and their contribution to the (visible) actions of humans, based on the Theory of Mind (ToM). The variables represented in the path diagram can be either static variables (i.e., variables without a memory or impact from their previous realizations) and dynamic variables (i.e., variables with a memory that are impacted by the past realizations). The static variables of the path diagram are real-life data, perceived data, rationally perceived knowledge, intention, and action, and the dynamic variables include beliefs, goals, emotions, perceived knowledge, and biases.

The path diagram is then formulated in Chapter 2 as a mathematical model, which we refer to as MMM, using systems theory. MMM includes three modules: perception, which models how real-life data is perceived by humans to impact their mental states; cognition, which represents the dynamic evolution of the mental states of humans; and decision-making, which determines the actions of the humans based on their mental states.

The dynamic processes of MMM have been developed in the state space framework. Thus, they are represented in terms of a set of inputs, outputs, state variables (i.e., dynamic variables that represent the internal mental states at any given time and that evolve according to a differential or difference equation), and dynamic auxiliary variables (i.e., dynamic variables that are additionally included in the model to allow for a realistic and precise bridging of the different modules and state variables). All the dynamic variables (i.e., beliefs, goals, and emotions, which are the state variables, as well as the biases and the perceived knowledge, which are the dynamic auxiliary variables) are updated within the cognition module.

The three modules of perception, cognition, and decision-making are sequentially integrated, i.e., the output variable of the perception module is injected as input into the cognition module, which updates all the dynamic variables of the model. Then, the updated state variables beliefs and goals are injected into the decision-making module, which outputs the action. Note that although the state variable emotion does not directly impact the decision-making processes, it does impact the updated state variables beliefs and goals via their inter-dynamics.

The state space representation of MMM allows the model to be embedded within control-theoretic approaches, which in turn enables steering SRs transparently with guarantees on performance and constraint satisfaction. Next, we describe how the MMM is leveraged for incorporation into a control-theoretic method for steering the behavior of SRs. The model is represented in a discrete time framework.

3.2.1. Leveraging MMM for Model-based Control of SRs

We introduce improvements to the formulation of MMM that make the resulting model suitable for adoption by control-theoretic methods in real-life implementations of Human-Social-Robot Interaction (HSRI).

Note that, according to the neuroscience findings, although perception, cognition, and decision-making are interconnected, their processes occur with different speeds (Benedetto et al., 2020). More specifically, the perception processes (i.e., the transition of the external stimuli into perceived information) occur almost three times faster than the cognition and motor processes (i.e., the evolution of mental states and their transition into actions) (Buzsáki & Draguhn, 2004). Moreover, the perception depends on the external stimuli and should be updated every time a new external stimulus is received. However, the cognition evolves not only due to the perception but also under the impact of the inter-dynamics of the mental states. Therefore, the cognition may be updated with a frequency different from the frequency of capturing the external stimuli. Consequently, in the following discussions, we have used different discrete time steps for the perception (where the discrete time steps are indicated by k^{p}) and for the cognition and decision-making (where k denotes the corresponding discrete time steps).

The improvements made in different modules of MMM and the motivation for introducing them are explained next.

Perception module

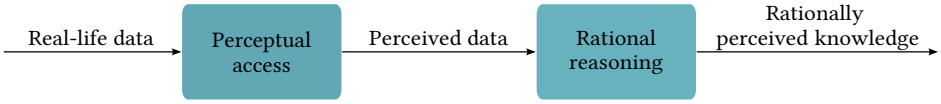


Figure 3.1: Perception module.

Perception is the process of translating real-life data into an internal representation of captured information. Perception in MMM is mathematically modeled with three variables, i.e., real-life data, perceived data, and rationally perceived knowledge, and through two linked subprocesses, perceptual access and rational reasoning. The perceptual access is the transformation of the real-life data into perceived data, which is the data captured by the human. The rational reasoning is the transition of the perceived data into rationally perceived knowledge, which is knowledge that would ideally be reasoned by that individual when not biased by current mental states. Figure 3.1 illustrates these subprocesses and their input-output pairs.

In order to leverage MMM for model-based control of SRs when used in the context of HSRI, we propose the following improvements (highlighted in the boxes and elaborated in the text) for the perception module of MMM.

Specifying the perceptual access mathematically

In Chapter 2, the perceptual access subprocess is represented via a generic function $f^{\text{PA}}(\cdot)$ without specification regarding its class or characterization. We fill this gap to enable deployment on SRs by representing the perceptual access as a filter where the output (i.e., perceived data) equals the input (i.e., real-life data) when the input is captured, and, otherwise, equals the last captured value. Our solution presents a transparent way to mathematically represent the perceptual access, enhancing the white-box nature of MMM, and facilitating its usage by SRs.

Suppose that $\mathcal{R}$ is an ordered set which includes all the real-life data in a given HSRI context. At the discrete time step k^{p}, the value of the elements of $\mathcal{R}$ are denoted by $u_i(k^{\text{p}})$ with $i = 1, \ldots, |\mathcal{R}|$. Moreover, $y_i^{\text{PA}}(k^{\text{p}})$, with $i = 1, \ldots, |\mathcal{R}|$, is the i^{th} element of the output of the perceptual access (i.e., i^{th} element of the perceived data) at time step k^{p}, which is determined based on $u_i(k^{\text{p}})$. Whenever the human captures the i^{th} element of real-life data, then $y_i^{\text{PA}}(k^{\text{p}})$ is replaced by the value of this element, $u_i(k^{\text{p}})$. Otherwise, the value of $y_i^{\text{PA}}(k^{\text{p}})$ keeps its most recent value. The mathematical representation of the perceptual access is:

$$y_i^{\text{PA}}(k^{\text{p}}) = \begin{cases} u_i(k^{\text{p}}), & \text{if } u_i(k^{\text{p}}) \text{ was captured} \\ y_i^{\text{PA}}(k^{\text{p}} - 1), & \text{otherwise} \end{cases} \tag{3.1}$$

Decoupling & parameterizing the function that models the rational reasoning

In Chapter 2, the rational reasoning subprocess is represented via a generic function $f^{\text{RR}}(\cdot)$, which remains abstract, without specifying its class or characterization. We address this gap for deployment on SRs, which should understand how humans reason about perceived data, by decoupling and parameterizing the function that maps the input of the subprocess (i.e., the perceived data) to its output (i.e., rationally perceived knowledge). Our solution further enhances the white-box nature of MMM, thus its transparency and traceability, and yields a practical and computationally efficient way for personalizing the perception process to users.

The perceived data is injected as input into the rational reasoning subprocess to generate as output the rationally perceived knowledge (see Figure 3.1). Thus, the outputs $y_i^{\text{PA}}(k^{\text{p}})$ of the perceptual access are the inputs to the rational reasoning. The output of the rational reasoning, i.e., rationally perceived knowledge, is composed of elements $y_j^{\text{RR}}(k^{\text{p}})$, where $j = 1, \ldots, |\mathcal{K}|$ and $\mathcal{K}$ is the ordered set that embeds all potential realizations of the rationally perceived knowledge in the given HSRI context.

We consider HSRIs that require selective attention from humans (e.g., a particular educational or entertaining joint task by the SR and the human). Since the human focuses on a specific stimulus or input at a time while suppressing the others, each piece of perceived data affects rationally perceived knowledge independently. This

allows us to decouple the input-output mapping (i.e., the mapping that shows the impact of input $y_i^{\mathrm{PA}}(k^{\mathrm{p}})$ on output $y_j^{\mathrm{RR}}(k^{\mathrm{p}})$) in the mathematical modeling of the rational reasoning. Considering the additivity condition (which allows for simplicity and efficiency of the computations) on this input-output mapping, we have:

$$y_j^{\mathrm{RR}}(k^{\mathrm{p}}) = \sum_{i=1}^{|\mathcal{P}|} f_{ij}^{\mathrm{RR}}(y_i^{\mathrm{PA}}(k^{\mathrm{p}}); \boldsymbol{\theta}_{ij}^{\mathrm{RR}}), \quad \forall j \in \mathcal{K} \tag{3.2}$$

In (3.2), the subprocess is modeled as the summation of various parameterized functions, each mapping one input of the subprocess to its corresponding output. More specifically, the contribution of the i^{th} element $y_i^{\mathrm{PA}}(k^{\mathrm{p}})$ of $\mathcal{P}$ through the rational reasoning process into the j^{th} element $y_j^{\mathrm{RR}}(k^{\mathrm{p}})$ of $\mathcal{K}$ (i.e., the j^{th} element of the rationally perceived knowledge at time step k^{p}) is modeled by the function $f_{ij}^{\mathrm{RR}} : \mathcal{P} \to \mathcal{K}$. Note that the function $f_{ij}^{\mathrm{RR}}(\cdot)$ is defined as parameterized, i.e., $\boldsymbol{\theta}_{ij}^{\mathrm{RR}}$ is a parameter vector that should be identified per user for all indices i and j. This allows for more flexibility in the mathematical model and for the ease of personalization per user.

Remark 7 *Sets $\mathcal{P}$, and $\mathcal{K}$ are defined as ordered sets, because the inputs and outputs in (3.1) and (3.2) need to be distinguishable to be associated to the corresponding parameterized function of the model.*

Cognition module

Cognition refers to the processes that use the output of the perception process, i.e., the rationally perceived knowledge, to update the mental states that will determine the decision-making of the human. In Chapter 2, using a state space framework, the state and dynamic auxiliary variables (shown by $x(k)$ and distinguished by subscript indices) of the cognitive module are updated at each discrete time step $k+1$, based on their values at the previous discrete time step k. Note that different indices do not only refer to different dynamic variables (e.g., belief versus emotion) but also to different elements for one variable (e.g., different beliefs that play a role in the given HSRI context). We have[1]:

$$\begin{aligned} x_i(k+1) = f_i\left(x_i(k)\right) &+ \sum_{i \neq j} w_{ji}(k) x_j(k) \\ &+ \sum_{\ell \in \mathcal{K}} z_{\ell i}(k) y_\ell^{\mathrm{RR}}(k) \end{aligned} \tag{3.3}$$

where the first term on the right-hand side of (3.3) models the impact of the previous realization of the variable, i.e., of $x_i(k)$, on its updated realization, i.e., on $x_i(k+1)$.

[1]Note that indices used in different sections of the chapter (e.g., i and j) should be treated as local variables, i.e., there is no relevance between these indices used, for instance, in the mathematical discussions for the cognition module and the same indices used elsewhere in the mathematical discussions for the perception or decision-making modules.

The second term on the right-hand side of (3.3) represents the inter-dynamics of the state and dynamic auxiliary variables, with $w_{ji}(k)$ the relative weight showing the impact of variable $x_j(k)$ (called the influencing variable) on updated variable $x_i(k+1)$ (called the influenced variable) at discrete time step k. The third term on the right-hand side of (3.3) represents the influence of those inputs that are external to the cognitive module, with $z_{\ell i}$ the influence of the output of the perception module, the rationally perceived knowledge, on the dynamic variable perceived knowledge (see Figures 3.1 and 3.2).

Remark 8 *Note that (3.2) estimates $y_\ell^{RR}(k^p)$ more frequently than the update frequency of the cognition module. Thus, in (3.3) $y_\ell^{RR}(k)$ is the most recent value of $y_\ell^{RR}(k^p)$ that has been estimated prior to time step k.*

The weights $w_{ji}(k)$ in (3.3) are, in general, functions of both the influencing and influenced variables at the current time step k, as these interactions occur within the cognition module and are state-dependent. In contrast, the weight $z_{\ell i}(k)$ only depends on the influencing variable, since the rationally perceived knowledge acts as an external input to the cognition module, without a closed-loop feedback from the cognitive variables. Accordingly:

$$w_{ji}(k) = f_{ji}^{\mathrm{W}}\left(x_i(k), x_j(k)\right) \tag{3.4a}$$

$$z_{\ell i}(k) = \tag{3.4b}$$

$$\begin{cases} f_{\ell i}^{\mathrm{Z}}\left(y_\ell^{\mathrm{RR}}(k)\right), & x_i(k) \text{ is perceived knowledge} \\ 0, & \text{otherwise} \end{cases}$$

with $f_{ji}^{\mathrm{W}}(\cdot)$ denoting the function that estimates the weights $w_{ji}(k)$ based on the values of the influencing and influenced variables at each time step, and $f_{\ell i}^{\mathrm{Z}}(\cdot)$ denoting the function that estimates $z_{\ell i}(k)$ based solely on the influencing variable, i.e., the rationally perceived knowledge $y_\ell^{\mathrm{RR}}(k)$.

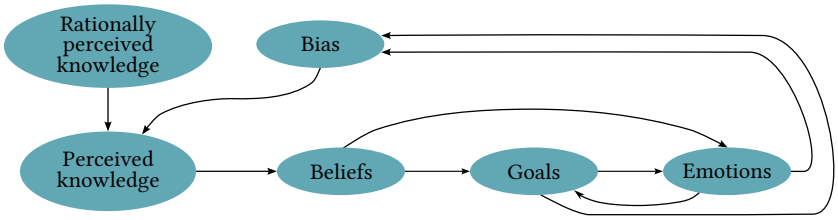


Figure 3.2: Cognition module.

We propose the following improvements (highlighted in the boxes and elaborated in the text) to leverage the cognition module of MMM for real-life applications in HSRI.

Incorporating the different update frequencies of state or dynamic auxiliary variables

Not all dynamic variables in MMM are, in general, updated with the same frequency. This difference should thus be incorporated in the update equation (3.3). Accordingly, we propose to represent the first term, i.e., $f_i\left(x_i(k)\right)$, as a weight w_{ii} multiplied by the most recent realization of the variable, $x_i(k)$. While each weight w_{ii} models how fast a variable responds to its last realization, the relative values of all the weights in (3.3) incorporate the difference of the update frequencies for different variables.

In MMM, the representation of function $f_i(\cdot)$, as well as the relative update frequencies of the dynamic variables were not discussed. As the processes that result in the formation of beliefs[2] are included in the perception module, and the perception processes are faster than the cognition processes (Buzsáki & Draguhn, 2004), the beliefs are in general updated more frequently than the emotions and goals. Accordingly, we propose:

$$f_i(x_i(k)) = w_{ii}x_i(k) \tag{3.5}$$

By adjusting the value of w_{ii}, one models how intensely the variable $x_i(k)$ changes per discrete time step (i.e., how quickly the variable responds to its previous realization). Moreover, the relative value of w_{ii} and $w_{ji}(k)$ in (3.3) incorporates the difference in the frequency of the self-impact and the impact by other variables.

For the beliefs, in particular, we use $w_{ii} = 0$ since the beliefs per time step depend on the current stimuli, and are generated by the perceived knowledge (see Figure 3.2). Any internal impacts that may affect the beliefs (e.g., the impact of the most recent goals and emotions) are incorporated through the impact of the auxiliary variable bias on the perceived knowledge (see Figure 3.2).

If the weight w_{ii} tends to 1, it leads to instability of (3.3) (Mahmoud & Singh, 2012). To ensure the stability of the model, all weights w_{ii} and w_{ji} should be chosen as to assure that all the eigenvalues of the corresponding matrix W (i.e., a matrix with weights w_{ii} in its diagonal element positions (i, i) and weights w_{ij} with $i \neq j$ in positions (i, j)) are within the unit circle (Mahmoud & Singh, 2012).

[2]Beliefs correspond to the internal representations of stimuli, arising from the perception of a rational agent, and thus vary in time as the perception does. Knowledge (e.g., moral values, political beliefs, etc.) that is fixed or that varies very slowly is deemed pieces of general world knowledge of the agent.

Modeling the inter-dynamics of mental variables via piecewise-constant functions

In (3.4a), as is given in Chapter 2, $f_{ji}^{\mathrm{W}}(\cdot)$ is presented as generic and abstract, while specific characteristics in formulating $f_{ji}^{\mathrm{W}}(\cdot)$ for meeting desired criteria by MMM remain vague. We address this gap by proposing a mathematical representation for $f_{ji}^{\mathrm{W}}(\cdot)$ that provides a balanced trade-off between computational efficiency and adequate flexibility for illustrating the interpersonal differences of humans, that is always striven for in HSRIs. Accordingly, we propose to define $f_{ji}^{\mathrm{W}}(\cdot)$ as a piecewise function with respect to the influenced variable $x_i(k)$ and influencing variable $x_j(k)$, where these constant values should be identified per human.

Suppose that $\mathcal{X}_i$ and $\mathcal{X}_j$ are the admissible sets of all the potential realizations of the state/auxiliary variables x_i and x_j, respectively. Moreover, $\overline{\mathcal{X}}_i \subseteq \mathcal{X}_i$ and $\underline{\mathcal{X}}_i \subseteq \mathcal{X}_i$, with $\overline{\mathcal{X}}_i \cup \underline{\mathcal{X}}_i = \mathcal{X}_i$ and $\overline{\mathcal{X}}_i \cap \underline{\mathcal{X}}_i = \emptyset$, include two distinct ranges for the influenced variable. For instance, they may be interpreted as ranges of undesirable and desirable status for the variable (e.g., spectrum of feeling bored and spectrum of feeling interested for the emotion state variable "boredom"). We also consider $\mathcal{X}_{j,1}, \ldots, \mathcal{X}_{j,n} \subseteq \mathcal{X}_j$ with $\cup_{m=1}^{n} \mathcal{X}_{j,m} = \mathcal{X}_j$ and $\mathcal{X}_{j,m} \cap \mathcal{X}_{j,o} = \emptyset$ for $m, o = 1, \ldots, n$ and $m \neq o$ as distinct ranges of the influencing variable, where in each of these sub-sets the impact of the influencing variable on the influenced variable may be considered as constant. Then for $x_j(k) \in \mathcal{X}_{j,m}$ with $m = 1, \ldots, n$, we propose the following representation for the function $f_{ji}^{\mathrm{W}}(\cdot)$:

$$w_{ji}(k) = \begin{cases} \zeta_{j,m}, & x_i(k) \in \overline{\mathcal{X}}_i \\ \xi_{j,m}, & x_i(k) \in \underline{\mathcal{X}}_i \end{cases} \tag{3.6}$$

where $\zeta_{j,m}$ and $\xi_{j,m}$ are fixed parameters that should be identified per user of the SR for all values of the indices j and m. Note that (3.6) provides a simple, easily implementable, and computationally efficient formulation for real-life applications on the model for SRs, while any potential nonlinearity in the inter-dynamics of the influencing and influenced variables will properly be represented by proper tuning of the number n of subsets $\mathcal{X}_{j,m}$. In practical implementations, the subsets $\mathcal{X}_{j,m}$ may be defined either based on theoretical thresholds or derived empirically from data. For example, the subsets may correspond to conceptually meaningful ranges, such as low, medium, and high levels of an emotion or belief, or they can be obtained through data-driven partitioning methods, such as quantile-based segmentation or clustering, during model identification. The choice of the relevant approach depends on the availability of domain knowledge and data, as theory-based partitioning offers greater interpretability but requires more design effort.

Decision-making module

Decision-making is the process of deciding about actions, depending on the mental states, namely beliefs and goals. The decision-making process in MMM has been

modeled via two subprocesses that are linked through a static auxiliary variable, called the intention. These two subprocesses are the rational intention selection (i.e., the conversion of the beliefs and goals into intentions) and the rational action selection (i.e., the conversion of the intentions into actions). Figure 3.3 shows a simplified illustration of the decision-making module in MMM.

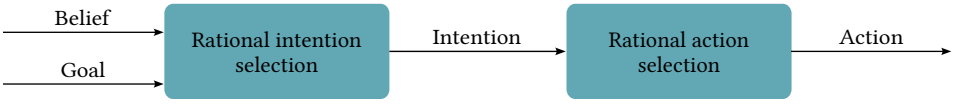


Figure 3.3: Decision-making module.

In order to be able to model the decision-making of humans in real-life HSRIs using MMM, we propose the following improvements (highlighted in the box and elaborated in the text).

Parameterized formulation of rational intention selection based explicitly on beliefs and goals

In Chapter 2, an explicit formulation based on the beliefs and goals for the rational intention selection subprocess has not been provided. We address this gap by proposing an affine parameterized formulation that outputs the values for the intensity of intentions, based on the values realized for the beliefs and goals. This formulation provides a functional way to individually identify the impact of the state variables on the intention, and to estimate their weighted summation to obtain the intentions. This yields an explainable, straightforward, and easily personalizable model for representing the rational intention selection of humans for SRs.

Suppose that $y_i^{\text{RIS}}(k)$ is used to show the i^{th} element of the intention at time step k, with $i \in \mathcal{I}$ and $\mathcal{I}$ a set including all integer indices for the intention elements. In fact, per time step k, the rational intention selection subprocess outputs vector $\boldsymbol{y}^{\text{RIS}}(k)$ that includes $y_i^{\text{RIS}}(k)$ for all $i \in \mathcal{I}$. Element $y_i^{\text{RIS}}(k)$ quantifies the strength of intention i that depends on the realized values of the beliefs and goals at time step k that influence that intention. For each $i \in \mathcal{I}$ we have:

$$y_i^{\text{RIS}}(k) = \sum_{j \in \mathcal{B}} \theta_{ij}^{\text{RIS}} x_j(k) + \sum_{\ell \in \mathcal{G}} \theta_{i\ell}^{\text{RIS}} x_\ell(k) + \theta_i^{\text{RIS}} \tag{3.7}$$

where $\mathcal{B}$ and $\mathcal{G}$ are the sets of all integer indices for the beliefs and goals, respectively. Moreover, θ_{ij}^{RIS} and $\theta_{i\ell}^{\text{RIS}}$ are parameters that relatively weigh the influence of, respectively, one's beliefs and goals. The parameter θ_i^{RIS} quantifies the (fixed) neutral intention (i.e., the intention that would be realized when beliefs and goals are zero) on the i^{th} intention.

Finally, the rational action selection subprocess receives $\boldsymbol{y}^{\text{RIS}}(k)$ as input, and outputs vector $\boldsymbol{y}^{\text{RAS}}(k)$, which includes all elements of the action taken by the human at time

step k. Each element of the action vector is simply a binary variable, with 1 implying that the action is selected, and 0 implying that the action is excluded. The process of transitioning the quantified strength of the intentions into binary actions via rational action selection depends on various practical aspects of the HSRI setup, such as the level of abstraction of the actions (e.g., whether only one or multiple actions are required to fulfill an intention) and any constraints that restrict the actions. In general, this process is defined depending on the HSRI setup.

3.2.2. Identification of the Model

The formulations given in Section 3.2.1 for MMM show that the model includes parameters that need to be identified (preferably personalized for each user) using approaches based on, e.g., gradient descent (Bishop & Bishop, 2024) or genetic algorithms (Kramer, 2017). These parameters of MMM arise from the theoretical formalization of the model, which is grounded in cognitive and psychological theories, such as ToM (see Chapter 2 for details). In particular, these theories describe how mental states interact with each other, how they are formed through perception, and how they lead to actions via decision-making processes. The mathematical formulation of MMM expresses these mechanisms through parameterized functions, where the parameters determine how strongly variables influence one another and, in the case of the cognition module, how quickly the affected states respond to those influences. MMM has been presented comprehensively so that it can be adopted in different contexts involving HSRIs. To achieve this, it includes multiple parameters. For a particular context, however, not all the modeled inter-dynamics, and thus parameters, are relevant. Therefore, a crucial consideration in the identification of MMM for a particular context is to avoid redundancy of the parameters, particularly because these parameters mostly correspond to real-life concepts related to the mental processes.

This redundancy particularly occurs when the structure of the model is too complex for the size of the dataset that is available and relevant in a particular context to train the parameters. While performing longer and wider ranges of interactions may allow the collection of more data, this is often not desirable/possible in HSRIs. Moreover, larger datasets do not necessarily guarantee to circumvent the issue, especially when the collected data is not diverse enough for the complexity level of the model.

Hence, based on the context and the concepts that define the realizations of the variables, linkages, and parameters of MMM, one may need to gradually, and systematically, simplify the model structure. While this is context-dependent, we propose two main steps that together allow evaluating and reducing the redundancies in the model every time it is simplified for a given context. This simplification, which includes removing some linkages, variables, and parameters, alongside these two steps can ensure proper identification of the MMM. These steps include parameter redundancy evaluation, given in Algorithm 3.1, and a multi-stage identification procedure with warm-start, explained via Algorithms 3.2–3.4.

Evaluating the presence of redundant parameters in MMM

In order to evaluate whether the model needs to be further simplified, the parameters that are unequivocally identifiable in the current structure of MMM are established using Algorithm 3.1. The MMM will be simplified by gradual, systematic removal of variables and linkages whose corresponding parameters have not been identified unequivocally. This continues until all remaining parameters are unequivocally identifiable.

Note that the identification procedure involves solving generally non-convex optimization problems (i.e., minimizing the error of the values that are generated by the model with regard to the corresponding values collected from real-life HSRIs). Therefore, the values that are returned by the optimizer may correspond to local, rather than global, optima. To circumvent the issue properly, the identification optimization is performed n^{run} independent times, each time with different starting values for the parameters. Then, the n^{opt} (where $n^{\text{opt}} \leq n^{\text{run}}$) sets of parameters that yielded the least realized value for the error function in the validation stage are chosen. In case the standard deviation of the n^{opt} chosen values per parameter is small enough (i.e., less than a small threshold), the parameter is deemed identified.

Algorithm 3.1: Procedure to determine the parameters of MMM that are unequivocally identified

Local variables

r: Index of current identification run
p: Index of parameter
n^{run}: Number of independent identification runs
n^{opt}: Number of optimal identification runs considered in assessment
n^{par}: Number of parameters considered
$\theta_{p,r}$: Identified value of parameter of index p in run r
$\boldsymbol{\theta}_p^*$: Vector of dimension n^{opt} including the identified values of parameter of index p in the n^{opt} runs with minimum optimization cost

Procedure

1: Run identification procedure n^{run} times, each time with different, random starting values for all n^{par} parameters
2: $\boldsymbol{r}^* \leftarrow$ indices of n^{opt} runs with minimum optimization cost in the validation stage
3: **for** $p \in \{1, ..., n^{\text{par}}\}$ **do**
4: $\quad \boldsymbol{\theta}_p^* \leftarrow \theta_{p,r} \qquad \forall r \in \boldsymbol{r}^*$
5: $\quad \sigma_p \leftarrow$ standard deviation of $\boldsymbol{\theta}_p^*$
6: $\quad$ **if** $\sigma_p \leq$ threshold **then**
7: $\quad\quad \theta_p$ was unequivocally identified
8: $\quad$ **else**
9: $\quad\quad \theta_p$ was not unequivocally identified

Multi-stage optimization with warm-start

After collecting measurements for the beliefs, goals, emotions, and actions of a person, for the sake of simplicity, the identification optimization may be decoupled for the

perception, cognition, and decision-making modules. By knowing the beliefs, goals, and actions (i.e., the inputs and outputs of the decision-making module, as it is illustrated in Figure 3.3), it is straightforward to decouple this module from the other two modules. Contrarily, the perception and cognition modules are less trivial to decouple since the output of the perception module that is then injected into the cognition module is an auxiliary variable, and thus cannot be measured, e.g., by asking people about it in the course of HSRIs.

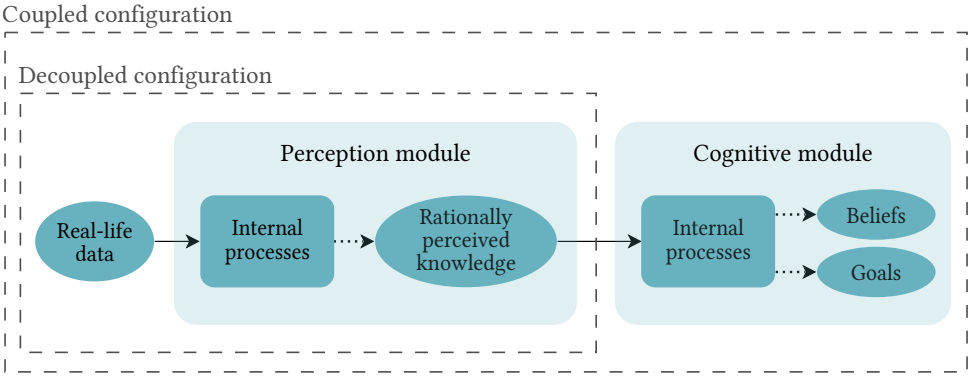


Figure 3.4: The two configurations used in the two-stage identification of the ToM model. The *coupled configuration* includes both the perception and cognition modules, while the *decoupled configuration* includes only the perception module.

Therefore, we propose dividing the identification procedure into multiple optimization stages. To achieve this, we introduce two approaches, A and B, to perform the identification optimization. These approaches are given in Algorithms 3.3 and 3.4. Both approaches start with a preliminary identification of the parameters of the perception module, given by Algorithm 3.2, whose outcome is then used as a warm-start. In this case, the perception module of the model is decoupled from the cognition module and is solely considered (called the *decoupled configuration* in Figure 3.4). In the *decoupled configuration*, the inputs and outputs are, respectively, real-life data and rationally perceived knowledge. Since the values of the rationally perceived knowledge are not measurable, the pre-identification for the *decoupled configuration* is performed under the assumption that each element of the rationally perceived knowledge for time step k is the same as the corresponding element of the belief for the same time step k (i.e., assuming the bias is null).

After obtaining a warm-start for the parameters of the perception module, one of the following two approaches is adopted to identify the entire model.

Approach A The perception and cognition modules are simultaneously identified, considering the *coupled configuration* shown in Figure 3.4. The starting values in the identification optimization for the parameters of the perception module are those obtained in the pre-identification stage explained above. The starting values for the

Algorithm 3.2: Generating the warm-start for the parameters of the perception module

Local variables

r: Index of current identification run
r^*: Index of run with minimum optimization cost
n^{run}: Number of independent runs
$\boldsymbol{\theta}_r^{\text{per}}$: Vector containing identified values for parameters of perception module in run r
$\boldsymbol{\theta}^{\text{per-ws}}$: Vector containing warm-start of parameters of perception module
$y_\ell^{\text{RR}}(k)$: Realized value of element ℓ of rational reasoning at time step k
$x_\ell(k)$: Realized value of element ℓ of belief at time step k
$\mathcal{B}$: Set of all integer indices of beliefs.

Procedure

1: Assume $y_\ell^{\text{RR}}(k) = x_\ell(k) \qquad \forall \ell \in \mathcal{B}$
2: **for** $r \in \{1, ..., n^{\text{run}}\}$ **do**
3: Randomly initialize $\boldsymbol{\theta}_r^{\text{per}}$
4: Optimize $\boldsymbol{\theta}_r^{\text{per}}$ within *decoupled configuration*
5: $\boldsymbol{\theta}^{\text{per-ws}} \leftarrow \boldsymbol{\theta}_{r^*}^{\text{per}}$ for run r^* with minimum optimization cost

parameters of the cognition module are randomly selected. Approach A is described in detail in Algorithm 3.3.

Algorithm 3.3: Model identification with warm-start - Approach A

Local variables

r: Index of current identification run
n^{run}: Number of identification runs
$\boldsymbol{\theta}^{\text{per-ws}}$: Vector containing warm-start of parameters of perception module obtained in Algorithm 3.2
$\boldsymbol{\theta}_r^{\text{per}}$: Vector containing identified values for parameters of perception module in run r
$\boldsymbol{\theta}_r^{\text{cog}}$: Vector containing identified values for parameters of cognition module in run r

Procedure

1: **for** $r \in \{1, ..., n^{\text{run}}\}$ **do**
2: $\boldsymbol{\theta}_r^{\text{per}} \leftarrow \boldsymbol{\theta}^{\text{per-ws}}$ (initialization)
3: Randomly initialize $\boldsymbol{\theta}_r^{\text{cog}}$.
4: Optimize $\boldsymbol{\theta}_r^{\text{per}}$ and $\boldsymbol{\theta}_r^{\text{cog}}$ within *coupled configuration*
5: $\boldsymbol{\theta}_{r^*}^{\text{per}}, \boldsymbol{\theta}_{r^*}^{\text{cog}} \leftarrow \boldsymbol{\theta}_r^{\text{per}}, \boldsymbol{\theta}_r^{\text{cog}}$ for run r with minimum optimization cost

Approach B Before identifying all parameters of the perception and cognition modules together, a pre-identification for the parameters of the cognition module is performed considering the *coupled configuration* in Figure 3.4. At this stage, the parameters of the perception module are fixed to the values obtained in the warm-start,

and only the parameters of the cognition module are identified (see steps 1–4 of Algorithm 3.4). Afterward, both perception and cognition modules are warm-started by the values of their pre-identified parameters and a final identification optimization is performed on both modules (see steps 6–10 of Algorithm 3.4).

Algorithm 3.4: Model identification with warm-start - Approach B

Local variables

r: Index of current identification run
n^{run}: Number of identification runs
$\boldsymbol{\theta}^{\text{per-ws}}$: Vector containing warm-start of parameters of perception module obtained in Algorithm 3.2
$\boldsymbol{\theta}_r^{\text{per}}$: Vector containing identified values for parameters of perception module in run r
$\boldsymbol{\theta}_r^{\text{cog}}$: Vector containing identified values for parameters of cognition module in run r
$\boldsymbol{\theta}_1^{\text{cog}*}, \ldots, \boldsymbol{\theta}_{n^{\text{opt}}}^{\text{cog}*}$: Vectors containing identified values for parameters of cognition module in the n^{opt} runs with the minimum optimization cost until step 6

Procedure

1: **for** $r \in \{1, \ldots, n^{\text{run}}\}$ **do**
2: $\quad \boldsymbol{\theta}_r^{\text{per}} \leftarrow \boldsymbol{\theta}^{\text{per-ws}}$ (initialization)
3: $\quad$ Randomly initialize $\boldsymbol{\theta}_r^{\text{cog}}$
4: $\quad$ Optimize $\boldsymbol{\theta}_r^{\text{cog}}$ with *coupled configuration*
5: $\boldsymbol{\theta}_1^{\text{cog}*}, \ldots, \boldsymbol{\theta}_{n^{\text{opt}}}^{\text{cog}*} \leftarrow \boldsymbol{\theta}_r^{\text{cog}}$ of n^{opt} runs with smallest realized values for optimization cost
6: **for** $r \in \{1, \ldots, n^{\text{opt}}\}$ **do**
7: $\quad \boldsymbol{\theta}_r^{\text{per}} \leftarrow \boldsymbol{\theta}^{\text{per-ws}}$ (initialization)
8: $\quad \boldsymbol{\theta}_r^{\text{cog}} \leftarrow \boldsymbol{\theta}_r^{\text{cog}*}$ (initialization)
9: $\quad$ Optimize $\boldsymbol{\theta}_r^{\text{per}}$ and $\boldsymbol{\theta}_r^{\text{cog}}$ with *coupled configuration*
10: $\boldsymbol{\theta}_{r^*}^{\text{per}}, \boldsymbol{\theta}_{r^*}^{\text{cog}} \leftarrow \boldsymbol{\theta}_r^{\text{per}}, \boldsymbol{\theta}_r^{\text{cog}}$ for run r with optimal optimization cost

On the one hand, approach A has the advantage of being comprised of less steps, which entails less computational load. Moreover, this approach allows for a more flexible search through the parameter spaces, since all parameters are optimized together, without imposing any constraints.

On the other hand, approach B is first constrained with fixed parameters considered for the cognition module. This systematic way of exploration for the optimizer can lead to more parameters being uniquely identified, at the expense of diminishing the flexibility of the search by the optimizer.

3.2.3. Implementation of MMM

To facilitate the use of MMM, this section summarizes how the model is configured and executed for real-world applications. The implementation involves two complementary procedures: (1) an offline setup, which outlines the procedural steps to prepare and initialize MMM for a specific context, and (2) an online execution, which demonstrates

how MMM computes perception, cognition, and decision-making variables per discrete time step during its real-time execution. Together, these two procedures illustrate how the mathematical formulation of MMM presented in Section 3.2.1 is used in practice, relying on the parameter identification process described in Section 3.2.2.

The offline setup, presented in Algorithm 3.5, outlines the procedural steps to configure and instantiate the MMM for a specific context. It begins with defining the variables relevant to the intended application. Then, the mathematical expressions that describe the rational reasoning process (see (3.2)), the equation describing the rational action selection, and the ranges $\mathcal{X}_{j,m}$ in (3.6) are specified. Although most components of MMM are already provided in a general parameterized formulation that can be directly applied in practice, as introduced in Section 3.2.1, the three components mentioned above depend strongly on the context and are therefore tailored at this stage. Finally, the offline setup also includes the identification of the model parameters (detailed in Section 3.2.2).

Algorithm 3.5: Offline setup of MMM

1: For each variable group (e.g., perceived data, beliefs, intentions), declare the variables relevant in the context
2: Specify the functions $f_{ij}^{\mathrm{RR}}(\cdot)$ of the rational reasoning process, introduced in (3.2)
3: For each cognitive variable, define the set of distinct ranges $\mathcal{X}_{j,m}$ of the influencing variables, in (3.6);
4: Define equation of rational action selection;
5: Identify parameters of the model, following the algorithms explained in Section 3.2.2.

The online execution demonstrates how MMM processes one discrete time step during real-time interactions, explicitly linking the sequence of equations across perception (via (3.1)—(3.2)), cognition (via (3.3)—(3.6)), and decision-making (via (3.7). MMM follows a discrete-time state space representation, in which beliefs, goals, emotions, and auxiliary variables evolve as state variables through difference equations. At each step, real-life data is measured, the perception module produces rationally perceived knowledge, the cognition module updates the mental states, and the decision-making module produces the actions of the human, based on those mental states. The procedure is summarized in Algorithm 3.6 and in Figure 3.5.

In accordance with Remark 8, perception is considered instantaneous with respect to cognition and decision-making. Thus, per update of MMM, perception is executed once (i.e., (3.1)–(3.2) are computed), and its output is fed into the cognitive module. Subsequently, the cognitive and decision-making modules are executed. The frequency of this loop corresponds to the internal update rate of MMM, which governs the evolution of the state variables. This internal update rate may differ from the interaction or control frequencies, as these external processes may run at slower rates and therefore sample the output of MMM less frequently than its internal update rate.

Algorithm 3.6: Online execution of MMM

1: Measure real-life data $u_i(k^{\mathrm{p}})$
2: Execute perceptual access: Compute (3.1) for $i = 1, \ldots, |\mathcal{R}|$
3: Execute rational reasoning: Compute (3.2) for $j = 1, ..., |\mathcal{K}|$
4: Calculate current realizations of the weights of the cognitive module
   $z_{\ell i}(k)$: Compute (3.4b) for all i and all ℓ
   $w_{ij}(k)$: Compute (3.6) for all i and for all j except $i \neq j$
5: Calculate next realization of each variable $x_i(k+1)$: Compute (3.3) for all i
6: Execute rational intention selection: Compute (3.7) for each $i \in \mathcal{I}$
7: Execute rational action selection

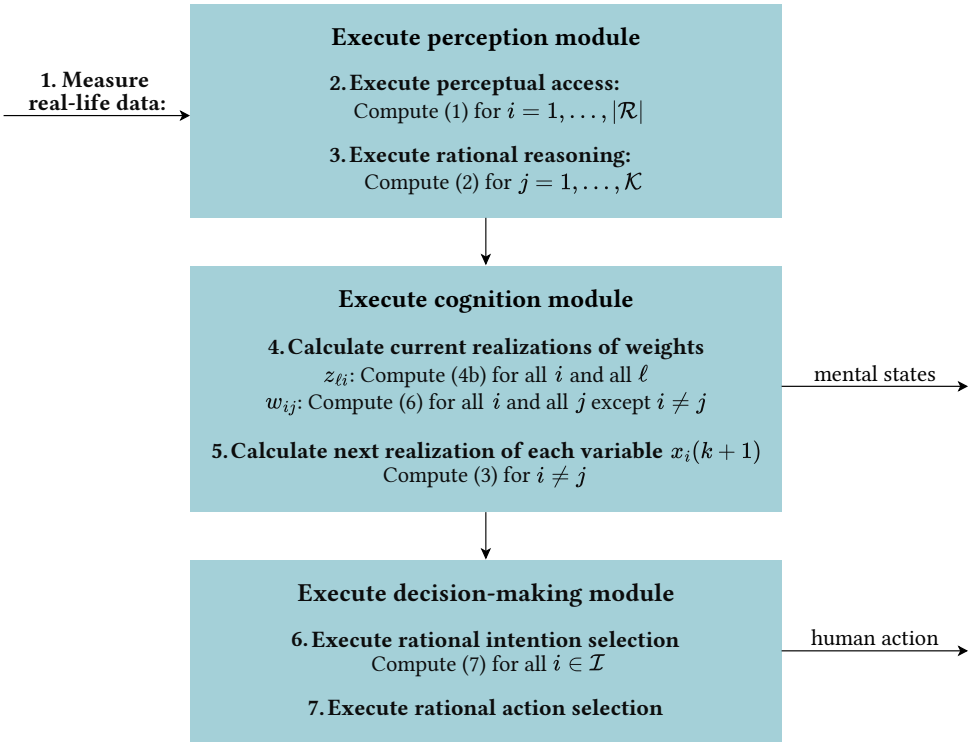


Figure 3.5: Overview of the online execution of the MMM. At each discrete time step k, real-life data is measured, the perception module produces rationally perceived knowledge (via (3.1)–(3.2)), the cognition module updates the mental states (via (3.3)–(3.6)), and the decision-making module generates human actions (via (3.7)).

3.2.4. Closed-loop Model-based Control of SRs

Once the model is mathematically and dynamically formulated, after identifying its parameters for a particular human, the model is embedded within a closed-loop control system for SRs to be used in interactions with the human. Figure 3.6 shows the architecture of the Model-Based Controller (MBC) system and illustrates how the

dynamic ToM model, MMM, is integrated within the control loop.

The MBC system generates a control input $\boldsymbol{u}_k$, which includes all the controlled interactive actions of the robot, such that given criteria for the HSRI are met. In general, satisfying such criteria is associated with optimizing given objectives (e.g., maximizing the interaction time, the user engagement, or satisfaction). To do so, an optimizer may be used in the control loop (see Figure 3.6) to suggest candidate control inputs. Note that such a controller, in general, works upon a prediction window, across which candidate control inputs are generated (for the current and all future time steps within the prediction window). In case the size of the prediction window is unity, the MBC system looks only one step ahead (i.e., considers only the impact of the current control input on the HSRI).

In general, every HSRI is subject to various constraints that impact the actions of the SR. For instance, if the SR should offer any joint activities to a person, it should consider the physical and cognitive limitations of the person in making the suggestion. Therefore, the optimization problem of the MBC system is in general a constrained one (see Figure 3.6).

Subsequently, the candidate control inputs are fed into the dynamic ToM model, which predicts the mental states (shown by x in the figure) of the human as a result of these stimuli (i.e., suggested interactive actions of the SR), as well as the corresponding actions by the human (shown by y in the figure). In case these fulfill the desired objectives and satisfy the imposed constraints, the candidate control inputs are selected for the SR. Otherwise, the optimizer should adjust and propose new candidate control inputs, and the process is repeated.

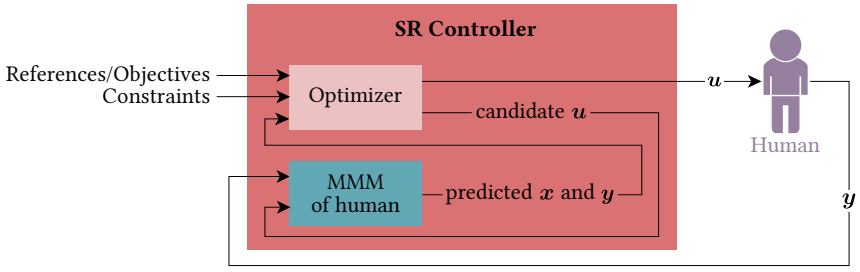


Figure 3.6: Block diagram of the model-based controller of the SR using the ToM model.

By following such an approach, it is possible to ensure that the HSRIs comply with pre-defined constraints (e.g., keeping the user engagement above a certain level) and that the interactive actions of the SR result in maximizing the given criteria (e.g., sustaining the HSRIs as long as possible), by systematically incorporating the current (and future) mental states and actions of the user within the control loop.

3.2.5. Assumptions Underlying MMM and the Proposed Framework

MMM and its integration within an MBC rely on a set of underlying assumptions that delineate the current scope of applicability of the framework and define operational boundaries. Therefore, clarifying these assumptions is essential for setting realistic

expectations regarding the performance of MMM and the framework, understanding the trade-offs involved in practical deployment, and identifying adjustments required for broader applicability across diverse users and contexts.

Assumption 1: Boundedness of cognitive variables

All cognitive variables of MMM are bounded to the interval $[-1, 1]$ to ensure stability and to provide a shared numerical scale across users. Nevertheless, personalization is preserved because the mapping between the intensity of a mental state and its numerical realization is user-specific. For example, a moderate level of engagement may be represented as 0.4 for one user and 0.6 for another. Similarly, the value 0.4 may not convey the same level of engagement across users, as it reflects their individually identified cognitive dynamics. However, this boundedness may limit the ability of MMM to capture atypical or extreme behaviors. For example, if users exhibit extreme emotional responses that deviate substantially from those observed in the interactions used for model identification, this may lead to realizations of the mental states that fall outside the interval $[-1, 1]$. This limitation reflects the fact that MMM is designed to capture representative cognitive patterns rather than unlikely outliers. If atypical or extreme responses do occur and play a central role in an application, the model can be re-identified to include such dynamics.

Assumption 2: Simplified update rules for state evolution

The dynamics of the state variables are represented mathematically using piecewise-linear approximations, rather than fully non-linear formulations (see (3.6)). This design choice keeps the model computationally tractable and suitable for real-time execution and control. However, it may limit the capacity of MMM to capture highly complex or abrupt cognitive shifts. The impact of this assumption may be mitigated by increasing the number of piecewise regions — supported by the proposed mathematical framework — to better capture the nonlinearity of the dynamics of the state variables. Such refinement, however, entails a trade-off between improved accuracy and higher computational demand of the model.

Assumption 3: Modular structure and decoupling for identification

MMM is structured based on distinct modules for perception, cognition, and decision-making processes, as is common in cognitive architectures. To address data scarcity in typical HSRI scenarios, the proposed identification algorithms (explained in Section 3.2.2) temporarily decouple perception and cognition modules in the first step of the multi-stage identification process (see Algorithm 3.2). Some parameters of the perception module are pre-identified under a zero-bias assumption (i.e., rationally perceived knowledge and beliefs are considered the same) to provide warm-start values for the subsequent identification process. A final joint optimization stage refines all parameters in the fully coupled configuration, effectively restoring perception-cognition dependencies and improving accuracy. However, in extreme cases – particularly for users with strong emotional biases — the initial warm-start may skew the search space toward local optima. Nevertheless, the identification process inherently mitigates

this effect by performing multiple independent gradient-descent runs from different initialization points and by bounding parameter ranges.

Assumption 4: Neurotypical user modeling

MMM is grounded in cognitive theories that reflect neurotypical users. When applied to users with different cognitive profiles (e.g., neurodivergent or cognitively impaired individuals), adjustments to some components of the model (such as parameter tuning or minor structural refinements) may be required to better reflect individual cognitive dynamics. Thus, future studies involving neurodivergent or cognitively impaired users may be needed to refine the model's applicability to these user groups.

Overall, the four assumptions outlined above define the boundaries of MMM in its current representation, while providing a foundation for future refinements and for potential extensions of the framework.

3.3. Case study

To assess our framework, we carried out an experiment involving HSRIs with 10 volunteer human participants. In these HSRIs, each participant solved multiple chess puzzles while interacting with a Nao robot. Each participant interacted with Nao in 3 sessions of 45 to 90 minutes.

3.3.1. Experimental Setup

This section outlines the technical setup, interaction design, participant protocol, and measurement methods used in the experimental study.

Setup

The experimental setup included a Nao robot (a programmable humanoid robot), a display, a microphone, and a mouse, as shown in Figure 3.7. In each experiment session, the participant was seated at a table, with the Nao robot placed in front, facing them. A display, which was placed between the user and the robot, showed the chess puzzles, instructions, and questions to the participant during the session. A desktop microphone was placed on the side of the screen to facilitate the verbal communication of the participants with Nao, allowing more robust communication than when the microphones of the robot were used. Finally, a mouse was placed on the other side of the display to enable the participants to play the puzzles. A laptop was connected to both the Nao robot (via WiFi), the display, and the microphone (via a wired connection). This setup enabled the autonomous control of Nao through Python and facilitated data exchange during the sessions. On the third session with each participant, a camera was placed on top of the display pointing to the participant to record their facial expressions and status during the session. This data was used afterward to analyze the engagement of the participants. Finally, the chess puzzles were taken from the *lichess.org* database (Lichess.org, 2024).

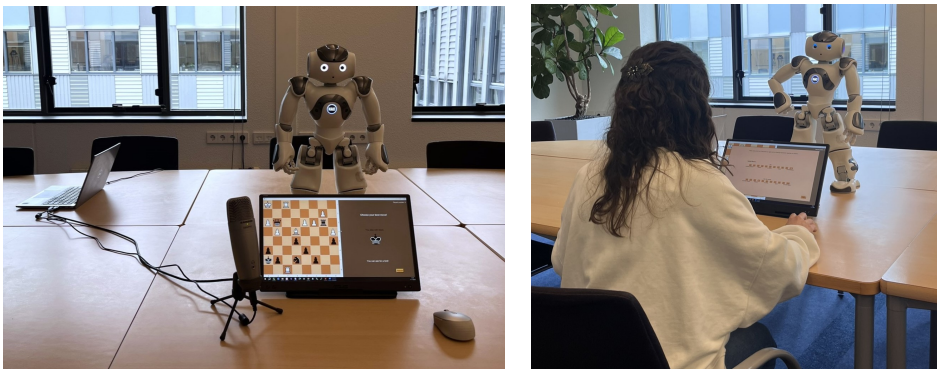
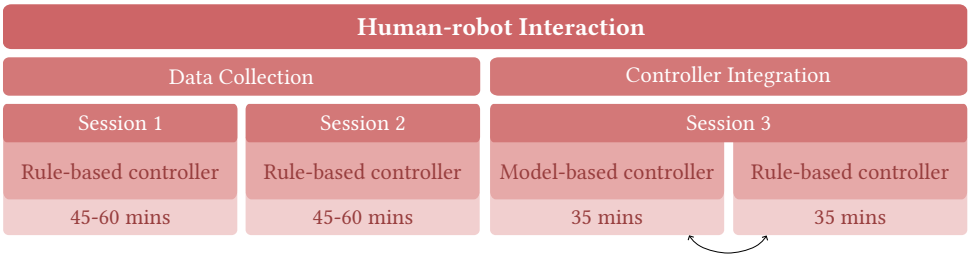


Figure 3.7: Experimental setup: The participant sat at a table in the room. The Nao robot was placed on top of that table, facing the participant. A display was placed on the table between Nao and the participant. The display showed the chess puzzles and the questions asked to the participant during the session. A mouse and a microphone were placed on the sides of the display. The microphone and the display were both connected to a laptop that managed the HSRIs and that was connected via Wi-Fi to Nao to steer its behavior via Python.

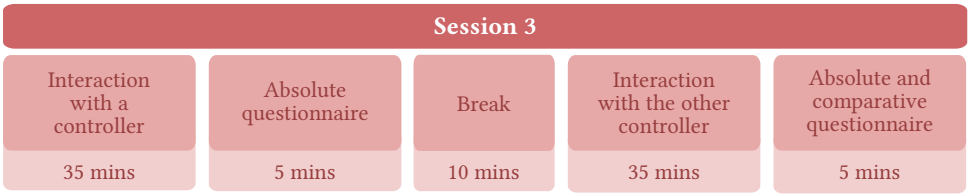
Design of interaction sessions

The experiment with each participant consisted of 3 sessions, where the details are given in Figure 3.8. The first two sessions were designed to collect data about the changes in the relevant mental states of the participant in order to identify the MMM for that participant (personalization). In the third session, the identified model was embedded in the controller of the robot, so that the robot interacted with the participant following the method described in Section 3.2.4 and based on the personalized MMM (see Figure 3.8a).

In the first two sessions, the participants were asked to solve at least 18 chess puzzles or to continue solving puzzles for at least 45 minutes. After playing 30 puzzles or solving puzzles for 60 minutes, the session was ended nonetheless. The participants were in all cases allowed to terminate or quit the session any time earlier, if they wished to. The last session consisted of 2 interactions of 35 minutes each. After each interaction, the participant was given a questionnaire to complete. Moreover, a ten-minute break was given between the two interactions (see Figure 3.8b). In one of the two interactions in this session, the participant interacted with a version of Nao that was steered via the MBC embedding the MMM identified for that participant. In the other interaction, Nao was controlled by a model-free, rule-based controller that resembled the current simplified steering systems used for SRs. This controller (specified as the standard controller in Figure 3.8) was also used to steer the behavior of Nao in the first two sessions designed for data collection. Half of the participants interacted with the standard controller in the first interaction of the third session, while the other half interacted with it in the second interaction of the third session. For each participant, the sessions were at least one week apart from each other, and the entirety of the three sessions was conducted within one month.



(a) Description of the experiment layout, including the three sessions, their purposes, and durations.



(b) Detailed layout of the third session of the experiment, where the HSRI using our proposed model-based steering framework, including MMM, was assessed.

Figure 3.8: Experiment layout including three separate sessions.

Execution of sessions

Before the start of the experiment with each participant, the participant was informed about how to interact with Nao and how to play the puzzles via a briefing document. Those participants who had little or no experience with chess could request an extra document that described the basic moves and rules of chess. Then, the experiment started with a brief introduction given by Nao which summarized the most essential points described in the briefing document. Subsequently, the first puzzle showed up on the screen. The debriefing documents can be found in Morão Patrício and Jamshidnejad (2025a).

Each puzzle consisted of a sequence of 2 to 6 most advantageous moves that the participant had to perform correctly. To perform a move, the participant had to click on the desired piece first, and then on the square that the piece was supposed to move to. If the move was correct, the piece was placed in the new square and the next move was made by the program. Otherwise, the move was refused and the participant had to try again until the right move was done. On the right-hand side of the screen, the color that the participant had to play with and the number of puzzles that the participant had completed so far were displayed while the participant was solving a puzzle.

In the first and second sessions, twice per puzzle, the participant was asked to answer questions about their mental states. More specifically, the participant was asked to quantify their beliefs, goals, and emotions relevant for the situation, in a discrete range from 0 to 10, additionally categorized by qualitative terms from “completely disagree” to “completely agree”. The questions showed up on the screen, and the participant had to reply by choosing a number using the mouse. These questions are presented in Appendix B.4.

While playing, participants could verbally ask Nao for a hint, to skip the current puzzle, or to quit the session. Nao would then confirm the request verbally with the participant. When participants asked Nao for the first hint about a puzzle, it provided a general tip about the objective of the puzzle (e.g., the objective was to check-mate the opponent). If the participant asked Nao for a second hint about the same puzzle, it would reveal which piece had to be moved. In response to requesting a third hint, if it concerned the same move, Nao would reveal the optimal move of the piece. If the hint concerned a new move, Nao would again reveal the piece to be moved. When a participant asked to skip the current puzzle, that puzzle disappeared and the next puzzle showed up on the screen. At the end of each puzzle, the participant could request more difficult or easier puzzles, by replying to a question about this that appeared on the screen. All this information was included in the briefing document that was given to the participants at the beginning of the sessions. This document can be found in Morão Patrício and Jamshidnejad (2025a).

Participants of the experiments

Ten volunteers aged between 25 and 35 years old participated in the experiment. The participant pool consisted of 30% women and 70% men.

The sample size was determined by the practical constraints of the experimental protocol and the objectives of the study. To obtain multiple and representative measurements for identification and validation of the model under realistic HSRI conditions, each participant completed three sessions of 45 to 90 minutes, spaced at least one week apart. Given the duration and intensity of this multi-session design, recruiting a larger participant pool was not feasible within the scope of this study. Comparable sample sizes are common in early-stage HSRI research, where experiments are time-intensive and primarily aimed at demonstrating feasibility and validating the methodological approach, rather than achieving broad statistical generalization.

Role of Nao

Besides giving general indications about the chess puzzles to the participants and socially interacting with the participants, Nao also decided the difficulty level of the puzzles and whether to give rewards to the participants, in order to keep them engaged.

Regarding the difficulty level of the puzzles, six levels of difficulty were defined based on the ratings of the puzzles given in the *lichess.org* database (Lichess.org, 2024) (where the rating of a puzzle is a measure of its difficulty). For each difficulty level, a range of the rating was considered and the puzzles with a rating in that range were included. The ratings that corresponded to each difficulty level considered in this case study can be found in Table B.1.1 of Appendix B.1.

To give rewards to the participants, Nao performed one of the following entertaining movements or gestures lasting around 10 to 30 s: pretending to play a guitar (posing as if the robot was holding a guitar, but there was no real guitar in the room), dancing, performing tai-chi, pretending to be an elephant, pretending to take a photo of the participant (with an imaginary camera). These movements are presented in

Appendix B.3.

During the first two sessions, the actions of Nao were carefully scheduled as to instigate different beliefs, goals, and emotions for the participants, and to provide a diverse dataset to be subsequently used to identify the MMM per participant. Thus, to stimulate the mental states as widely as possible, and to capture the dynamic effects of increasing or decreasing the difficulty level of the puzzles when the participant was in different mental states, we distributed the difficulty levels of the puzzles to include both increasing and decreasing sequences. The difficulty level was kept identical for three puzzles in a row to ensure that the data gathered truly captured the effects of solving puzzles of that difficulty for the participant. Then this difficulty level would increase or decrease with a step size of 2. More specifically, in the first session, the sequence of the levels of difficulties was $\{0, 2, 4, 4, 2, 0\}$, meaning that the difficulty level started at 0 (for three consecutive puzzles), increased to 2 (also for three consecutive puzzles), and then to 4. After playing two groups of three puzzles of difficulty 4, the difficulty level decreased to 2, and then to 0. In the second session, the sequence of the levels of difficulties was $\{5, 3, 1, 1, 3, 5\}$. In these sessions, if the number of puzzles that the participants solved surpassed 18 puzzles, then the difficulty level of puzzle number $18 + n^{\text{puzzle}}$, with n^{puzzle} an integer, was the same as the difficulty level of puzzle n^{puzzle} (taking into account that the maximum number of puzzles offered per session was 30). See Table B.1.2 of Appendix B.1 for the detailed distribution of the difficulty levels in the first two sessions of the experiments.

As for giving rewards to the participants, in the first two sessions, Nao randomly performed one celebratory movement or gesture for each set of three puzzles with the same difficulty level. Note that the main objective of this design choice was to capture the dynamic effects of giving a generally encouraging/entertaining reward on the mental states of the participants, rather than the specific design of such rewards.

Measurement methods

Three complementary measurement methods were used in this case study: (1) self-reported feedback during interactions, (2) post-interaction questionnaires, and (3) automated video analysis. Self-reports were used to obtain the ground truth of user mental states, whereas all three methods contributed to assessing user engagement and overall interaction quality.

The first method — self-reported feedback during interactions — consisted of participants providing self-reported ratings of their mental states (i.e., beliefs, goals, emotions, and intentions) at regular intervals upon request during the interaction sessions. Responses were recorded on a discrete 0–10 scale with qualitative anchors ranging from completely disagree to completely agree (see Appendix B.4 for the complete list of questions).

In the first two sessions, self-reports were collected twice per puzzle and after hint requests, providing the data required for identifying and validating the MMM parameters. In the third session, feedback was requested if more than 150 s had passed since the last feedback request or every three puzzles. In this session, the responses were used both to reinitialize the user's mental states estimated by MMM, which are required by the controller, and to assess the quality of the ongoing interaction between

the participant and Nao. To minimize unreliability and inconsistency, participants were briefed on the importance of honest responses, and the questionnaire combined numerical scales with qualitative anchors. The frequency of the feedback requested from participants was carefully selected to balance the need for sufficient data for model identification against the risk of distracting participants or causing fatigue, which could otherwise compromise response quality.

The second method consisted of post-interaction questionnaires. In the final session, after each of the two interactions, participants completed a standardized 7-point Likert-scale questionnaire assessing their engagement level and their perceived robot awareness, adaptability, and personalization (available in the experimental dataset Morão Patrício and Jamshidnejad (2025a)). These questionnaires provided a complementary subjective evaluation of engagement and interaction quality.

During preliminary testing, a scale drift was observed between the two questionnaires, which were administered approximately 45 minutes apart, suggesting that the time gap between administrations may have affected the consistency of responses. Therefore, a comparative questionnaire was added at the end of the third session to allow the participants to directly compare the two interaction conditions, after first completing the independent absolute questionnaires for each condition. This approach enhanced the reliability and interpretability of the questionnaire results.

Finally, the third method — an automated, computer-vision-based engagement detection tool, presented in Duchetto et al. (2020) — was additionally used in the final session as a complementary, non-intrusive measurement of engagement that did not rely on user self-reports. This tool estimated engagement levels of participants on a continuous scale from 0 (no engagement) to 1 (full engagement). However, since the engagement detection tool had been trained on robot-centric interactions with a single focus point, whereas participants in our setting alternated attention between the robot and the puzzle screen, its application involved a multi-focus interaction context differing from the original training conditions. Accordingly, the results obtained from this method are reported together with their statistical significance and are interpreted with caution in Section 3.4.

The integration of multiple complementary measurement methods in our experiments enabled cross-validation of the model and controller, ensuring a reliable evaluation of the proposed framework.

3.3.2. Model-based Controller for Nao

The MBC steering system of Nao used in the third session of this case study had to decide, at the end of each puzzle, about the level of difficulty of the next puzzle and about whether to give a reward to the participant. This decision had to optimize the mental states of the participant according to a cost function that depended on the model identified for each participant. Next we describe the details for designing the MBC steering system of Nao.

MMM for participants

One of the crucial elements of an MBC steering system is a model of the process (i.e., the perception, cognition, and decision-making of the participants) that should be impacted in desired ways by the designed steering system. This section describes the mathematical modeling of the perception, cognition, and decision-making modules for the participants of this case-study, based on the MMM detailed in Section 3.2.1. A graphical representation of each module is given in Appendix B.2.

Table 3.1: Concepts that represent the real-life data and perceived data of the perception module for the case study.

Real-life data	Perceived data (corresponding to the real-life data in the same row of this table)	Concept	Type
$u_1(k^{\mathrm{p}})$	$y_1^{\mathrm{PA}}(k^{\mathrm{p}})$	Puzzle difficulty level	Integer
$u_2(k^{\mathrm{p}})$	$y_2^{\mathrm{PA}}(k^{\mathrm{p}})$	Number of hints	Integer
$u_3(k^{\mathrm{p}})$	$y_3^{\mathrm{PA}}(k^{\mathrm{p}})$	Number of wrong attempts	Integer
$u_4(k^{\mathrm{p}})$	$y_4^{\mathrm{PA}}(k^{\mathrm{p}})$	Time to solve a puzzle	Continuous
$u_5(k^{\mathrm{p}})$	$y_5^{\mathrm{PA}}(k^{\mathrm{p}})$	Skipped puzzle?	Boolean
$u_6(k^{\mathrm{p}})$	$y_6^{\mathrm{PA}}(k)$	Reward given?	Boolean

Perception module The perception module, as explained before (see Figure 3.1), is in general composed of perceptual access and rational reasoning subprocesses. Table 3.1 shows the pieces of real-life data and perceived data that are relevant for and used in the case study. The list of the pieces of rationally perceived knowledge is given in Table 3.2. In this case study, due to the nature and setup of the experiments, it is sensible to assume that the participants perfectly observe all relevant pieces of real-life data. Thus, based on (3.1), for all perception time steps k^{p}, for the perceptual access we have $y_i^{\mathrm{PA}}(k^{\mathrm{p}}) = u_i(k^{\mathrm{p}})$, where y_i^{PA} is a piece of perceived data (i.e., an output from the perceptual access subprocess) and $u_i(k^{\mathrm{p}})$ is its corresponding input real-life data. As given in Table 3.1, for this case study, $i = 1, \ldots, 6$.

To model the rational reasoning subprocess, which is formulated via (3.2) in MMM, the relationships between all pieces of the perceived data $y_i^{\mathrm{PA}}(k^{\mathrm{p}})$ for $i = 1, \ldots, 6$ (i.e., the inputs to the subprocess) and the corresponding piece of rationally perceived knowledge $y_j^{\mathrm{RR}}(k^{\mathrm{p}})$ (i.e., the outputs of the subprocess) should be defined by providing relevant expressions for function $f_{ij}^{\mathrm{RR}}(\cdot)$ in (3.2). Figure B.2.1 illustrates the pieces of perceived data that influence each rationally perceived knowledge, where $y_1^{\mathrm{RR}}(k^{\mathrm{p}})$ in Table 3.2 is influenced by $y_1^{\mathrm{PA}}(k^{\mathrm{p}}), \ldots, y_5^{\mathrm{PA}}(k^{\mathrm{p}})$ given in Table 3.1, and $y_2^{\mathrm{RR}}(k^{\mathrm{p}})$ is influenced by $y_6^{\mathrm{PA}}(k^{\mathrm{p}})$.

In order to model function $f_{ij}^{\mathrm{RR}}(\cdot)$ to mathematically represent the processes that are

illustrated in Figure B.2.1, three types of mathematical expressions were considered, i.e., affine (3.8a), exponential (3.8b), and boolean (3.8c). The choice of each expression depended on the nature of the input variable $y_i^{\text{PA}}(k^{\text{p}})$ as is explained further when the particular use cases of these functions are discussed. The expressions for these functions are:

$$f_{ij}^{\text{RR,F}}\left(y_i^{\text{PA}}(k^{\text{p}});\theta_{ij,0},\theta_{ij,1}\right) = \\ = \theta_{ij,0}y_i^{\text{PA}}(k^{\text{p}}) + \theta_{ij,1} \tag{3.8a}$$

$$f_{ij}^{\text{RR,E}}\left(y_i^{\text{PA}}(k^{\text{p}});\theta_{ij,0},\theta_{ij,1}\right) = \\ = \theta_{ij,1}\exp\left(\theta_{ij,0}y_i^{\text{PA}}(k^{\text{p}})\right) + 1 \tag{3.8b}$$

$$f_{ij}^{\text{RR,B}}\left(y_i^{\text{PA}}(k^{\text{p}});\theta_{ij,0},\theta_{ij,1}\right) = \\ = \begin{cases} \theta_{ij,0}, & y_i^{\text{PA}}(k^{\text{p}}) = 0 \\ \theta_{ij,1}, & y_i^{\text{PA}}(k^{\text{p}}) = 1 \end{cases} \tag{3.8c}$$

We have selected the affine relationship (3.8a) to estimate the impact of the inputs that are integer, with an upper bound. These inputs include *puzzle difficulty level*, $y_1^{\text{PA}}(k^{\text{p}})$, and *number of hints*, $y_2^{\text{PA}}(k^{\text{p}})$, both being integers with an upper bound of, respectively, 5 and $1 + 2n^{\text{moves}}$ where n^{moves} is an integer that represents the number of moves in the puzzle.For the inputs of the rational reasoning subprocess that are integers but have no upper bound, i.e., for *number of wrong attempts*, $y_3^{\text{PA}}(k^{\text{p}})$, and *time to solve a puzzle*, $y_4^{\text{PA}}(k^{\text{p}})$, the impact on the output of the subprocess was modeled using an exponential function, as given in (3.8b). Finally, for those inputs to the subprocess that are of a Boolean nature, i.e., for *skipped puzzle?*, $y_5^{\text{PA}}(k^{\text{p}})$, and *reward given?*, $y_6^{\text{PA}}(k^{\text{p}})$, the impact was modeled using (3.8c).

The first output $y_1^{\text{RR}}(k^{\text{p}})$ of the rational reasoning subprocess, i.e., *the puzzle is difficult*, is generated by all first 5 pieces of inputs from the perceived data indicated in Table 3.1, i.e., by $y_1^{\text{PA}}(k^{\text{p}}),\ldots,y_5^{\text{PA}}(k^{\text{p}})$, as is shown in Figure B.2.1. The second output $y_2^{\text{RR}}(k^{\text{p}})$, i.e., *Nao offered a reward*, is generated solely by $y_6^{\text{PA}}(k^{\text{p}})$, i.e., by the perceive data *reward given?* (see Table 3.1). Using (3.2) and (3.8), we have:

$$\begin{aligned} y_1^{\text{RR}}(k^{\text{p}}) = {} & \theta_{11,0}y_1^{\text{PA}}(k^{\text{p}}) + \theta_{11,1} + \\ & \theta_{21,0}y_2^{\text{PA}}(k^{\text{p}}) + \theta_{21,1} + \\ & \theta_{31,1}\exp\left(\theta_{31,0}y_3^{\text{PA}}(k^{\text{p}})\right) + 1) + \\ & \theta_{41,0}\exp\left(\theta_{41,1}y_4^{\text{PA}}(k^{\text{p}})\right) + 1) + \\ & f_{51}^{\text{RR,B}}\left(y_5^{\text{PA}}(k^{\text{p}});\theta_{51,0},\theta_{51,1}\right) \end{aligned} \tag{3.9a}$$

$$y_2^{\text{RR}}(k^{\text{p}}) = f_{61}^{\text{RR,B}}\left(y_6^{\text{PA}}(k^{\text{p}});\theta_{61,0},\theta_{61,1}\right) \tag{3.9b}$$

Remark 9 *Note that in* (3.9a) *the term* $\theta_{11,1} + \theta_{21,1} + f_{51}^{RR,B}\left(y_5^{PA}(k^p);\theta_{51,0},\theta_{51,1}\right)$ *may be identified as one single parameter.*

Cognition module The cognition module (see Figure 3.2) is composed of three main state variables, i.e., beliefs, goals, and emotions. In the setup of this case study, the

Table 3.2: Concepts that represent the rationally perceived knowledge, perceived knowledge, beliefs, goals, emotions, and biases related to the cognitive module for the case study. Note that the number of the relevant beliefs, goals, emotions, and biases involved in the experiment are, respectively, 2, 4, 2, and 1.

Rationally perceived knowledge	Perceived knowledge (corresponding to the rationally perceived knowledge in the same row of this table)	Beliefs (corresponding to the same row of this table)	Concept
$y_1^{\mathrm{RR}}(k)$	$x_1^{\mathrm{PK}}(k)$	$x_1^{\mathrm{b}}(k)$	The puzzle is difficult
$y_2^{\mathrm{RR}}(k)$	$x_2^{\mathrm{PK}}(k)$	$x_2^{\mathrm{b}}(k)$	Nao offered a reward
Goals			**Concept**
$x_1^{\mathrm{g}}(k)$			Quit the session
$x_2^{\mathrm{g}}(k)$			Skip the puzzle
$x_3^{\mathrm{g}}(k)$			Get help
$x_4^{\mathrm{g}}(k)$			Change difficulty
Emotions			**Concept**
$x_1^{\mathrm{e}}(k)$			Boredom
$x_2^{\mathrm{e}}(k)$			Frustration
Bias			**Concept**
$x^{\mathrm{bias}}(k)$			The puzzle is difficult

number of beliefs, goals, and emotions involved is, respectively, 2, 4, and 2, where the corresponding concepts have been given in Table 3.2. The realization of these variables are real values which can vary between -1 and 1. Figure B.2.3 and Figure B.2.2 show the structure of the cognitive module, including variables and linkages. Figure B.2.3 corresponds to the complete model used in the first and second sessions. Following the strategy described in Section 3.2.2, a simplified version of the previous model was used to integrate the model-based control session. Figure B.2.2 represents this simplified model.

To generate the perceived knowledge, (3.3) is used. The first term of this equation is null (i.e., $f_i(x_i(k)) = 0$), and $z_{\ell i}(l) = 1$. As shown in Figure 3.2, the second term of this equation reflects the influence of the biases.

In order to update the beliefs, goals, and emotions, (3.3)–(3.6) are used, with weight w_{ii} set to 0 for beliefs and to 0.9 for both goals and emotions. The weights $w_{ji}(k)$ are

given by:

$$w_{ji}(k) = \begin{cases} w_{ji}^{-}, & x_j(k) \leq 0 \\ w_{ji}^{+}, & x_j(k) > 0 \end{cases} \tag{3.10}$$

where the parameter values w_{ji}^{+} and w_{ji}^{-} should be identified for each participant.

Remark 10 *To assess the validity of the hypothesis that the goals and emotions are updated with a lower frequency than the beliefs, as stated in Section 3.2.1, in addition to considering the values indicated above for w_{ii}, an additional identification procedure for the MMM has been conducted for all participants of the case study using $w_{ii} = 0$ for both the beliefs and the goals. The estimation errors have been shown and compared in Section 3.4, and the validity of the hypothesis has been discussed there.*

Remark 11 *Since the biases are auxiliary variables used to describe the effect of emotions on new beliefs, we represent them without dynamics.*

Decision-making module The decision-making module (see Figure 3.3) is composed of two main subprocesses, rational intention selection and rational action selection. Table 3.3 includes the concepts that define the intentions and actions relevant for the decision-making module in this case study.

Regarding the rational intention selection, the intensity of each intention y_i^{RIS} is given by (3.7). In the context of the case study, each intention is only influenced by one goal and is not influenced by any belief, yielding a particular case of (3.7). Table 3.3 lists the goal that influences each intention, where the 4 goals, described earlier in this section, result in 5 possible intentions. This information is also graphically presented in Figure B.2.4.

As for the action selection, we assume that each action is performed if the intention corresponding to the same concept is positive (and not performed otherwise). Hence, the rational action selection is mathematically represented as:

$$y_i^{\text{RAS}}(k) = \begin{cases} 0, & y_i^{\text{RIS}}(k) \leq 0 \\ 1, & y_i^{\text{RIS}}(k) > 0 \end{cases} \tag{3.11}$$

Measurement frequency The MBC should determine whether or not the participant receives a reward from Nao, as well as the difficulty level of the next puzzle, every time a puzzle is finished. Since the measurements of the cognitive variables are gathered by asking the participants about their mental states (see Appendix B.4), there needs to be a trade-off between collecting a sufficient number of measurements and not disturbing or distracting the participants, as it may affect the reliability of the measurements. Therefore, to balance the number of questions asked, they were posed twice per puzzle and after the participant requested a hint.

Every time a measurement of the mental states was received through the feedback of the participant, the inputs given in Table 3.1 were also collected. These inputs were directly accessible in the code. Then, the estimates for the variables of the MMM were

Table 3.3: Concepts that represent the intentions and actions of the decision-making for the case study.

Intention	Action	Concept	Cognitive variable that influences the intention
y_1^{RIS}	y_1^{RAS}	Quit the game	$x_1^{\text{g}}(k)$: Quit the game
y_2^{RIS}	y_2^{RAS}	Skip the puzzle	$x_2^{\text{g}}(k)$: Skip the puzzle
y_3^{RIS}	y_3^{RAS}	Ask for help	$x_3^{\text{g}}(k)$: Get help;
y_4^{RIS}	y_4^{RAS}	Ask for an easier puzzle	$x_4^{\text{g}}(k)$: Change difficulty
y_5^{RIS}	y_5^{RAS}	Ask for a more difficult puzzle	$x_4^{\text{g}}(k)$: Change difficulty

updated to incorporate the new measurements. However, the MMM, as explained in Section 3.2.3, may be updated at a higher frequency than the frequency of sampling the measurements. In this case study, the variables of the MMM were updated two times in between every two consecutive measurements.

Model identification

Preliminary HSRI were carried out with two volunteer participants, where in addition to the concepts presented in Section 3.3.2 (see Table 3.2), the preliminary model included two extra concepts for the belief, i.e., *Nao hinders solving the puzzle* and *participant made progress*, one for the goal, i.e., *get a reward*, and one for the emotion, i.e., *happiness*. Moreover, in addition to determining the difficulty level of the next puzzle and whether or not to offer a reward to the participant, Nao had a choice to hinder the participant by purposely giving incorrect clues or to help the participant. Hindering was hypothesized to induce frustration in the participants, generating more diverse data and improving the identification process. Note that these two volunteer participants differed from the 10 participants of the experiments, and their data has not been included in the results. Their participation aimed to assess the designed sessions by repeating the experiments multiple times, collecting feedback, and assessing the estimations. By doing so, the design of the sessions was enhanced as much as possible before involving the participants.

It was found out that, with the data collected in two sessions lasting 45 to 60 minutes per participant, it was not possible to identify all parameters unequivocally, deploying the identification and assessment methods presented in Section 3.2.2. Performing a third session with each participant (i.e., extending the training dataset) slightly improved the number of parameters that were unequivocally identifiable, but did not completely address the issue. It was concluded that the control input concerning Nao yielding false hints and its corresponding belief could create confusion, especially for participants with no or limited experience with chess playing. This introduces undesirable noise into the data, hindering the identification process. By analyzing the data collected in the preliminary sessions, the belief *participant made progress*, the goal

get a reward, and the emotion *happiness* were found to be redundant. Furthermore, the other concepts for these variables were more relevant to be tracked and optimized by Nao. Consequently, the model was simplified according to the approach that has been presented in Section 3.2.2.

After the preliminary phase, the model described in Section 3.3.2, which we refer to as *complete model* and is shown in Figure B.2.3, was obtained. This *complete model* was identified for both preliminary participants in order to assess its accuracy in tracking their mental states. Then, two extra simplifications were designed to reduce the model that was embedded in the MBC used in the last session. The first simplification was the removal of the goals *get help* and *change difficulty* (see Figure B.2.2 for a representation of the *simplified model*), as these goals did not directly reflect the quality of the interactions (and, consequently, did not need to be optimized), nor did they influence the other mental states. For the ten volunteer users, the choice to embed the complete or the simplified model in the MBC was done per user, by selecting the model that achieved a smaller validation error and that corresponded to a larger number of unequivocally identifiable parameters. The second simplification was the replacement of the scheduled weights in (3.10) by just one scalar parameter (i.e., $w_{ji}(k) = w_{ji}^{-} = w_{ji}^{+}$). This second simplification was considered only for participants whose simplified model could not be fully unequivocally identified.

Note that the weight parameters w_{ii} in (3.10) were set to 0.9 whenever $x_i(k)$ represented a goal or an emotion. For the participants for whom setting $w_{ii} = 0.9$ did not result in a satisfactory optimization cost, w_{ii} was gradually reduced with a step of 0.1 until the optimization cost was acceptable. Finally, to facilitate and improve the model identification, the inputs of the real-life data were normalized per participant between 0 and 1.

Optimizer

The integration of the MMM into an MBC is based on Figure 3.6. The control input is composed of two variables, where the first one, i.e., the difficulty level of the next puzzle, is an integer variable with 6 possible realizations and the second one, i.e., *reward given?*, is a Boolean with 2 possible realizations (1 when reward is given, and 0 otherwise). Therefore, 12 possible combinations and 12 possible control inputs exist.

A cost function was formulated based on the belief, goals, and emotions that are predicted at time step k for the upcoming time step $k+1$ by the MMM, when the current state vector $\boldsymbol{x}(k) = [x_1^{\mathrm{b}}(k), x_2^{\mathrm{b}}(k), x_1^{\mathrm{g}}(k), x_2^{\mathrm{g}}(k), x_3^{\mathrm{g}}(k), x_4^{\mathrm{g}}(k), x_1^{\mathrm{e}}(k), x_2^{\mathrm{e}}(k)]^\top$ is measured:

$$J(\boldsymbol{x}(k)) = |x_1^{\mathrm{b}}(k+1)| + w^{\mathrm{g}} \sum_{i=1}^{2} x_i^{\mathrm{g}}(k+1) + w^{\mathrm{e}} \sum_{i=1}^{2} x_i^{\mathrm{e}}(k+1) \tag{3.12}$$

The estimation of the upcoming state variables based on the current measured states has been detailed in Section 3.3.2.

The formulation of (3.12) balances a trade-off in minimizing the cognitive variables whose rise will negatively impact the participants, including the absolute value of the first element of the belief, i.e., *the puzzle is difficult* (see Table 3.2), the first two

elements of the goal (i.e., *quit the session* and *skip the puzzle*), and both elements of the emotion (i.e., *boredom* and *frustration*). The lower the value of the mentioned goals and emotions is, the better for the HSRI. As for the belief, since the minimum and maximum values of this variable correspond to the belief that the puzzle is too easy or too difficult, respectively, it is ideal to keep this belief as neutral as possible (i.e., as close to 0 as possible). The parameters w^{g} and w^{e} weigh these concepts relatively. In this case study, we gave an equal level of importance to all these concepts and tuned them as $w^{\mathrm{g}} = w^{\mathrm{e}} = 1$. The control input that yields in the lowest value of the cost function given by (3.12) is selected by the MBC for time step k.

3.4. Results and Discussions

Next, we present the results and discussion on identifying the MMM for the participants and using the identified models in MBCs to steer behavior of Nao in the final interaction session.

3.4.1. Results for Model Identification

In order to assess the accuracy of the model in estimating the mental states, we divided the data points of the first two sessions into training data (67%) and test data (33%). Thus, the training data was used to identify the model and the test data to assess it. For this purpose, we used the complete model composed of the variables shown in Tables 3.1, 3.2, and 3.3, and displayed in Figure B.2.3. Note that in general, the preferences and personality traits, which remain constant in the short term (e.g., during the three interaction sessions), regulate the variations in the mental states (see Chapter 2 for more details). In this chapter, due to the nature of the HSRIs, the overall interest of the participant in playing chess (which is a general preference) may regulate the speed for reaching the goal "quit the game". Moreover, how focused and confident a participant is in general (as a trait), as illustrated in Figure B.2.3, impacts the evolution of the emotions "boredom" and "frustration", respectively. Thus, these fixed parameters have been included in MMM and are identified for the participants.

The average Mean Squared Error (MSE) in the estimation of the mental states is presented in Table 3.4 for when the goals and emotions are updated with a frequency lower than the frequency of updating the beliefs, and when the goals and emotions are updated with the same frequency as the beliefs (see Remark 10, Section 3.3.2). These average MSE values are presented for three cases concerning the identification procedure: (1) **Conventional**: When the coupled configuration (see Figure 3.4) is used to identify the perception and cognition modules simultaneously, using a multi-start optimization but no warm-start. (2) **Approach A**: When the perception and cognition modules are identified following Algorithms 3.2 and 3.3 of Section 3.2.2. (3) **Approach B**: When the perception and cognition modules are identified following Algorithms 3.2 and 3.4 of Section 3.2.2.

The average MSE values achieved for both the training and test data sets are very satisfactory, all presenting a value under 0.1, where the maximum possible value for the MSE is 4. These values indicate the excellent capacity of MMM to estimate the invisible cognitive procedures of various participants in a personalized way for

the given HSRIs. Furthermore, they demonstrate that the identification was carried out successfully. Furthermore, the hypothesis that the goals and emotions should be updated with a lower frequency than the beliefs is supported by the results, where applying this hypothesis results in a decrease in the average MSE of around 16% and 11% for the training and test data sets, respectively.

Finally, the identification of the perception and cognition following approaches A and B improves the percentage of parameters that are unequivocally identified when compared with the conventional approach, as shown in Table 3.5. Therefore, these two approaches contribute to making the model more interpretable, while not impacting the accuracy of the estimations made by the identified models, as shown in Table 3.4.

Next, we present the results obtained when using the MBC for behavioral control of Nao.

Table 3.4: Average MSE obtained when estimating the mental states (emotions, goals, beliefs) using the training and the test data sets in the model identification procedure. The error is normalized over all the mental states (beliefs, goals, and emotions) and time steps. Given that the mental states are bounded in $[-1, 1]$, the maximum MSE is 4. The values presented are an average over all the participants.

Data set	Different frequencies[1]	Identification of perception and cognition		
		Conventional	Approach A[2]	Approach B[2]
Training	No	0.063	0.064	0.065
	Yes	0.053	0.053	0.053
Test	No	0.075	0.075	0.082
	Yes	0.067	0.067	0.067

[1] Goals and emotions are once updated with a lower frequency than beliefs and once with the same frequency, as explained in Remark 10, Section 3.3.2

[2] Approaches A and B are explained in Section 3.2.2

Table 3.5: Percentage of parameters unequivocally identified during the optimization process with the different identification approaches. The values presented are an average over all the participants.

Different frequencies	Identification of perception and cognition		
	Conventional	Approach A	Approach B
No	59.44%	76.94%	80.56%
Yes	73.61%	96.39%	97.78%

3.4.2. Results for Model-based Controller

To showcase the performance of the proposed controller, which embeds MMM for one-step predictive decision-making (see Section 3.3.2 for details), we display in

Figure 3.9 and Figure 3.10 the evolution of the belief, goals, and emotions of two participants over time, as well as the corresponding decisions (inputs) made by the controller. For this purpose, we used the simplified model (see Figure B.2.2), since this model provided better results for each participant, as described in Section 3.3.2.

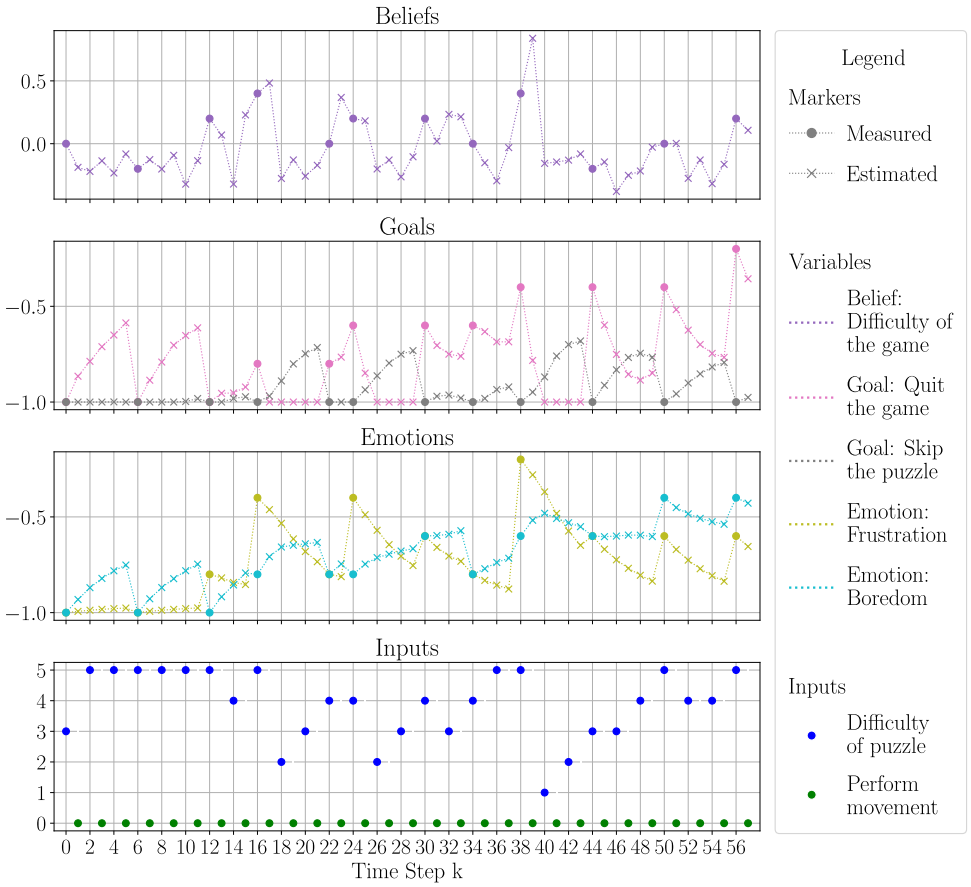


Figure 3.9: Values corresponding to the mental states and inputs (i.e., decision variables of the controller) for one participant over time. The vertical lines correspond to the start of a new puzzle. After the introduction of a puzzle, two time steps were considered, such that the first time step was in the middle of playing the puzzle and the second time step coincided with the start of the next puzzle. Around time step $k = 16$, the participant suddenly became frustrated, as is deduced from the plot for emotions. Thus, in the next time step, the controller lowered the puzzle difficulty level, which, over time (i.e., by reaching time step $k = 30$), led to a decrease in the frustration. The same happens again at time step $k = 38$. Moreover, when the participant presented the same level of frustration with a higher level of boredom (see time steps $k = 44$ and $k = 40$), the difficulty level of the puzzle for the next time step was chosen to be higher by the controller. The controller effectively prevents the engagement of the participant from increasing excessively during interactions by appropriately responding to their mental states. As for the reward, the controller has inferred that the participant preferred not to receive rewarding movements from Nao.

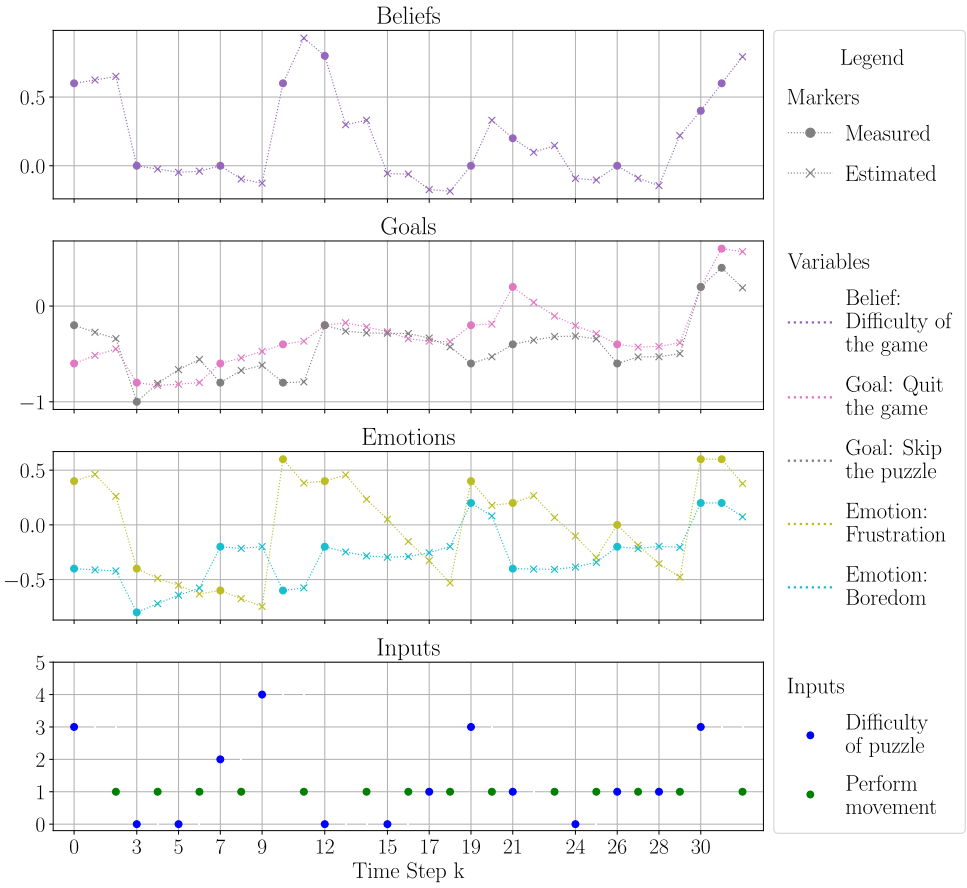


Figure 3.10: Values corresponding to the mental states and inputs (i.e., decision variables of the controller) for one participant over time. The vertical lines correspond to the start of a new puzzle. Right after the start of the interaction, the participant felt frustrated, as illustrated in the plot for emotions. Hence, the controller immediately decreased the difficulty level of the next puzzle, which resulted in a decrease in the frustration of the participant. At time step $k = 7$, the participant started becoming bored. Thus, for the next time step $k = 9$, the controller has increased the level of difficulty of the puzzle. Again, the participant became frustrated, and the controller decreased the difficulty level of the puzzle accordingly. The controller shows to be able to sustain the engagement of the participant stably throughout the interactions by properly reacting to their mental states. As for the reward, the controller has inferred that the participant prefers to always receive rewarding movements from Nao.

To evaluate the HSRIs when our control framework is used, both objective and subjective metrics were deployed. As explained in Section 3.3.1, in the third session, the participants interacted with two versions of Nao — once steered by a controller based on our proposed framework and once steered by a rule-based controller that selected the inputs based on a uniform distribution. During each interaction in this session, we tracked the following metrics: the values for the self-reported goals and emotions of the participants, the results from self-reported questionnaires completed after the

interaction (available in Appendix B.5), and the engagement level of participants estimated from a recording of the interaction.

The answers given by the participants regarding their emotions of "boredom" and "frustration", as well as their goals to "quit the game" and "skip the puzzle", were collected with the following frequency: After the first move by the participant in the first puzzle, the questions appeared on the display and the participants were asked to provide their response. In the subsequent puzzles, after the first move of the participant in the puzzle, if more than 150 s passed or if the participant played 3 puzzles since the last set of questions, then the questions were posed again. The values provided by the participants were used both to reset the values of the mental states used by MMM within the MBC and to assess the quality of the interactions. The average emotions and goals reported by each participant over the interactions with the MBC and with the Rule-Based Controller (RBC) are shown in Figure 3.11.

For the majority of the participants, interacting with our MBC resulted in lower boredom (thus higher engagement) and a lower desire to quit the game or skip the puzzle. In fact, 8 out of 10 participants reported lower levels of boredom throughout the interaction with the MBC, and only 1 participant reported higher levels of boredom throughout this interaction. Regarding the frustration level, there does not seem to be a clear tendency when interacting with the MBC. Some participants showed higher frustration levels, but less boredom with the MBC approach (see participants P3 and P8 in Figure 3.11). This suggests that, for some participants, a trade-off between frustration and engagement should be considered. An appropriate increase in the frustration level may be necessary to sustain engagement (e.g., for highly competitive people) and, thus, a longer-term interaction.

Additionally, Table 3.6 compares the average values of the goals and emotions obtained with each controller. On average, when interacting with the Nao controlled by the MBC approach, the desire of the participants to quit the game and to skip the puzzles were, respectively, 7.70% ($p = 0.032$) and 8.86% ($p = 0.029$) less than when the RBC was deployed. Furthermore, participants felt 15.98% ($p = 0.009$) more engaged when interacting with Nao steered by the MBC. There were in general no indications that using the MBC approach was able to reduce the frustration of the participants ($p = 0.827$).

Table 3.6: Comparison between the average values of the goals and emotions for all participants when interacting with our MBC and with the RBC. The improvement of the average values per variable when the participants interacted with the MBC, as well as the p-values, are also specified.

Variable	Average values		Difference (%)	p-value
	MBC	RBC		
$x_1^{\mathrm{g}}(k)$: Quit the game	-0.645	-0.491	7.70	0.032
$x_2^{\mathrm{g}}(k)$: Skip the puzzle	-0.619	-0.441	8.86	0.029
$x_2^{\mathrm{e}}(k)$: Frustration	-0.472	-0.448	1.17	0.827
$x_1^{\mathrm{e}}(k)$: Boredom	-0.460	-0.140	15.98	0.009

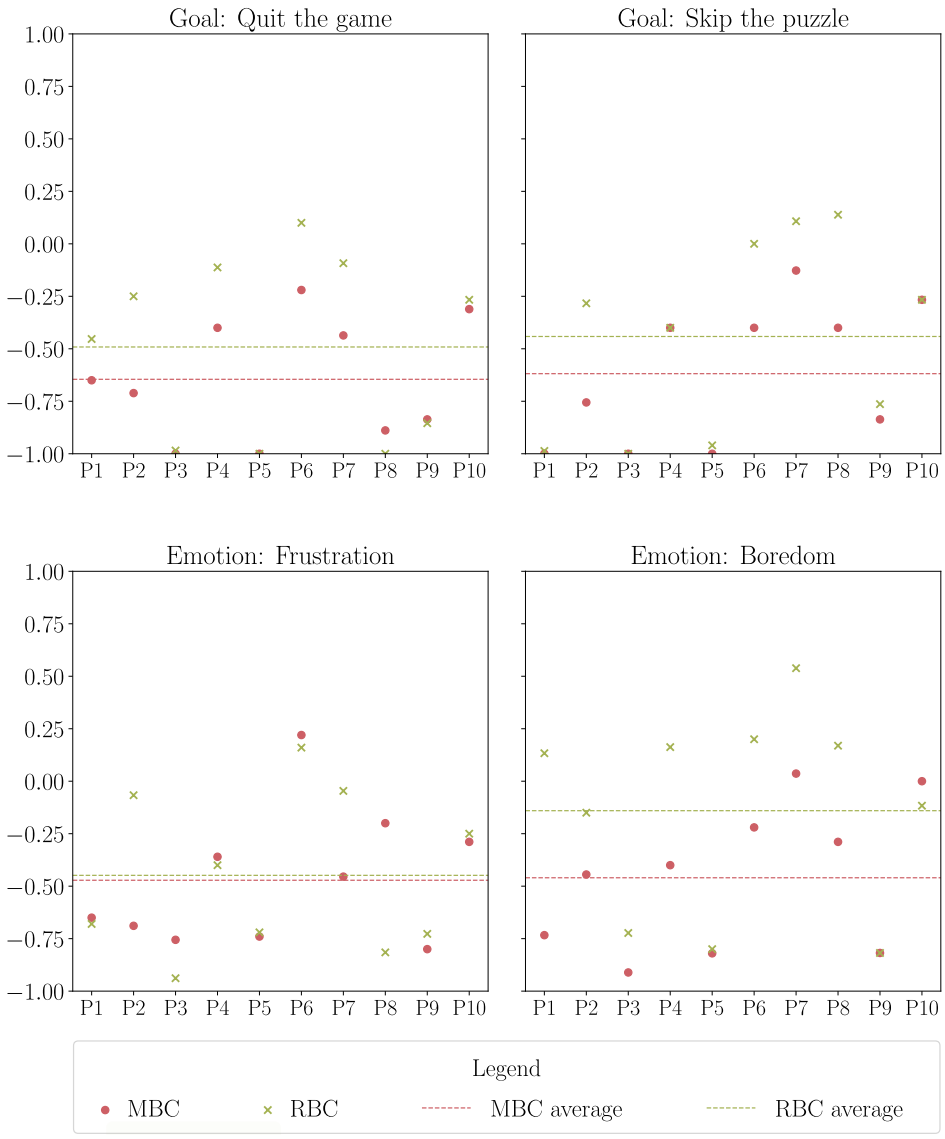


Figure 3.11: Average self-reported values of the emotions and goals of each participant over the interactions with the MBC and the RBC. Each graph corresponds to one variable. For a participant, the average value of that variable over the interaction with each controller is represented by a marker, whereas the dotted line represents the average value over all participants for that variable and controller (RBC stands for RBC).

After interacting with Nao using each controller, the participant was given a questionnaire to assess the HSRI (see Figure 3.8b for the structure of the session). The questionnaire, which is included in Morão Patrício and Jamshidnejad (2025a),

consisted of five questions regarding the engagement, adaptability, personalization, and awareness of the robot regarding the mental states of the participant. Two questions concerned the engagement, asking participants how bored and how engaged they felt in the course of the interactions. The participants answered these questions using a 7-point Likert scale. During the debrief, preliminary participants (2 participants whose input was used to improve the setup of the experiments, but whose data was not included in the results) reported perceiving a (significant) difference in some of these categories between the two interactions deploying the MBC and the RBC. However, their absolute answers to the questionnaires did not reflect this difference. When questioned about this, the preliminary participants reported that they were unsure whether they considered the same scale to answer both questionnaires due to the time gap between the sessions. They reported that, when answering the second questionnaire, they did not remember in detail the answers given in the first interaction, and that, during the second interaction, they experienced a change of perception of the assessed aspects (i.e., engagement, adaptability, etc.), which could have led them to use a different scale. To prevent this undesirable effect, after answering the questionnaire per session (once after each interaction), the participants were given an additional questionnaire where they could provide their responses in a relative sense, comparing their experiences about the two sessions. This comparative questionnaire is available in the experimental dataset Morão Patrício and Jamshidnejad (2025a). Table 3.7 shows the results from the questionnaires.

Although the average answers regarding the mental states favored the proposed MBC, the answers given by the participants in the absolute questionnaires regarding the engagement, awareness shown by the robot about their mental states, personalization, and boredom are not conclusive ($p > 0.05$). Only the answers regarding the personalization of the robot demonstrate that participants experienced our proposed approach as tailored to themselves ($p = 0.04$). Regarding the other four aspects, the comparative answers must be analyzed: The results indicate that 7 participants reported feeling more engaged and less bored when interacting with the MBC, while 3 participants reported the opposite. Furthermore, 8 participants perceived that the robot controlled by the MBC approach was more aware of their mental states, 1 participant had no opinion, and the other participant perceived the RBC to display more awareness. Finally, 6 participants perceived our MBC approach to be more adapted to them, whereas 3 participants stated having no opinion.

Finally, a video stream was used to capture the engagement of the users, using the tool presented in Duchetto et al. (2020). This tool quantified the level of engagement between 0 and 1, where a value close to 1 indicates that the participant is engaged. Although this tool correctly measured the engagement from the perspective of the robot in Duchetto et al. (2020), the dataset where it was trained included only robot-centric interactions, i.e., the robot was the focus point. In our case, the main objective was for the robot to keep the participants engaged in solving the puzzles. Thus, we placed the camera right above the screen where the puzzles were displayed to record the participants, rather than on the robot. Hence, the tool indicated a high level of engagement when the participant gazed at the screen. Nevertheless, it indicated a low score regardless of whether the participant looked away from the interaction or

Table 3.7: Results obtained from the two questionnaires given to participants in the third session. The absolute answers refer to the two questionnaires posed at the end of each interaction of the third session, while the relative answers correspond to the final questionnaire given at the end of the third session (see Figure 3.8b). Note that the absolute answers were given in the range $[1, 7]$. Thus, the closer the value of the positive indicators (i.e., engagement, awareness, adaptability, and personalization) are to 7, the better. The value of the boredom should be as low as possible. In the comparative questionnaire, participants could choose between one of the two controllers or neither for each question. For each criteria, the number of participants who have voted for each of controller (or neither) is counted. Therefore, the numbers per row add up to the total number of participants, i.e., 10.

Question	Absolute Answers			Comparative Answers		
	MBC	RBC	p-value	MBC	RBC	No opinion
Q1: Engagement	4.7	4.4	0.496	7	3	0
Q2: Awareness	4.0	2.9	0.057	8	1	1
Q3: Adaptability	3.6	2.5	0.040	5	1	4
Q4: Personalization	3.4	3.0	0.479	6	1	3
Q5: Boredom	3.5	4.9	0.158	2	7	1

at the robot. The average engagement of each participant, according to this tool, is presented in Table 3.8. The results show non-significant differences between the two interactions of each participant ($p = 0.106$) and are not coherent with the self-reported answers to the questions asked during the interactions or the questionnaire. This discrepancy is likely due to the engagement measurement tool not being suitable for our application since it did not account for a second focus point. As a result, this metric was not representative of the real engagement of the user in the current HSRI.

Table 3.8: Results obtained for the participants using the engagement measurement tool provided in Duchetto et al. (2020).

Controller	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Average
MBC	0.948	0.962	0.903	0.859	0.945	0.841	0.968	0.884	0.931	0.923	0.916
RBC	0.948	0.942	0.930	0.880	0.951	0.864	0.970	0.919	0.924	0.932	0.926

Overall, these results demonstrate the effectiveness of the proposed model-based control framework in adapting robot behavior to the mental states of users. However, the number of participants in this case study was limited, which constrains the generalizability of the findings. As explained in Section 3.3, this is a common characteristic of early-stage HSRI experiments that involve multi-session protocols and personalized modeling. In such cases, the primary goal is to establish feasibility and to validate the underlying approach rather than to achieve broad statistical generalization. Broader validation of the proposed framework requires future studies with larger participant pools and more diverse scenarios, as discussed in Section 3.5.

3.4.3. Computational Efficiency

To assess the computational feasibility of the proposed framework, we analyzed the run times and memory requirements associated with the parameter identification and the real-time deployment of the MBC.

Computational efficiency of model identification

The run time of model identification is determined mainly by three factors: the number of gradient descent runs, the number of iterations per run, and the dimensionality of the identification problem (i.e., the number of model parameters and data points used for their identification).

Each gradient descent run corresponds to an independent process with different initialization values. Consequently, the runs can be executed in parallel across available processor cores. Thus, the total identification time increases linearly with the number of runs and decreases inversely with the number of parallel workers.

The number of iterations per run also contributes linearly to the total run time, as each iteration involves recomputing the gradients and updating the model parameters. Empirical measurements were consistent with this linear relationship across multiple runs, participants, and model configurations. The number of iterations per run, however, varies across runs because a pruning strategy has been implemented to terminate runs with poor or stagnated performance after a predefined number of iterations, improving overall efficiency.

Finally, the identification time also scales approximately linearly with the problem complexity, which is determined by the number of model parameters and the number of data points, as both factors proportionally affect the duration of each iteration. Empirical measurements across participants and model configurations confirmed this relationship, with a normalized run time per iteration of $0.204\,\text{ms}$ on average and $0.250\,\text{ms}$ at maximum, using an Intel Xeon Gold 6448Y (2.1 GHz) CPU.

As a reference, the maximum observed identification time corresponded to the participant with the largest training dataset (including 87 data points) and the most complex model (including 36 parameters). On a machine equipped with two Intel Xeon Gold 6448Y (2.1 GHz) processors, using 48 parallel workers, this configuration required $964\,\text{s}$ for 250 independent runs with up to 120 iterations per run. The multi-stage identification procedures described in Section 3.2.2 showed comparable run times: $983\,\text{s}$ for Approach A (i.e., pre-identification of the perception module, followed by a joint optimization), and $921\,\text{s}$ for Approach B (i.e., pre-identification of the perception and cognition modules, followed by a joint optimization).

Since the model identification is intended for long-term HSRIs and can be performed offline between sessions, these run times remain well within practical limits.

In terms of memory requirements, the baseline memory consumption per parallel process, after environment initialization, was at most 65.10 MB. This was measured with the most complex model (including 36 parameters) and the largest dataset (including 87 data points). Although memory usage increases slightly with the number of model parameters and the size of the training dataset, the variation is insignificant in practice. For example, reducing the number of parameters from 36 to 24 or the number of data points from 87 to 35 yielded marginal decreases to 65.06 MB and

64.87 MB, respectively. During active identification, each gradient-descent process required an additional 1.04 MB at peak usage, for the largest model and training dataset. Consequently, when multiple workers are executed in parallel, the total memory overhead scales approximately linearly with the number of active processors.

Overall, these requirements are negligible relative to modern computational capacities, confirming that memory is not a limiting factor for the proposed identification procedure.

Computational efficiency of real-time deployment

For real-time deployment, we evaluated the run time and memory requirements of the MBC for different model configurations, corresponding to varying numbers of parameters. Across all tested configurations, the controller computed a new action within 5.4 ms on a laptop equipped with an Intel i7-1185G7 (3.0 GHz) CPU. Additionally, the controller required less than 8.14 MB of RAM across all configurations, with only negligible variations due to model complexity. The runtime environment, including the Python 3 interpreter and supporting libraries, occupied 66.53 MB.

Because the control loop operates in discrete time steps of a few seconds, consistent with the natural timescale of HSRIs, these run times are negligible in practice. Hence, the controller runs reliably in real time, without introducing perceptible delays. Combined with the low memory footprint, these results confirm the computational efficiency and suitability of the proposed framework for real-time interactive social robotic applications.

3.4.4. Limitations and Mitigation Strategies

While MMM and its integration within an MBC offer a mathematically grounded, transparent, and adaptive framework for controlling SRs based on the mental states of their users, certain limitations must be discussed, along with corresponding mitigation strategies. This section clarifies the main factors that may affect the effectiveness of the model and highlights the mechanisms through which their impact can be mitigated within the proposed design.

A primary source of potential underperformance of the presented framework arises from model mismatch between MMM and the actual cognitive dynamics of the user. Such a mismatch may occur when the data available for parameter identification is scarce or lacks sufficient diversity, limiting the ability to achieve accurate model identification and personalization. It may also occur when users display highly unpredictable or atypical behaviors, for which MMM may fail to capture the corresponding cognitive dynamics accurately. Additionally, if users fail to provide accurate and reliable self-reports during the initial sessions when data is collected for identifying MMM, the resulting parameter estimates may be imprecise, thereby reducing the accuracy of the identified model. These conditions reduce the accuracy of the estimations made by the model and may consequently lead to sub-optimal behavioral adaptation by the robot during deployment. Nevertheless, model mismatch is a challenge inherent to all model-based approaches and is not unique to MMM.

In addition to model mismatch, underperformance may also occur due to state

estimation drift. Since mental states are not directly visible or measurable, the controller relies on MMM to propagate state estimates between intermittent self-reports. Such self-reports should not be requested too frequently to avoid disruption in natural HSRIs. Therefore, the estimates may gradually drift from the actual mental states of the user over time, and potentially degrade the performance and effectiveness of the control system, which relies on estimates provided by MMM.

While the challenges discussed in this section are inherent to any model-based approach, they can be effectively mitigated through several complementary strategies. First, regular re-identification of model parameters over time helps maintain alignment between MMM and the cognitive dynamics of the user. Second, if necessary, the structure of the model may be extended to capture user-specific or previously unmodeled behavioral patterns, which enhances its representational accuracy. Third, integrating a state estimator within the control loop is expected to improve the accuracy of mental state estimates between self-report updates.

These mitigation strategies are particularly feasible in long-term HSRIs, where extended interaction times provide sufficient data for iterative refinement and progressive improvement of the model.

3.5. Conclusions and Topics for Future Research

This chapter introduces a novel systems-and-control theoretic framework for designing and sustaining Human-Social-Robot Interactions (HSRIs), ensuring desired subjective outcomes (e.g., long-term human engagement) and enriched human experiences (e.g., feeling understood by the robot), while enabling systematic adaptability driven by the robot in the interactions.

Leveraging the mathematical model of human perception, cognition, and decision-making introduced in Chapter 2 — Mathematical Model of Mind (MMM) — and grounded in the principles of Theory of Mind (ToM), we have adapted MMM for Social Robots (SRs) to exhibit ToM-like behavior. This model has been embedded into a controller, enabling SRs to track and adapt their interactive behavior based on the evolving mental states of their human counterpart. This critical capability addresses a key limitation in current SRs, empowering them to sustain meaningful social interactions with humans (Leite et al., 2013; Matarić & Scassellati, 2016) and to engage them for extended periods. This is expected to result in significant societal impacts, particularly for vulnerable populations (Clabaugh et al., 2019; Khosla et al., 2019; M. H. Lee et al., 2023; Luperto et al., 2024; Scassellati et al., 2018; Tapus & Mataric, 2008).

We carried out a case study with 10 volunteer participants who solved chess puzzles on a screen while interacting and being guided by a Nao robot. The case-study consisted of 3 sessions lasting 45 to 90 minutes. During these sessions, the robot dynamically adjusted the difficulty level of the puzzles and rewarded participants with entertaining movements. In the first two sessions, data on mental states of the participants was collected while they played the chess puzzles and interacted with Nao. This data was then used to personalize the parameters of MMM for each participant. In the final session, participants interacted with Nao under two conditions: once, when

the robot was steered by a model-based controller embedding their personalized MMM, and once when Nao was guided by a conventional rule-based controller that did not systematically consider their mental states.

MMM achieved an average Mean Squared Error (MSE) of 0.067 in tracking the beliefs, goals, and emotions of the participants within a normalized range of $[-1, 1]$. Compared to the rule-based controller, the MMM-based controller increased participant engagement by 16% ($p = 0.009$) and decreased the goal of quitting the game by 8% ($p = 0.032$), as measured objectively in the final sessions. Responses to a post-interaction questionnaire further confirmed that most participants perceived Nao, when controlled by the MMM-based controller, to be more engaging, more aware of and adaptive to their mental states, and more personalized to their needs.

Although the MMM-based controller successfully adjusted the difficulty level of the puzzles based on the mental states of the participants in the final session, it did not do the same for rewarding them. One possible reason is that during the first two sessions, while the novelty effect (i.e., a temporary increase in the engagement and interest of individuals when exposed to new experiences, e.g., interacting with a Nao robot) was still present, participants consistently reacted positively or negatively to the entertaining movements of Nao. This may have led MMM to infer that these movements always had either a positive or a negative influence on the participants, regardless of their actual mental states. Future work should explore the impact of the novelty effect on collected data, quantifying its duration and magnitude to determine when user responses stabilize. This will enable the development of identification strategies less affected by novelty effects in the training data, such as re-identifying the model once the novelty effect has subsided.

Moreover, the present case study involved a relatively small and homogeneous group of participants, as it was designed mainly to demonstrate the feasibility and effectiveness of the proposed framework under controlled conditions. Future studies should expand the participant pool through larger-scale, multi-phase, and longer-duration experiments to strengthen the statistical validity and generalizability of the findings. Additionally, the current design of MMM and the experiments focused on neurotypical users. Future work should include neurodivergent or cognitively impaired participants to examine whether the formulation of MMM generalizes across different cognitive profiles, and, if needed, to refine the model for those user groups.

Given the central role of engagement in HSRIs, future developments should also improve the objective engagement measurement method used in this study. In the present case study, this measurement method relied on a single camera view and thus lacked precision in multi-focus settings. Future implementations should include one camera per focus point and merge the results from all cameras to gather an accurate estimate of the engagement of the user. Additionally, a personality test could be administered to the participants prior to the interaction to establish which mental states are relevant to optimize (e.g., whether the frustration might boost or hinder the engagement of that participant).

Finally, while MMM demonstrated excellent performance in short-term predictions of mental states, its long-term estimations declined for some participants. Future work will focus on integrating dedicated state estimators into the closed-loop control system

to mitigate this estimation drift. This will reduce reliance on direct self-reports from users and enable greater robustness and improved real-time adaptation during extended HSRIs. Overall, this chapter is the first to explore systems-and-control-theoretic methods for HSRIs, providing guarantees on performance and safety. Our findings demonstrate how cognitively grounded, model-based control and control-theoretic principles can enable SRs to exhibit adaptive, cognition-aware, and personalized behavior, leading to more meaningful HSRIs and sustaining long-term engagement. In doing so, our work charts a new trajectory toward fundamentally new forms of SRs control that could enable sustained and personalized long-term support across fields such as healthcare, education, and assisted living, all of which face pressing challenges related to increasing demand for long-term, individualized support.

4

A Systems-Theoretic Approach to Mental State Estimation for Theory-of-Mind-Aware Social Robots

The previous chapter demonstrated the capability of a cognitive-aware, model-based controller to optimize users' mental states by leveraging the Mathematical Model of Mind (MMM), proposed in Chapter 2, to adapt robot behavior to the user's evolving mental states. The effectiveness of this system, however, relies on the accuracy of the mental state estimation that feeds the controller. The open-loop prediction of mental states generated by the standalone MMM, as employed in Chapter 3, can be prone to error accumulation, particularly when real-time measurements are scarce or noisy. The work presented in this chapter addresses this crucial limitation by proposing and validating three model-based state estimators that integrate the MMM. Specifically, this chapter introduces three variants of the Extended Kalman Filter tailored for Human-Social-Robot Interactions to refine state predictions based on available measurements.

This chapter is based on the following publication:

Morão Patrício, M. L., & Jamshidnejad, A. (2025b). A systems-theoretic approach to mental state estimation for theory-of-mind-aware social robots. *IEEE Access*, *13*, 158467–158482. https://doi.org/10.1109/ACCESS.2025.3607165

Abstract

Social Robots (SRs) are increasingly deployed in fields such as health care and education to support users through social interactions. Nonetheless, these robots mostly rely on black-box machine learning methods that lack awareness of the mental states of their users, which often leads to unnatural behavior. To address this, we propose three model-based techniques for real-time estimation of invisible mental states of humans. Each method adapts the extended Kalman filter and incorporates a validated dynamic model of human mental states. These mental state estimators are designed for human-robot social interactions and personalize their parameters using initial user data. When tested with 10 human participants interacting with a NAO robot, the mental state estimators reduced the average error in estimation and prediction of mental states across all participants by 3% (i.e., from 12% to 9%), with improvements of up to 13% for individual participants. These results demonstrate the potential of integrating such state estimators into the behavioral control systems of SRs to enhance their awareness of the mental states of users.

4.1. Introduction

In recent years, Social Robots (SRs) have increasingly been deployed to assist, care for, and entertain users across a wide range of applications. These include deployment in education for tutoring purposes (Arshad et al., 2020; Gordon et al., 2016), enhancing social and cognitive skills for children with autism (Ascensão & Jamshidnejad, 2022; Clabaugh et al., 2019; Scassellati et al., 2018), administering post-stroke rehabilitation (Feingold Polak & Tzedek, 2020; M. H. Lee et al., 2023), distracting patients during pediatric medical interventions (Rossi et al., 2020), and providing care for older adults and individuals with cognitive disabilities (Giorgi et al., 2022; Kubota et al., 2020; Luperto et al., 2023; Schrum et al., 2019).

Despite the promise of these applications in enhancing the quality of life of users, the systems that govern the behavior of SRs are often still perceived as rudimentary and limited (Céspedes et al., 2021; Tapus et al., 2007). Most SRs rely on black-box machine learning approaches to control their behavioral responses according to metrics related to directly measurable performance of their users, without awareness of their mental states. For example, Shiarlis et al. (2017) and Ko et al. (2020) applied deep neural networks to learn proper control policies for SRs directly from human demonstrations. Furthermore, Maroto-Gómez et al. (2024) and Gao, Yang et al. (2018) employed Deep Reinforcement Learning for optimal action selection in social robots. While Maroto-Gómez et al. (2024) and Gao, Yang et al. (2018) incorporated some modeled elements — such as biological functions or social norms — into the input or reward function, the main control approach was the machine learning module, and it did not account for the internal mental states of users. Only recently have a few projects started incorporating cognitive states of the users, such as emotions, beliefs, and goals, into behavioral control of SRs (Bagheri et al., 2021; Morão Patrício & Jamshidnejad, 2026b; Rossi et al., 2020), or to attribute such states to SRs by embedding cognitive models directly into their behavioral steering systems (Dell'Anna & Jamshidnejad, 2024).

Meaningful interactions require that SRs respond to users, reflecting an understanding of their mental states (Bagheri et al., 2021; Matarić & Scassellati, 2016; Tahir et al., 2020; Tapus et al., 2007). This, in turn, requires accurate modeling and real-time estimation of the mental states of users (Bagheri et al., 2021; Tahir et al., 2020; Tapus et al., 2007). A promising approach is to model the dynamical evolution of mental states, based on the inputs (i.e., stimuli perceived by users) and outputs (i.e., actions performed by users), as proposed in Chapter 2. This model, Mathematical Model of Mind (MMM), which has been developed in Chapter 2 and subsequently extended for human-robot social interactions in Chapter 3, can be used to directly estimate the mental states of users, as illustrated in Figure 4.1a. However, when measurements are scarce or inaccurate, the estimation based on MMM alone is prone to errors (see Chapter 3 for details).

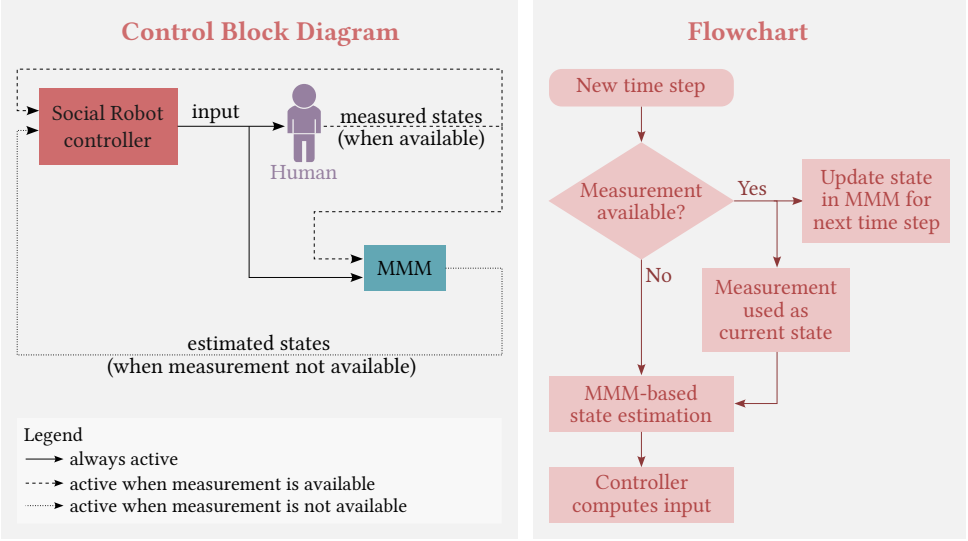
To enhance the estimation of human mental states, we propose embedding the MMM within a state estimator, as illustrated in Figure 4.1b. This approach is motivated by a known limitation of the standalone MMM: in the absence of frequent measurements, prediction errors tend to accumulate over time, degrading the quality of the mental state estimates. This limitation is common among open-loop predictive models operating in dynamic, uncertain environments, and does not reflect a fundamental flaw in MMM. Consequently, the issue cannot be fully resolved by replacing the model. Moreover, as demonstrated in Chapter 2, MMM achieved an average mental state estimation accuracy of 81.55%, making it a highly promising candidate. Additionally, it is, to our knowledge, the only dynamic cognitive model formulated within a state space framework. This motivates us to retain the model, while introducing additional mechanisms to mitigate the observed accumulation of estimation error. Embedding the model within a state estimator enables us to refine predictions based on available measurements, thereby maintaining higher estimation accuracy over time (Welch & Bishop, 1995).

Improved accuracy in mental state estimation enables the control systems of SRs to respond more effectively to the varying needs of their users (Matarić & Scassellati, 2016; Tapus et al., 2007). This, in turn, promotes natural, human-like interactions between SRs and their users.

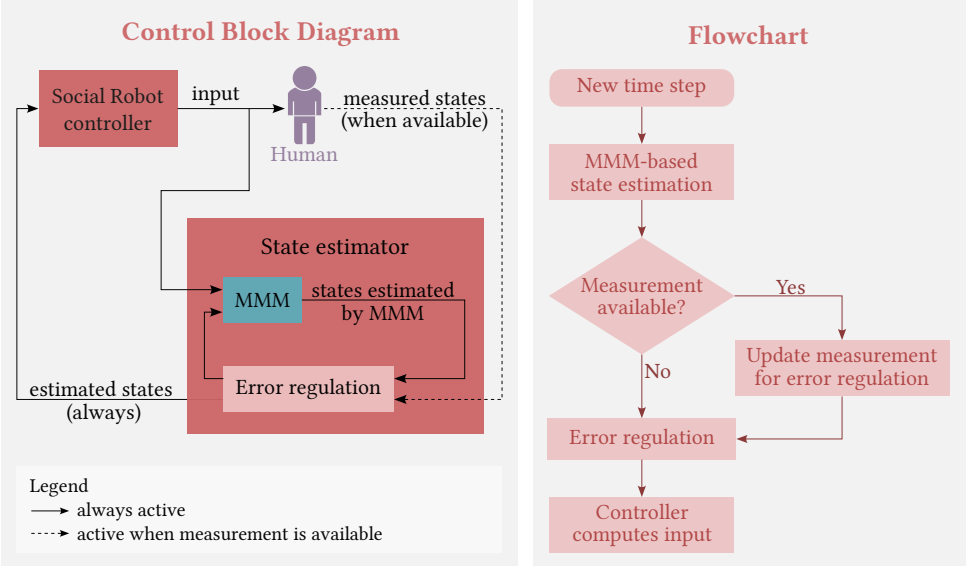
The remainder of this chapter is organized as follows: We describe the main contributions of this chapter in Section 4.1.1. Section 4.1.2 reviews relevant work on model-based control of SRs that forms the foundation for our approach, as well as existing state estimation techniques relevant to the purposes of this chapter. Section 4.2 describes the adaptations and enhancements made to tailor existing state estimation methods to the domain of human-robot social interactions. The experimental setup and assumptions are detailed in Section 4.3. Section 4.4 presents and discusses the main results of the chapter. Finally, Section 4.5 concludes the chapter and gives directions for future research.

4.1.1. Main Contributions

This chapter presents three main contributions that advance Human-Social-Robot Interaction (HSRI) beyond the work presented in Chapter 3, in which Mathematical Model of Mind (MMM) was used in an open-loop estimation configuration (i.e.,



(a) Control block diagram and flowchart representing the system without a state estimator, as proposed in Chapter 3. When a measurement is not available, the MMM computes an estimation of the current state based on the control input and previous state. Then, this estimation is fed to the controller. When a measurement is available, that measurement is fed to the controller of the SR as the current state, and it is used to update the state of the MMM for the next time step.



(b) Control block diagram and flowchart representing the system with a state estimator. The state estimator begins by predicting the system state using the MMM. This prediction is then refined through an error regulation process that merges the model-based estimate with available measurements to minimize the estimation error. When a new measurement becomes available, it is incorporated into this regulation step. The resulting estimated current state is then provided to the controller of the SR.

Figure 4.1: Comparison of the control architectures proposed in Chapter 3 and in this chapter, illustrating how the current state is determined for use by the predictive controller. Each subfigure represents one architecture through a control block diagram and a flowchart.

mental states were predicted solely based on model dynamics, without correction from measurements), leading to error accumulation over time. Core contributions of this chapter are outlined next:

- This chapter presents, to our knowledge, the first deployment of systems-theoretic, model-based state estimators — specifically Extended Kalman Filter (EKF) variants — for estimation of invisible mental states of humans in HSRIs. By adapting state estimation techniques (traditionally used solely for physical and engineering systems) to abstract, dynamic cognitive states, our work bridges a gap between systems and control theory and social robotics. Key technical adaptations include the construction of personalized noise covariance matrices and the modification of EKF computations to handle the challenge of sparse measurements inherent in HSRIs, resulting in three EKF-based estimator variants. This model-based approach offers interpretable, real-time mental state tracking, in contrast with black-box and heuristic methods commonly deployed in HSRIs that typically lack dynamic tracking capabilities.

- We address the degradation problem identified in earlier deployments of MMM (see Chapter 3 for details), in which estimation accuracy declines with sparse user feedback. While Chapter 2 introduced MMM as a theoretical framework for mental state estimation and Chapter 3 applied this model within HSRIs without a state estimator (see Figure 4.1a), our work closes this gap by embedding MMM within a personalized, model-based state estimation framework (see Figure 4.1b).

- We evaluate whether a state estimator based on MMM can improve the accuracy of the mental state estimations for users in HSRIs. Using data collected from 10 human participants interacting with a social robot (see Chapter 3 for details), we evaluate how the proposed EKF-based estimators can potentially improve mental state predictions. Our results show that, compared to the baseline MMM estimator used in Chapter 3, the EKF-based estimators reduce prediction errors by an average of 3% across participants, with improvements of up to 13% for individual cases. This is a crucial step toward improving the efficacy of model-based controllers that steer the behavior of Social Robots (SRs) in HSRIs based on the evolution of mental states of their users, as previously proposed in Chapter 3.

4.1.2. Related Work

In Chapter 2, a model — referred to as the Mathematical Model of Mind (MMM) — is proposed for SRs to represent the dynamic, interconnected processes of human perception, cognition, and decision-making. The MMM is formulated using an input-output state space representation, which enables its integration with established systems and control techniques to enhance Human-Social-Robot Interactions (HSRIs). In particular, the MMM can be embedded within a model-based controller, enabling SRs to autonomously select actions that optimize users' mental states (e.g., minimizing anxiety or maximizing engagement). To do so, accurate estimation of mental states of the users in real time is essential for SRs. Thus, MMM can also be incorporated into

state estimators — mathematical tools for inferring internal state variables of a dynamic system, based on available observations/measurements and a model of the system — to enhance the inference of SRs about the beliefs, goals, and emotions of users.

In Chapter 3, MMM was embedded within a predictive model-based controller that steered the behavior of a NAO robot during interactions with users who were solving chess puzzles. The robot dynamically selected the difficulty of the puzzles and chose whether to display entertaining behaviors to optimize the engagement and frustration of participants, and to stimulate prolonged interactions. Parallel to its predictive role, MMM also estimated the evolution of the mental states (i.e., beliefs, goals, emotions) of participants throughout the interaction session. Due to potential errors in these estimations over time, participants were occasionally asked to self-report their beliefs, goals, and emotions. To minimize disruption to participants, however, these self-report occasions were limited to, on average, 10 reports per 35 minutes (roughly one measurement every 3.5 minutes).

Although the proposed approach outperformed a conventional rule-based controller, it was observed that, for some participants, the errors by MMM in estimating the beliefs, goals, and emotions increased significantly between consecutive measurements. Therefore, the performance of the controller degraded over time in the absence of new measurements, leading to sub-optimal behavior by the robot in that interval. For these participants, increasing the accuracy of the estimation of mental states is critical to maintaining a consistently high-quality interaction. To tackle this issue, we propose model-based state estimation methods that integrate MMM, which was previously validated in Chapter 2 and Chapter 3, to track the beliefs, goals, and emotions of users.

Among state-of-the-art state estimation methods, the Kalman Filter (KF) remains one of the most widely used algorithms, mainly due to its computational efficiency and its optimal state estimation in linear systems affected by Gaussian noise (Kim & Bang, 2018; Simon, 2006). Although the KF is limited to linear systems, the EKF is an adaptation of this algorithm for non-linear models (Kim & Bang, 2018; Simon, 2006). Other alternatives, such as the unscented KF and the particle filter, are widely used in highly non-linear systems, especially when, due to its linearization, the EKF introduces significant approximation errors (Simon, 2006). Nonetheless, these approaches come at the cost of substantially increased computational demands, compared to EKF. In our case, while the MMM is a nonlinear model, the degree of nonlinearity is not severe enough to justify the use of more advanced filters (Simon, 2006) (see Section 4.2.2 for model description). EKFs capture the necessary dynamics with sufficient accuracy, without incurring the computational complexity of unscented KF or particle filtering — an essential requirement in HSRIs. Furthermore, while model-free estimators, such as deep neural networks, can be used for state estimation, our goal is to leverage a validated and interpretable model of human mental state dynamics. Additionally, training such model-free approaches typically requires large datasets, which are generally unavailable in HSRIs. For these reasons, we focus on model-based estimation techniques.

All in all, the EKF offers a suitable balance between performance and computational cost. Consequently, it was selected for integration with MMM to estimate the mental states of humans in HSRIs.

4.2. Methodology

In this section, we describe the main steps required to develop and implement a model-based state estimator for estimating the invisible mental states of humans, enhancing various human-centered applications, particularly Human-Social-Robot Interactions (HSRIs). In Section 4.2.1, we detail how the process and measurement noise covariance matrices, which are essential tuning parameters of Kalman Filters (KFs), are constructed for a state estimator in the context of Mathematical Model of Mind (MMM). In Section 4.2.2, we explain the procedure for adapting Extended Kalman Filters (EKFs) to estimate the mental states of humans, while dealing with model uncertainties and scarce measurements. We specifically present three versions of EKF developed for this purpose.

4.2.1. Construction of the Noise Covariance Matrices

In order to deploy an EKF, the process noise covariance matrix Q and the measurement noise covariance matrix R, including key parameters of KFs (Grewal, 2008; Kim & Bang, 2018), must be defined. A suitable choice of these matrices is crucial for a satisfactory performance of KFs (Nakamura, 1982; Wu et al., 2021). We note that the construction of the noise covariance matrices Q and R is not presented as a general-purpose methodological contribution. Rather, it is a domain-specific adaptation developed to support the use of EKF in HSRIs, where mental state measurements are subjective, sparse, and limited in number. The following discussion motivates such an adaptation by reviewing existing methods and demonstrating their incompatibility with the HSRI settings.

Although various approaches have been proposed in the literature to estimate these matrices from data (see, e.g., Feng et al. (2014); Matisko and Havlena (2013); Mehra (1970); Odelson et al. (2006); Wu et al. (2021)), these methods mostly suit physical systems, i.e., systems that follow laws of physics, and thus, the state estimator should track a quantified physics-based variable as the system state. For example, Mehra (1970) developed an adaptive approach based on performing a statistical analysis of the KF to improve the estimation of the noise covariance matrices. This algorithm can be used when initial estimates of the two matrices are available, but the KF that results from applying these matrices is not optimal (i.e., the KF does not provide the minimum mean-squared error estimate of the states). They demonstrated a practical implementation of their approach in an inertial navigation system. Also in the context of navigation, Wu et al. (2021) implemented a learning-based adaptive approach to estimate the measurement and process noise covariance matrices for a KF that integrated the signals from a global navigation satellite system with signals from an inertial navigation system. Furthermore, Feng et al. (2014) proposed a recursive algorithm for estimating the process noise covariance matrix in real time and demonstrated its effectiveness in a simulation of a simple positioning system that modeled position, velocity, and acceleration.

Given that these approaches have been developed for physical systems, they are based on assumptions or have characteristics that make them unsuitable for EKFs designed to estimate the mental states of humans. Most state-of-the-art approaches rely on statistical analyses of the signals and measurements retrieved from the system

(Kim & Bang, 2018). For the sake of accuracy, these approaches require a large number of data samples (Matisko & Havlena, 2013; Mehra, 1970; Odelson et al., 2006; Wu et al., 2021), whereas in HSRI contexts, collecting a sufficiently large number of samples for mental states from humans is not feasible. Moreover, some of these approaches assume linear models (see, e.g., Feng et al. (2014); Matisko and Havlena (2013)), whereas MMM requires nonlinear functions to accurately capture the complex processes of perception, cognition, and decision-making of humans (see Chapter 2 and Chapter 3 for details).

Apart from the initial offline construction of the noise covariance matrices, it is also possible to adapt those matrices online to account for errors in the initial estimation. However, such approaches typically require a large amount of data. For example, the adaptive KF proposed in Mehra (1970) used 10 batches of 950 points in their experiments, and explicitly stated the need for large amounts of data to ensure good performance. Odelson et al. (2006) presented another statistics-based adaptive approach that improved upon Mehra (1970) to estimate Q and R matrices online. In this approach, the accuracy increases with data size, requiring several thousand samples for reliable estimation. Wu et al. (2021) adapted the noise covariance matrices online using neural networks that adjust the noise characteristics in real time. The amount of samples used for training is around 30 times larger than that in Mehra (1970). Feng et al. (2014) proposed a recursive adaptive approach that can estimate Q with significantly fewer samples. This, however, is realized at the expense of certain requirements — including linearity and very accurate knowledge of system dynamics and measurement noise — that make this approach impractical for most HSRI settings. Hence, similarly to the methods used to construct the initial values of Q and R, the adaptive approaches in the literature are well-suited for physical systems with high-frequency and abundant measurements, but are not applicable in settings that involve abstract internal mental states and infrequent, subjective measurements. Therefore, relying on state-of-the-art adaptive approaches for estimating or correcting Q and R online is also not feasible in our domain.

Finally, another approach commonly followed in practice is to generate an initial estimate of the two noise covariance matrices based on intuition and fine-tune the parameters based on performance when running the EKF (Welch & Bishop, 1995). Therefore, in order to construct the noise covariance matrices for EKFs that estimate the mental states of humans, it is necessary to delve into the physical meaning of the elements of these matrices. This allows the creation of a parallel between the typical systems for which these approaches have been developed and MMM, so we can adapt the design of the EKF accordingly. In the context of human cognition, the measurements for estimating the mental states (beliefs, goals, emotions) may be taken in two ways: (i) using physical sensors that indirectly measure psychological states of users (e.g., measuring physiological signals, such as heart rate, to infer the anxiety levels of a user, or using visual recognition to quantify user engagement); (ii) asking users to directly report their beliefs, goals, and emotions. On the one hand, the first approach is less intrusive and less disruptive in the context of HSRIs. On the other hand, because these measurements are indirect indicators of mental states, they may not accurately reflect the true mental states estimated by MMM, which can lead to increased errors in the estimation process. Moreover, not all relevant mental states

are always retrievable from such measurements. Therefore, Chapter 3 applied the second approach (i.e., directly receiving feedback about the mental states from users) to measure the mental states of users during experimental HSRIs, where data gathered in those experiments was used in this chapter to tune and validate the developed state estimators.

To construct the two matrices, which are personalized per participant, we use the sequence of data points $\mathcal{D}$ collected in two HSRIs that were part of the experiment described in Chapter 3 (the dataset collected in these experiments can be found in Morão Patrício and Jamshidnejad (2025a)). On average, 41 data points were collected per participant per session. Each data point contains the values of the input vector $\boldsymbol{u}(k)$ (i.e., environmental data related to the interaction that was accessible in the real world) and the measurement of the state vector $\boldsymbol{x}(k)$ (i.e., the beliefs, goals, and emotions of each participant) at time step k. Data is stored in sequence $\mathcal{D}$ in order of collection, i.e., based on time step k.

Process noise covariance matrix

The process noise covariance matrix Q reflects the uncertainty of the estimations made by the model in a KF about the values of the dynamic state variables (Kim & Bang, 2018; Simon, 2006). In order to construct the Q matrix for a KF that estimates mental states of humans, we assume that the noise on the different variables is uncorrelated, a common assumption in literature when constructing this matrix (Kim & Bang, 2018). The assumption implies that matrix Q is diagonal and that each diagonal element $Q_{i,i}$ reflects the uncertainty of MMM when estimating the time evolution of mental state $x_i(k)$. To determine this uncertainty, we evaluate the prediction error of MMM per variable $x_i(k)$ using data from real-life HSRI experiments. Specifically, we estimate each element $Q_{i,i}$ by calculating the Mean Squared Error (MSE) between the prediction made by the identified MMM for the value of mental state $x_i(k)$ and the corresponding value from the collected data. Therefore, Q is given by:

$$Q = \text{diag}(\sigma_1^{\text{Q}}, \ldots, \sigma_n^{\text{Q}}) \tag{4.1}$$

where n is the number of mental states (i.e., the cumulative number of possible beliefs, goals, and emotions) and σ_i^{Q} for $i = 1, \ldots, n$ is determined via:

$$\sigma_i^{\text{Q}} = \frac{1}{|\mathcal{D}|} \sum_{k \in \{1,\ldots,|\mathcal{D}|\}} (\hat{x}_i^{\text{MMM}}(k) - x_i^{\text{m}}(k))^2 \tag{4.2}$$

where $\mathcal{D}$ is the sequence of data gathered from real-life experiments and stored in chronological order, $\hat{x}_i^{\text{MMM}}(k)$ is the value that is estimated by MMM for mental state $x_i(k)$ for time step k, and $x_i^{\text{m}}(k)$ is the measured value of the variable (captured from real-life experiments for mental state $x_i(k)$). In (4.2) all values measured for the i^{th} mental state at all time steps included in the experimental dataset $\mathcal{D}$ are used to determine σ_i^{Q}. Moreover, $|\cdot|$ represents the cardinality of the sequence.

Measurement noise covariance matrix

The measurement noise covariance matrix R represents the noise in the system measurements (Kim & Bang, 2018). In physical systems, the system measurements are often obtained via sensors. Apart from using data-driven approaches, such as those discussed earlier in this section, the measurement noise covariance matrix is often extracted based on the characteristics of the sensors or via statistical analyses of the measurements obtained from the system (Grewal, 2008; Welch & Bishop, 1995).

In the current context, since the measurements were provided directly by each user (rather than collected through physical sensors), it is not possible to construct matrix R based on sensor information, as is common for this matrix. Moreover, since the number of samples captured during the experimental HSRIs is limited, it is not possible either to conduct complex statistical analyses on the data to deduce meaningful patterns on the characteristics and evolution of the data, which could assist in constructing R.

Therefore, an alternative approach was adopted to construct the measurement noise covariance matrices for each user. Specifically, the elements of the R matrix were estimated using state measurements collected under steady-state conditions, for which it is necessary to keep the inputs to the system constant. Under such conditions, any variation observed in the measured states is assumed to arise from measurement noise rather than changes in the underlying system. Accordingly, the variance of these measurements was used to estimate the measurement noise covariance.

Given the aforementioned limitations in data collection inherent to experiments with humans, it was not straightforward to re-create such steady-state conditions while collecting measurements for two reasons. First, prolonged constant conditions may lead to distraction or boredom of participants, which increases the measurement noise. Second, due to the scarcity of the measured data, dedicating many measurements solely to identifying the noise covariance matrix is inefficient. Therefore, the data used to identify MMM was reused to identify the noise covariance matrix R, by isolating sequences of at least three consecutive data points with similar values for the inputs. As mentioned earlier in this section, each data point includes the vector of all mental states in a time step. Sequences with fewer than three data points were excluded because they provided too little data to reliably estimate the variability caused by measurement noise. The threshold of three points was chosen as a practical compromise, balancing the need for reliability with the limited amount of available data. The variance of the i^{th} measured mental state per sequence $\mathcal{S}_\ell$ is computed by:

$$\sigma^{\text{R}}_{i,\ell} = \frac{1}{|\mathcal{S}_\ell|} \sum_{x^{\text{m}}_i(k) \in \mathcal{S}_\ell} \left(x^{\text{m}}_i(k) - \bar{x}_{i,\ell}\right)^2 \tag{4.3}$$

where $\bar{x}_{i,\ell}$ is the average value of $x^{\text{m}}_i(k)$ across all time steps within sequence ℓ. Assuming a total number n^{seq} of such isolated sequences, for each mental state, the variance $\sigma^{\text{R}}_{i,\ell}$ is averaged across all the sequences, yielding a representative value of the variance for that mental state. We have:

$$\sigma^{\text{R}}_i = \frac{1}{n^{\text{seq}}} \sum_{\ell \in \{1,\ldots,n^{\text{seq}}\}} \sigma^{\text{R}}_{i,\ell} \tag{4.4}$$

As it is common in practical implementations of KFs, we assume that the measurement noise covariance matrix R is diagonal (Grewal, 2008). Thus, the R matrix is given by:

$$R = \text{diag}(\sigma_1^{\text{R}}, \ldots, \sigma_n^{\text{R}}) \tag{4.5}$$

4.2.2. Implementation of the Extended Kalman Filter

Given the central role of the process and measurement models in the EKF, we begin this section by presenting their general form, followed by an overview of the classical EKF, and a description of the three EKF variants proposed in this work.

Process and measurement models

In our implementation, the state transition function $f(\cdot)$ of the process model used in the EKF is derived from the MMM, originally presented in Chapter 2 and subsequently refined in Chapter 3. The state transition function captures the dynamic evolution of three groups of mental state variables — beliefs, goals, and emotions — where each category can include multiple variables. The measurement model links these internal states to user self-reports.

The general formulations of the process and measurement models are, respectively, given by:

$$\boldsymbol{x}(k) = f\left(\boldsymbol{x}(k-1), \boldsymbol{u}(k-1)\right) + \boldsymbol{w}(k-1) \tag{4.6a}$$

$$\boldsymbol{x}^{\text{m}}(k) = h\left(\boldsymbol{x}(k)\right) + \boldsymbol{\nu}(k) \tag{4.6b}$$

where $\boldsymbol{x}(k)$ is the vector of all mental states at time step k, and $\boldsymbol{x}^{\text{m}}(k)$ is the vector of all measured (e.g., self-reported) mental states at time step k. The vector $\boldsymbol{u}(k)$ includes the inputs at time step k, i.e., the real-world data that affects the cognition of the user. The vectors $\boldsymbol{w}(k)$ and $\boldsymbol{\nu}(k)$ are, respectively, the process and measurement noise, and are assumed to follow a Gaussian distribution with zero mean and Q and R covariance matrices, respectively. The function $f(\cdot)$ is the state transition function of MMM (see Chapter 2 and Chapter 3 for details), and $h(\cdot)$ is the measurement function. Both $f(\cdot)$ and $h(\cdot)$ are, in general, vector functions.

Based on the formulation in Chapter 3, we briefly summarize the structure of the state transition function $f(\cdot)$, focusing on its dependence on the state variables. This is the portion relevant for the linearization step in the EKF, which requires computing the Jacobian with respect to $\boldsymbol{x}(k)$ (explained in detail in Section 4.2.2 and employed via (4.9)). For a detailed description of the full model, we refer the reader to Chapter 3. The following equations presents $f_i(\cdot)$, the i^{th} component of the nonlinear state transition function $f(\cdot)$, corresponding to the update of the i^{th} state variable x_i:

$$\begin{aligned} f_i\left(\boldsymbol{x}(k-1), \boldsymbol{u}(k-1)\right) = w_i x_i(k-1) + \\ \sum_{i \neq j} w_{ji}(k) x_j(k-1) + f^{\text{input}}(\boldsymbol{u}(k-1)) \end{aligned} \tag{4.7}$$

where w_i is a scalar weight that regulates the influence of $x_i(k)$ on its next realization. The terms $w_{ji}(k)$ are weights that represent the influence of other mental state

variables $x_j(k-1)$, with $j \neq i$, on the evolved value $x_i(k)$ of mental state i. These weights are defined as piecewise constant, depending on the region of the state space. For example, the influence of the emotion "bored" on the goal of "quitting the interaction" may differ depending on whether the emotion is positive (the user is bored) or negative (the user is engaged). In other words, $w_{ji}(k) = w_{ji}^{+}$ when $x_j(k) > 0$, and $w_{ji}(k) = w_{ji}^{-}$ when $x_j(k) \leq 0$. This structure results in a globally nonlinear function composed of multiple locally affine components. The function $f^{\text{input}}(\cdot)$ accounts for the influence of external inputs (i.e., real-world data) perceived by the user. However, its detailed structure is omitted here, as it does not contribute to the linearization process used in the EKF.

Additionally, in our application, the measurements directly correspond to the mental states, as they are measured through self-reports from users. Consequently, the measurement function is given as:

$$h\left(\boldsymbol{x}(k)\right) = \boldsymbol{x}(k) \tag{4.8}$$

Adaptation of Extended Kalman Filter

Following Kim and Bang (2018), we divide the estimation made by the EKF into two phases: the prediction phase, represented via (4.9), and the correction phase, represented via (4.10).

Prediction phase The prediction phase is formulated by:

$$\hat{\boldsymbol{x}}^{\text{MMM}}(k) = f\left(\hat{\boldsymbol{x}}(k-1), \boldsymbol{u}(k-1)\right) \tag{4.9a}$$

$$P^{-}(k) = F(k-1)P(k-1)F^{\top}(k-1) + Q \tag{4.9b}$$

where $\hat{\boldsymbol{x}}^{\text{MMM}}(k) = [x_1^{-}(k), \ldots, x_n^{-}(k)]^{\top}$ is the vector of all estimated mental states by the MMM at time step k, $f(\cdot)$ is the state transition function given by the state space representation of MMM, $F(k)$ is the Jacobian matrix of $f(\cdot)$ evaluated for realized value of the state vector $\hat{\boldsymbol{x}}$ at time step k, and $P(k)$ is the error covariance matrix predicted for time step k.

Correction phase The correction phase of the EKF is formulated via:

$$K(k) = P^{-}(k)H^{\top}(k)\left(R + H(k)P^{-}(k)H^{\top}(k)\right)^{-1} \tag{4.10a}$$

$$\hat{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}^{\text{MMM}}(k) + K(k)\left(\boldsymbol{x}^{\text{m}}(k) - h\left(\hat{\boldsymbol{x}}^{\text{MMM}}(k)\right)\right) \tag{4.10b}$$

$$P(k) = \left(I - K(k)H(k)\right)P^{-}(k) \tag{4.10c}$$

where $P^{-}(k)$ is computed via (4.9), $h(\cdot)$ is the output function of the state space representation of MMM, $H(k)$ is the Jacobian matrix of $h(\cdot)$ evaluated at $\hat{\boldsymbol{x}}(k)$, $\boldsymbol{x}^{\text{m}}(k)$ is the vector of all measured mental states at time step k, and $P(k)$ is the error covariance matrix.

While the prediction phase, shown in (4.9), depends only on the model prediction (i.e., on $\hat{\boldsymbol{x}}^{\text{MMM}}(k)$), the correction phase, given by (4.10), depends also on the

measurements (i.e., on $\boldsymbol{x}^{\mathrm{m}}(k)$) captured for the mental states. Therefore, while the prediction phase can be carried out in each discrete time step and is only potentially limited by the computation speed of MMM, the correction phase can only be performed when a measurement of the mental states becomes available.

In practical implementations of MMM, measurements of mental states may be unavailable at every discrete time step, as explained in Section 4.2.1. Therefore, to directly apply both phases of the EKF at each time step k, prior adjustments are required.

We next propose and implement three approaches to circumvent this issue. Figure 4.2 illustrates the functioning of these three methods.

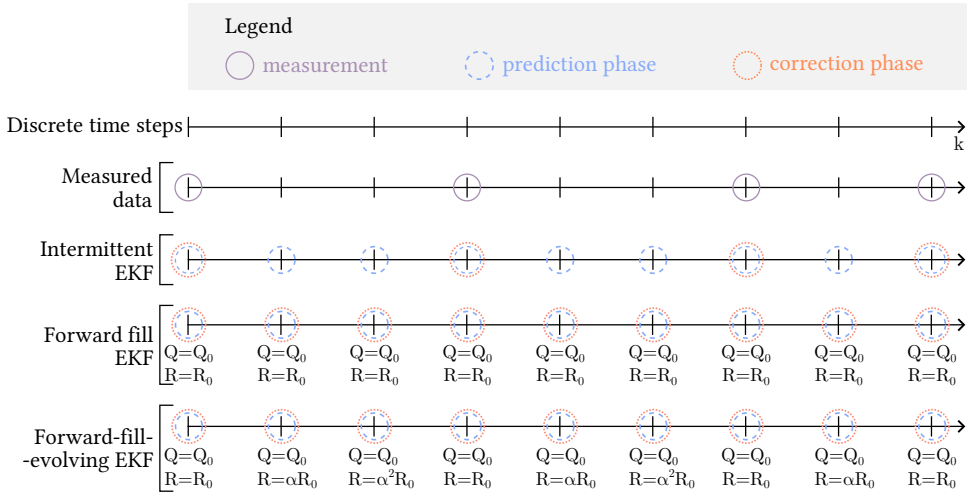


Figure 4.2: Illustration of the functioning of the three proposed versions of the EKF depending on the availability of measurements. The top axis represents the discrete time steps, with green solid circles showing time steps when measurements for mental states are available. The bottom three axes show when the prediction and correction phases are applied for each variant of EKF, depending on measurement availability. Furthermore, the evolution of the Q and R matrices over time is depicted below each axis.

Intermittent EKF

The intermittent EKF employs a common approach for implementing EKFs when measurements are less frequent than the estimation updates. In this approach, the prediction phase is executed at every time step. When measurements are available, the correction phase is also performed, and the estimation is obtained via (4.10b). In the absence of measurements, the correction phase is skipped, and the state estimate is obtained from (4.9a), which relies only on the prediction made by MMM (see Figure 4.2). Figure 4.3 illustrates the operation of the intermittent EKF, showing when the prediction and correction phases are applied based on the algorithm and the availability of measurements.

Despite its simplicity and common deployment, this approach faces limitations when applied to HSRIs. In such applications, the sparsity of measurements may lead the EKF

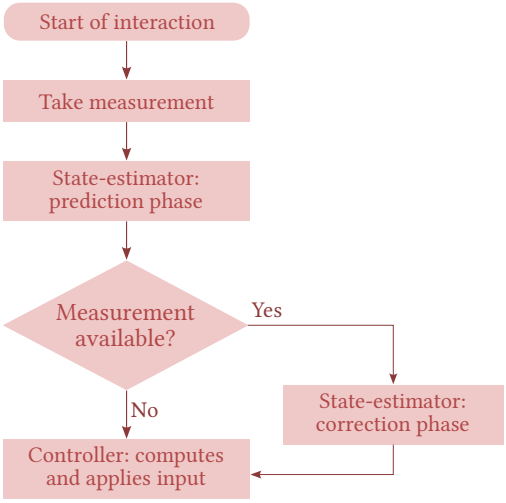


Figure 4.3: Flowchart illustrating the control loop steps for the intermittent EKF. The diagram outlines the sequential steps of this state estimator, depending on the availability of measurements.

to rely only on model predictions for most time steps. As a result, it provides limited improvements compared to approaches that do not use a state estimator and estimate the states directly from the model, as in Chapter 3. To overcome this limitation, we propose an alternative approach that more effectively leverages the information gathered from available measurements.

Forward-fill EKF

The second approach, called the forward-fill EKF, addresses the limitations of the intermittent EKF by incorporating the measurements more frequently, thereby enabling both phases of the EKF to be executed every discrete time step k. To achieve this, the measurements are forward-filled, i.e., at each time step k, the value of measurement $\boldsymbol{x}^{\mathrm{m}}(k)$ is set to the most recent available measurement. Consequently, if no new measurement is available for q consecutive time steps, the measurement $\boldsymbol{x}^{\mathrm{m}}(k)$ corresponding to time step k is re-used for $\boldsymbol{x}^{\mathrm{m}}(k+1), \ldots, \boldsymbol{x}^{\mathrm{m}}(k+q)$. Figure 4.4 illustrates the operation of the forward-fill EKF, demonstrating its behavior depending on the availability of measurements.

Although this approach benefits from continuous incorporation of measurement information, it may introduce errors when mental states change frequently or abruptly. Nevertheless, we expect that the contribution of the model to the EKF estimation — combined with the personalized construction of measurement and process noise covariance matrices based on the noise characteristics of the model and measurements per participant — will help to mitigate such measurement errors over time.

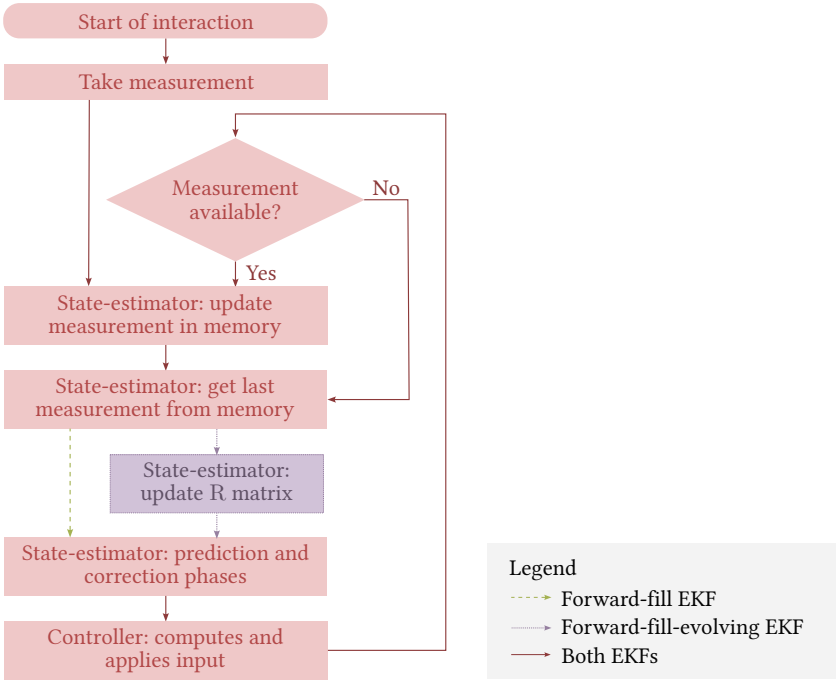


Figure 4.4: Flowchart illustrating the control loop steps for the forward-fill EKF and forward-fill-evolving EKF. The diagram outlines the sequential steps of these two state estimators, depending on the availability of measurements. The additional step specific to the forward-fill-evolving EKF is distinguished with a red dotted line.

Forward-fill-evolving EKF

The third and final approach, the forward-fill-evolving EKF, accounts for the decreasing reliability of measurements over time in the absence of new measurements. Since the measurement noise covariance matrix $R(k)$ reflects the expected measurement error, we propose increasing the value of the elements of this matrix during time steps when no new measurements are available. Therefore, matrix $R(k)$ is updated by:

$$R(k) = \begin{cases} R_0, & \text{if mental states are measured at} \\ & \text{time step } k \\ \alpha\, R(k-1), & \text{otherwise} \end{cases} \tag{4.11}$$

where $\alpha > 1$ is a scalar parameter and R_0 is the measurement noise covariance matrix calculated offline. This strategy ensures that whenever a new measurement is unavailable and the previous one is reused, the entries of the $R(k)$ matrix are increased with respect to the previous time step. As a result, compared to a fixed $R(k)$ matrix, the Kalman gain is lower (see (4.10a)), and the influence of the measurement in the estimation of the updated mental states is reduced (see (4.10b)). Furthermore, the longer the gap between capturing two measurements, the less the KF relies on

the measurements and the more it relies on the model estimation. Figure 4.2 (bottom plot) depicts how the $R(k)$ matrix evolves with this approach based on measurement availability. The value of parameter α determines the growth of matrix $R(k)$ per time step when there are no new measurements. Thus, α effectively models the rate of decay of the confidence in reused measurements.

This strategy constitutes a simple, domain-specific form of adaptive filtering, as it heuristically modifies the measurement noise covariance $R(k)$ based on the availability of new measurements. Figure 4.4 illustrates the operation of the forward-fill-evolving EKF, demonstrating its behavior depending on the availability of measurements and how its behavior differs from the forward-fill EKF.

Based on the configuration of these three approaches, we formulate the following three hypotheses:

Hypothesis 1 *Intermittent EKF will perform similarly to the estimator in Chapter 3.*

Hypothesis 2 *Forward-fill EKF will outperform the estimator in Chapter 3.*

Hypothesis 3 *Forward-fill-evolving EKF will outperform forward-fill EKF, as it better balances the predictions generated by the MMM with the varying reliability of measurements over time.*

These hypotheses are further discussed in Section 4.4.

Convergence Considerations

Formal convergence guarantees for EKF usually require restrictive assumptions, and EKF-based applications in nonlinear and adaptive settings often rely on empirical assessment of convergence behavior (Bonnabel & Slotine, 2015; Boutayeb et al., 1997; Simon, 2006; Sueki et al., 2022). In the context of this chapter, the specific formulation of MMM is customized per participant and HSRI scenario, making a general formal convergence proof for the proposed EKFs infeasible.

Nonetheless, some theoretical insight can be gained by examining the general structure of the state transition function $f(\cdot)$ of MMM. Although $f(\cdot)$ is nonlinear, it is piecewise affine with respect to the state $\boldsymbol{x}(k)$ within local regions of the state space (see (4.7)). In these regions, the EKF behaves as a standard KF, for which local convergence can be established under classical conditions, such as observability, noise properties, and accurate initial estimates (Bonnabel & Slotine, 2015; Simon, 2006). While these conditions are likely to hold in practice given the structure of the model and the assumptions adopted (see Section 4.2.2), they should be explicitly verified in each context-specific instantiation of MMM to formally guarantee local convergence. However, even if local convergence is satisfied, it has limited practical use in the present context. During interaction, dynamic cognitive state variables (i.e., beliefs, goals, and emotions) may transition across these affine regions of the state space. As a result, convergence guarantees within those regions do not generalize globally.

Similarly, a general global convergence proof is not tractable, due to the context-specific and personalized nature of cognition modeled through MMM. Each participant-specific model yields unique dynamics, requiring separate convergence

analyses. Moreover, any adaptation of the model during deployment — as expected in realistic, long-term HSRIs — invalidates previously established guarantees, making global guarantees impractical in real-world applications.

Given these theoretical limitations, we adopt an empirical approach to assess convergence behavior, since this aligns better with the practical requirements of our setting. In Section 4.4, we present experimental results that demonstrate how estimation uncertainty evolves by analyzing the trace of the covariance matrix of the proposed state estimators for each participant over time. This provides a practical indication of whether the filters maintain bounded uncertainty over time, supports the safe and effective use of EKFs, and offers an interpretable and adaptive measure for the reliability of estimations in real time during personalized HSRIs.

4.3. Case Study

In this section, we present the case study used to evaluate the state estimators proposed in this chapter. First, we describe the experimental setup in Section 4.3.1, followed by an explanation of the assessment methodology in Section 4.3.2.

4.3.1. Experimental Setup

This experiment assesses the extent to which the proposed model-based mental state estimators improve the online predictions of the mental states of participants, compared to estimations made by MMM without a state estimator (see Figure 4.1 for a comparison between the two architectures).

The state estimators were evaluated on data we collected within an extensive human-robot interaction study, in which the behavioral control framework of the robot, task design, and system implementation have been detailed in Chapter 3. The dataset includes self-reported mental state measurements from 10 human participants, collected across three sessions each. For completeness, we provide a concise summary of the experimental protocol and robot behavior in this section. Readers interested in the full system design and implementation details are referred to Chapter 3.

The experiments were conducted using a NAO V6 humanoid robot (SoftBank Robotics, n.d.), a platform widely used in HSRI research due to its versatility, ease of programming, and suitability for social applications (Robaczewski et al., 2021), with research applications spanning health-care (Alhmiedat & Alotaibi, 2022; M. H. Lee et al., 2023) and educational contexts (D. Pandey et al., 2022). The high-level behavior control of NAO was implemented in Python using the NAOqi SDK, and was executed on a connected laptop running Windows 10 with an Intel CPU (i7–1185G7 3.0GHz), and 32 GB of RAM. Built-in modules provided by the SDK were employed for speech synthesis and motion control. Motion control also included adapted versions of movements originally designed in Choreographe.

In the HSRI sessions, 10 participants interacted with a NAO Robot (SoftBank Robotics, n.d.) while solving chess puzzles (see Figure 4.5). As outlined in Section 4.1.2, NAO dynamically selected the difficulty level of the puzzles and decided whether to display entertaining behaviors during the interaction, aiming to keep the participants engaged. Meanwhile, NAO evaluated and tracked the mental states, i.e., one value

for the belief (related to the difficulty level of the puzzle), two values for the goals (related to quitting the interaction and to skipping the puzzle), and two values for the emotions (related to the frustration and the boredom) per participant. The objective for NAO was to select actions that minimized the values of goals and emotions, and brought the belief value as close to zero as possible — indicating that the perceived difficulty of the game was suitably balanced, neither too easy nor too hard.



Figure 4.5: Experimental setup of HSRI with NAO robot (SoftBank Robotics, n.d.). Reproduced from Chapter 3.

The experiment consisted of three sessions. The first two sessions, each lasting 45 to 60 minutes, were used to collect measurements of the five variables through participant self-reports. On average, 41 measurements were collected per session. The data was then used to identify the parameters of MMM to personalize the model per participant. In the third session, the personalized MMM was used by the robot to track the mental states of the participants. It was also embedded in a predictive model-based controller to select optimal actions for NAO. To mitigate the estimation error accumulation over time that occurs due to a lack of measurements, the participants were requested to provide self-reports of their mental states over time (approximately 10 measurements per participant during the 35-minute interaction).

In our deployment, data from the first two sessions was used to construct the process and the measurement noise covariance matrices Q and R for each participant according to the approach described in Section 4.2.1. The parameter α of the forward-fill-evolving EKF (see (4.11)) was set to 2. A data-driven tuning approach was not pursued, as it would have required an additional dataset specifically for tuning α after the initial values of the noise covariance matrices Q and $R(k)$ had been obtained. Instead, α was selected as a trade-off between enabling a meaningful comparison between forward-fill EKF and forward-fill-evolving EKF, and maintaining a conservative value (i.e., close to 1) to avoid a substantial increase in $R(k)$. Data from the third session (which we refer

to as the validation dataset) was then used as online input to simulate and assess the estimations that each EKF would have produced during the HSRI. These predictions were compared against the predictions done in Chapter 3, i.e., the predictions produced by the standalone model. After analyzing the results of the three proposed EKFs, the value of parameter α of the forward-fill-evolving EKF was varied to assess its impact and determine whether the data-driven personalization of this parameter influences the performance of the state estimator.

The intermittent EKF, forward-fill EKF, and forward-fill-evolving EKF were implemented in Python 3.9. The assessment of the EKFs was conducted in an offline setting using a laptop running Windows 10 with an Intel CPU (i7–1185G7 3.0GHz), and 32 GB of RAM.

4.3.2. Estimation Assessment

In order to assess the performance of the three EKF variants proposed in this article and to compare them with the baseline estimations from Chapter 3, we used the same validation dataset (Morão Patrício & Jamshidnejad, 2025a) from that study, treating it as data that is being captured sequentially and online.

Since our goal is to reduce the cumulative prediction error that builds up over time in the absence of frequent measurements, we assessed the accuracy of the estimations made by the EKFs based on the prediction errors that accumulate between consecutive measurements.

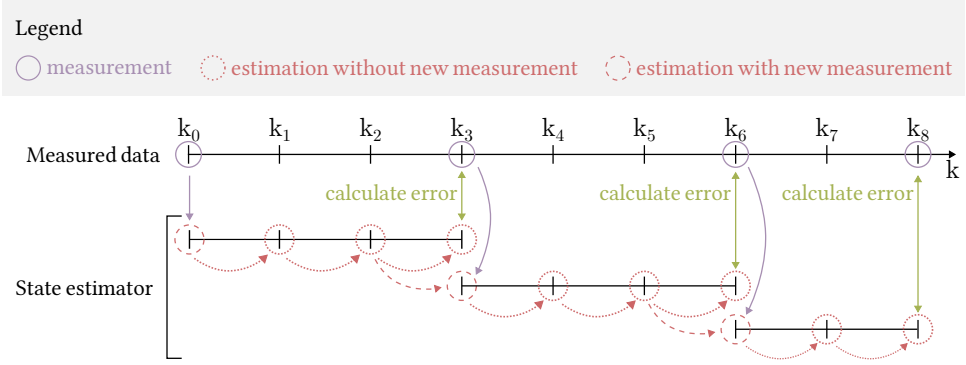


Figure 4.6: Example of computing the cumulative prediction errors between measurements: The measurement at time step k_0 initializes the EKF. For the next time steps k_1 and k_2 with no new measurements, predictions are made using only the model in the case of the intermittent EKF, and using the forward-filled value of the last measurement for the forward-fill-evolving EKF and forward-fill-evolving EKF. This is represented by the blue dotted arrows and circles. At time step k_3, a new measurement is available, prompting two parallel computations. First, an estimation is performed assuming no new measurement (blue dotted arrow and circle) that is compared to the actual measurement to compute the cumulative error between time steps k_0 and k_3. Second, an estimation is made for time step k_3 utilizing the measurement $\boldsymbol{x}^{\mathrm{m}}(k_3)$ (specified by the green arrow, where its transition from the previous time step has been shown via a dashed blue arrow). This estimation is then used for future estimations. The same process is repeated at time steps k_6 and k_8, when cumulative errors are also computed. The average cumulative error is given by averaging the cumulative prediction errors at time steps k_3, k_6, and k_8.

To ensure a fair comparison, all estimators were evaluated using the same dataset. The evaluation process starts from a time step k, when a measurement $\boldsymbol{x}^{\mathrm{m}}(k)$ is available (e.g., k_3 in Figure 4.6). The EKF estimates the state based on the previous estimation $\hat{\boldsymbol{x}}(k-1)$ and the current measurement $\boldsymbol{x}^{\mathrm{m}}(k)$, applying the prediction and correction phases as described in (4.9) and (4.10). From this time step until a new measurement is available, for all time steps $\kappa > k$ (e.g., at time steps k_4 and k_5 in Figure 4.6), the estimations are conducted differently. For the intermittent EKF only the prediction phase is executed, whereas for the forward-fill EKF and forward-fill-evolving EKF, both prediction and correction phases are applied using the last available measurement $\boldsymbol{x}^{\mathrm{m}}(k)$.

When a new measurement becomes available at time step $k+m$ (e.g., at time step k_6 in Figure 4.6), two operations are performed. First, an estimation is made as if no new measurements were available, as was done for the time step $k+m-1$. The resulting estimation is compared to the actual measurement $\boldsymbol{x}^{\mathrm{m}}(k+m)$. This yields the cumulative prediction error since the last measurement. Second, the measurement $\boldsymbol{x}^{\mathrm{m}}(k+m)$ is incorporated to update the state estimation, which is then used for subsequent predictions.

This process is repeated at each time step when a measurement is available, excluding the first measurement, which serves as initialization. For each participant, the average cumulative error is computed as the mean absolute value of all cumulative errors across their session.

4.4. Results and Discussions

In this section, we present and discuss the estimations made by the EKFs for the mental states of the human participants who interacted with NAO while solving chess puzzles. In the absence of established state estimators for internal cognitive states in HSRIs, we compare our proposed methods against the standalone open-loop estimator based on MMM used in Chapter 3, which represents the current state of practice in this domain. Although advanced state estimation methods exist for physical systems, they are not directly applicable here, due to fundamental differences in measurement quantity and the abstract nature of mental state variables.

4.4.1. State Estimation Accuracy

The average cumulative prediction errors between consecutive measurements for each state estimator are presented in Figure 4.7. For each state estimator and participant, these errors were computed as described in Section 4.3.2. Figure 4.7 presents the distribution of errors across the ten participants using box plots for each state estimator. The minimum, maximum, median, and mean errors are also provided in Table 4.1. As expected, the intermittent EKF did not improve prediction accuracy over the baseline results from Chapter 3 (see Hypothesis 1 in Section 4.2.2). Its performance is nearly identical to that of the original MMM when no state estimator is used (see Figure 4.7 and Table 4.1). In contrast, the forward-fill EKF demonstrates a clear prediction enhancement over the baseline where estimations were made by MMM alone. In fact, while the minimum prediction error remains similar, the maximum, median, and

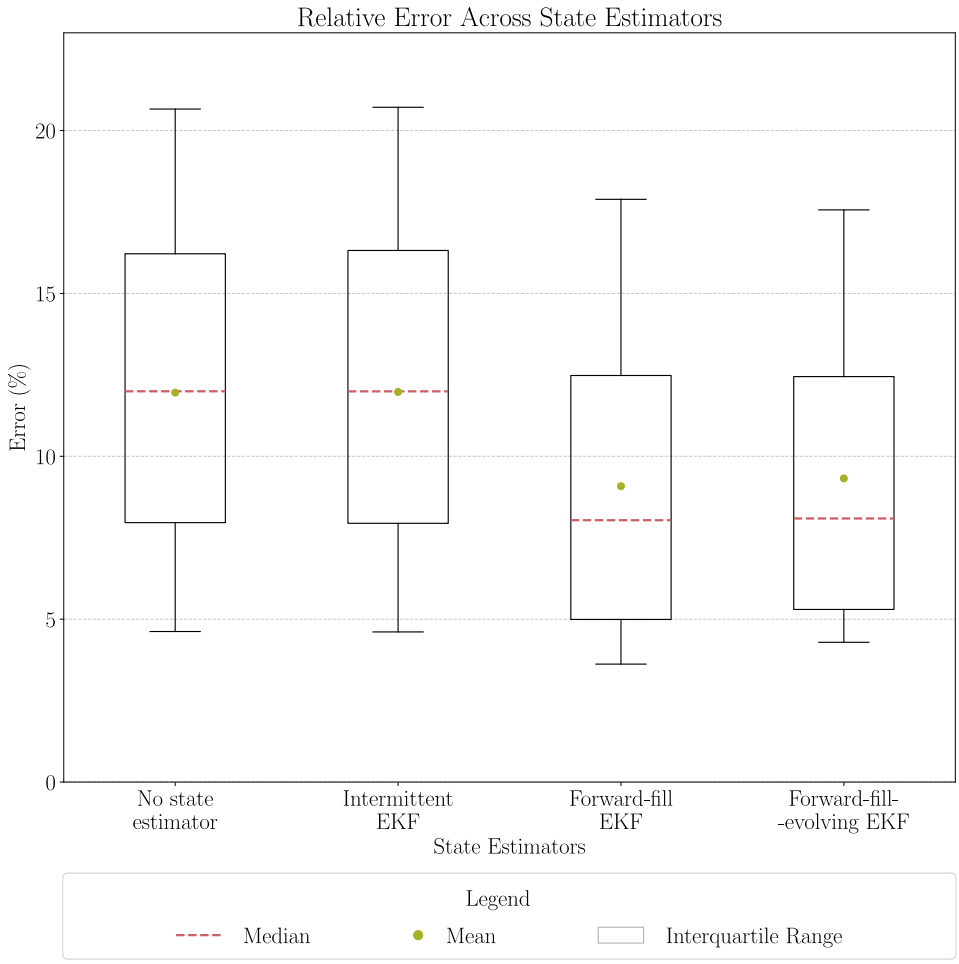


Figure 4.7: Relative prediction error of the mental states of the ten participants across different state estimators. The average prediction errors per state estimator obtained during the interaction with each participant are displayed as box plots. The intermittent EKF, forward-fill EKF, and forward-fill-evolving EKF, which are adapted versions of EKFs (Simon, 2006) developed in this chapter and described in Section 4.2.2, are compared with the "No state estimator" baseline used in Chapter 3.

mean prediction errors decrease by 2.8%, 4.0%, and 2.9%, respectively. This confirms Hypothesis 2.

In Hypothesis 3, we theorized that increasing the values of the measurement noise covariance matrix $R(k)$ during periods without new measurements, as implemented in the forward-fill-evolving EKF, would enhance the state estimator performance. However, the initial results did not support this hypothesis. The performance of forward-fill-evolving EKF was comparable to that of forward-fill EKF, as shown in Figure 4.7 and Table 4.1. Several factors may explain this outcome: First, the forward-fill EKF already yields significant performance improvements, leaving limited

Table 4.1: Relative prediction errors of the mental states for the ten participants across different state estimators. The intermittent EKF, forward-fill EKF, and forward-fill-evolving EKF are three variants developed in this chapter, adapted from the standard EKF (Simon, 2006) and whose technical design and implementation are described in Section 4.2.2. These are compared against the "No state estimator" baseline used in Chapter 3. The minimum, maximum, median, and mean values of the average prediction error per participant and state estimator are presented.

State estimator		No state estimator (from Chapter 3)	Intermittent EKF	Forward-fill EKF	Forward-fill-evolving EKF
Minimum error (%)	Error	4.621	4.608	3.620	4.290
	Improvement	—	-0.013	-1.001	-0.331
Maximum error (%)	Error	20.660	20.714	17.888	17.564
	Improvement	—	0.054	-2.772	-3.096
Median error (%)	Error	11.996	11.994	8.038	8.091
	Improvement	—	-0.002	-3.958	-3.905
Mean error (%)	Error	11.956	11.977	9.083	9.322
	Improvement	—	0.021	-2.873	-2.634

room for further prediction error reduction in this particular HSRI context. In general, mental states of humans are inherently difficult to estimate with high precision, making it unrealistic to expect negligible prediction errors for such states. Second, the mental states captured from the participants in this dataset are relatively stable for most participants, meaning that measurements remain informative and reliable over longer periods. Consequently, penalizing the degradation of measurement accuracy over time by increasing the elements of the $R(k)$ matrix — which distinguishes forward-fill-evolving EKF from forward-fill EKF — may be unnecessary in this context. Finally, the parameter α, which governs how quickly matrix $R(k)$ evolves over time, was set to 2 for all participants, as explained in Section 4.3.1. Since data from the first two sessions was used to estimate the R and Q matrices, and data from the third session was used to assess the performance of the state estimators, a fixed, non-personalized value was used for α, which may explain why the original hypothesis was not supported by the results. The possibility of enhancing the performance of the forward-fill-evolving EKF by personalizing α is, thus, discussed later in this section.

Overall, the modifications offered by the forward-fill EKF proved to be the most effective enhancements over the original estimation method. The results indicate that the average estimation error across all participants is reduced by 2.9% when using the forward-fill EKF. This provides preliminary evidence that the forward-fill EKF enhances estimation accuracy. However, there are considerable differences in the magnitudes of the original estimation errors among the ten participants. Therefore, we

further analyze the estimation errors of each participant individually to assess more thoroughly the degree of improvement provided by the proposed EKFs.

To gain more insight into the impact of each state estimator, we analyze the average prediction errors over time for each participant individually. As shown in Figure 4.8 and Table 4.2, forward-fill EKF yielded a lower prediction error than the baseline error for all participants except for participant P8. For participants whose baseline prediction error was already below 10% (i.e., participants P3, P7, P9, and P10), the gains were understandably minor (see Figure 4.8). For participants P1, P2, and P4, forward-fill EKF led to significant improvements for mental state prediction error compared to estimations conducted without a state estimator, with decreases in prediction error of 4.0%, 13.1%, and 7.6%, respectively.

Table 4.2: Relative prediction error of the mental states for each participant per state estimator (the EKF variants developed in this chapter, whose technical design is described in Section 4.2.2) vs. no state estimator (introduced in Chapter 3).

Participant	State estimator			
	No state estimator (from Chapter 3)	Intermittent EKF	Forward-fill EKF	Forward-fill-evolving EKF
P1	11.449	11.443	7.472	7.813
P2	16.689	16.814	3.620	4.899
P3	9.147	9.127	8.604	8.369
P4	20.660	20.714	13.047	14.396
P5	12.542	12.545	12.455	12.411
P6	14.798	14.833	12.489	12.459
P7	4.621	4.608	4.231	4.290
P8	16.973	16.987	17.888	17.564
P9	5.109	5.149	4.469	4.521
P10	7.569	7.548	6.554	6.496

While the forward-fill EKF outperformed the forward-fill-evolving EKF for some participants, the opposite occurred for others. This variation suggests that the performance of the forward-fill-evolving EKF may be enhanced by personalizing parameter α per participant. Table 4.3 presents the average cumulative prediction error per participant using the forward-fill-evolving EKF under different values for α. The results show that the value of α that results in optimal performance of the forward-fill-evolving EKF varies across participants, further supporting the need for personalizing α.

4.4.2. Computational Efficiency

In addition to assessing the estimation accuracy, we also evaluated the computational efficiency of each state estimator. The average running times for one iteration of each

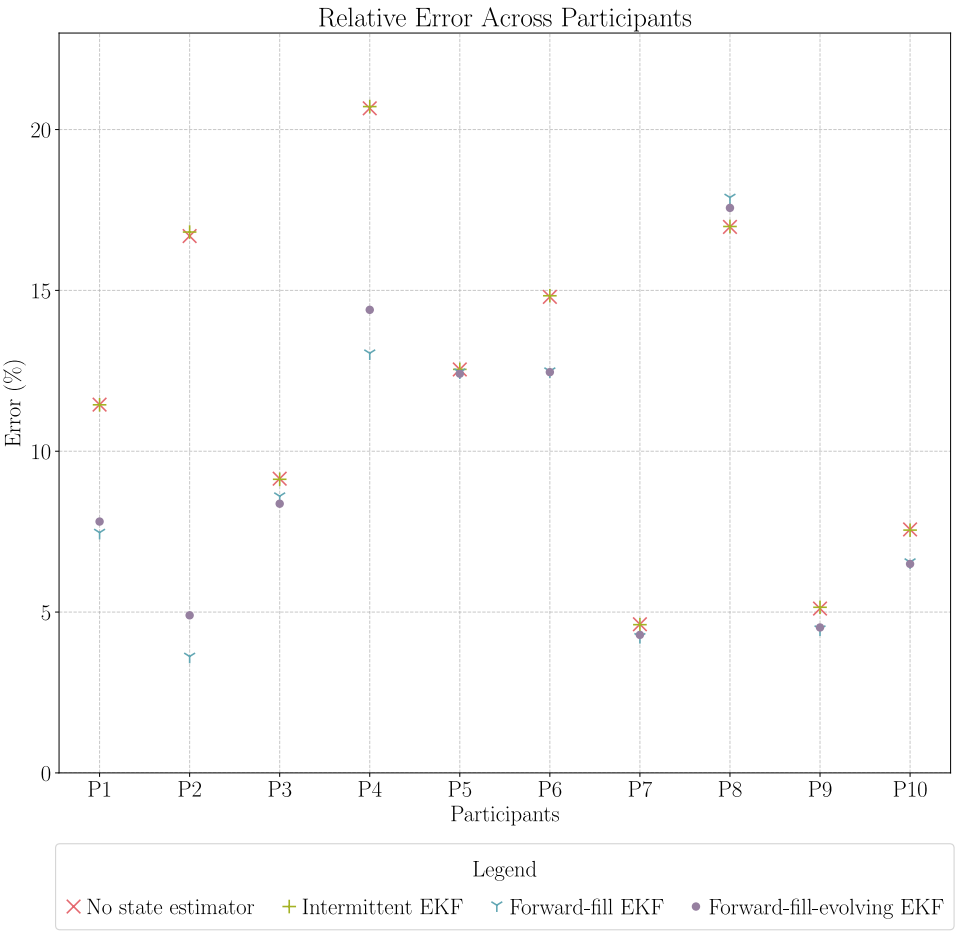


Figure 4.8: Relative prediction error of the mental states for each participant for each state estimator, compared to the case where no state estimator is deployed, introduced in Chapter 3. The intermittent EKF, forward-fill EKF, and forward-fill-evolving EKF are adapted versions of EKFs (Simon, 2006) developed in this chapter and described in Section 4.2.2. Participants are represented by the letter "P" and a number on the horizontal axis.

state estimator were approximately 1.6 ms, 1.7 ms, and 1.9 ms, for the intermittent EKF, forward-fill EKF, and forward-fill-evolving EKF, respectively. Additionally, the maximum time for one iteration was 3.2 ms, 4.1 ms, and 5.1 ms for the intermittent EKF, forward-fill EKF, and forward-fill-evolving EKF, respectively. Such low computation times indicate that all filters are suitable for real-time applications in HSRI settings, where sampling intervals are typically on the order of several seconds (see Chapter 3 for details).

Table 4.3: Impact of varying parameter α on the prediction error of the mental states for each participant when using the forward-fill-evolving EKF. The value of α that minimizes the overall prediction error for each participant is shown in bold.

α	1[1]	1.1	2	5	10	100
P1	**7.472**	7.482	7.813	9.742	10.541	11.173
P2	**3.620**	3.689	4.899	8.925	12.138	16.035
P3	8.604	8.592	8.369	**8.120**	8.325	8.89
P4	**13.047**	13.093	14.396	17.838	19.143	20.449
P5	12.455	12.453	12.411	12.011	**11.658**	11.941
P6	12.489	12.489	**12.459**	13.096	14.804	16.972
P7	4.231	4.234	4.290	4.449	4.401	**4.167**
P8	17.888	17.878	17.564	17.190	17.415	**17.054**
P9	**4.469**	4.474	4.521	4.644	4.772	5.069
P10	6.554	6.552	6.496	5.965	**5.765**	5.911

[1] When $\alpha = 1$, the forward-fill-evolving EKF reduces to the forward-fill EKF.

4.4.3. Convergence Analysis

As discussed in Section 4.2.2, general formal convergence guarantees are not tractable in our setting due to the nature of HSRI settings, which involve nonlinearity and depend on both context and user-specific personalization. In practice, empirical assessment of convergence behavior is more practical and informative in HSRIs. Therefore, we analyze the evolution of the trace of the covariance matrix $P(k)$ over time to evaluate whether the filters maintain bounded uncertainty during deployment. For the case of intermittent EKF, in steps where measurement is not available, the correction phase is not performed, and the actual covariance matrix $P(k)$ is not available (see (4.10c)). For these time steps, we report the predicted covariance matrix $P^{-}(k)$ (see (4.9b)) instead. The results are presented in Figure 4.9.

The covariance matrix of the intermittent EKF does not seem to converge for most participants, as its trace fluctuates over time. Between consecutive measurements, the trace of the predicted covariance matrix $P^{-}(k)$ increases: since the correction phase is not performed, process noise accumulates, and the predicted covariance matrix increases. Then, the trace of $P(k)$ only decreases when the correction phase is performed. This fluctuating behavior prevents the filter from reaching a stable error covariance and indicates that the filter does not exhibit practical convergence in the absence of frequent updates.

In contrast, the trace of the covariance matrix $P(k)$ for forward-fill EKF shows convergence for all participants. After an initial phase where the trace of $P(k)$ increases — likely due to a low initialization of $P(k)$ — it then decreases within the first 10 time steps and remains stable over time for all participants. This indicates that the filter maintains bounded uncertainty and exhibits converging behavior, making it suitable for HSRIs.

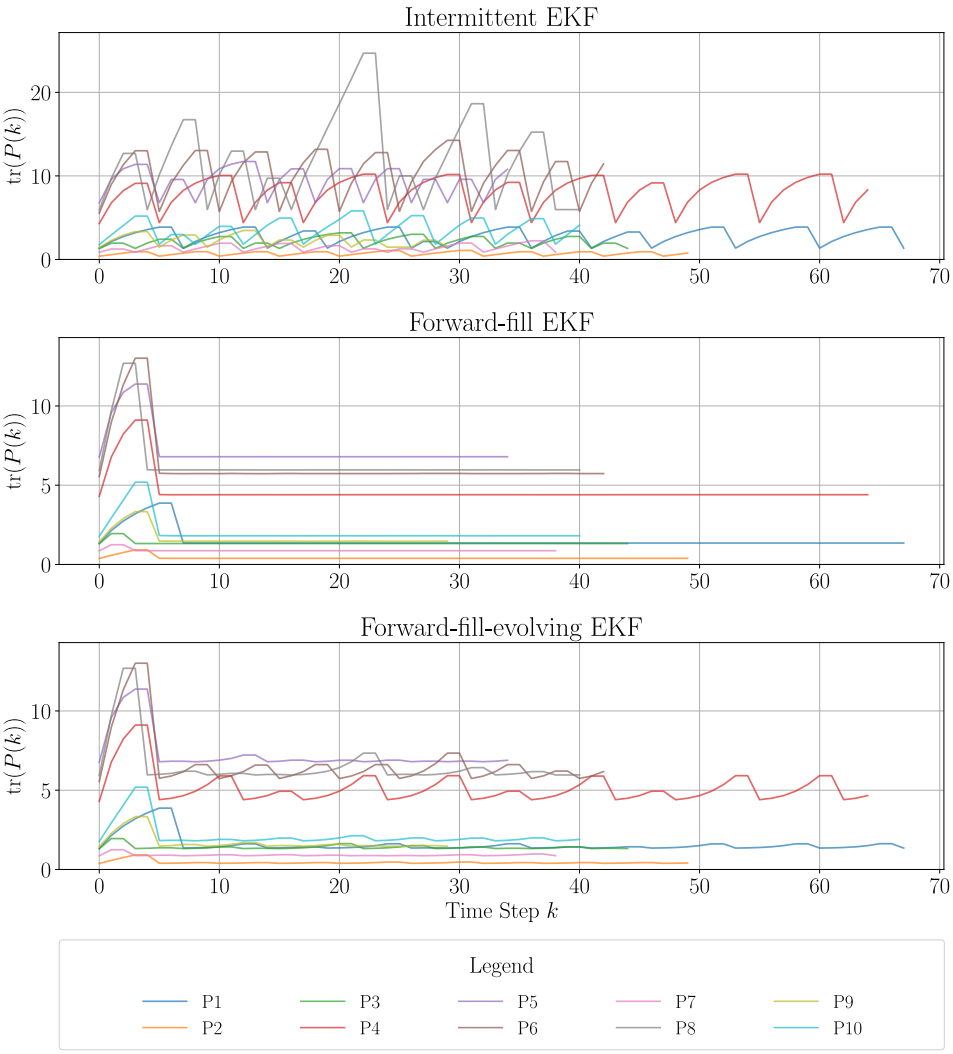


Figure 4.9: Trace of the state covariance matrix over time for 10 participants under three EKF variants: (a) intermittent EKF, (b) forward-fill EKF, and (c) forward-fill-evolving EKF. Each curve represents the trace of $P(k)$, for an individual participant — or the trace of $P^{-}(k)$ for intermittent EKF, when no correction is applied. A bounded or decreasing trace indicates stable estimation and practical convergence: (a) the intermittent EKF shows a sawtooth pattern and fails to converge without frequent updates; (b) the forward-fill EKF stabilizes rapidly after an initial transient and maintains bounded uncertainty; (c) the forward-fill-evolving EKF exhibits intermediate behavior, balancing adaptivity of noise weighting with convergence stability.

Finally, the forward-fill-evolving EKF exhibits better convergence behavior than intermittent EKF, but not as clear as forward-fill EKF. Despite applying both prediction and correction phases in each time step, forward-fill-evolving EKF still shows small fluctuations in $P(k)$ between measurements, likely due to the increase in

the measurement noise covariance matrix $R(k)$ when measurements are unavailable. Given that a bounded $R(k)$ is often required for convergence of KFs, prolonged intervals without measurements can inflate uncertainty and reduce stability of forward-fill-evolving EKF. This reflects a trade-off between adaptivity and convergence in the estimation process that should be balanced in real-world applications.

Overall, these results suggest that forward-fill EKF is the most stable and practically suitable estimator for use in HSRIs, while intermittent EKF does not offer reliable convergence, and forward-fill-evolving EKF presents a trade-off between adaptivity and convergence. However, across all variants, the behavior of state estimators may vary depending on measurement sparsity, user-specific dynamics, and task conditions. As such, real-time heuristics that monitor the trace of the covariance matrix can serve as an important safeguard: they can provide early indicators of estimator degradation and enable adaptive responses, such as re-initialization or increased measurement frequency. Incorporating such heuristics will be critical for deploying model-based estimators in real-world HSRIs.

In summary, the forward-fill EKF was the best-performing state estimation method on the current dataset. It significantly reduced the prediction error of the mental states for most participants whose prediction error exceeded 10% without deploying a state estimator, and it showed converging behavior. While the forward-fill-evolving EKF achieved comparable performance, we discussed several factors that may explain why it did not outperform the forward-fill EKF in this case study. Furthermore, its convergence was slightly less stable than that of the forward-fill EKF and may degrade if measurements are significantly more infrequent than in the present study. Since both state estimators are simple to implement and improve the estimation of mental states of users, we recommend their use in HSRIs when measurements are infrequent, ideally in combination with heuristics that monitor filter reliability and adapt filter behavior accordingly. Specifically, when personalization of α is not feasible or measurements are particularly infrequent, the forward-fill EKF is preferable. Otherwise, the forward-fill-evolving EKF may provide improved performance.

Finally, while the results support the effectiveness of the proposed state estimators, we acknowledge that the number of participants is limited. This reflects a common constraint in HSRI experiments that involve multi-session protocols and personalized modeling. Additionally, since the evaluation was conducted offline, the performance of the estimators during live interactions remains to be validated. These factors should be considered when interpreting the generalizability of the findings.

4.5. Conclusions and Topics for Future Research

Just as humans maintain an awareness of each other's mental states during social interactions, Social Robots (SRs) also require controllers that model the mental states of their users. In Chapter 3, Mathematical Model of Mind (MMM) — a dynamic mathematical model of human perception, cognition, and decision-making proposed in Chapter 2 — was integrated into a model-based controller to steer the behavior of an SR, enabling the robot to optimize the mental states of its users over the course of their interactions.

While the results of the proposed approach showed high promise for advancing social

robotics, the prediction errors in estimating the mental states of some participants highlighted a key area for improvement. Given that the effectiveness of a model-based controller relies on having an accurate estimation of the current states of the system (Morari & H. Lee, 1999), reducing the state estimation and prediction errors is crucial for enhancing the overall performance of the controller.

Leveraging the existing model MMM, which mathematically describes the dynamic evolution of the relevant mental states, we proposed incorporating a model-based state estimator into the control loop of SRs to improve the accuracy of estimating and predicting these mental states. To this end, we developed and evaluated three versions of Extended Kalman Filters (EKFs), tailored to estimate mental states of users in the context of human-robot social interactions.

We detailed how the parameters of these mental state estimators were derived and personalized to each participant, using data gathered in initial interactions with each participant. The proposed EKFs based on MMM were then evaluated on a separate portion of the same dataset (Morão Patrício & Jamshidnejad, 2025a) collected in experiments we previously conducted and reported in Chapter 3.

Since the evaluation of EKFs relied on previously collected data, our assessment was limited to quantifying improvements in the prediction error of mental states using the proposed EKFs. Although Chapter 3 demonstrated the potential of model-based control with MMM, it remains essential to investigate how improved state estimations actually affect the performance of the model-based controller and the overall quality of real-time interactions between humans and SRs.

As future work, we recommend implementing the proposed EKFs within the control loop during actual interaction sessions. This will allow assessing how the improved prediction of mental states impacts the behavior of SRs, as well as the interaction experience of their users. Additionally, a larger participant pool will be necessary to assess the generalizability of the proposed state estimators. Furthermore, we recommend integrating simple heuristics that can monitor convergence (e.g, establishing a threshold for the trace of the covariance matrix), and trigger adaptive responses such as re-initialization or increasing measurement frequency, if possible. Since EKFs assume Gaussian process and measurement noise, which may be a simplification given the bounded state space and the occasional larger errors present in self-report measurements, future work could also explore particle filters, which do not rely on this assumption and can represent bounded and asymmetric uncertainty. Further investigation into the personalization of parameter α of the forward-fill-evolving EKF is also recommended, as our results suggest that tailoring this parameter per participant may significantly improve the performance of this state estimator.

While this work presents the first application of a model-based state estimator for invisible cognitive states in Human-Social-Robot Interactions (HSRIs), future research should explore alternative estimation frameworks — such as unscented Kalman filters, particle filters, or adaptive Kalman filters — within this context. Although applying these techniques will require adaptation to accommodate sparse and subjective measurements, doing so will potentially enable more comprehensive benchmarking and provide deeper insights into the performance of state estimators in cognitively driven interactive systems.

5

Behavioral Control of Social Robots: State-of-the-Art and Future Directions

The preceding chapters demonstrated the potential of systems and control theory for generating the behavior of social robots by proposing a behavioral control framework built upon a dynamic model of user mental states (Chapter 2). Building on this foundation, the model was subsequently utilized to develop a cognition-aware, model-based controller (Chapter 3) and model-based state estimators (Chapter 4). To fully position this work and identify emerging opportunities for future research, this chapter systematically reviews and presents a forward-looking position on the control approaches currently employed in social robotics. This chapter provides a structured classification of approaches and investigates their current usage, inherent strengths, and limitations. Additionally, it evaluates control approach suitability across interaction lengths, functional roles, and applications. By offering a perspective on how existing and emerging control methodologies can be synthesized or strategically combined to advance social robotics, this chapter clarifies the current landscape of behavioral control in social robotics and outlines pathways toward more engaging and sustainable long-term interactions.

This chapter is based on the following publication:

Morão Patrício, M. L., & Jamshidnejad, A. (2026a). Behavioral control approaches in social robotics: A functional role synthesis and roadmap for sustained human-robot interaction. *ACM Transactions on Human-Robot Interaction* [*under review*]

Abstract

Social Robots (SRs) hold significant potential for supporting vulnerable populations, including older adults, individuals with cognitive impairment, and children with autism spectrum disorder. However, their real-world adoption remains constrained by difficulties in sustaining user engagement over extended periods. Since user engagement is strongly influenced by the adaptability and social appropriateness of the robot's behavior, advancing the behavioral control systems of SRs is crucial for sustaining long-term engagement. This survey analyzes state-of-the-art control frameworks used for behavior generation in SRs, organized into four main categories: symbolic and rule-based, learning-based and data-driven, modeling, and probabilistic. For each category, we analyze the underlying assumptions, strengths and limitations, and capacity to support socially appropriate human-social-robot interactions. Building on this analysis, we evaluate the suitability of different approaches across interaction duration, functional role within the behavioral control system, and application domain. Approaches commonly used as standalone behavioral controllers (particularly reinforcement learning and rule-based methods) prove insufficient for primary decision-making in long-term interactions, as they typically operate reactively without grounding behavior in internal reasoning or user models. For sustained, adaptive long-term interactions, cognitive and behavioral architectures should provide the structural foundation, with specialized methods fulfilling distinct roles: neural networks for perception, logic-based and probabilistic approaches for internal reasoning and user modeling, and model-based and bio-inspired methods for decision-making integrated with internal reasoning. Altogether, these findings highlight the need for principled integration of complementary approaches to enable meaningful, sustained Human-Social-Robot Interactions (HSRIs) in real-world settings.

5.1. Introduction

Vulnerable population groups that require specialized assistance, such as older adults, individuals with cognitive impairment, and children with autism spectrum disorder, are rapidly growing worldwide (Macalupu et al., 2025; Matarić & Scassellati, 2016). Meanwhile, healthcare and social care services face increasing pressure due to persistent and intensifying shortages in human resources, which limit their ability to provide adequate support to these populations (Bradwell et al., 2021; Macalupu et al., 2025). As a result, there is a rising demand for assistive solutions that are both affordable and widely accessible (Macalupu et al., 2025). Social Robots (SRs) have emerged as promising tools to enhance current care provision by delivering scalable, cost-effective, and personalized support to vulnerable groups, pottentially alleviating the pressure on overstretched care systems (Bradwell et al., 2021; Macalupu et al., 2025; Matarić & Scassellati, 2016).

Despite the potential of SRs to support vulnerable groups, their real-world adoption remains limited, primarily due to difficulties in sustaining user engagement over extended periods (Clabaugh et al., 2019; Maroto-Gómez, Alonso-Martín et al., 2023; Matarić & Scassellati, 2016). Most assistive applications inherently require prolonged interactions with users, since meaningful support cannot be achieved through short

or single-session interactions (Matheus et al., 2025; Tahir et al., 2020). However, SRs frequently struggle to maintain user engagement over the long term, particularly once initial novelty wears off (Clabaugh et al., 2019; Matheus et al., 2025). This decline in engagement can undermine the effectiveness of SRs, since users are unlikely to benefit from assistive robots they do not use consistently (Maroto-Gómez, Alonso-Martín et al., 2023; Matarić & Scassellati, 2016). Consequently, enhancing the ability of SRs to sustain long-term engagement with users represents a crucial challenge to enable their widespread deployment and real-world utility (Clabaugh et al., 2019; Matheus et al., 2025; Tahir et al., 2020).

User engagement in Human-Social-Robot Interactions (HSRIs) is closely tied to the behavior displayed by SRs. Research in social robotics highlights that sustaining engagement requires robots to display socially appropriate behaviors and establish trust with users over time (Matarić & Scassellati, 2016; Matheus et al., 2025). Evidence shows that robots that exhibit diverse, adaptive, and personalized behaviors achieve higher engagement and improved performance, whereas static or poorly matched behaviors diminish user interest (Macalupu et al., 2025; Matarić & Scassellati, 2016; Matheus et al., 2025). The ability to engage users, therefore, relies fundamentally on the social appropriateness, variability, and adaptability of the behavior of SRs.

The behavior expressed by a SR is strongly determined by its underlying behavioral control system (Maroto-Gómez, Alonso-Martín et al., 2023). Behavioral control systems directly define how a robot acts, moves, and communicates (Maroto-Gómez, Alonso-Martín et al., 2023; Matarić & Scassellati, 2016). Consequently, different control strategies enable distinct behavioral capabilities, thereby shaping responsiveness and interaction quality (Maroto-Gómez, Alonso-Martín et al., 2023; Matheus et al., 2025). Moreover, the adaptability and personalization capabilities of a SR, which shape its behavior and help sustain engagement, depend heavily on its behavioral control system (Matarić & Scassellati, 2016; Matheus et al., 2025).

Consequently, improving user engagement ultimately requires analyzing and advancing the control approaches that generate the behavior of SRs. Without control architectures designed to enable rich, adaptive, and socially appropriate behavior, user engagement cannot be reliably sustained (Maroto-Gómez, Alonso-Martín et al., 2023; Matheus et al., 2025).

In response to these challenges, this survey examines the landscape of control approaches used for behavior generation in SRs. Specifically, it identifies the dominant algorithmic frameworks currently employed, analyzes the assumptions that guide their design, and evaluates the ability of these frameworks to support adaptive, real-world HSRIs. Furthermore, this survey explores emerging and under-examined control strategies and discusses their potential contributions to enabling richer, more engaging social behavior for next-generation SRs.

5.1.1. Research Questions

To clarify the objectives of this survey and to structure the analysis of behavioral control approaches in SRs, we articulate the following research questions:

RQ1 What are the dominant algorithmic frameworks used for behavioral control in SRs?

RQ2 What assumptions underlie those control paradigms, and how do these assumptions shape design choices?

RQ3 How effectively do existing frameworks handle the dynamic, real-world nature of HSRIs, and where do they fall short?

RQ4 What unexplored or emerging control approaches of SRs exist, and how might they enable more adaptive, engaging, and socially appropriate SR behavior?

Addressing these questions provides insight into how different control frameworks may enable or constrain adaptive and socially appropriate behaviors of SRs, characteristics that are essential for sustaining user engagement in HSRIs. By examining the assumptions, limitations, and future potential of existing and emerging approaches, this survey aims to clarify which control paradigms may contribute to richer, more sustainable HSRIs.

The remainder of this chapter is structured as follows. Section 5.1.2 compares our survey with existing surveys in the field, highlighting key differences in scope and approach. Section 5.2 describes the methods used for literature identification, selection, and analysis. Section 5.3 presents the control approaches identified in the surveyed studies, grouped into symbolic and rule-based, learning-based and data-driven, modeling, and probabilistic categories. Section 5.4 discusses the findings and their implications, and Section 5.5 concludes the chapter.

5.1.2. Positioning Among Existing Surveys

This section situates our work within the context of prior surveys on the behavioral control of SRs. In particular, we outline the scope and limitations of those reviews and clarify how our survey differs from and complements them.

Although several surveys in the literature discuss behavioral control approaches in social robotics, they typically differ in scope — such as focusing on specific application domains or control paradigms — and address different research questions. For example, some surveys provide an in-depth review of specific classes of control methods (Akalin et al., 2021; Maroto-Gómez, Alonso-Martín et al., 2023). Akalin et al. (2021) focus exclusively on the application of Reinforcement Learning (RL) for behavioral control of SRs. Similarly, the survey by Nocentini et al. (2019) offers a detailed discussion of social robot behavior grounded in cognitive, emotional, and adaptive models — areas that are included in our review — but does not fully encompass the broader control landscape we address. Unlike our review, which adopts a more inclusive perspective on control paradigms across social robotics, Nocentini et al. (2019) concentrates on cognitively and emotionally driven behavioral control. Furthermore, while our inclusion criteria require implemented systems or concrete control frameworks, Nocentini et al. (2019) also incorporates conceptual or design-oriented studies that aim to inform the design of social robot behavior without necessarily proposing concrete control approaches or implementation details.

Other surveys narrow their scope to specific application areas of SRs. For instance, Cao, Esteban et al. (2017) provides an overview of state-of-the-art trends in behavioral control of SRs in healthcare, which makes its conclusions domain-specific.

Finally, Maroto-Gómez, Alonso-Martín et al. (2023) presented a historical evolution of the decision-making and behavioral control approaches of SRs over the past three decades. Although Maroto-Gómez, Alonso-Martín et al. (2023) offers valuable taxonomies and broad overviews of control strategies in SRs, its scope encompasses general control approaches — including physical control and navigation of robots — across various domains. In contrast, we focus on the behavioral control of SRs. Furthermore, while Maroto-Gómez, Alonso-Martín et al. (2023) emphasizes historical trends and application areas of SRs, e.g., manufacturing, healthcare, and education, its comparative analysis of algorithmic foundations remains limited. In contrast, our survey addresses this gap by systematically identifying and comparing dominant algorithmic frameworks, analyzing how their underlying assumptions shape system capabilities and limitations, and evaluating their ability to cope with the dynamic, uncertain nature of HSRIs. Beyond cataloguing the dominant methods, we explore emerging control approaches and their potential to enable richer, more adaptive, and socially appropriate behaviors for SRs.

5.2. Methods

This section describes the procedures used to identify relevant literature and analyze the selected papers. Section 5.2.1 outlines the process of identifying and selecting papers through search engines and predefined inclusion criteria. Section 5.2.2 details the procedure used to review these papers and establish the categories of control approaches.

5.2.1. Literature Identification and Selection

This section provides an overview of the article selection process, including the sources used (Section 5.2.1) and the criteria applied during screening (Section 5.2.1).

Search engines

This subsection describes the sources from which the articles in this survey were retrieved. Table C.1.1 in Appendix C.1 lists the sources, the papers collected from each source, and the search parameters used.

Scopus was used as the primary search engine due to its highly customizable interface, which allows for precise selection and combination of keywords. A single search using the keywords and filters presented in Table C.1.1 resulted in 90 papers. These papers were initially reviewed based on title and abstract, and, when necessary, based on the full content. Ultimately, 21 papers were deemed relevant for the scope of this survey according to the inclusion criteria described in Section 5.2.1. Figure 5.1 illustrates the selection process for the papers retrieved through the Scopus search engine.

Google Scholar was used only as a complementary source for three main reasons: First, it does not support complex keyword combinations. Second, its search algorithm is frequently updated, making keyword-based searches non-reproducible. Finally, using a similar number of keywords as in Scopus yields an unmanageably large number of results, making manual screening impractical. For both search engines, only papers published in the last 10 years were considered. In addition to these two search engines, a snowballing approach was also employed to include additional relevant studies that referenced or were referenced in the papers collected from the two search engines.

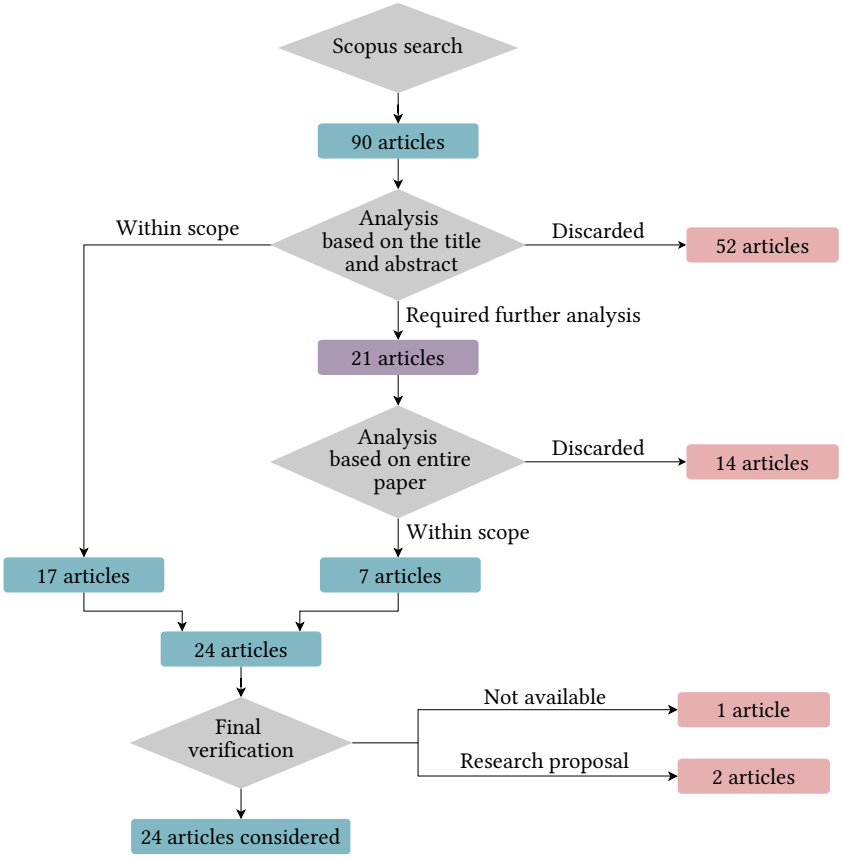


Figure 5.1: Flowchart of the paper selection process, illustrating inclusion and exclusion decisions across screening phases. The flowchart shows only the papers identified through the Scopus search engine.

Criteria for survey inclusion

Next, we outline the criteria used to include or exclude articles, detailing the methodological and thematic considerations that guided the selection process.

We included all articles that described the control or behavior generation of a robot designed to perform social interactions with users, regardless of whether the term

"social robot" was explicitly used. Specifically, papers were included if they proposed novel approaches for controlling the behavior of Social Robots (SRs) and provided detailed descriptions of those approaches.

Conversely, the following types of papers were excluded, even if they appeared within the defined keyword set or were collected through the methods described in Section 5.2.1 and summarized in Table C.1.1:

1. Articles that solely focused on technical capabilities (e.g., speech recognition, user detection, object manipulation), unless they also described how these capabilities were used to generate or control social behavior for the robot;
2. Papers that described only manually controlled systems (i.e., Wizard-of-Oz approaches);
3. Papers that only examined the effects of social interactions with SRs on humans, without presenting an algorithmic framework for controlling robot behavior;
4. Contributions from psychology that proposed purely conceptual frameworks aimed at informing the design of new SRs, without providing implementation details or control frameworks.

5.2.2. Review Procedure

To provide a structured overview of control approaches in social robotics and to analyze their purposes, advantages, shortcomings, and usage patterns, it was necessary to group the diverse approaches found in the literature. These groups were then arranged into a hierarchical structure. This organization allows for analyzing each subcategory in detail while identifying patterns, similarities, and trends across categories, providing a systematic understanding of the field rather than a collection of scattered, isolated approaches. A classification into categories and subcategories was, therefore, developed to enable systematic comparisons.

The review process followed three steps, which were used both to analyze the papers and to establish the control categories and subcategories:

1. **Paper processing:** Each paper was read and annotated, with particular attention to understanding and identifying the control approaches employed.
2. **Categorization:** The control approaches were first listed and grouped into broad categories, based on common principles. Frequently used approaches within each category were assigned to individual subcategories, while less common approaches were grouped together into broader subcategories to highlight general trends, rather than individual variations.
3. **Analysis within categories:** For each category, the relevant papers were examined to identify commonalities, differences, applications, and aspects relevant to the research questions.

This procedure yielded both a structured classification and a foundation for answering the research questions.

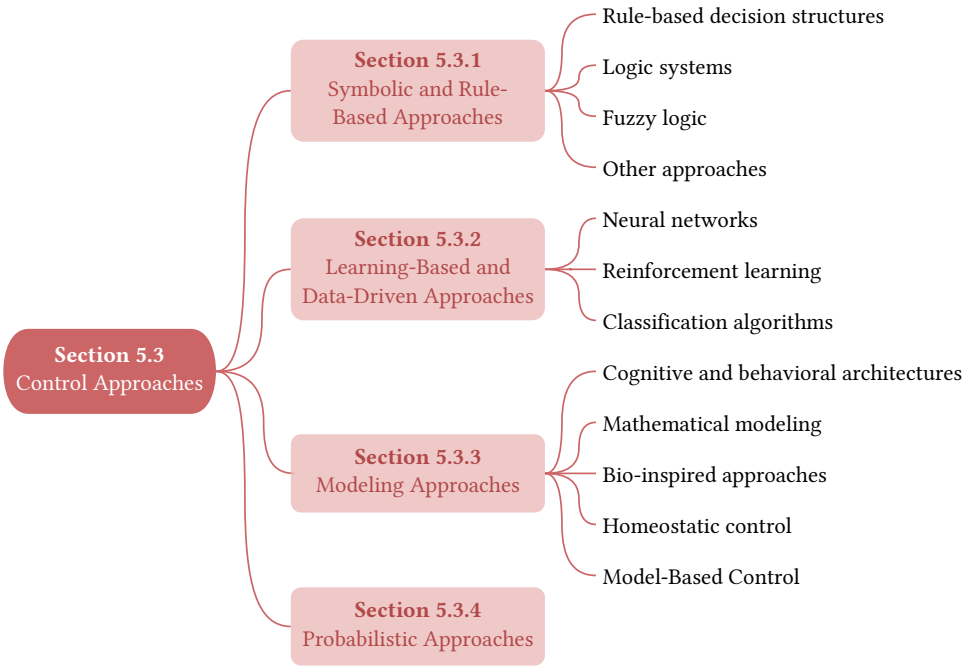


Figure 5.2: Categories and subcategories of control approaches for behavior generation in SRs. Section references indicate where each category is discussed in this survey.

Figure 5.2 visually represents the categories and subcategories of control approaches identified during the review process, and where the discussion of each category can be found within this chapter.

5.2.3. Relevant Concepts

This section provides a brief overview of key concepts that are commonly applied in social robotics and recur throughout this survey. These foundational concepts guide the design of multiple approaches reviewed in Section 5.3 for steering the behavior of SRs. They include cognitive models and theories rooted in psychology and neuroscience, as well as core mechanisms for social interaction.

Circumplex Model of Affect The circumplex model of affect (Russell, 1980) represents affective states on a two-dimensional axis: *valence*, which measures how positive or negative an emotion is; and *arousal*, which measures the intensity of the emotion. In social robotics, most studies that relied on the circumplex model of affect used it to simulate an affective state of the robot (Cao et al., 2019; Dziergwa et al., 2018; Maroto-Gómez, Malfaz, Castro-González & Salichs, 2023; Mazzei et al., 2014; Mitchinson & Prescott, 2016). Contrarily, Filippini et al. (2021) employed this model to identify the affective state of the user, which was then used to adapt the SR behavior. Furthermore, Dziergwa et al. (2018) and Cao et al. (2019) used an extension of this

model that also includes Dominance, which reflects the degree of control the agent perceives themselves to have.

Theory of Mind (ToM) Another cognitive theory often used in social robotics is ToM (Dissing & Bolander, 2020; Gray & Breazeal, 2014; J. J. Lee et al., 2019; Morão Patrício & Jamshidnejad, 2023a; Tani et al., 2020; Vossen et al., 2018). ToM refers to the ability to attribute mental states (e.g., beliefs, intentions, desires, knowledge) to both oneself and others, while understanding that others' mental states can differ from one's own. A key aspect of ToM is the recognition of *false beliefs* (Dissing & Bolander, 2020; Tani et al., 2020), i.e., understanding that another person can hold a belief that is incorrect, even when the observer knows the ground truth. This ability is commonly assessed through a *false belief task*, in which the assessed agent must predict another person's behavior based on that person's mistaken belief rather than the actual state of the world.

Belief-Desire-Intention (BDI) BDI is a cognitive framework used to steer the behavior of SRs (Dell'Anna & Jamshidnejad, 2024; Páez et al., 2020; Vossen et al., 2018). BDI is a model for rational agents where behavior is driven by beliefs (i.e., information about the world), desires (i.e., motivations or goals), and intentions (i.e., committed plans of action).

Gaze Control Systems Gaze control systems are mechanisms that select an attention target in multi-person interactions and orient the robot's head and eyes toward that target to support socially appropriate behavior. Typically, they follow a modular pipeline comprising social cue extraction (e.g., speech activity, gestures, distance, and orientation), target selection based on these cues, and gaze execution, which generates coordinated eye-head movements toward the chosen target (Aliasghari et al., 2020; Mishra & Skantze, 2022; Zaraki et al., 2014; Zhu et al., 2024). In the target selection process, some systems employ the concept of Elicited attention (EA), a scalar value that quantifies how strongly each individual attracts the robot's attention based on social cues and spatial factors (Aliasghari et al., 2020; Zaraki et al., 2014).

5.3. Control Approaches Used in the Literature

This section covers the main control approaches that are used in state-of-the-art research to steer the behavior of SRs. These approaches are organized into four main categories: symbolic and rule-based approaches, learning-based and data-driven approaches, modeling approaches, and probabilistic approaches (see Figure 5.2).

Within each category, approaches are further grouped into subcategories based on shared methodological principles. For each subcategory, we describe how the corresponding class of approaches has been used in SRs, based on the reviewed literature. Each subcategory concludes with a synthesis of insights from the reviewed works, highlighting key use cases, strengths, and limitations.

Table C.2.2 in Appendix C.2 summarizes the reviewed studies, highlighting their application domains, robotic platforms, and the control approaches employed across the four main categories.

5.3.1. Symbolic and Rule-Based Approaches

This section focuses on control approaches that use symbolic representations and explicitly defined rules to govern robot behavior. These approaches typically operate on discrete states or symbols and rely on predefined control logic. The control frameworks and approaches in this category include rule-based decision structures, logic systems, fuzzy logic, and heuristic and hard-coded rule-based systems.

Rule-based decision structures

The rule-based decision structures subcategory includes structured control mechanisms that use explicitly defined rules to map conditions or system states to actions or new states, thereby determining the behavior of SRs based on the current state of the environment, the user, the interaction, and/or the robot. These approaches, such as decision trees and Finite State Machines (FSMs), structure control inputs to the robot as a sequence of decisions that are governed by pre-established rules. FSMs are decision structures consisting of a set of possible states, the transitions between those states, and rules that govern those transitions. The system to be controlled, i.e., the behavior of the robot, can only be in one state at a time and changes state based on the inputs it receives or on varying conditions. FSMs are particularly useful for modeling sequential behavior and state-dependent decision-making (Skiena, 2008; Yannakakis & Togelius, 2018). Decision trees make decisions through a series of splits based on input values, narrowing down possibilities until reaching a final outcome or action. They provide interpretable rule-based reasoning by following a path from root to one of the leaves (Yannakakis & Togelius, 2018).

Mazzei et al. (2014) implemented the reasoning module of a custom cognitive architecture using C Language Integrated Production System (CLIPS). The module uses a structured rule-based approach with three hierarchical layers: analyzing rules that process input sensor data, planning rules that make behavioral decisions, and executing rules that send commands to actuators. Rules are organized in an agenda with priority-based execution, and the system uses template-based structured facts for knowledge representation. The usage of CLIPS allows researchers designing behavioral models (e.g., psychologists and neuroscientists) with little programming experience to describe robot behaviors through intuitive *if-then* rules.

Lazzeri et al. (2018) extended this work by formalizing two cognition layers: the primary layer processes sensory raw data and immediate robot state awareness, while the secondary layer performs deliberative reasoning over abstract social representations. Furthermore, Lazzeri et al. (2018) integrated meta-maps in the cognition of the SR. The meta-maps are an abstract way to represent the robot's perception of itself, the environment, and other humans, and they facilitate the modeling and reasoning about these aspects. Compared to Mazzei et al. (2014), the extensions introduced in Lazzeri et al. (2018) enable socially coherent behavior through explicit reasoning based on internal and social states.

Dziergwa et al. (2018) described the behavioral control architecture of the EMYS robot (Kedzierski et al., 2013) for long-term in-home cohabitation and assistance, as part of a study investigating how users' attachment styles influence human-robot interaction. The architecture includes three layers (hardware interfaces, reusable

competencies, and task coordination), and employs an FSM in the top layer for rule-based task coordination. The FSM represents each robot function (e.g., checking the weather, following the user) as a state, with state transitions primarily driven by user instructions. The FSM enables explicit task sequencing and reliable execution but offers limited support for adaptive or deliberative behavior. The behavioral control architecture structure is described in more detail in Section 5.3.3.

The control system of a SR that acts as a tutor to assist users with playing grid-based logic games, proposed by Gao, Barendregt et al. (2018), includes rule-based components. The controller consists of three main modules: a user model tracking indicators of task performance, a personalization module implemented with Reinforcement Learning (RL), and an action selection module implemented using a decision tree. The user model is implemented as a rule-based model that infers, based on predefined rules and performance indicators, whether the user is experiencing task-related difficulties during puzzle solving. If so, the personalization RL module is activated to select a category of supportive behavior, i.e., a style of social support. The decision tree of the action selection module receives the category chosen by the personalization module, checks whether the selected category is appropriate in the current situational context, and if appropriate, selects a specific supportive behavior from the available actions within that category. In this chapter, rule-based decision structures ensure predictable, interpretable behavior selection but limit flexibility and adaptation beyond predefined rules.

Rodríguez-Lera et al. (2018) proposed Hierarchy of needs, Motivational variables and Pool of FSM (HiMop), a framework that aims to produce natural and varied behaviors for SRs. In this architecture, each available task-level behavior (e.g., verbal interactions, play game) is represented by one FSM, characterized by states and transitions that depend on the motivations of the robot, external stimuli, and scheduling. The behaviors are categorized into three layers based on complexity (i.e., primitive, simple, or complex), where behaviors of a layer may rely on the behaviors from the same or lower layer. The collection of all available FSMs forms a "pool" from which behaviors are dynamically selected using a mathematical modeling approach, as explained in Section 5.3.3. Here, FSMs implement executable task behaviors, while the primary action selection is handled by a higher-level motivational architecture. This mitigates the rigidity typical of rule-based controllers.

Scassellati et al. (2018) designed a system for SRs supporting children with autism in developing social skills. In Scassellati et al. (2018), the SR accompanied the children in their homes for one month. Each day, the child listened to a story and played three games designed to foster social skills. A simple decision tree was used to adjust the difficulty of the games based on the score obtained by the child from the previous session. The thresholds that defined the different outcomes were fixed and not personalized for each participant. This rule-based approach enables transparent and reliable robot behavior but limits personalization and responsiveness to individual learning dynamics.

Clabaugh et al. (2019) developed Hierarchical Human-Robot Learning (hHRL), a framework for controlling the behavior of the robot SPRITE (Short et al., 2020) during long-term interactions with children with autism. These interactions aimed to develop

the children's mathematical skills via educational games. The hHRL framework includes a higher-level meta-controller that manages the main flow of the interaction by sequentially activating lower-level controllers (disclosure, promise, instruction, feedback, and inquiry) in a structured, rule-based manner. While the Instruction and Feedback controllers are implemented using RL, which is discussed in Section 5.3.2, the other three controllers do not involve learning or adaptation and are only described at a conceptual level. The rule-based decision structure used in Clabaugh et al. (2019) enables interpretable and predictable interaction sequencing for a specific application, while limited adaptation is delegated to selected lower-level controllers.

Kubota et al. (2020) proposed Just Express Specifications, Synthesize, and Interact (JESSIE), an approach that generates a logic-based FSM to control SRs based on high-level inputs provided by caregivers. The generated FSM is used at runtime to determine the next action of the SRs based on sensor inputs (e.g., touch detection) and activity completion states (e.g., whether a cognitive exercise has finished). The novelty of JESSIE lies in the fact that the FSM is constructed using Linear Temporal Logic (LTL), formally ensuring that the resulting behavior satisfies specified constraints, while enabling non-programmers to design complex robot behaviors through a tangible interface. The LTL mechanism is discussed in detail in Section 5.3.1.

M. H. Lee et al. (2023) designed a controller aimed at SRs that act as coaches in physical rehabilitation therapy. The general flow of the interaction is controlled by an FSM consisting of ten states, such as starting the interaction, demonstrating a movement, and giving feedback. The transition between states depends on user inputs and the output of a hybrid model that assesses the quality of the user's rehabilitation movements. This hybrid model merges the assessments produced by a Machine Learning (ML) algorithm and a rule-based model. The rule-based model consists of fifteen *if-then* parameterized rules that assess whether the rehabilitation movements of the user are correct, based on the position, speed, and trajectory of their joints. The rules are initially defined based on expert interviews, and the parameters are later tuned to each participant using a probabilistic approach (see Section 5.3.4). Similarly to Clabaugh et al. (2019), M. H. Lee et al. (2023) applies a rule-based decision structure to steer the interaction flow, enabling interpretable and predictable robot behavior, albeit at the cost of flexibility. Additionally, a second rule-based component enables expert-driven movement assessment, which indirectly informs robot behavior.

Bettosi et al. (2024) proposed a framework to systematically generate SR behavior offline from human expert demonstration, targeting assistive contexts where the robot acts as the care provider (e.g., education, coaching, rehabilitation). The approach begins by collecting data from human experts performing the assistive role and organizing it into high-level behaviors. Once the dataset is ready, clustering approaches identify relevant sequences of high-level behavior, which are then organized into a structured representation of behavior flow. The authors acknowledged that the specific approach to represent the behavioral flow depends on the application and suggested using decision structures (such as behavior trees, finite state machines, or graph-based models). Their case studies used a graph-based model based on transition matrices, i.e., a probabilistic representation of transitions between robot behaviors. Finally, the generated representation is assessed and refined through expert annotation. Three case

studies are presented where the approach is applied to squash coaching, post-stroke rehabilitation, and fracture recovery. Overall, this approach provides a structured and systematic methodology for translating expert demonstrations into reproducible SR behavior. Nevertheless, the resulting controller remains task-specific rather than domain-independent.

General appraisal of rule-based decision structures

- In general, rule-based decision structures are used in social robotics to represent predictable and deterministic behaviors under certain rules.
- Some of the papers discussed in this section use rule-based decision structures for specific or simple tasks as part of a more complex framework (Gao, Barendregt et al., 2018; Rodríguez-Lera et al., 2018; Scassellati et al., 2018). In particular, Gao, Barendregt et al. (2018) uses a behavior tree to select the final behavior of the SR from a set of options, whereas Rodríguez-Lera et al. (2018) implements several FSMs, each to implement a modular behavior that can be executed separately.
- More recent papers use rule-based decision structures to control the interaction flow by sequentially organizing low-level behaviors of SRs (Bettosi et al., 2024; Clabaugh et al., 2019; Dziergwa et al., 2018; Kubota et al., 2020; M. H. Lee et al., 2023). Apart from Kubota et al. (2020) and Bettosi et al. (2024), the controllers were developed for specific applications, since decision structures typically need to be tailored to contexts. In contrast, Kubota et al. (2020) and Bettosi et al. (2024) presented general-purpose frameworks that allow caregivers to generate decision structures tailored to specific applications. In the case of Kubota et al. (2020), FSMs are automatically generated from high-level specifications defined by the user. In the case of Bettosi et al. (2024), however, decision structures are derived using a data-driven approach from observed interactions.
- A few approaches implement complex behavioral controllers using rule-based decision structures (Lazzeri et al., 2018; Mazzei et al., 2014). However, in these cases, extensive rules have to be manually encoded.
- All in all, while rule-based decision structures allow for a clear and easily implementable representation of deterministic behaviors, in general, they either offer limited flexibility and are tailored to specific applications or they require extensive manual rule encoding. This makes them poorly suited for highly dynamic or unpredictable scenarios.

Logic systems

This subcategory encompasses control approaches based on formal logics, such as propositional (Smith, 2012), modal (Blackburn et al., 2001), deontic (Parent & van Der Torre, 2013), and temporal logic (Van Benthem, 1983). These approaches rely on structured symbolic representations and use defined inference rules and logical operators to perform reasoning and draw conclusions.

The deliberative subsystem of the HiMop architecture (Rodríguez-Lera et al., 2018) plans robot behaviors to achieve long-term goals by using Planning Domain Definition Language (PDDL), a symbolic planning framework. Accordingly, this subsystem represents its inputs in the form of *objects, predicates, initial state, goal specification,* and *actions*. For example, in a home assistance scenario where the robot is playing with a user, but needs to greet a visitor, the *objects* include entities like the robot, the user, the living room, the visitor, and the main door; the *predicates* describe the properties or relationships of objects, such as the robot being in a location or whether the game is finished; the *initial state* specifies starting conditions like the robot being in the living room and the game being incomplete; the *goal specification* defines desired outcomes such as the game being complete or the visitor being greeted; and *actions* are robot operations like navigating to door or playing the game. The formal structure of PDDL enables the subsystem to systematically organize the available behaviors from the behavior database of the robot, evaluate different action sequences, and generate long-term structured plans.

Vossen et al. (2018) developed an architecture to steer the behavior of Leolani, their Pepper-based robot acting as a SR. The system aims to learn information about the users (such as personal details like name, occupation, and interests, but also knowledge and emotions) and the surrounding environment, while resolving knowledge conflicts. One of the main components of this framework is the knowledge representation layer, implemented using a logic-based framework. In this layer, knowledge is stored in Resource Description Framework (RDF) triples, i.e., in the form of subject-predicate-object tuples. For example, to represent that Bram is laughing, the system stores *[laugh, has actor, Bram]*, where "laugh" is the event (subject), "has actor" is the relationship (predicate), and "Bram" is the person performing the action (object). This formalism allows the robot to connect events, entities, and their relationships in a structured way. In addition to storing the information in RDF, the representation layer also records the source of the information and its perspective (including the confidence level and emotional connotation). Furthermore, the entire framework is structured to solve conflicting information. Therefore, if the robot acquires contradictory RDF tuples, it resolves them by checking their sources and perspectives, and by questioning the sources or the subjects involved. This not only helps maintain an accurate environment representation, but also promotes active communication initiated by the robot.

The JESSIE framework (Kubota et al., 2020) generates formal, logic-based systems that steer the behavior of SRs based on high-level behavior defined by caregivers. The available behaviors and sensors of the robot are represented as physical cards, which caregivers arrange in sequences to specify how the robot should act and under what conditions based on sensor input. These rules, defined through the card arrangements, are then formalized as LTL formulae. LTL is a formal language used to describe temporal properties and sequences of events over time — for example, to specify that something must always hold, eventually happen, or continue until another event occurs. Furthermore, system requirements are also formulated using LTL. Finally, an LTL engine processes all formulae to synthesize an FSM that can autonomously steer the behavior of the SR during the interaction. The use of LTL ensures that

the generated FSM satisfies all caregiver specifications, regardless of the caregiver's programming expertise.

Dissing and Bolander (2020) used Dynamic Epistemic Logic (DEL) — a logic framework that models how knowledge changes when information is shared — to enable SRs to pass the false belief task (see Section 5.2.3 for details). DEL represents knowledge per agent as a set of possible worlds, where each world corresponds to a different configuration of facts that may be true from the perspective of the agent. The more an agent knows about the real world, the fewer worlds it considers as possible. When an epistemic event occurs (e.g., a SR observes an action by a human), DEL models this event as an epistemic action that specifies which agents observe which pieces of information. This action updates the set of possible worlds, i.e., agents who observed the event refine their beliefs and can distinguish between the previous and updated states of the world, while agents who did not observe the event retain their original set of possible worlds and, thus, maintain their prior beliefs. This mechanism allows the agent to reason about what other agents know or believe (including false beliefs) based on who has access to which information.

In their implementation of false belief, Dissing and Bolander (2020) categorize physical entities into objects (cubes or boxes) and agents. Four event types are defined for these entities: appear, disappear, pick up, and put. While events *appear* and *disappear* can be applied to any entity, the event *pick up* involves both a person and a cube, and the event *put* consists of a person placing a cube into a box. Additionally, factual information is represented as predicates (e.g., facts such as "the blue block is in box 1") or observability atoms (e.g. "Sally can see Anne"). Then, whenever an event occurs, the corresponding action (i.e., content and who observes it) is used to update the possible worlds for each agent. Finally, a query is used to extract the beliefs of the agents based on their possible worlds, and to pass the false belief task. DEL captures not only first-order beliefs (i.e., what an agent believes to be true about the world), but also beliefs of an agent about the beliefs of other agents, and how these both evolve in response to information flow.

The cognitive architecture proposed by Benedictis et al. (2023) is centered on a user model implemented as an ontology, i.e., a formal, structured knowledge base that organizes information using logic systems. The model contains standard information about user abilities and disabilities based on the International Classification of Functioning, Disabilities and Health framework (ICF). The ICF is a framework developed by the World Health Organization to systematically describe health and disability using multiple indicators. The model adopts a subset of these indicators, including attention, memory, perception, and cognition indicators, as long-term measures of functioning. The impairment level of each ICF indicator (called functioning quality) is rated on a five-level scale from no impairment to full impairment. Besides these long-term ICF indicators, the model also includes short-term elements, such as affective state and user intents. All information is stored as an ontology, enabling the SR to adjust its behavior based on all model elements in a structured, practical, and scenario-independent way. For example, for users with hearing impairments, the sound level is adjusted accordingly.

General appraisal of logic systems

- Logic-based approaches systematically represent domain knowledge, user models, and interaction rules using formal logic systems, e.g., propositional, modal, deontic, temporal, and dynamic epistemic logic. These structured representations enable deliberate, interpretable, and explainable reasoning over acquired knowledge.
- In social robotics, logic systems are deployed to collect, structure, and process environmental and user-related information. This often results in the construction of belief databases that reflect the understanding of the robot of its surroundings, as well as the perspectives of different agents. Such systems can support the revision of internal knowledge as new information becomes available, and promote active communication initiated by the robot (Dissing & Bolander, 2020; Vossen et al., 2018).
- Furthermore, these systems support systematic reasoning, long-term planning, and behavioral adaptation in response to long-term goals, context changes, and user characteristics (Benedictis et al., 2023; Kubota et al., 2020; Rodríguez-Lera et al., 2018). For example, Planning Domain Definition Language (PDDL) and Linear Temporal Logic (LTL) have been used to synthesize behavior plans that satisfy formal specifications (Kubota et al., 2020; Rodríguez-Lera et al., 2018). Additionally, logic-based user models enable the robot to perform formal inference over user traits and states, guiding its behavior selection (Benedictis et al., 2023).
- Key advantages of logic-based methods include high interpretability, traceability of decisions, and formal correctness guarantees. However, they require substantial upfront effort to define domain knowledge and rules, are less robust to incomplete or noisy data, and exhibit limited adaptability to unforeseen situations.
- In summary, logic-based control frameworks are well-suited for applications with predictable environments, clearly defined tasks, and structured interactions, where domain knowledge and interaction rules can be explicitly defined. Nevertheless, their reliance on predefined rules and explicit representations constrains their generalizability to novel or unpredictable scenarios.

Fuzzy logic

Inspired by human reasoning, fuzzy-logic-based systems are designed to represent and reason over imprecise, ambiguous, or incomplete information by using continuous degrees of truth. Unlike classical logic, which operates on binary true/false values, fuzzy logic allows inputs to partially satisfy conditions of being true or false. This enables nuanced inference based on uncertain inputs. In fuzzy-logic-based systems, knowledge is encoded through fuzzy *if-then* rules that map relationships between inputs and outputs. These rules are evaluated using fuzzy inference mechanisms, which apply the rules even when inputs only partially match the antecedent (i.e.,

conditions) of a rule. The degree of match is quantified by *membership functions*, which assign values between 0 and 1 to each variable or input to indicate how well it fits a given fuzzy category. Rules are typically expressed using linguistic variables (e.g., "low," "medium," or "high"), which are associated with corresponding membership functions that define graded category boundaries (Zadeh, 1988).

Páez et al. (2020) introduced the Fuzzy Cognitive Binary Tree (FCBT), a model to assess the user's strategy in solving puzzle-type tasks, based on inputs such as whether learners correct their own mistakes or make relevant moves. The proposed FCBT consists of a Fuzzy Cognitive Map (FCM) (i.e., a graph-based model that represents uncertain causal relationships between concepts using fuzzy logic) arranged in a binary tree structure. Unlike conventional binary trees, which follow deterministic root-to-leaf paths, the FCBT aggregates weighted inputs from the leaves upward to the root to produce an output, i.e., the user strategy. Subsequently, this output informs the interventions of the SR, such as providing feedback, hints, or motivational cues. Therefore, the FCBT allows the robot to adjust its behavior to each user's unique learning strategy, which is essential in educational contexts. Nevertheless, it is tailored to the specific case of puzzle solving, and its performance depends on the accuracy of the defined causal relationships.

The work proposed in Chapter 2 used an extended version of FCMs to model the inter-dynamics between mental states, such as beliefs, goals, emotions, and biases. This approach captures the non-linear and recursive interdependencies among mental variables in an intuitive way, making it well-suited for modeling human reasoning in social interactions. Despite these strengths, FCM parameters require manual specification or data-driven learning, and computational complexity scales with the number of mental states and their connections. These considerations shape how FCMs can be applied in different social robotics contexts. The approach is further explained in Section 5.3.3.

Dell'Anna and Jamshidnejad (2024) used fuzzy rules to represent social norms that steer the behavior of SRs through a multi-agent architecture called SOcial Norm Aware Robots (SONAR). The fuzzy rules enable both inferring social cues based on data collected from the environment and user (e.g., inferring relationships from proxemics) and shaping how the robot should behave within a social context (e.g., selecting suitable voice tone or pitch). Fuzzy logic is particularly suitable for modeling social norms because they are inherently vague and are usually expressed through ambiguous linguistic terms (e.g., "suitable distance"). Since the interpretation of social norms depends on cultural and social contexts, using fuzzy logic with adaptable membership functions yields a practical way to adjust the rules to different contexts by tuning these functions based on data collected during interactions. The adaptation algorithm of SONAR collects interaction data until a sufficient amount of data is available. It then adjusts the trapezoidal membership functions by resizing their domain to fit observed data, shifting their center to align with the mean value of the collected data, and resizing their width based on the standard deviation of the collected data. This approach enables automatic adaptation from unlabeled interaction data, allowing robots to learn context-specific interpretations of social norms, such as "medium distance". However, the need to collect sufficient data before adaptation may slow responsiveness

to new contexts, reflecting a trade-off between reliable learning and rapid adjustment. The SONAR architecture is described in detail in Section 5.3.3.

Munster and Jamshidnejad (2025) proposed a behavioral controller for SRs that combines fuzzy logic with RL to enable personalized and adaptive robot behavior. Fuzzy inference systems are used to translate user personality traits (e.g., extroversion, agreeableness, neuroticism) into behavioral parameters (e.g., speech modulation, gesture intensity, and interpersonal distance), yielding a candidate robot action that warm-starts the RL algorithm. In addition, fuzzy rules estimate the user's state-of-mind from emotional cues, which shape the reward function for the RL agent. This integration allows socially meaningful, linguistically interpretable reasoning while supporting adaptation over time. However, the approach depends on manually designed rule bases and membership functions, which may require manual tuning to handle diverse user preferences or social contexts.

General appraisal of fuzzy logic

- The approaches covered in this section apply fuzzy logic to represent and reason over imprecise or uncertain concepts, which are common in social contexts (Dell'Anna and Jamshidnejad (2024); Munster and Jamshidnejad (2025); Páez et al. (2020) and Chapter 2)
- Fuzzy logic has been used to personalize the behavior of SRs to individual users (Munster and Jamshidnejad (2025); Páez et al. (2020) and Chapter 2) and to adapt robot behavior to varying social contexts (Dell'Anna & Jamshidnejad, 2024).
- By modeling ambiguity and partial truth, fuzzy logic aligns well with human reasoning in social interactions. Fuzzy logic also enables the representation of non-linear relationships between variables and supports user personalization and context adaptation.
- Despite its strengths, fuzzy logic shares limitations with other symbolic and rule-based systems. The design process typically requires expert knowledge, which limits generalizability to new contexts. While data-driven methods can assist in tuning membership functions, expert input remains essential for defining the rule base and overall system structure.
- Additionally, computational complexity may grow exponentially with the number of rules and variables, posing scalability challenges in high-dimensional applications.
- Another limitation of many fuzzy-logic-based control systems is the lack of systematic methods to ensure stability, guarantee performance, and enforce dynamic safety constraints. Unlike model-based control frameworks, fuzzy-logic-based systems do not inherently provide formal guarantees. This makes it challenging to verify safe and reliable behavior, especially in dynamic or safety-critical environments.

- Overall, fuzzy logic provides a useful framework for modeling social ambiguity and supporting adaptive behavior in SRs. However, its effectiveness often depends on expert-driven design and may be constrained in complex, dynamic scenarios.

Heuristic and hard-coded rule-based systems

This subcategory includes control approaches in which robot behavior is governed by manually designed rules, heuristics, or threshold-based decision logic. Robot actions are selected through explicit condition–action mappings, such as *if–then* rules, score-based prioritization schemes, or fixed parameter thresholds.

Kidd and Breazeal (2008) integrated a SR into a weight loss program to encourage users to engage with the program more frequently and for longer. During the long-term interaction, the SR aims at establishing and maintaining a supportive coaching relationship with the user. The speech of the robot is generated based on a set of pre-defined rules that adapt the dialogue according to the time of the day, the stage and evolution of the relationship between the SR and the user, and predefined user inputs provided via touchscreen selections related to diet and exercise. Kidd and Breazeal (2008) illustrate how rule sets can generate and tailor dialogue in task-specific applications, offering a transparent and straightforward design, but with limited adaptability, scalability, and generalization to broader social interactions.

Zaraki et al. (2014) developed a gaze control system for the FACE robot (Pioggia et al., 2004) that, while predominantly implemented through mathematical models of human attention and eye gaze, also incorporates additional hard-coded components and heuristics. In modeling attention, the proxemics space and field of view are discretized into zones, each assigned a different weight in the equations modeling attention. Additionally, specific constants are hard-coded, including three discrete speed levels for both eye and head movement, and latency values for eye and head responses. Zaraki et al. (2014) illustrate how rule-based heuristics can be used in a supplementary role to handle specific cases rather than implementing the main control logic. The main gaze control logic, grounded in mathematical modeling, is discussed in detail in Section 5.3.3.

Sequeira et al. (2016) developed a framework to generate an autonomous SR controller by replicating the control strategies of a Wizard-of-Oz (WoZ) controlled SR, i.e., where a human operator remotely controls the robot during interaction. One component of the generated controller is a rule-based module. The core novelty of this work lies in restricting the perception and action space of the wizard to match those of the robot, enabling the control policy chosen by the wizard to be replicable for the robot. The process of developing the controller begins with a mock-up study where a human takes the role of the robot. The key task-relevant perceptual features (e.g., face recognition) and behavioral tasks (e.g., pointing at users or playing a game) are selected during system design and implemented as standalone components. Subsequently, WoZ interactions are performed, where the wizard only has access to the selected perceptual features and behavioral tasks. The wizard's policy is extracted by encoding observed practices as explicit rules and training ML classifiers on the WoZ interaction data, resulting in a hybrid controller that combines a rule-based and an ML module to map

perception to behavior. This hybrid approach balances the transparency and reliability of rule-based logic with the capability of learning-based methods to identify complex behavioral patterns.

Schicchi and Pilato (2018) developed a control approach for a Pepper robot (A. K. Pandey & Gelin, 2018) that plays portmanteau games with children to assist them with language learning. The system supports three modes of play (i.e., generating portmanteau words, guessing source words, or inferring words from definitions) and two roles for the SR (generator or answerer). The controller includes three modules: the conversation engine (for dialogue management), the creation engine (for portmanteau word creation), and the definition engine (for definition creation and guessing the source words of a portmanteau from its definition). After converting the child's spoken input to a text string using an external library, the conversation engine uses RiveScript — a rule-based scripting language — to match the obtained text against pre-defined question-answer rule pairs and to generate appropriate game-related replies. Depending on the combination of game and role chosen by the user, the conversation engine activates a different set of question-answer modules. The definition engine also relies on rule-based approaches; however, their detailed description is out of scope for this survey, as they concern task-specific linguistic processing rather than social interaction mechanisms. Overall, this work exemplifies the use of hard-coded rule sets for dialogue and language processing, yielding transparent but inflexible behavior with limited applicability beyond task-specific scenarios.

Schrum et al. (2019) also developed a hard-coded approach to regulate the behavior of a SR that encourages users to dance as a form of exercise. The robot proposes dance movements of varying intensity that aim to keep the heart rate of the user between 20% to 30% of their resting heart rate. To achieve this, after a given time interval (30 to 50 seconds), if the heart rate was below or above the target range, the SR increases or decreases the intensity, respectively. If the heart rate is within the defined range, the intensity remains unchanged. Although this controller adapts to a user state, its non-personalized, limited rule-based adjustments make the controller task-specific and unsuitable for broader or long-term interaction.

The DEL-based approach presented in Dissing and Bolander (2020) for enabling robots to recognize false beliefs, explained earlier in Section 5.3.1, also relies on general rule-based components. First, complex events such as "agent picking up cube" or "placing cube in box" are detected based on manually defined sets of *if-then* rules. For example, if the hand of a user occupies the same position in space as a cube, the "agent picking up cube" logic event is triggered. Second, the position of objects (cubes and boxes) is detected through markers. In this case, hard-coded rules are employed to implement perceptual and event detection mechanisms that complement the primary logic-based reasoning framework.

Rossi et al. (2020) developed a simple, hard-coded approach to adapt the behavior of a SR that aims at distracting children during vaccination procedures. The behavioral steering system activates pre-programmed behaviors based on two variables: age and anxiety level of the child. The latter is assessed by a nurse prior to the procedure and is discretized into low or high. Depending on whether the age is above or below a threshold, the robot selects one of two age-tailored conversation scripts focusing

on either child-oriented or school-related topics. Similarly, depending on whether the anxiety level is low or high, it displayed one of two pre-programmed emotional behaviors (positive vs. negative valence). As in Schrum et al. (2019), this controller relies on a small, hard-coded rule set for decision-making, yielding rigid behavior; however, because vaccination involves brief, one-off interactions, such rigidity is acceptable.

Céspedes et al. (2020) presented a semi-autonomous system that integrates the Nao robot into Lokomat, a robotic device widely used for gait rehabilitation, to reduce caregiver workload. During the interaction, the SR aims at motivating the user, monitoring their performance, and providing feedback, while the therapist configures system parameters, supervises the session, and intervenes when necessary. The robot's behavior is governed by a set of deterministic rules based on heart rate, spinal posture angles, and self-reported perceived exertion of the user. When the heart rate exceeds an age-based threshold, the robot alerts both therapist and patient. When spinal inclination angles exceed $15°$ from neutral, the robot verbally corrects posture and demonstrates proper alignment. During proper posture, the robot provides motivational encouragement. Additionally, the robot periodically queries perceived exertion. Because rule activation relies on static, therapist-defined thresholds, the system does not adapt or personalize its behavior across users. Céspedes et al. (2021) extended the work presented in Céspedes et al. (2020) by introducing a graphical interface for therapists and by evaluating the same rule-based SAR system in a multi-month clinical deployment involving 15 therapy sessions. This study assessed the long-term use of the system in clinical practice, while the underlying control logic — a small set of static *if-then* rules based on therapist-defined thresholds — remained unchanged.

The gaze control system proposed by Aliasghari et al. (2020) uses rule-based methods to infer social cues, such as detecting the current speaker by matching the sound source with the position of one user, detecting pointing gestures by checking whether a user's hand direction targets another person, and determining whether or not users are engaged by analyzing if they are looking at the SR. To select the gaze target, these cues are combined into a scalar Elicited attention (EA) score per user, representing how strongly that user attracts the robot's attention. The user with the highest EA is selected as the attention target; if two users have the same EA, the closest person to the robot is chosen. Additionally, if all EAs are below a threshold, the SR looks away. Additional rule-based behaviors govern gaze execution: head movements during gaze shifts are triggered only if the angular difference between the robot's gaze and the selected target exceeds a threshold, and the SR is programmed to briefly gaze at users entering or exiting the room. Overall, this approach relies on hand-crafted rules and is largely threshold-driven, which, even for gaze control, limits adaptability and the range of achievable natural behaviors.

Filippini et al. (2021) developed an approach to adjust the behavior of a SR to the affective engagement of children during entertaining or educational interactions with the robot. While the emotion recognition is performed by an ML module, the paper incorporates basic rule-based decision logic for robot behavior responses. When the emotion recognition module detects the affective state to be positive or negative, the

robot seeks confirmation by asking the user if they are amused or bored, respectively. Then, the SR continues performing the current activity if the user is amused, or switches activities otherwise.

Mishra and Skantze (2022) proposed a gaze control system for the Furhat robot (Al Moubayed et al., 2013) that plans gaze behavior over a future time horizon instead of solely reacting to current social cues, allowing gaze duration and eye-head coordination to be planned in advance. The system has two components: a gaze planner that uses rule-based methods to assign priority scores to potential gaze targets, and a gaze controller that uses a mathematical approach to direct eye and head movements based on these scores (discussed in Section 5.3.3). The gaze planner assigns priority scores to potential gaze targets, including users, objects, and the environment. The environment serves as a default gaze-away target with a fixed baseline score, while scores for people and objects are defined by static rules based on social events, such as the robot or a user mentioning a target, the robot listening to a user, or the robot pretending to think. Priorities for each gaze target are computed over a ten-step window (two seconds). Incorporating these event-triggered scores facilitates social behaviors of the robot, such as turn-taking, joint attention, intimacy regulation, and eye-head coordination. However, the authors note that the numerical values of the priority scores were assigned arbitrarily and recommend optimizing them in future work.

Within the cognitive architecture for SRs presented in Benedictis et al. (2023), a reactive layer deploys *if-then* rules to adapt the communication style of the robot (e.g., adjusting word choice and verbal tone) to a set of variables that define the current context. These variables are either inferred from the user utterances (e.g., their intent and sentiment) using a speech processing module or originate from other components of the architecture — the user model (e.g., extraversion) or the reasoning layer (e.g., current exercise). A set of *if-then* rules is then applied to these variables to determine the SR's communication style and generate an appropriate spoken response.

General appraisal of heuristic and hard-coded rule-based systems

- Heuristic and hard-coded rule-based systems are widely employed in social robotics due to their simplicity and ease of implementation, and have been deployed for diverse purposes.
- Rule-based engines are used to generate speech text (Kidd & Breazeal, 2008; Schicchi & Pilato, 2018) and to tailor verbal responses of the robot based on the Human-Social-Robot Interaction (HSRI) context or user state (Benedictis et al., 2023; Rossi et al., 2020).
- Some of the approaches reviewed in this section rely on manually defined rules for detecting social behaviors of users (Aliasghari et al., 2020; Zaraki et al., 2014) and for triggering robot behaviors (Sequeira et al., 2016).
- Several works designed the main logic of the behavioral controller as direct mappings from a small set of input variables (e.g., age, anxiety level, physiological data, or emotional state of the user) to fixed robot actions (e.g.,

display pre-programmed behavior or increase exercise intensity), often using static thresholds (Céspedes et al., 2020; Céspedes et al., 2021; Filippini et al., 2021; Rossi et al., 2020; Schrum et al., 2019).

- Additionally, rule sets are employed to implement supplementary features that complement the primary control method (Aliasghari et al., 2020; Dissing & Bolander, 2020; Schicchi & Pilato, 2018), or to manage specific modules within distributed architectures (Benedictis et al., 2023; Mishra & Skantze, 2022; Sequeira et al., 2016).
- Most approaches in these categories rely on rule-based logic, either as a complement to the main controller or as the primary controller in very simple applications.
- While heuristic and hard-coded rule-based systems are prevalent, because they are straightforward and computationally inexpensive, they exhibit significant limitations. In particular, they are inflexible and difficult to scale, and perform poorly in complex or unexpected scenarios. Their lack of adaptability limits their effectiveness in dynamic social interactions.
- Although supplementing main controllers with rule sets is common and often necessary, relying solely on rule sets for primary control of the SR behavior typically results in simplistic, non-adaptive behaviors that fail to meet the demands of rich, long-term HSRIs.

5.3.2. Learning-Based and Data-Driven Approaches

In this section, we discuss statistical and machine learning techniques used in social robotics to extract patterns or control strategies from data for Social Robot (SR) behavior generation.

Neural networks

Neural Networks (NNs) are Machine Learning (ML) architectures inspired by biological neurons that are capable of approximating non-linear relationships which map input data to corresponding output data. NNs are composed of elements, called neurons, which are organized into multiple layers. Each neuron applies a weighted sum of its inputs (plus a bias value) followed by an activation function to produce an output. The associated weights are tuned during the learning process so that the outputs of the networks best fit the training data (Abdi et al., 1999).

Shiarlis et al. (2017) proposed Deep Behavioural Social Cloning (DBSoC), an NN-based architecture designed to automate the low-level movement of a SR in telepresence scenarios by learning socially acceptable behaviors from human demonstrations via supervised learning. DBSoC addresses two major identified challenges. First, the variable number of people interacting with the SR poses a challenge for learning-based controllers that require fixed-size inputs. This was addressed by converting the robot's observations of nearby people into two-dimensional images where the positions

and velocities of people are graphically encoded and processed by a Convolutional Neural Network (CNN) for spatial feature extraction. Second, the use of single-frame observations limits the ability to capture motion dynamics and makes detection more sensitive to noise. This was addressed by employing a type of recurrent NNs, Long short-term memory (LSTM), which is capable of detecting temporal patterns in sequences of data points. The system was trained on approximately two hours of human pilot demonstrations across two social tasks (maintaining appropriate positioning during group interactions and following people as they move). While this approach produces socially appropriate motion patterns learned from demonstrations, the resulting behavior remains limited to the type of scenarios represented in the training data, reflecting the task-specific nature of applying NNs for end-to-end behavior learning.

NNs are also commonly used in control architectures for SRs to support perceptual processing and to transform sensory inputs into structured representations. For example, the Leolani robot architecture (Vossen et al., 2018) relies on NNs for perceptual processing (e.g., object recognition), whose outputs are converted into symbolic representations for subsequent reasoning. In the Dynamic Epistemic Logic (DEL)-based system that allows SRs to recognize false beliefs (Dissing & Bolander, 2020) (explained earlier in Section 5.3.1), the robot's perception is assisted by NNs. These NNs are used for user recognition, speech recognition, and to convert high-dimensional camera sensor data into structured low-level world representations (e.g., positions of faces and body poses) suitable for processing by the DEL-based reasoning subsystem. Furthermore, Benedictis et al. (2023) used off-the-shelf IBM Watson modules, which rely on different types of NNs, to convert speech to text and to identify user intentions and sentiments from user utterances. These examples demonstrate the frequent role of NNs as perceptual inference systems that provide high-level inputs to other control components rather than driving the SR behavior directly.

Bagheri et al. (2021) used a real-time emotion detection system based on facial expressions, developed in earlier work (Bagheri et al., 2020), that relies on a Stacked Auto Encoder NN. The system classifies facial expressions into the six basic emotions and determines the most dominant emotion over the previous 10 seconds. This emotion detection system serves as a key supporting component of the SR controller developed in Bagheri et al. (2021) by providing the emotional state needed by the Reinforcement Learning (RL) module to select appropriate empathic behaviors. Here, the NN serves as a supporting module that supplies affective state estimates to the main behavioral controller.

In a brain-inspired architecture for a SR manifesting Theory of Mind (ToM), Tani et al. (2020) used NNs to model relevant brain regions. While most regions are implemented through simple NNs with predefined, non-trainable weights, two regions are implemented through voltage-driven plasticity-centric spiking NNs. The types of NNs, the number of NNs per brain region, and the number of layers per NN were biologically inspired by the corresponding brain region. These Spiking NNs were trained offline using a combination of supervised and unsupervised learning. This work illustrates a task-specific use of NNs that support internal processing rather than directly steering the SR behavior.

Filippini et al. (2021) used a multi-layer perceptron NN to detect the valence and arousal of users based on thermal images. The combination of thermal and visible-light images, when processed by a computer vision program, allows measuring the temperature of the tip of the user's nose. The gradient of this temperature serves as input to the NN, which outputs the valence and arousal of the user in terms of negative, neutral, or positive arousal. Based on this output, a simple rule-based system is used to generate the behavior of the SR. This again reflects the common use of NNs in SRs as supporting perceptual components rather than primary behavioral controllers.

The controller proposed in M. H. Lee et al. (2023) for SRs that coach physical rehabilitation therapy also includes an ML module to assess the quality of movements performed by users. M. H. Lee et al. (2023) evaluated several supervised learning algorithms and selected a feedforward NN based on performance. The NN is trained offline using annotated videos from multiple participants. M. H. Lee et al. (2023) also proposed fine-tuning the pre-trained model to best suit each participant, but found that this approach degraded performance and thus did not adopt it. The assessment produced by the ML module is combined with the assessment produced by a rule-based model using a weighting approach. This reinforces the supporting role of NNs in hybrid controllers rather than as primary behavior mechanisms.

In the SOcial Norm Aware Robots (SONAR) architecture developed by Dell'Anna and Jamshidnejad (2024), GPT-based language models — deep NNs designed for processing and generating language — are integrated to provide natural conversational responses during Human-Social-Robot Interactions (HSRIs). This enables the robot to engage in open-ended dialogue, while the SONAR architecture handles social norm awareness and behavioral control. This work reflects the recent expansion of deep NNs in supporting language-based interactions.

General appraisal of neural networks

- NNs are mostly used in social robotics for perception, including emotion recognition (Bagheri et al., 2020; Bagheri et al., 2021; Filippini et al., 2021), speech recognition (Benedictis et al., 2023; Dissing & Bolander, 2020), object recognition (Vossen et al., 2018), and user recognition (Dissing & Bolander, 2020).

- They are also used as function approximators to mimic specific processes — in Tani et al. (2020) to emulate the functioning of certain brain regions within a bio-inspired approach, and in Shiarlis et al. (2017) to learn social movements from demonstration.

- More recently, they have been applied to language understanding and generation (Benedictis et al., 2023; Dell'Anna & Jamshidnejad, 2024).

- Based on the reviewed literature and inherent capabilities of NNs, they tend to be most effective in social robotics when applied to components that require approximating complex, non-linear relationships, extracting patterns from high-dimensional data, and learning from examples. Accordingly, perception,

movement imitation, and language processing emerge as application areas where NNs are particularly well suited

- However, NNs require large training datasets, are computationally intensive, operate as black-box models, and may fail in situations not represented in training data. For these reasons, they are in general less suitable as standalone or primary control mechanisms for steering SR behavior, and are more commonly integrated as supporting components, particularly in scenarios involving vulnerable or sensitive users or highly variable environments.

Reinforcement learning

RL is an ML framework in which an agent learns an optimal policy to perform a certain task by interacting with its environment in a trial-and-error fashion (Sutton, Barto et al., 1998). The current condition of the agent is mathematically described by a state, which encapsulates all relevant variables necessary to characterize the agent and environment system at a given time. The RL agent learns the best actions to take in each state, a strategy known as its policy. A reward function defines the task objective by providing feedback (either positive or negative) based on the actions the agent takes. The goal of the agent is to learn a policy that maximizes the expected cumulative return over time, even when individual rewards are sparse or delayed (Sutton, Barto et al., 1998).

Among classical RL methods, Q-learning is frequently used in SR because it is simple to implement and directly learns the action-value function, i.e., the function that estimates the expected cumulative return of selecting a certain action in a certain state. The learned policy then selects actions that maximize the action-value function (Sutton, Barto et al., 1998). In practice, Q-learning is most suitable for small and discrete state-action spaces.

Tapus and Mataric (2008) conducted one of the earliest studies using RL to control SRs, in which the RL algorithm adapts the SR's extraversion level in response to the performance of post-stroke patients during rehabilitation exercises. In this setting, the SR motivates and monitors the users. The state space of the RL agent is not explicitly specified in the paper. The action space included behavioral parameters, such as verbal content, volume, speech rate, proxemics, and the speed and amplitude of body movements. The reward function is defined as the number of exercises performed by the user in the previous 15 seconds. The behavior policy is updated every 60 seconds: the average reward during this period is computed, and the gradient of the reward function with respect to each parameter in the action space is calculated. A new policy is then derived by adjusting each parameter in the direction of the gradient. Although each participant interacted with the robot for only 15 minutes — often insufficient to converge to an optimal policy — the authors noted that the approach could be extended for multi-session adaptation across therapy sessions. This early application of RL in SRs illustrates a pattern that remains common, where simple RL algorithms operate in low-dimensional action spaces to select the interaction strategy of SRs.

Magyar and Vircikova (2015) proposed using interactive RL to generate an autonomous controller that learns from demonstration by mimicking the action

selection behavior of a wizard in a Wizard-of-Oz (WoZ) setting (i.e., where a human manually controls the SR during interaction). However, despite claiming to use RL, the approach described to learn from the wizard is more accurately described as a supervised learning mapping problem rather than an RL problem: the wizard controls the SR based on the reactions of the user, the pairs of states and actions are recorded as rules, and a policy is approximated based on the recorded rules to eventually replace the wizard.

In a later work, Magyar et al. (2016) presented a new approach for SRs to learn an interaction policy by interacting with human experts. The proposed method defines the state space as two user variables (engagement and motivation), while the robot can choose from high-level action categories, including motivating the user, engaging the user, or asking task-related questions. However, instead of learning to mimic the behavior of a human wizard, as proposed in Magyar and Vircikova (2015), Magyar et al. (2016) used interactive RL where the human wizard provides reward signals to evaluate the actions selected by the robot (i.e., the reward is provided by the human). The RL agent employs Q-learning to learn the policy, but with a key modification: action selection is interactive, allowing the wizard to override the action selected by the RL agent when necessary. On the one hand, combining RL with human intervention allows the robot to safely learn over time and gradually become more autonomous, as human-provided rewards directly steer policy learning and action overrides constrain undesirable actions. On the other hand, the approach is limited to a small state space and a limited set of actions, reflecting practical scalability constraints of interactive RL due to reliance on human feedback.

The tutor SR proposed by Gao, Barendregt et al. (2018) to assist users playing grid-based logic games aims at personalizing the style of verbal social support using a Multi-Armed Bandit (MAB) approach, an RL-based algorithm: MAB allows discovering an optimal action (according to a reward function) amongst a finite set of actions by balancing exploration and exploitation, without modeling a state space. In Gao, Barendregt et al. (2018), four categories of supportive robot behaviors form the action space, and the reward function combines task performance (i.e., time the participant takes to complete the puzzles) and positive user reactions (i.e., verbal feedback) to the robot's interventions. An experiment with 24 adult university participants showed that users performed better when tutored by a SR displaying random supportive behaviors than when tutored by the SR with the proposed personalized controller. Accordingly, authors concluded that personalization does not necessarily benefit learning and may, in some cases, hinder it. However, because MAB does not model a state space and the identified policy converges to always selecting the same action, each user repeatedly receives the same supportive behavior, regardless of their current performance or mental states. This outcome points not to the ineffectiveness of personalization per se, but to a deeper algorithmic misalignment: users often prefer varying and adaptive behaviors (Leite et al., 2013; Matarić & Scassellati, 2016), but MAB is designed to find the single best action. This illustrates how simplified RL-based methods without state representation can produce static policies that fail to adapt to the evolving needs of users.

The Hierarchical Human-Robot Learning (hHRL) framework (Clabaugh et al., 2019)

was used to control the SPRITE robot (Short et al., 2020) to interact with children with autism by playing mathematical games (for details see Section 5.3.1). This framework includes two submodules, referred to as the *challenge controller* and *feedback controller*, implemented using RL. The two controllers manage the *level of challenge* (i.e., difficulty of the mathematical tasks) and the *level of feedback* (i.e., the explicitness of the hints and guidance provided to the user). The state space of each controller consists of the ten possible games, while the actions consist of five levels of challenge and four levels of feedback, respectively. Even though a state space is formally defined, the state only indicates which game is currently active and does not represent the evolving performance or mental states of users. As a result, each controller effectively learns an independent stateless policy per game, rather than a strategy that adapts to the evolving performance and mental states of users across games. The reward function of the *challenge controller* maximizes the *level of challenge* unless the mistake threshold (5 mistakes) is exceeded, at which point the reward becomes negative. The reward function of the *feedback controller* minimizes the *level of feedback* while penalizing excessive mistakes and help requests by the user. Both controllers used Q-learning to learn personalized policies during long-term interactions consisting of at least 20 sessions over a minimum of 30 days.

Patacchiola and Cangelosi (2020) developed Thrive, an RL-based framework for trust-based learning. In trust-based learning, a learner agent acquires information under the guidance of both reliable and unreliable informants, indirectly learning whom to trust. Motivated by biological findings reviewed in Patacchiola and Cangelosi (2020), which the authors interpret as suggesting that trust-based learning resembles an actor-critic RL algorithm, the system was designed following this structure. In such algorithms, the actor selects actions based on a policy, while the critic evaluates that choice via a value function, which is then used to improve the actor's policy. Accordingly, Patacchiola and Cangelosi (2020) designed a behavior steering system based on actor-critic RL for their robot. The framework is designed for a case study where the SR learns the labels of objects from two informants, one reliable and one unreliable. During the interaction, the SR perceives an object and receives a label from one of the informants. Then, the actor generates a belief about the label that best fits the object. Unlike classical RL, the agent does not rely on external rewards. Instead, inspired by the learning processes of the human brain, an intrinsic environment computes the cost of keeping the current belief based on the confidence level, the trustworthiness of the informant, and whether the informant's suggestion aligns with the belief. This cost is used to update the actor and the critic parameters using temporal-difference learning. The trustworthiness of each informant is updated when the agent is confident in its belief: it increases when the belief matches the informant's suggestion, and decreases otherwise. In contrast to many simplified RL applications for SRs, this work employs a more technically sophisticated algorithm. Nevertheless, it is tailored to a highly specific interaction scenario (i.e., trust-based learning) and does not directly generalize to broader SR interactions.

Bagheri et al. (2021) developed a SR behavioral controller that aims at showing empathy to users by first identifying the emotion of the user and then selecting and displaying a suitable utterance. A contextual bandit algorithm, a special case of RL

with a single-step horizon and no state transitions, is used to select the best utterance type based on the emotion displayed by the user. Twelve states are defined in the RL algorithm, combining three levels of extroversion and four basic emotions. While emotions are detected online by a separate emotion recognition module (Bagheri et al., 2020) and evolve during the interaction, the level of extroversion is determined through a questionnaire and is considered fixed for each participant. Accordingly, the algorithm effectively learns a separate policy for each extroversion level, adapting online only to changing emotions. This is comparable to training a distinct RL agent for each extroversion level. The actions of the SR consist of one of four categories of verbal empathic utterances: mimicking the user's emotion, motivating the user, distracting the user from negative emotions, or alleviating the impact of negative emotions. A simple binary reward function is used that returns $+1$ when the resulting emotion of the user is positive or neutral, and -1 when it is negative. A single policy shared across all participants was learned online during the interactions using a one-step Q-learning algorithm. During the interactions, after the RL module output the utterance category, one of two pre-defined utterances was randomly sampled and spoken to the user. This approach reinforces the pattern of applying simple RL algorithms with low-dimensional state and action spaces for behavioral control of SRs, while such simplifications often result in limited adaptability of the SR.

Hussain et al. (2022) trained two separate offline batch RL agents to generate non-verbal conversational backchannel behaviors, specifically head nods and smiles, aimed at enhancing user engagement. The RL agents share the same state space, which represents the short-term dynamics of the user speech: a 209-dimensional vector of statistical features computed from 1 second of user speech audio. The action space is binary (perform the backchannel or do nothing). The reward is based on an objective estimate of engagement inferred offline from observable interaction behaviors in annotated human-human conversation data, aggregated over a 15-second sliding window. Training is performed entirely offline using annotated human-human interaction data and the proposed Sequential Random Deep Q-Network algorithm. The RL agents are integrated within a larger interaction system that handles dialogue flow and turn-taking. Because the actions selected by RL are simple and discrete (nod or smile), and their effects on the reward (user engagement) are measured objectively, the interaction data is suitable for annotation. This enables training the model offline rather than during online interactions, which allows the use of a complex RL algorithm.

Munster and Jamshidnejad (2025) proposed a hybrid framework combining fuzzy logic control and RL to achieve both personalization and adaptation in SRs. In their approach, the RL module adapts the robot's social behavior to the evolving state-of-mind of the user. The state space consists of five discrete categories representing the user's estimated state-of-mind (very bad, bad, neutral, good, very good), inferred from eight basic emotions through a fuzzy inference system. The action space corresponds to adjustments of seven behavioral parameters, including amount and volume of speech, number of gestures, types of interactive and motivating comments, feedback style, and proxemics. Then, Q-learning is used to learn an action-value function over this discrete state–action space. Importantly, rather than designing the reward function as an analytical function, it is generated through fuzzy rule-based modules

that evaluate both the quality of the resulting user state-of-mind and the intensity of its change. Finally, the RL module is warm-started by a fuzzy personalization module based on user personality traits, reducing the exploration burden and accelerating convergence compared to conventional RL approaches. Although the RL algorithm relies on small state and action spaces and Q-learning (similarly to many approaches described in this section), its integration with fuzzy logic for state abstraction and warm-start initialization promotes adaptability and personalization.

General appraisal of reinforcement learning

- In social robotics, RL is primarily used to learn policies that determine the interaction strategy of robots, such as when to execute different high-level social behaviors (Bagheri et al., 2021; Clabaugh et al., 2019; Gao, Barendregt et al., 2018; Magyar et al., 2016; Munster & Jamshidnejad, 2025; Tapus & Mataric, 2008). This enables robots to adapt to individual users and to optimize interactions over time, while reducing the need for extensive manual programming typical of rule-based approaches.

- However, most implementations of RL in this domain operate in low-dimensional, discrete state and action spaces (often with fewer than ten states or actions) (Bagheri et al., 2021; Clabaugh et al., 2019; Gao, Barendregt et al., 2018; Magyar et al., 2016; Munster & Jamshidnejad, 2025). Some approaches omit a state space entirely and explicitly formulate the problem as a multi-armed bandit (Gao, Barendregt et al., 2018), meaning that the policy does not depend on any representation of the user or environment, and actions are chosen without considering context. Others, such as Clabaugh et al. (2019) and Bagheri et al. (2021), fix one state dimension per participant, effectively learning separate policies for each fixed value while adapting only along the remaining dimensions. This is equivalent to running multiple RL agents with simplified state spaces.

- Many methods produce static policies per participant, with no adaptation to dynamic mental states (Clabaugh et al., 2019; Gao, Barendregt et al., 2018; Tapus & Mataric, 2008). Even when mental-state-dependent policies are considered, the number of states remains small (Bagheri et al., 2021; Magyar et al., 2016), which limits both personalization and generalization across users and contexts. []

- Furthermore, several approaches optimize immediate rewards rather than cumulative discounted returns. Therefore, they do not fully exploit the potential of RL to learn long-term strategies (Bagheri et al., 2021; Gao, Barendregt et al., 2018; Tapus & Mataric, 2008).

- Finally, the RL algorithms used are generally basic, such as Q-learning (Bagheri et al., 2021; Clabaugh et al., 2019; Magyar et al., 2016; Munster & Jamshidnejad, 2025) or one-step RL variants (Bagheri et al., 2021; Gao, Barendregt et al., 2018), which are suitable mainly for small-scale or highly structured interaction tasks. Only recently have studies applied state-of-the-art RL algorithms without simplifying them to fit the problem (Hussain et al., 2022; Patacchiola & Cangelosi,

2020). The key difference from prior work is that, in these two studies, RL is confined to narrowly defined internal components rather than full behavioral control. This separation from real-time user interaction makes more extensive training possible: simulation enables extensive online learning in Patacchiola and Cangelosi (2020), while annotated data supports offline training in Hussain et al. (2022). As a result, the use of more sophisticated RL algorithms becomes feasible.

- Overall, RL allows policies to be learned from interaction experience without full manual design. However, most applications for SRs employ simplified RL algorithms in environments with small state and action spaces, resulting in simple policies that are insufficiently adaptive to user and environmental states.

- These limitations stem from the inherent challenges of applying RL to social interactions: accurately representing users and environments requires high-dimensional state spaces, which make exploration less efficient; rewards are often sparse or ambiguous; and complex algorithms increase training difficulty. Furthermore, these challenges cannot be addressed through extended training in real-world settings, because interactions with human users are limited. Consequently, most approaches favor simplified algorithms, low-dimensional spaces, and immediate rewards, as these make training easier, but also restrict adaptability and long-term planning.

- These observations raise the question: *Is RL always suitable for SR control?* Rather than using it for overall control, RL may be better suited for specific tasks that naturally fit the RL paradigm (especially if they enable offline or simulation-based training) or for high-level decision-making when states and actions are limited.

Classification and clustering algorithms

This category of ML-based approaches includes supervised and unsupervised methods that assign input data to predefined classes or discover clusters based on shared characteristics within elements of the input data.

The gaze control system proposed by Zaraki et al. (2014) integrates a facial recognition engine for user identification. This engine uses Principal Component Analysis, a linear dimensionality-reduction method, for feature extraction from facial images, followed by identity assignment through comparison with a pretrained face database. Within the gaze control system, user identification enables the robot to track individuals over time and adjust its gaze based on prior gaze history. In this system, classification supports the perceptual processing of the robot, functioning as a component that assists the main behavioral controller rather than driving behavior directly.

Tahir et al. (2020) used classification algorithms to estimate the social state of speakers during two-person conversations, thereby improving the social perception of SRs. Four social states, describing interaction-level speaking behavior, are considered: normal, uninterested, overly talkative, and aggressive. The proposed framework estimates three high-level indicators (i.e., the level of interest, dominance, and

agreement) based on low-level conversational and prosodic cues, including turn-taking, speaking rate, volume, and acoustic features. First, the most relevant conversational and prosodic cues were automatically selected through a feature selection algorithm. Then, four different classification algorithms (K-Nearest Neighbor (Cover & Hart, 1967), NN, Naive Bayes (G. H. John & Langley, 1995), and Support Vector Machine (Cortes & Vapnik, 1995)) were compared, and the best one was selected. The classifiers were trained and assessed based on 150 manually annotated audio files. Support Vector Machine achieved the highest precision for estimating the three indicators and was then integrated into the SR controller. Finally, the high-level indicators were converted into the social state of the speaker using a pre-defined conversion table. Based on the estimated social state, the SR acts as a social mediator, providing feedback to the speaker through speech and gestures (e.g., encouraging participation, discouraging excessive talking, or calming aggressive behavior). This again reflects the role of classification algorithms as perceptual components that support the behavioral control system of SRs.

The hybrid controller proposed by Sequeira et al. (2016), which aims to extract interaction strategies from WoZ demonstrations of experts, incorporates an ML module that operates in parallel with the rule-based component detailed in Section 5.3.1. In the ML module, classification and clustering algorithms are used to map inputs to the behaviors of the SR, using the features extracted from the WoZ demonstrations as training data. This module complements the policy defined by the rule-based component, which encodes explicit interaction strategies, by capturing complex and subtle social behaviors that are hard to detect manually. Here, classification contributes directly to behavior generation by approximating mappings between interaction inputs and behavioral responses that are difficult to formalize manually, enabling richer behavior but also risking reduced interpretability and predictability.

In Dziergwa et al. (2018), the EMYS robot (Kedzierski et al., 2013) was deployed as an autonomous in-home assistant during a long-term cohabitation study. To motivate users to interact often with the SR, they were instructed to teach EMYS to recognize colors by showing colored objects (e.g., cards or clothing) and stating the color name. To model the learning from the robot's point of view, the k-nearest neighbors classifier algorithm (Cover & Hart, 1967) is employed. After teaching colors to the robot, users were able to test its knowledge by showing the robot colored objects and having it guess the color name. While useful for demonstration-based learning, this application of classifiers is highly task-specific, offering little support for broader adaptive behavioral control of SRs.

The Human-Robot Scaffolding architecture (HRS-EDU) framework introduced in Páez et al. (2020) uses K-Means clustering to classify the emotional state of users as boredom, curiosity, anxiety, or comfort, although the input vector of the clustering algorithm is not explicitly mentioned in the paper. This reinforces the common use of classification and clustering algorithms to support the perception of SRs.

In the trust-based learning architecture *Thrive* proposed by Patacchiola and Cangelosi (2020), Self-organizing Maps (SOMs) are employed as function approximators, while the overall control is based on an actor–critic RL framework (see Section 5.3.2 for details). SOMs are unsupervised learning algorithms that cluster similar inputs and organize

them on a low-dimensional grid so that nearby units represent similar data. Both the actor and critic are built using units containing two SOMs: a vision SOM that processes visual inputs (e.g., observed objects) and a vocabulary SOM that processes auditory inputs (e.g., informant's suggestions). Their training occurs through Hebbian learning, a biologically inspired rule that strengthens or weakens connections between SOM based on whether their units activate together. Overall, in this framework, clustering methods act solely as internal representation mechanisms rather than driving robot behavior, thereby limiting their role to a highly specific learning task within the architecture.

Bettosi et al. (2024) proposed using clustering to infer patterns of behavior sequences displayed by human experts in assistive contexts, and then using them to derive structured behavior representations for SRs. The input dataset includes labeled high-level behaviors used in assistive sessions, such as questioning users and providing instruction or feedback. This approach follows a similar idea to the one presented by Sequeira et al. (2016), where clustering approaches are used to extract behavioral patterns from human demonstrations.

General appraisal of classification and clustering algorithms

- In social robotics, classification and clustering algorithms are primarily used to support perception tasks, such as user recognition (Zaraki et al., 2014) and identification of social or emotional states of users (Páez et al., 2020; Tahir et al., 2020).
- Classification algorithms can simulate natural learning processes on the robot, enabling it to acquire knowledge directly from user interactions (Dziergwa et al., 2018).
- Classification and clustering methods are also applied to capture complex behavioral sequences and patterns from human demonstrations or interactions. These learned patterns, which are difficult to detect and formalize manually, are used to drive robot behavior and often complement rule-based systems (Bettosi et al., 2024; Sequeira et al., 2016).
- Clustering mechanisms have also been used as function approximators within RL-based architectures, using sensory data (e.g., audio and visual inputs) to estimate internal states and thereby implicitly incorporating a perception component (Patacchiola & Cangelosi, 2020).
- Similarly to NNs, classification and clustering algorithms can handle high-dimensional sensory inputs, learn from examples, and capture intricate or subtle behaviors that are challenging to formalize manually.
- However, the effectiveness of these algorithms depends on the quality of the data. Furthermore, supervised learning methods require labeled data, which is often difficult to obtain in social robotics. Additionally, these algorithms may exhibit limited generalizability across HSRIs and reduced interpretability.

- Although classification and clustering algorithms may be suited as perception modules, as decision approximators, or as components of hybrid control systems, they rarely operate as standalone mechanisms for full robot behavioral control.

5.3.3. Modeling Approaches

This section discusses control approaches that rely on explicitly defined models to represent aspects of the Human-Social-Robot Interaction (HSRI), such as the task, the robot's behavior, or the user's behavior. These models may serve as the foundation for decision-making and planning, or act as supporting mechanisms that inform or regulate behavior selection within social robotics systems.

Cognitive and behavioral architectures

In this section, we discuss control approaches that rely on structured models of internal cognitive processes to represent the reasoning or decision-making of the agents participating in HSRI.

Mazzei et al. (2014) designed a cognitive architecture intended as a general-purpose behavioral control system for Social Robots (SRs) and tested it on the FACE robot (Pioggia et al., 2004). The architecture is implemented as a modular system of distributed services allowing real-time integration, and comprises three main modules: sense, plan, and act. The *sense* module extracts high-level verbal and non-verbal social cues from a camera stream, such as speaking activity, facial expressions, gestures, and body position. The *plan* module, considered the "brain" of the SR, is implemented as a rule-based structure (discussed in Section 5.3.1) that performs high-level deliberative reasoning and behavior selection. Finally, the *act* module executes robot behaviors, including computing and displaying facial expressions based on the circumplex model of emotions, controlling gaze, and resolving conflicting movement requests. The controller employs a hybrid approach, in which a deliberative layer manages complex, slower tasks such as symbolic reasoning and attention selection, while a reactive layer handles fast, simple responses to stimuli such as gaze control and low-level action execution. While the deliberative layer makes use of all three modules, the reactive layer creates a direct *sense-act* connection that bypasses the *plan* module. This layered hybrid structure enables real-time interaction by preventing slower processes from delaying quick reactions. However, its dependence on extensive manual rule design may limit scalability.

Mitchinson and Prescott (2016) proposed a brain-inspired architecture for the control of an animal-like companion robot, Miro (Collins et al., 2015). The architecture is structured into three layers (i.e., spinal cord, brainstem, and forebrain), where higher layers can modulate the responses of lower layers, while lower layers operate independently in the absence of higher layers. The lowest layer, the spinal cord, behaves as a reactive controller that produces reflexes when external stimuli are detected (e.g., stopping wheel motion if no surface is detected). The middle layer, the brainstem, mainly regulates the affective state, generates simple spatial behaviors, and coordinates motor generation. The emotion regulation is modeled according to the circumplex model of emotions and is influenced by internal factors (such as the

circadian rhythm) and the user's behavior toward the robot. The spatial behavior system allows the robot to display certain behaviors (e.g., approaching or avoiding stimuli) depending on the position and type of the perceived stimuli. Finally, a basal ganglia model selects between competing behaviors, and the motor pattern generation converts the selected behavior into actuator commands. The highest layer, called the forebrain, models cognition and aims at generating higher-level behaviors by altering or influencing the work of the lower layers. Overall, the Miro platform illustrates how a biologically grounded layered architecture can support general-purpose social behavior without being tailored to a specific task. However, while user behavior influences the affective state and action selection of the robot through immediate sensory input, no explicit user modeling or long-term adaptation is implemented. Additionally, because of Miro's animal-like design, its behavioral complexity is intentionally simpler and avoids the expectations associated with human-like cognitive systems.

The architecture developed by Vossen et al. (2018) for social interaction and communication in the Leolani robot consists of four layers: the signal processing layer, which handles vision and speech inputs (e.g., face detection, object recognition, speech recognition); the conversation flow layer, which handles decision-making; the natural language layer, which converts language into basic knowledge blocks (i.e., structured subject–predicate–object representations) and generates language based on the output of the other layers; and the knowledge representation layer, which stores all knowledge and perspectives obtained through sensory input and communication with users. The framework is also inspired by Theory of Mind (ToM) and Belief-Desire-Intention (BDI), and implicitly incorporates elements from both. The knowledge representation layer (explained in detail in Section 5.3.1) reflects a key principle of ToM: the understanding that rational agents can hold different beliefs over the same fact. Hence, the information acquired by the robot is stored not as absolute truth, but in terms of who holds which beliefs. Additionally, the conversation flow layer is structured as a BDI architecture, where beliefs are the output of other layers; desires are focused on acquiring social information and resolving knowledge uncertainties; and intentions are chosen from a pre-defined list to satisfy the desires, given the current beliefs. This architecture demonstrates the value of incorporating perspective taking and ToM into decision-making of SRs for managing complex social interactions. However, its reliance on predefined belief structures may limit scalability to more diverse or unanticipated interaction domains.

Rodríguez-Lera et al. (2018) proposed the Hierarchy of needs, Motivational variables and Pool of FSM (HiMop) architecture to generate behaviors for SRs that vary based on the role of the robot. The architecture follows three principles: (1) a "Hierarchy of Needs" with three levels (i.e., existence, relatedness, and growth), where lower-level needs have priority over higher-level needs; (2) motivational variables, which drive the behavior of the robot based on its role to ensure satisfaction of the robot's needs; (3) a pool of Finite State Machines (FSMs), where each FSM defines one possible behavior (for details see Section 5.3.1). The architecture is organized into three modules: a deliberative system that plans to achieve long-term goals of the robot; a reactive system that handles unexpected external stimuli, such as obstacles or a ringing doorbell; and a motivational system that tracks the motivational variables

and influences both deliberative planning and reactive behavior selection to prioritize actions according to the robot's needs. Behavior generation starts with defining the current goals of the robot (based on either the motivational subsystem, the reactive subsystem when external triggers occur, or user commands). The deliberative system then identifies all behaviors that support those goals and generates a pool of all relevant FSMs. Finally, the reactive system selects behaviors from the pool based on the motivational variables and external stimuli. The key strength of this motivational architecture is its role-dependent modulation of behavior, which allows the robot to exhibit more varied behavior patterns under similar conditions. However, because this architecture selects behaviors from a predefined pool, its flexibility and capacity for true emergence remain limited.

Cao et al. (2019) presented a cognitive architecture that semi-autonomously steers the behavior of SRs through shared control with therapists. The architecture stands out for producing behaviors that can be reused across different therapeutic scenarios and robot platforms, enabled by its modular design that separates abstract behavior generation, scenario management, and robot-specific behavior execution. The architecture consists of six modules. (1) The perceptual system converts raw data into interaction events, such as "face detected". (2) The user manager stores information about the user. (3) The affect system generates the emotional component of the robot's behavior, shaped according to the circumplex model of emotions and modulated by the robot's personality. This personality can be configured using the Big Five framework by the therapist or system designer. (4) The behavior generation system creates the behavior of the robot based on the aforementioned modules and follows a three-layer architecture. A deliberative layer generates task-based behaviors depending on the therapeutic scenario. A reactive layer generates reactions to stimuli, such as the emotions and touches of the user, and adapts the robot's gaze to the position of the user. Finally, a reflective layer acts as a buffer that ensures that the behaviors generated by the other layers do not violate predefined ethical and clinical requirements. The last two modules are (5) the supervision interface, through which therapists interact with the system, and (6) the behavior realization system, which maps the generic behavior created by the behavior generation system into platform-specific actions using kinematic conversion of joint positions. This modular organization enables the system to generate both socially expressive and therapeutically appropriate behaviors while integrating therapist supervision. Furthermore, such a structure allows the architecture to be used across different therapeutic contexts and platforms, thereby facilitating real-world deployment.

The EMYS robot (Kedzierski et al., 2013) adapted for long-term in-home assistance in Dziergwa et al. (2018) employs a three-layer control architecture that enables fully autonomous operation. This hierarchical architecture features interdependent layers, with each layer building upon the functionality of those below it. The lowest layer manages all modules related to actuators, sensors, and external software interfaces. The middle layer implements the functional competencies that the robot is able to perform, such as sending emails, checking weather forecasts, and engaging in natural dialogue with the user. The highest layer handles the robot's decision-making processes, determining when and how to trigger each competency from the middle

layer. This layered architecture enables robust long-term home autonomy, though its focus on scheduling functional competencies and its reliance on scripted responses provide limited support for deeper social adaptivity.

The Human-Robot Scaffolding architecture (HRS-EDU) framework developed by Páez et al. (2020) was designed to control the behavior of a SR that assists users in solving block-manipulation puzzles, i.e., puzzles where a sequence of actions transforms an initial configuration into a target configuration, according to specific rules. The controller, which is grounded in the BDI framework and established pedagogic theories (flow theory and scaffolding strategy), consists of four modules: sensory, beliefs, desires-intentions, and action-planner. The sensory module uses microphones and cameras to extract information regarding the emotional and cognitive state of the user and task-related information. The beliefs module estimates this state based on knowledge of the problem rules, problem-solving ability, sequence of moves made, and emotional state of the user. The desire-intention module selects a goal among six categories (problem-solving skills, cognitive support, emotional support, challenge control, life signals, and immediate support). The score of each goal is calculated based on the current beliefs of the robot about the cognitive and emotional state of the user, and the goal with the highest score becomes the current intention of the robot. Finally, the action-planner module selects the SR's movements and expressions according to the generated intention and the SR's emotional state, which is determined based on the user's performance and need for guidance through predefined rules grounded in established pedagogical and psychological principles. This cognitively grounded control architecture enables SRs to generate adaptive support during structured learning tasks. Nevertheless, its tight coupling to scaffolding principles and well-defined task models limits its direct applicability to broader social interaction settings.

Tani et al. (2020) developed a brain-inspired ToM model for SRs using a brain-structured architecture that enables learning from self-experience to solve false-belief tasks. The architecture incorporates four conceptual pathways critical to this task: the self-experience learning pathway, the motivation understanding pathway, the reasoning about one's own belief pathway, and the reasoning about other people's belief pathway (Tani et al., 2020). These pathways, which are computational constructs, are composed of computational modules inspired by specific brain regions known to be active during false belief reasoning, selected based on prior experimental studies involving human participants reported in the literature. While effective for modeling false-belief reasoning, the architecture is centered on a narrow subset of ToM constructs and does not incorporate affective, motivational, or decision-making processes, which would likely limit its applicability to more general, open-ended social interactions.

The work presented in Chapter 2 proposed a ToM mathematical model that allows SRs to estimate the evolution of the mental states of users. The model is structured into interconnected perception, cognition, and decision-making modules, reflecting key components of human reasoning. The perception module models two processes: converting raw world data into perceived data, and then transforming the perceived data into higher-level knowledge. The cognition module models the evolution of fast-dynamics state variables (beliefs, goals, and emotions) and incorporates slow-dynamics variables, which are long-term user traits that influence fast-dynamics

mental states. This structure enables both short-term prediction and long-term personalization in socially assistive contexts. The cognition module is formulated as a state-space representation, i.e., a mathematical framework that describes how states evolve over time based on inputs, which in this case are the higher-level knowledge produced as output by the perception module. This enables integration of the model with systems and control approaches, such as model-based control and state estimation methods. Finally, the decision-making module selects intentions based on the updated beliefs and goals, then determines specific actions based on those intentions. Since the processes and mental state dynamics in this model are defined through parameterized equations, the model can be personalized to each user based on data collected during HSRIs. The authors validated the model in simulated emergency evacuation scenarios, demonstrating its ability to estimate intentions and predict actions. The work presented in Chapter 3 extends the evaluation of the model to real-world HSRI settings, supporting the earlier validation results by demonstrating that the model remains effective at estimating user mental states during prolonged HSRI. This model exemplifies the structured, modular organization common to cognitive architectures while demonstrating strong potential for personalization and adaptability in social interactions.

Benedictis et al. (2023) developed an architecture for SRs that provides both long-term user personalization and adaptation of communication style to real-time contexts. The architecture is designed to be general enough for any type of healthcare application of SRs. The architecture consists of three modules: a user model, a reasoning layer, and an interactive layer. The user model is based on a standard health-care framework that describes user disabilities, and is implemented using a logic system (see Section 5.3.1 for details). Similarly to the work presented in Chapter 2, it includes both fast-changing variables, such as emotions, and slowly varying variables, such as user disabilities. The reasoning layer handles personalization by scheduling and shaping assistive behavior based on the long-term user traits, and is implemented using a mathematical approach (detailed in Section 5.3.3). The interactive layer generates the SRs's actions and ensures real-time adaptation by adjusting the communication style of the robot according to fast-changing variables, using *if-then* rules (detailed in Section 5.3.1). Both layers rely on the user model, and interact with each other: the reasoning layer determines the content of the SRs's behavior, which the interactive layer generates; the interactive layer provides feedback on interaction outcomes and user performance to the reasoning layer and updates the model when new user information is captured. This architecture reflects the trend toward general, layered frameworks that integrate long-term personalization with real-time adaptation.

The SOcial Norm Aware Robots (SONAR) architecture, proposed by Dell'Anna and Jamshidnejad (2024), is a multi-agent system with three groups of agents, each handling one level of abstraction. Worker agents receive and process low-level information from sensors and send commands to actuators of a robot. Manager agents aggregate information from multiple sources to form high-level knowledge, such as beliefs. The BDI agent reasons about new beliefs and finds a suitable robot action. When reasoning, the BDI agent identifies social norms relevant to the context, defines which desires and intentions are allowed or required, and chooses an action accordingly.

Additionally, social awareness is incorporated into the architecture through three types of rules: social interpretation rules, which infer social context from environmental signals (e.g., inferring affective state from speech volume); behavior qualification rules, which encode how the robot should act in a given social context; and finally, prohibition and obligation rules, which enforce basic social norms. The integration of these norms is distributed among the different architecture agents. The prohibition and obligation rules are implemented using classic logic. Social interpretation and behavior qualification are implemented using fuzzy logic, and a norm-adapter agent adjusts them to different social or cultural contexts based on data collected during the interactions. This architecture distinguishes itself by explicitly modeling social norms and enabling context-sensitive adaptation, emphasizing the importance of norm-aware reasoning in social interactions. In addition, the multi-agent design facilitates scalable and distributed deployment.

General appraisal of cognitive and behavioral architectures

- Cognitive and behavioral architectures typically integrate reasoning, knowledge representation, perception, and decision-making into a unified framework.
- These architectures enable SRs to engage in general social interactions rather than being restricted to a single application (Benedictis et al. (2023); Cao et al. (2019); Dell'Anna and Jamshidnejad (2024); Dziergwa et al. (2018); Mazzei et al. (2014); Mitchinson and Prescott (2016); Rodríguez-Lera et al. (2018); Vossen et al. (2018) and Chapter 2).
- Common design patterns include layered control, typically organized into three or more layers. Many frameworks combine a reactive layer for fast, stimulus-driven responses with a deliberative layer for complex planning and long-term behavior (Cao et al., 2019; Mazzei et al., 2014; Mitchinson & Prescott, 2016; Rodríguez-Lera et al., 2018). Some architectures employ modules that each operate at a distinct level of abstraction, from low-level sensing and execution to higher-level reasoning and planning (Benedictis et al., 2023; Dziergwa et al., 2018; Mitchinson & Prescott, 2016), while others separate perception (Cao et al. (2019); Dell'Anna and Jamshidnejad (2024); Mazzei et al. (2014); Vossen et al. (2018) and Chapter 2) and decision-making (Benedictis et al. (2023); Cao et al. (2019); Dell'Anna and Jamshidnejad (2024) and Chapter 2) from other reasoning or cognitive processes.
- Only two frameworks reviewed in this section were designed for specific applications (Páez et al., 2020; Tani et al., 2020), with modules focused on application-specific subtasks rather than following the general design patterns described above.
- The modular and layered nature of these architectures facilitates integration of multiple cognitive functions, enhances interpretability, and equips SRs to handle general-purpose social interaction across task domains. Such controllers support personalization, long-term engagement, and adherence to social norms and ethical constraints.

- Despite their complexity and demanding design requirements, these architectures enable SRs to generalize across applications, making them more versatile and reusable in diverse social contexts.

Mathematical modeling

This subcategory includes control approaches that use explicit mathematical models to represent aspects related to the HSRI and inform robot behavior. Specific groups of mathematical modeling, such as bio-inspired approaches, homeostatic control, or control based on cognitive architectures are discussed in separate categories.

A SR designed to encourage users to use a weight loss program, proposed by Kidd and Breazeal (2008), estimates the status of its relationship with the user (i.e., whether the relationship is in an initial, normal, or repair state) using a mathematical approach. During a six-week interaction with each user, starting after the first week, the robot estimated the status of the relationship based on user responses to predefined relationship-assessment questions. The robot rotated through a set of eight questions and asked two questions per day. The most recent eight answers (i.e., obtained during the previous four days) were averaged, and the same averaging was applied to the preceding set of eight answers. Then, the difference between the two averages was computed, and if the result was below a fixed threshold, the relationship was considered to be in need of repair. This relationship state was used by a rule-based control system to guide robot behavior, as discussed in detail in Section 5.3.1. The mathematical computation used in Kidd and Breazeal (2008) provides a transparent and computationally efficient way to estimate relationship status from structured user feedback. However, its reliance on fixed thresholds and short-window averaging limits adaptability and makes the estimate sensitive to short-term variability in user responses.

Rodríguez-Lera et al. (2018) proposed the HiMop architecture, which aims to produce less repetitive robot behavior based on the robot's role in assistive service contexts. To select the behavior of the SR from a pool of behaviors, the architecture relies on a mathematical formulation inspired by Hull's Global Theory of behavior. Based on this theory, the likelihood of producing each behavior (referred to as EP) is computed based on four factors: the frequency of interaction with users, the urgency of internal needs, the magnitude of the goal, and the frequency with which the behavior has been previously activated. In particular, EP is obtained by multiplying these four factors, normalizing the result, and adding a random noise term. Although it is not explicitly described how EP is used to select behaviors, the reported examples consistently indicate that the behavior with the highest EP at each time step is selected. The mathematical approach used in this framework provides an efficient way to translate theoretical motivational constructs into an implementable behavior selection mechanism. Nevertheless, the paper remains vague regarding crucial elements, such as how EP values arbitrate behavior selection and the intended role of the noise term.

M. H. Lee et al. (2023) designed a controller for SRs acting as rehabilitation coaches for post-stroke survivors, with the robot guiding and providing feedback during physical therapy exercises. To evaluate the quality of users' movements during these

exercises, the controller relies on the independent assessments generated by a Machine Learning (ML) and a rule-based module. The outputs of the two modules are then combined through a weighted average, where the weight of each module is derived from an offline evaluation of its performance against expert-labeled data. Consequently, the module that better matches expert assessments influences the final assessment more strongly. The mathematical approach used in this system is well-suited for balancing the outputs of two modules built on different paradigms, enabling effective integration without added complexity. However, the mathematical approach used here is constrained to a supporting role rather than implementing the decision-making of the robot.

Maroto-Gómez, Malfaz, Castro-González and Salichs (2023) proposed an architecture for driving the behavior of SRs, in which simulated neuroendocrine substances regulate internal biological functions of the robot and generate biological deficits, which in turn activate motivations that steer the robot's actions. The architecture relies on explicit mathematical equations to model these mechanisms. The level of each neuroendocrine substance is modeled as a summation of its influencing factors, including circadian and ultradian rhythms of the robot and external stimuli. The evolution of the circadian rhythm over time is modeled using cosine and sigmoid functions, chosen to reflect characteristic biological oscillatory dynamics. Finally, the biological deficits are computed with simple subtraction formulae, and the intensity of each motivation is computed by multiplying the sums of biological deficits, biological processes, and stimuli. This approach highlights the suitability of mathematical modeling for transparently implementing biologically inspired internal mechanisms, which facilitates inspection and validation of internal states and may support more predictable behavior. However, its complexity and reliance on many hand-tuned parameters may pose challenges for wider deployment.

The gaze control system proposed in Zaraki et al. (2014) is structured into three layers: perception, attention, and gaze control. The perception layer uses external libraries to extract high-level data from audiovisual input. The attention layer processes this high-level data to identify the most relevant gaze target. First, a measure of how much the robot should be interested in each user, Elicited attention (EA), is computed by summing the contribution of four components: social features, proxemics (distance to the robot), orientation (relative to the robot's field of view), and a memory component modeling habituation. The social features component is modeled via a weighted summation of normalized social cues (e.g., speaking activity and gestures), with weights derived from analyses of human gaze behavior in an offline eye-tracking study. The effects of proxemics and orientation on EA are modeled via mathematical decay functions that reflect how spatial dynamics influence attention. To model habituation, the memory component applies a decay function to reduce EA over time whenever eye contact is established with a user. Then, at each time step, the user with the highest EA is selected as the gaze target. Finally, the gaze control layer directs the robot's eye movements toward the target and only moves the head if a predefined threshold is exceeded. This coordination relies on mathematical models based on human motion data, enabling the system to mimic natural human eye-head coordination.

Aliasghari et al. (2020) proposed a gaze control system based on the controller design described in Zaraki et al. (2014), with the goal of improving naturalness. Similarly to Zaraki et al. (2014), the proposed gaze control system relies on mathematical modeling to support gaze selection and execution and consists of three units: perception, logic, and action. The perception unit uses external libraries to extract social cues from microphone and video streams. The logic unit is largely based on the attention layer developed in Zaraki et al. (2014), with a few modifications. The EA of each user is computed by multiplying — rather than summing — three of the four components proposed in Zaraki et al. (2014) (i.e., social features, proxemics, and orientation). This formulation reflects the assumption that a high EA is obtained only when all components are sufficiently relevant. Each component is modeled as in the original work, but the parameters were empirically estimated from gaze data collected in experiments with human participants. The gaze target is selected using rules based on EA values, as discussed in Section 5.3.1. The action unit differs from Zaraki et al. (2014): if the target angle is below a given threshold, the gaze shift is performed only with the eyes; otherwise, 20% of the gaze is achieved with eye movement and 80% with head movement. Finally, a blinking feature is included to better emulate natural human gaze behavior. A probabilistic approach is used, in which the likelihood of a blink decreases exponentially with time since the last blink, making blinks most likely shortly after the previous one. Both Zaraki et al. (2014) and Aliasghari et al. (2020) represent an early stage in gaze control, relying on mathematical modeling to incorporate biologically and psychologically inspired behavioral components into the gaze control system. This strengthens the theoretical grounding and enhances perceived naturalness; however, mathematical modeling remains a complementary component rather than the main control mechanism.

Within the gaze control system presented by Mishra and Skantze (2022), the gaze control unit integrates mathematical formulations that complement the main logic to direct the gaze of the SR. The mathematical mechanism operates on priority scores, which indicate the relative importance of each potential target, across the next ten time steps for all candidate targets (i.e., users, objects, and a designated environment target for gaze aversion). These scores are computed by the rule-based system described in Section 5.3.1. The mechanism determines the target of each time step in the planning window by selecting the candidate target with the maximum priority score in that step. Then, a slack variable regulates head movement: the slack, lower-bounded by zero, decreases linearly with the number of times the target appears in the ten-step sequence. The head moves only if the angular deviation between the current head orientation and the current target (i.e., the target of the immediate next time step) exceeds the slack. This mechanism establishes a trade-off between the planned gaze duration and spatial deviation, increasing the naturalness of the robot's gaze behavior. Similarly to Zaraki et al. (2014) and Aliasghari et al. (2020), mathematical modeling contributes to improved theoretical grounding and perceived naturalness, but remains confined to a supporting role within the control architecture.

In the cognitive architecture developed by Benedictis et al. (2023) for SRs providing personalized assistive services, the reasoning layer generates a long-term exercise plan

for each user using a mathematical approach. An incidence matrix encodes which functioning qualities (i.e., cognitive and physical capabilities such as attention, memory, and perception) each of the possible exercises stimulates. For each user, a vector represents the degree of disability for each functioning quality. Multiplying the matrix and vector yields a ranking vector that indicates the relevance of each exercise for the patient. The highest-ranked exercises are then considered by a planner to generate the long-term plan for the user. This is formulated as a mathematical planning problem, with inputs including user preferences (e.g., start time and interaction duration), degree of disability for each functioning quality, and the effect of each exercise on the user. The planner is constrained by initial and desired states (characterized by impairment levels) and the available time to achieve the desired states. Based on these constraints and inputs, a timeline-based planner (PLATINUm) generates the exercise schedule. It is not specified whether the planner finds optimal solutions or simply produces any feasible sequence that satisfies the constraints. In this architecture, mathematical modeling provides an effective method for generating personalized, long-term therapeutic plans. However, it functions as a planning subsystem rather than contributing to the generation of the social behavior of the robot.

Zhu et al. (2024) proposed a system that evaluates the social intention of human candidates (i.e., their willingness to interact with the robot) and uses this evaluation to control the gaze behavior of a SR. The quantification of social intention of users from their social cues is implemented using the entropy weight method, a multi-criteria evaluation approach where the importance of each criterion is based on the variability of that criterion. In Zhu et al. (2024), the criteria correspond to the social cues (e.g., head orientation, distance, voice volume, and facial cues). Starting from a matrix encoding the values of all social cues across participants in the field of view of the robot, this matrix is normalized, and the entropy of each cue is computed to define its corresponding weight. Then, for each participant, the Euclidean distances from their weighted social cues to the ideal (positive) and undesirable (negative) reference social cues are computed. Finally, a closeness coefficient combining the two distances is computed, and the participant with the highest coefficient is deemed to have the highest social intention. In parallel with the distance-based evaluation, gray correlation (i.e., a measurement of similarity between two sets of values) is applied to the weighted social cues to measure their similarity to the reference social cues. The results from both methods are fused using least squares optimization. This presents a more objective and robust weighting approach to assess social intention than previous methods (Aliasghari et al., 2020; Mishra & Skantze, 2022; Zaraki et al., 2014). Two sets of differential equations model the dynamics of eye and head movements, and are combined into a single state-space (input-output) model of the entire system. This model is then integrated into a model-predictive controller, which is used to steer the gaze of the robot, as described in detail in Section 5.3.3. This system involves more detailed mathematical modeling than previous gaze control approaches, enabling more realistic gaze dynamics. However, like the other gaze controllers discussed in this section, the application of mathematical modeling remains specific to this single component rather than supporting broader behavioral control.

General appraisal of mathematical modeling

- Mathematical approaches are commonly used in social robotics to estimate high-level social variables or indicators from low-level cues and often leverage pre-existing models from psychology or social sciences (Aliasghari et al., 2020; Kidd & Breazeal, 2008; Mishra & Skantze, 2022; Zaraki et al., 2014; Zhu et al., 2024).
- Several authors applied mathematical modeling for decision-making to select robot behaviors based on low-level indicators through mathematical formulations (Benedictis et al., 2023; Rodríguez-Lera et al., 2018).
- Mathematical models are also employed to implement well-defined biological functions that can reliably be described by mathematical expressions, such as differential equations, sinusoidal functions, or decay functions (Maroto-Gómez, Malfaz, Castro-González & Salichs, 2023; Zaraki et al., 2014; Zhu et al., 2024).
- Finally, some approaches used mathematical modeling to implement technical or physical subcomponents of the system, such as weighted fusion of module outputs and gaze control dynamics (Aliasghari et al., 2020; M. H. Lee et al., 2023; Mishra & Skantze, 2022; Zaraki et al., 2014; Zhu et al., 2024).
- Overall, mathematical modeling enables the direct embedding of theories from psychology, neuroscience, or control engineering in a transparent and interpretable manner. Additionally, these approaches are typically computationally efficient and do not require training on large datasets.
- However, across the works reviewed in this survey, the application of mathematical modeling is generally limited to subcomponents of the control architectures, serving as a complement to the main controller.
- This limitation likely stems from the reliance on simplified assumptions and fixed parameters, which restricts their ability to capture the variability and complexity of real-world social interactions and reduces flexibility and generalizability across different users and contexts.

Bio-inspired approaches

This subcategory includes approaches that draw inspiration from biological systems to model and steer robot behavior.

The cognitive architecture proposed by Mazzei et al. (2014) for the FACE robot (Pioggia et al., 2004), designed for expressive social interaction with humans, is inspired by the human nervous system, which processes information at two speeds: slower, complex planning and fast reflexes that operate independently to avoid delays. In a follow-up work, Lazzeri et al. (2018) extended the architecture by simulating internal physiological states of the SR (e.g., heartbeat, respiratory rate, and stamina). These simulated internal states are processed as if they were sensed by the robot, enabling it to integrate bodily awareness into its cognition and regulate its behavior in a more

human-like manner. These works illustrate how bio-inspired mechanisms can increase naturalness and bodily realism of SRs, functioning as enhancements to existing control systems rather than full behavioral controllers.

Mitchinson and Prescott (2016) drew inspiration from the brain of mammals to design the control system of the animal companion robot Miro (Collins et al., 2015). The architecture of the control system is inspired by the structure of the mammalian central nervous system. The system that selects between multiple, competing behaviors is inspired by the basal ganglia, a part of the brain of mammals considered to be essential for action selection. The spatial behavior system creates salience maps and selects spatial behaviors (approach, avert, orient) based on the superior colliculus, a part of the brain that manages spatial attention in mammals. Finally, the implementation of spinal reflexes, affective state regulation, and circadian rhythm are other bio-inspired elements of this architecture that further enhance its realism. Grounding the architecture in well-studied biological mechanisms supports natural and realistic behaviors. However, the presented bio-inspired elements are already complex to implement, even though the resulting system is intended for an animal-like robot with more modest behavioral expectations than those of humanoid robots.

The HiMop architecture (Rodríguez-Lera et al., 2018) generates behaviors for SRs based on bio-inspired concepts. The architecture incorporates motivational variables inspired by human psychology (such as rest, fear, curiosity, comfort, and frustration) that drive robot behavior to satisfy its needs. Furthermore, Rodríguez-Lera et al. (2018) incorporates additional bio-inspired mechanisms, such as circadian rhythms and habituation, to regulate and modulate these motivational variables over time. Thus, this architecture leverages bio-inspired motivational variables and physiological mechanisms to enrich variation and realism in behavior. However, these operate as modulatory components rather than a complete control system.

The architecture proposed by Tani et al. (2020) to attribute ToM to SRs (described earlier in Section 5.3.3) is inspired by the human brain, particularly the brain regions involved in the false-belief task. Both the internal structure of each brain region and the overall inter-regional organization inspire the computational implementation. Thus, each architectural module is inspired by a brain region. Additionally, brain plasticity (i.e., how the strengths of neural pathways in children evolve through learning) is incorporated in the architecture via specific types of Neural Networks (NNs) capable of analogous learning-based adaptation. This bio-inspired capability is implemented only in architectural modules whose corresponding human brain region supports experience-dependent learning. Grounding this architecture in human brain mechanisms results in a biologically faithful model of false-belief reasoning. However, its bio-inspired design targets a highly specialized capability and does not explicitly model social elements such as emotions or decision-making, which may limit its suitability for broader social interactions.

Following a similar approach, Patacchiola and Cangelosi (2020) structured their trust-learning architecture *Thrive* based on the brain components involved in trust-based learning in humans: the dorsal and ventral striatum form an actor-critic circuit where the dorsal striatum is associated with action selection, while the ventral striatum is associated with outcome evaluation. Furthermore, as in human trust-based learning,

the architecture incorporates internal assessments of informant credibility (i.e., the inferred reliability of an informant based on past interactions) rather than relying on external rewards. To identify objects from visual and auditory inputs, *Thrive* uses a type of neural networks called Self-organizing Maps (SOMs). The connections between SOMs are trained using Hebbian learning, a biologically inspired learning rule stating that when neurons fire together, their connections are reinforced. Thus, the inter-SOM connections are strengthened when units are active simultaneously and weakened otherwise. Similarly to Tani et al. (2020), the architecture is inspired by human brain mechanisms, resulting in biologically plausible and realistic behavior within its intended task. However, its design around a highly specific skill, such as trust-based learning, limits its suitability for HSRIs in broader social contexts.

Maroto-Gómez, Malfaz, Castro-González and Salichs (2023) presented a control architecture that integrates multiple bio-inspired processes to generate realistic and natural behavior for SRs. The architecture encompasses neuroendocrine responses, physiological processes, psychological processes, and motivations of the SR. These bio-inspired processes are influenced by both internal factors (e.g., circadian and ultradian rhythms of the robot) and external factors (e.g., stimuli). The neuroendocrine module implements the evolution of twelve neuroendocrine substances (e.g., melatonin, dopamine, cortisol) similarly to how they evolve in humans. The substances vary based on external stimuli (e.g., light and noise intensity, touch contact, user presence) and on the circadian and ultradian rhythms. These substances then regulate biological functions such as the sleep-wake cycle, social behavior, emotional responses, and stress management, emulating how these processes are regulated in human bodies. Furthermore, the SR has certain motivations (e.g., sleep, play, dance) that are activated by biological functions and external stimuli. The overall behavior of the SR is commanded by the motivations, i.e., whenever a motivation becomes dominant, the respective action is triggered. As in Mitchinson and Prescott (2016), this approach demonstrates the potential of bio-inspired architectures to generate natural behavior in SRs. However, the architecture remains primarily focused on detailed internal regulation of the robot, treating the user mainly as a passive source of stimuli, which may limit the extent to which interactions can become reciprocal and personalized.

The gaze control system by Zaraki et al. (2014) integrates bio-inspired mechanisms related to attention and gaze control. The discretized zones for proxemics and field of view are defined based on psychological models and human eye anatomy, respectively. Human behavior also informs the values assigned to head movement latency and the habituation effect, i.e., the decline in attention to repeated stimuli. Moreover, the process of determining the attention target based on EA (i.e., how much the robot should gaze at each human) is inspired by human attentional processes.

Similarly, the gaze control system proposed by Aliasghari et al. (2020), based on Zaraki et al. (2014), incorporates bio-inspired elements, such as proxemics, human attention patterns, and a habituation effect that reduces attention to a selected target over time. However, the parameters of the equations were calibrated by analyzing human gaze behavior in controlled experiments, rather than relying primarily on bio-inspired models as in Zaraki et al. (2014). Furthermore, Aliasghari et al. (2020) introduced a blinking feature to make the gaze behavior more realistic.

The gaze control system proposed by Mishra and Skantze (2022) incorporates human social behaviors, such as turn-taking, gaze aversion (i.e., intentional gaze disengagement), and joint attention. The system also models human eye-head coordination patterns, where the decision to move both eyes and head versus eyes alone is based on planned gaze duration, mimicking natural human gaze behavior.

Zhu et al. (2024) proposed a gaze controller that integrates a model of head and eye movement as functions of muscle forces and physical properties similar to those of the human body. Furthermore, the control strategy for moving the head and eyes is designed to minimize neural noise (i.e., fluctuations in brain activity), mirroring the behavior observed in humans.

Taken together, these gaze-control systems (Aliasghari et al., 2020; Mishra & Skantze, 2022; Zaraki et al., 2014; Zhu et al., 2024) demonstrate how bio-inspired mathematical modeling can significantly increase the naturalness and realism of gaze behavior, yet they remain highly specialized components that require extensive manual design and are limited to controlling gaze rather than broader social behavior.

General appraisal of bio-inspired approaches

- Bio-inspired approaches in social robotics often shape the structure of control architectures, where the overall organization and coordination of modules are modeled after biological systems (Lazzeri et al., 2018; Maroto-Gómez, Malfaz, Castro-González & Salichs, 2023; Mazzei et al., 2014; Mitchinson & Prescott, 2016; Patacchiola & Cangelosi, 2020; Tani et al., 2020).

- These approaches can also guide the design and implementation of specific components that emulate biological functions, such as circadian rhythms, endocrine regulation, motivational drives, or learning mechanisms (Aliasghari et al., 2020; Maroto-Gómez, Malfaz, Castro-González & Salichs, 2023; Mishra & Skantze, 2022; Mitchinson & Prescott, 2016; Patacchiola & Cangelosi, 2020; Rodríguez-Lera et al., 2018; Tani et al., 2020; Zaraki et al., 2014; Zhu et al., 2024).

- Additionally, bio-inspired mechanisms are frequently added as standalone elements to complement existing controllers and improve behavioral realism, for example, through gaze control, blinking, or habituation (Aliasghari et al., 2020; Lazzeri et al., 2018; Mishra & Skantze, 2022; Mitchinson & Prescott, 2016; Zaraki et al., 2014).

- Overall, bio-inspired methods can enhance the naturalness and realism of robot behavior. By incorporating principled mechanisms and patterns inspired by human and animal biolog biology, these approaches enable robots to exhibit more lifelike, context-sensitive, and adaptive behaviors, such as responding to internal states and external stimuli. These behavioral characteristics may positively influence user perception and acceptance and, consequently, support interaction quality and sustained engagement (Cuijpers & Knops, 2015; Li et al., 2023), though empirical validation of these effects remains limited and context-dependent.

- Despite these benefits, bio-inspired approaches introduce additional complexity and require extensive manual parameter tuning.
- Bio-inspired approaches are most often used in a complementary role, where biological mechanisms inform the design of architectures, components, or behavioral enhancements, while implementation typically relies on other paradigms such as mathematical modeling, rule-based systems, or machine learning.

Homeostatic control

Homeostatic control approaches use explicitly defined regulatory mechanisms to maintain internal system variables within desired ranges. These models are based on the concept of homeostasis from biology but are formalized using internal drives, set points, or control loops. Control actions are selected or modulated based on the current state of internal variables relative to their target values (Cao, Beir et al., 2017).

Although homeostatic control is rooted in biological principles and is in fact a bio-inspired approach, we have treated it separately here, due to its distinct focus on internal regulation mechanisms rather than broader biologically inspired cognitive or behavioral modeling.

Cao, Beir et al. (2017) developed a homeostatic controller for SRs in social therapies, where therapists configure needs and drives of the robot through a graphical interface to generate semi-autonomous interactive behaviors. The controller relies on a set of internal needs of the robot, including social requirements, that are homeostatically regulated. Whenever the value of a need is out of its desired range, it activates a drive, i.e., a state that signals the lack of equilibrium and can trigger an action to reestablish that equilibrium. Each drive can be influenced by several needs, and for each need-drive pair, one coefficient regulates how much the deficit of each need influences each drive. In this architecture, the drive with the highest value is chosen and activates its corresponding drive satiator, which contains a prioritized list of actions with associated feasibility conditions. The final action is then selected as the highest-priority action whose conditions are satisfied. This demonstrates how homeostatic regulation can be implemented for an interpretable, therapist-configurable system where robot actions are selected based on internal drive dynamics.

The control architecture proposed by Maroto-Gómez, Malfaz, Castro-González and Salichs (2023) based on modeling the neuroendocrine system, biological processes, and motivations also includes regulatory homeostatic mechanisms for decision-making. Each robot action is associated with a motivation that is influenced by biological function deficits, current biological states, and external stimuli. The deficits in biological functions corresponds to the deviation from their current value to their ideal value, and since they trigger related motivations, a self-regulatory homeostatic mechanism emerges. Each motivation is considered active if it surpasses a pre-defined threshold, and in case multiple motivations are active, only the most intense motivation is executed. Similarly to Cao, Beir et al. (2017), this architecture applies homeostatic control to action selection, resulting in life-like and dynamic behavior. However, the complexity of manually tuning biological parameters may limit practical deployment

across diverse interaction settings.

The brain-inspired architecture proposed in Tani et al. (2020) for supporting ToM reasoning in SRs includes a homeostatic mechanism within the NNs that implements two brain regions (namely, the precuneus/posterior cingulate cortex and the anterior cingulate cortex). This mechanism aims to maintain neuronal stability during NN training. In this case, homeostatic regulation plays a supporting role by maintaining neural stability during learning, rather than driving behavior selection as in the other approaches reviewed in this section.

General appraisal of homeostatic control

- Homeostatic control, a specific bio-inspired approach, is primarily used in social robotics for decision-making (Cao, Beir et al., 2017; Maroto-Gómez, Malfaz, Castro-González & Salichs, 2023).
- By designing architectures based on internal regulation of internal states, these methods enable robots to automatically generate complex and adaptive behaviors without requiring explicit rule specification for every situation. Such adaptability may be particularly valuable for sustaining engagement in long-term HSRIs (Matarić & Scassellati, 2016).
- Furthermore, grounding the homeostatic regulation in relevant biological variables leads to lifelike and biologically plausible behavior. As discussed in Section 5.3.3, this lifelike behavior may positively influence user perception, trust, and acceptance, thereby supporting interaction quality in HSRIs (Li et al., 2023).
- Despite the benefits mentioned, tuning and validating homeostatic models across diverse scenarios can be challenging. Their reliance on predefined drives and thresholds may limit their flexibility and generalization, especially in highly dynamic social contexts.

Model-based control

Model-based control approaches use an explicit model of the system or environment to simulate or predict outcomes for planning or control, in order to optimize certain aspects of the interaction. By incorporating forward simulation into the control loop, model-based approaches enable action selection based on anticipated consequences.

Inspired by ToM, Gray and Breazeal (2014) proposed an approach to influence user mental states by modeling and manipulating how perception updates beliefs about objects and their properties. For each user, the SR creates a copy of the world model to represent the world as viewed by that user. This model is updated by geometrically simulating the user's visual perspective based on their position, viewing angle, and occlusions, allowing the robot to infer the beliefs of the user. Then, the SR uses this model to look ahead in the future and check how each possible robot action is expected to impact the real world and, consequently, the beliefs of the user. Since the simulations evaluate potential actions without physically executing them, a simplified world model is used to simulate how actions may affect the environment and the

beliefs of users. Having a predefined goal (e.g., wanting all users to have a particular belief), all world models are checked to find the ones where this goal is true. Then, the sequence of actions that were applied to that virtual world is selected as the robot's action plan. The approach was evaluated through an experiment where participants watched videos of the robot playing a competitive game against a human opponent and evaluated the robot's social skills. In the game, both players place objects in their goal areas and may hide part of their movements. The robot wins if the two objects are different; otherwise, the opponent wins. This framework represents an early and promising application of predictive control in social robotics, using embodied simulation to model and influence user mental states. However, the framework relies on a deterministic representation of the physical world, and the mental states are limited to perceptual beliefs about object properties. This constrains its applicability to physically grounded scenarios and limits its extension to social or affective mental states. Nonetheless, the approach provides a valuable foundation for enabling SRs to reason about users' belief-like mental states.

J. J. Lee et al. (2019) presented a non-verbal communication framework for a speaker and a listener SR in a storytelling context, inspired by ToM. The framework intentionally models the speaker and listener as abstract interaction roles, without specifying whether they are humans or robots. For example, in the experimental validation, these roles are instantiated as a human storyteller and a robot listener. Formulated using probability models, which are further discussed in Section 5.3.4, both the listener and speaker act based on the estimated mental states of the other. The speaker infers the engagement level of the listener when choosing non-verbal cues. The listener, who aims to convince the speaker that they are paying attention, models the speaker's belief regarding their own attention. The speaker controller was tested in an offline experiment, where its performance was assessed using human–human interaction data. It outperformed baseline models that lacked action-conditioned transitions or observations, highlighting the importance of modeling the speaker's influence on state and response when selecting actions. The listener controller was tested in a HSRI with the Tega robot, where the speaker was a human child. The proposed controller was compared with a baseline approach that signaled engagement using only high-positive responses and without belief estimation, and it outperformed the baseline in terms of listening skills, based on comparative questionnaires. Although this framework is highly task-specific, it demonstrates the value of predicting user mental states when selecting actions, reinforcing the potential of model-based control to improve social awareness in SRs.

The gaze control system proposed by Zhu et al. (2024) employs a model-predictive controller that integrates a state-space model of eye and head movements of the robot. In models of biological motor control, neural signals are inherently noisy, and the variability of this noise increases with the amplitude of the control signal. Inspired by this principle, Zhu et al. (2024) models neural noise as a quadratic function of the control input. Then, the gaze controller is designed to minimize this noise while steering the robot's gaze toward a target, in order to promote smooth and stable gaze behavior. In contrast to the ToM-based approaches, this work applies model-predictive control to optimize robot motor dynamics rather than user mental states, showing how

model-based control can enhance natural behavior in narrowly focused tasks.

Similarly to Gray and Breazeal (2014), the work presented in Chapter 2 proposed a ToM-based framework to represent and influence the mental states of users. Unlike Gray and Breazeal (2014), the model proposed in Chapter 2 extends beyond beliefs to also include goals, emotions, and biases, which are essential for the complex social interactions in which SRs partake. Furthermore, this model explicitly captures the dynamics of mental states and supports personalization. In Chapter 3, the model is integrated in a model-based controller to predict the effect of SR actions on the mental states of users, and choose actions that bring those states closer to desired ones. While Gray and Breazeal (2014) applied this mechanism for belief deception in a competitive scenario, Chapter 2 focused on socially assistive interactions, aiming to promote beneficial states of the users such as engagement and comfort. By modeling a broader set of mental states and predicting their evolution, this approach illustrates how model-based control can support goal-oriented social assistance and sustain longer-term adaptive interaction.

General appraisal of model-based control

- Model-based control approaches are an emerging trend in social robotics, with increasing adoption in recent years.
- Most approaches discussed in this section leverage explicit models of internal mental states of users to predict and influence these states through the robot's behavior (Gray and Breazeal (2014); J. J. Lee et al. (2019) and Chapter 2). Grounded in ToM, these approaches enable the robot to plan sequences of actions to achieve social objectives or interaction goals, similarly to how humans reason. Despite sharing this principle, these methods differ in the complexity of the models, types of mental states represented and influenced, and being static or dynamic (i.e., presenting the evolution of mental states as well).
- Only one model-based control approach reviewed in this section, i.e., Zhu et al. (2024), models a component of the robot rather than the user. In particular, it applies model-based control to optimize motor dynamics and to achieve human-like natural behavior.
- While model-based control approaches require accurate models to ensure good performance, they enable predictive, goal-directed robot behavior that goes beyond reactive control. Importantly, approaches that model and influence human mental states create the perception that the robot understands the user, which has been highlighted as essential for maintaining long-term engagement in HSRI (Matarić & Scassellati, 2016).

5.3.4. Probabilistic Approaches

This category includes control approaches that use probabilistic models to represent uncertainty in perception, state estimation, or decision-making by Social Robots (SRs).

These models define probability distributions over variables and use inference or optimization techniques to guide robot behavior.

The controller proposed by M. H. Lee et al. (2023) for SRs coaching physical therapy for post-stroke patients uses Gaussian probability density functions to model the movement patterns of the unaffected side of the patients, enabling personalized assessment of the affected side. The parameters of the Gaussian distribution (mean and standard deviation) are estimated using Maximum Likelihood Estimation based on measurements from the unaffected side. The estimated parameters are used to personalize the threshold values in the rules of a rule-based model that assesses the quality of movement, as explained in Section 5.3.1. This probabilistic approach accounts for inter-user variability by adapting rule-based thresholds and, thus, supports controller personalization. Nonetheless, it is intentionally limited to operate within a broader control architecture, and it steers robot behavior only indirectly through its influence on movement assessment.

The gaze control system proposed in Zhu et al. (2024) uses an online Gaussian Process observer for disturbance prediction and compensation in the control system. The Gaussian Process predictions are then fed into a model predictive controller for proactive compensation (detailed in Section 5.3.3). By accounting for uncertainty in disturbances, this approach improves robustness and naturalness of gaze control, although its contribution is limited to a specific subsystem rather than broader social behavior.

Bettosi et al. (2024) presented a four-step method to generate SR behavioral models (i.e., structured representations of robot behavior) from human experts in assistive contexts. In their case studies, the behavioral models were implemented using transition matrices. Transition matrices are square matrices used to describe the probabilities of a system moving from one state to another. In this case, the states correspond to high-level interaction behaviors, and the transitions encode the likelihood of moving from one robot behavior to the next during an assistive interaction. The use of a probabilistic approach based on transition matrices yields an interpretable representation of SR behavior flow grounded in expert demonstrations that can be directly used for action selection. However, transition matrices model only local behavior-to-behavior transitions and do not capture longer-term temporal dependencies or hierarchical structure in social interaction. Furthermore, they provide limited mechanisms for adapting robot behavior to user states.

In Patacchiola and Cangelosi (2020), the intrinsic environment used to provide a cost to the actor-critic Reinforcement Learning (RL) environment in the *Thrive* architecture (see Section 5.3.2 for details) is implemented as a Bayesian Network (BN). BNs are graphical models that represent probabilistic relationships between random variables, allowing reasoning under uncertainty by encoding dependencies in a directed graph. In *Thrive*, the BN computes the cost of keeping the current belief (as estimated by the actor) based on the confidence of this belief, the reputation of the informant, and whether the belief and suggestion match. Although the conditional probabilities within the BN are manually designed based on psychological theory, the authors show that by defining different conditional probabilities, they can reproduce trust learning behaviors that are consistent with empirical data from children at different levels of maturity.

Here, probabilistic reasoning supports trust evaluation under uncertainty, but remains a secondary component rather than the primary driver of the robot behavior.

J. J. Lee et al. (2019) presented speaker and listener controllers based on Theory of Mind (ToM) for storytelling contexts, enabling each to model their peer's mental state to improve non-verbal communication. The speaker's controller is formulated as a Partially Observable Markov Decision Process (POMDP) to both infer and influence the attentive state of the listener. A POMDP models decision-making when an agent has incomplete information about the world, maintaining beliefs over hidden states based on observations, and choosing actions to maximize expected long-term reward. In the speaker's case, the hidden state represents whether the listener is paying attention, and the action set includes gazing, giving weak/strong cues, and a virtual action that signals goal achievement (needed for training, but disabled during real-time interaction). Observations consist of discrete listener responses (e.g., nods, smiles, gaze), categorized into five valence levels ranging from negative to positive. The transition and observation functions model how states and responses dynamically evolve as a function of actions. These functions are learnt from human demonstration using an approach based on maximum likelihood estimation that iteratively refines their parameters to explain observed behavior. The reward function is manually designed to represent the social cost of taking each action in each state, and the policy results from applying a softmax function to the state-value function. The listener aims at making the speaker believe they are engaged (desired belief), and its controller consists of two parts: a Dynamic Bayesian Network (DBN), which infers the speaker's belief regarding the listener's attention; and a myopic policy selector, which chooses the listener response to display based on the DBN output. The DBN is based on ToM, and it corresponds to the inverted POMDP used to represent the speaker. This allows inferring the speaker's beliefs from their actions, based on the assumptions that the speaker wants the listener to pay attention and is inferring the listener's attentive state. Then, the myopic policy chooses the response expected to move the belief of the speaker to the desired belief, based on the current belief and the belief transition function. By enabling each agent to reason about uncertain mental states, the model supports adaptive and socially appropriate behavior. Its applicability, however, is largely constrained to the structured storytelling scenario for which it is designed.

General appraisal of probabilistic approaches

- Probabilistic approaches in social robotics serve diverse purposes, ranging from perception to decision-making.
- They are commonly used to represent uncertainty in perception (J. J. Lee et al., 2019; Zhu et al., 2024), to estimate task-related indicators of users (M. H. Lee et al., 2023), to model behavioral transitions and interaction patterns (Bettosi et al., 2024), to infer internal variables (J. J. Lee et al., 2019; Patacchiola & Cangelosi, 2020), and to support decision-making under uncertainty (J. J. Lee et al., 2019; Patacchiola & Cangelosi, 2020).
- By incorporating uncertainty and stochasticity into control, probabilistic approaches can support more robust and adaptive behavior compared to purely

deterministic, rule-based methods. This capability is particularly relevant in social contexts, where user states and environmental conditions are inherently variable.

- Similarly to the mathematical modeling approaches discussed in Section 5.3.3, probabilistic methods typically operate as supporting components within hybrid architectures rather than as standalone controllers.

5.4. Discussion and Synthesis

Section 5.3 demonstrated that although numerous control approaches exist for generating SRs behavior, many are not well aligned with the demands of real-world, long-term interactions. The previous section examined the strengths and limitations of each approach in detail, revealing important differences in their underlying assumptions, adaptability, and suitability for dynamic Human-Social-Robot Interactions (HSRIs). These analyses also highlight that most existing control methods are insufficient to enable sustained, socially appropriate interaction on their own. Many existing approaches perform well only within narrow domains or for short-term structured tasks, as they rely on rigid control logic or simplified representations that limit personalization and adaptability over time.

These findings motivate a broader synthesis that moves beyond evaluating approaches in isolation and instead examines how effectively they address the practical demands of real-world interactions, where robot behavior should adapt to evolving mental states of users and long-term engagement requirements. Therefore, this section focuses on the suitability of different control methods across interaction time frames, functional roles, and application contexts. Additionally, it explores how complementary strengths of different methods may be combined to overcome current limitations and enable richer, more adaptive future SRs capable of sustained long-term social interactions.

5.4.1. Suitability of Control Approaches by Interaction Length

The distinction between short-term and long-term interactions is particularly relevant for SRs, as interaction duration strongly influences the requirements placed on robot behavioral control. Controllers that are effective in brief, goal-oriented interactions with humans may become inadequate when interactions extend over time and require sustained user engagement.

Short-term interactions are typically associated with specific purposes or well-defined tasks. Examples include distracting children during medical procedures or providing information to visitors in public spaces, such as museums, hospitals, or shopping centers. In these scenarios, interactions are limited in scope and duration, and the novelty effect generally persists long enough to maintain user engagement without a need for highly sophisticated control mechanisms. Consequently, investing in advanced behavioral control methods is often unnecessary in such contexts, since simpler, task-specific controllers already enable the robot to accomplish its intended goals effectively.

For such short-term interactions, simpler controllers are generally adequate. Rule-based and heuristic approaches are well suited due to their transparency, interpretability, computational efficiency, and ease of development when the state and action spaces are small. Similarly, RL can be used for decision-making in these interactions by learning policies that directly map environmental inputs to robot actions. However, in these simple settings, RL typically offers limited practical advantages over rule-based solutions of comparable complexity. Since the interaction dynamics are typically simple and the state and action spaces are small, effective behaviors can often be encoded directly through rules without requiring extensive data-driven policy learning. Moreover, RL introduces additional challenges related to explainability, training data requirements, and safety during learning. These concerns are largely unnecessary when rule-based approaches already achieve the desired functionality. Classification and clustering methods can also support the design of controllers for short-term interactions by structuring behavior as sequences of actions learned from expert demonstrations.

Nevertheless, behavioral control systems designed for short-term interactions exhibit important limitations when applied beyond their intended scope. Their reliance on predefined structures or policies learned from limited training settings often leads to rigid, repetitive behavior that lacks adaptability and personalization, which can cause user engagement to decline over time. Thus, when used in isolation, these approaches are poorly suited for long-term interactions.

By contrast, long-term interactions impose substantially higher demands on SR behavioral control. Sustained engagement requires sensitivity and responsiveness to evolving context, awareness of and appropriate responses to dynamic user states (including desires, emotions, and intentions), and mechanisms for incorporating relevant information from previous interactions.

Based on these considerations, more advanced control approaches are better suited for long-term HSRIs. Cognitive and behavioral architectures, model-based control, mathematical modeling, and probabilistic approaches provide mechanisms for structured reasoning, long-term adaptation, and integration of diverse information sources. Learning-based methodscan further support these approaches by offering adaptability and learning from interaction data, provided that their deployment is embedded within a broader framework that guarantees safety and interpretability. When properly integrated, these approaches can address the limitations of controllers designed for short-term use and generate robot behavior that evolves according to the user states and interaction context.

Building on this perspective, Sections 5.4.2 and 5.4.3 focus exclusively on long-term interactions and examine how the suitability of control approaches varies depending on their functional role within the controller and the application domain in which they are deployed. Figure 5.3 provides a visual overview of this assessment across functional roles and application domains.

5.4.2. Suitability of Control Approaches by System Role

The suitability of a behavioral control approach is not absolute. Instead, it depends on the role the approach plays within the overall behavioral control system of the

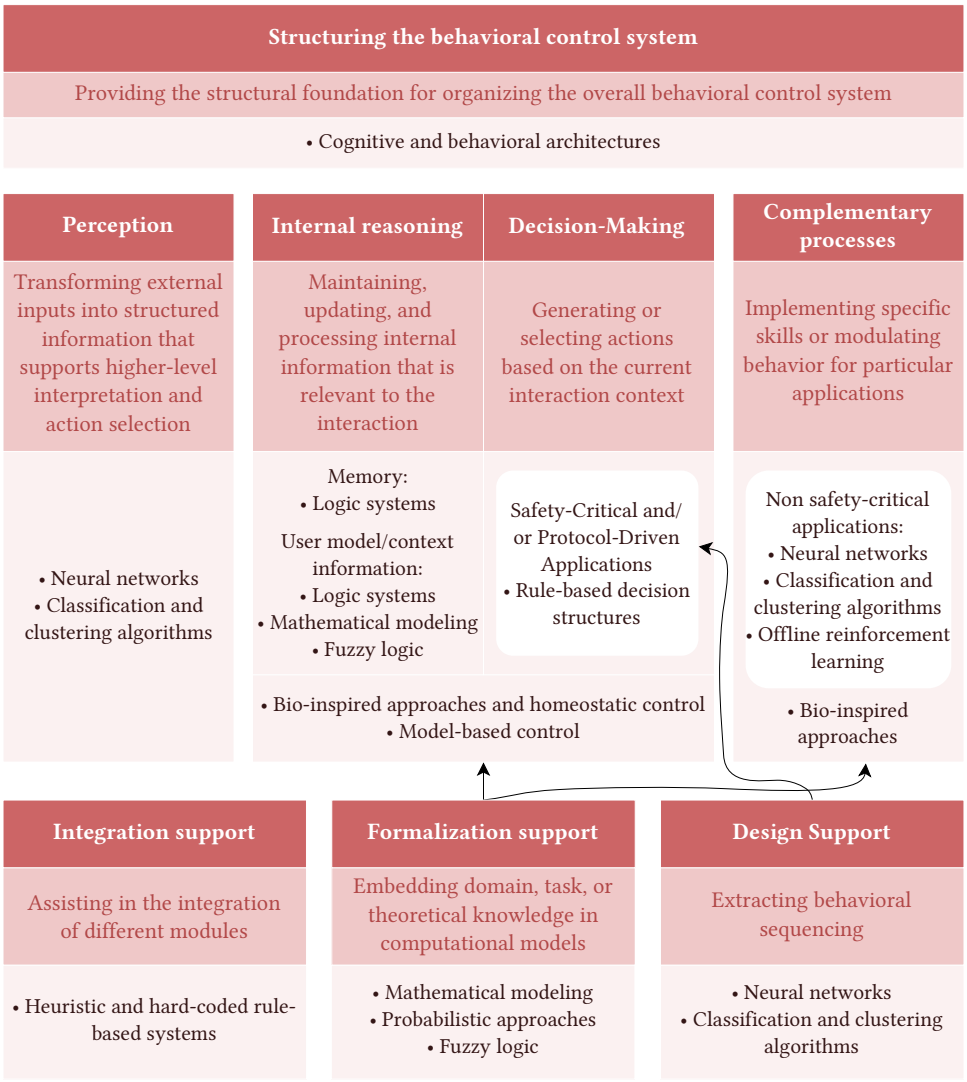


Figure 5.3: Conceptual overview of the suitability of different control approaches for long-term HSRIs. The figure summarizes the role-based suitability discussed in Section 5.4.2 and also reflects application-based considerations discussed in Section 5.4.3, where relevant.

robot. A control approach that fails when used as a primary mechanism for behavior generation may be highly effective as a complementary component within the control system (e.g., perceptual processing or context interpretation). Moreover, some methods are appropriate only for very specific and narrow roles, whereas others are more flexible and can support multiple components of the overall behavioral control system. Thus, the key question is which control approaches are suitable for which roles within the overall behavioral control system of SRs to support sustained interaction over time.

To address this, we distinguish between three major categories of roles for control approaches. The first concerns approaches that define the structure of the overall behavioral control system and specify how different components are organized and integrated. The second includes approaches suited for implementing functional components of the behavioral control system, such as perception, internal reasoning, and behavior generation or selection modules, as well as complementary processes that enable the robot to perform socially appropriate cues or task-relevant skills. The third concerns approaches that play a design-support or formalization role, without directly generating robot behaviors during HSRIs. In particular, this role includes methods used to embed theoretical or domain knowledge into the behavioral control system, support component integration, and extract patterns from data to inform decision-making design.

The discussion that follows argues which methods are suitable or unsuitable for each of the identified roles and how different methods may be combined to enable long-term adaptive interaction. Figure 5.3 summarizes this role-based perspective for long-term interactions, organizing control approaches into three levels that correspond to the role categories introduced above.

Approaches for structuring behavioral control systems

Before discussing the suitability of different approaches for specific functional roles within the overall behavioral control system, it is important to establish the structural perspective needed to guide the definition of such roles.

Cognitive and behavioral architectures should provide the structural foundation for organizing the overall behavioral control system, integrating heterogeneous modules, and supporting long-term adaptation. These architectures enable behavior generation in a principled and psychologically grounded way, moving beyond simple input–output mappings to support richer reasoning and decision-making processes. Furthermore, they allow task-specific modules, such as therapeutic routines, educational activities, or assistive behaviors, to be incorporated into the architecture and shaped according to changing user needs and interaction contexts. Altogether, these characteristics make cognitive and behavioral architectures well suited as the central organizing framework for long-term interactions, within which more specialized approaches can be positioned appropriately rather than being deployed as isolated ad-hoc components.

Approaches for implementing functional roles in behavioral control

While architectural frameworks determine the overall organization and coordination of the control system, the practical generation of behavior relies on selecting appropriate control approaches for different functional roles such as perception, reasoning, decision-making, and complementary processes. The proposed roles are grounded in recurring functional demands in SR behavioral control, as observed throughout Section 3, and provide a role-based lens for comparing approaches. In this section, we discuss which approaches are best suited for implementing each role and which commonly adopted methods fall short when used for these roles in long-term adaptive interactions. Importantly, the intent is not to prescribe a detailed architectural decomposition, but

to assess which control approaches are best suited to implement each functional role, considering the requirements of long-term HSRIs.

Perception Perception refers to the process through which raw sensory inputs from the environment are transformed into structured information that supports higher-level interpretation and action selection. In HSRIs, perception typically involves detecting and identifying relevant social cues, such as speech, emotional expressions, gestures, gaze direction, or the identity and physical actions of users.

Current methods employed for perception are well suited for this role, as Neural Networks (NNs) and classification and clustering algorithms have been widely used for perceptual processing in SRs with effective results. NNs are particularly suitable for perception because they transform raw, high-dimensional sensory data, such as audio signals, images, or motion trajectories, into compact, structured representations that can be leveraged by internal reasoning, decision-making, and complementary components. Similarly, classification and clustering methods map these representations into discrete groupings (e.g., emotion labels, gesture classes), making the perceived information more interpretable and actionable for other components.

In contrast, rule-based, symbolic, modeling, or probabilistic approaches typically rely on predefined rules, logic, or knowledge, or explicitly modeled relationships and therefore struggle with the high-dimensionality and ambiguity of raw sensory inputs. Consequently, these approaches are generally unsuitable for perceptual processing. However, these approaches become highly valuable after perception, once information has been converted into structured representations. At that stage, the aforementioned approaches can support internal reasoning, decision-making, and other complementary processes.

Internal reasoning Internal reasoning refers to the processes through which the robot maintains, updates, and organizes internal information that is relevant to the interaction (including knowledge about the user, task, and environment), and may involve inferring higher-level concepts that support decision-making and other complementary modules.

Our objective is to examine the suitability of different control approaches for implementing the internal reasoning of the robot. However, the processes encompassed by internal reasoning are inherently diverse, and the literature presents a broad range of cognitive architectures that differ both in the specific subprocesses they include and in how these subprocesses are organized and interconnected (see Section 5.3.3). As proposing or advocating for a specific architecture lies beyond the scope of this work, we instead identify a set of high-level components that capture core functional roles of internal reasoning in long-term HSRIs. This level of abstraction enables a systematic analysis of how different control approaches can support these roles, independently of particular architectural implementations.

In line with this abstraction, to support long-term and socially meaningful HSRIs, internal reasoning should include environmental context, a user model that reflects individual characteristics and preferences, and memory of past interactions, which allows the robot to maintain coherence across interactions over time. Furthermore,

when applicable, the internal reasoning should include domain knowledge such as therapeutic or educational frameworks.

Approaches that support explicit and structured knowledge representation are particularly well suited for implementing the internal reasoning of the robot. In particular, logic-based systems allow encoding and processing domain information through interpretable structures. This makes them a strong candidate for maintaining user models, long-term memories, and internal knowledge bases. The capacity of logic-based systems to represent nested belief states also makes them well aligned with implementing core elements of ToM, such as explicit reasoning about others' beliefs in social interactions. Fuzzy logic extends these capabilities by handling linguistic uncertainty and continuous variations in human-related variables (e.g., intensity of emotions, level of engagement, degree of agreeableness), both typical of human behavior. In doing so, fuzzy logic offers representations that more closely resemble human reasoning patterns. Additionally, mathematical modeling and probabilistic approaches can be used to represent theoretical principles or relationships between concepts, which are often inspired by cognitive or affective theories. This yields internal reasoning processes that are typically grounded in established models from psychology or neuroscience while remaining computationally efficient.

Decision-making Decision-making refers to the process through which the robot generates or selects actions based on the current interaction context. In long-term HSRIs, decision-making should support behavior that is coherent and adaptive over time, rather than rigid or repetitive.

Rule-based decision structures can effectively organize SR behavior in domains with clearly defined protocols (e.g., rehabilitation sessions or educational tasks), since they impose an explicit structure on robot behavior by mapping well-defined conditions (e.g., task completion, user performance) to appropriate actions or state changes. However, outside such constrained contexts, rule-based decision structures may be excessively rigid and lead to repetitive and non-adaptive robot behavior. In turn, heuristic and hard-coded rule-based systems rely on hand-crafted rules, typically without systematic organization of decision logic. This often results in purely reactive behavior (i.e., actions triggered by stimuli without coordination or planning), which may lead to more rigid behavior compared to rule-based decision structures. Consequently, both structured and heuristic rule-based approaches become inadequate for implementing decision-making beyond short-term interactions and narrowly defined tasks. Nevertheless, rule-based decision structures may remain appropriate for highly structured sessions due to their explicit flow organization. However, in these cases, they should be combined with adaptive strategies, especially in long-term interactions.

In social robotics, RL is widely used for decision-making. In particular, it is commonly used for learning to select robot actions directly from environmental inputs. In principle, RL appears suitable for this task due to its conceptual alignment with how humans and animals learn policies that yield the optimal action per state through trial-and-error interactions. Nevertheless, in practice, the application of RL in social robotics is prone to significant limitations. Learning effective policies for complex interaction requires numerous episodes, which is impractical to achieve with human

users in the loop. Furthermore, this extended trial-and-error learning is likely to negatively impact the engagement or satisfaction level of users and may expose them to inappropriate or unsafe behaviors by the robot. Simulation-based training may mitigate such issues, but it requires high-fidelity world models, since even small model inaccuracies can strongly bias learning and degrade the performance of the learnt policy. Modeling human social behavior with such precision is particularly challenging, which may explain why the literature reviewed in this survey does not report any applications of simulation-based training for RL. Finally, offline (batch) RL can alleviate some of the limitations of online RL, but only when annotated datasets are available, which inherently restricts the action space and the scope of actions that can be learned. As a result, offline RL is typically applicable only to well-defined aspects of behavior or to internal components of the control architecture that do not directly generate robot actions, rather than to full decision-making in open-ended interactions.

Therefore, most RL controllers used for decision-making of SRs are limited to short-term interactions with small, simplified action and state spaces, thereby producing behaviors with little adaptability. However, such behaviors can typically be implemented using rule-based and heuristic approaches, where these methods offer improved transparency and explainability over RL. All in all, current evidence suggests that RL is unsuitable as the primary decision-making mechanism for long-term interactions. Instead, its effective use in SRs is confined to supplementary roles (e.g., handling specific action dimensions or supporting internal reasoning components).

Taken together, the limitations of rule-based decision structures, hard-coded rule-based systems, heuristics, and RL demonstrate that decision-making based on immediate inputs without processing them through internal reasoning remains mostly reactive, which may lead to rigid and non-adaptive robot behavior. Accordingly, to generate behavior that adapts to the user and the interaction context, decision-making should be informed by internal reasoning processes that capture user models, contextual knowledge, and interaction history. Consequently, this requirement favors approaches in which decision-making is grounded in, or is tightly coupled with, internal models and reasoning dynamics, rather than operating as an isolated stimulus–response mechanism.

Integrated internal reasoning and decision-making Approaches that integrate internal reasoning and decision-making typically generate behavior as a function of internal states that describe the robot, the user, and the interaction context, rather than as a function of immediate perceptual input. Additionally, in such integrated approaches, internal representations directly shape action selection, without requiring a strict separation between reasoning and decision-making. This tight coupling enables behavior to emerge from the current values of these internal states, thereby providing a natural mechanism for adaptation and personalization.

Bio-inspired decision-making architectures draw on concepts such as drives, motivations, and homeostasis to couple internal state modeling and/or regulation directly with behavior selection. In these systems, actions are influenced by internal variables (e.g., motivational priorities, simulated physiological states) that are computed from sensory input, interaction history, and bio-inspired regulatory mechanisms,

thereby going beyond simple stimulus-response. This provides a mechanism for generating coherent, adaptive behavior over time that follows from the robot's physiological and motivational state and from relevant aspects of the interaction context. Moreover, by grounding behavior generation in principles inspired by human and animal biology, these architectures often promote robot behavior that appears more natural and socially plausible for users.

Model-based control integrates internal reasoning and decision-making by explicitly incorporating models of the interaction, the user, or the robot's own state, and by selecting actions that optimize given objectives defined as functions of expected outcomes. In these approaches, robot behavior is not chosen solely in response to current inputs and states, but also informed by how robot actions are expected to affect future states of the user, the robot, and the interaction. This tight coupling between modeling internal states and action selection makes model-based control particularly suitable for social interactions, where appropriate behavior often depends on evolving states that reflect the interaction context and longer-term goals, rather than on immediate observations alone. By grounding decision-making in predictive models, model-based control enables more deliberate, context-aware robot behavior. This directly addresses the limitations of reactive approaches, including rule-based systems, heuristic approaches, and RL as commonly applied for decision-making in social robotics, in long-term, real-world interactions.

Complementary processes Beyond perception, reasoning, and decision-making, SR controllers often require complementary modules that implement specific skills or modulate behavior for particular applications. These modules do not determine high-level behavior, but shape how robot actions are performed or adapted to task and interaction requirements. For example, rehabilitation or educational settings may require dedicated components for demonstrating movements or for emulating learning behavior. Moreover, behavioral realism may be increased by including modules that implement gaze control or physiological functions (e.g., blinking, breathing).

A range of approaches can be appropriate for implementing such complementary processes, depending on their purpose. Heuristic and hand-coded rule-based methods can be effective when behaviors are simple and tightly constrained to a specific task. NNs, classification methods, and clustering methods, may be appropriate for tasks that require learning patterns from high-dimensional data, such as reproducing movements from demonstration or emulating learning skills. Finally, bio-inspired mechanisms are particularly suited for processes that aim to enhance behavioral realism and social acceptability of SRs through the display of physiological functions. These considerations serve as guidelines, rather than prescriptions, for selecting approaches to implement complementary processes, as suitability ultimately depends on the application and purpose.

Approaches for supporting design, integration and formalization

Other control approaches may contribute effectively to behavioral control in SRs, even if they do not directly fit within a specific architectural role, such as perception, internal

reasoning, or decision-making. Instead, these approaches may support behavioral control by either informing the design of the controller, integrating heterogeneous components, or providing formal tools for implementing specific mechanisms of the controller.

Heuristic and hard-coded rule-based systems may be required to support the organization and coordination of heterogeneous controller components. While cognitive architectures define the structure of the different modules, heuristic and hard-coded rule-based systems are often necessary to connect these components in practice. For example, they may arbitrate between competing outputs or handle exceptional cases that fall outside the operating conditions of the main controller. In this role, the usage of heuristic and hard-coded rule-based systems is acceptable, as they contribute to system-level coherence rather than directly shaping robot behavior.

Classification and clustering approaches are commonly used to extract structure from data, such as identifying typical sequences of movements or interaction patterns from human demonstrations. Thus, they are suited to inform the selection of behavior repertoires or the organization of interaction flow, rather than directly generating robot behavior online. This makes classification and clustering approaches particularly suitable for designing decision-making mechanisms in task-specific or protocol-driven applications.

Mathematical and probabilistic methods are frequently used as formal tools to implement mechanisms of the controller, such as representing internal states, modeling regulatory dynamics, or embedding uncertainty. Rather than constituting standalone controller components, they provide principled ways to translate theoretical or domain knowledge into computational processes embedded within the internal reasoning or decision-making modules. In this sense, mathematical and probabilistic methods bridge the gap between abstract theories and practical control of SRs.

5.4.3. Suitability of Control Approaches by Application

Beyond considerations related to the interaction length and functional role, the suitability of control approaches for SRs also depends on the application domain. Specifically, application-specific requirements, such as safety constraints, interaction structure, or domain-specificity, directly determine whether and to what extent an approach is appropriate for a given role.

Many SRs operate, often without continuous supervision, in contexts involving vulnerable individuals, such as in-home assistance for older adults, cognitive training for individuals with impairment, or interventions for children with autism spectrum disorder. In these settings, safety and transparency of robot behavior are paramount. This requires control approaches that are interpretable and can be reliably validated.

When safety and predictability take priority over adaptability and sustained user engagement, rule-based decision structures may be suitable approaches for controlling SRs, as they enable predictable and interpretable behavior. However, this comes at the cost of rigidity and limited adaptability inherent to most rule-based methods. Contrarily, RL is generally not appropriate in such settings, due to difficulties in validating and explaining its decision-making processes. Moreover, when used as the main controller, RL often yields simple controllers, which offer limited advantages over

more transparent rule-based alternatives. Similar concerns regarding validation apply to classification-based methods when used directly for behavior generation, unless their outputs are transformed into representations that are straightforward to inspect, such as rule-based decision structures or probabilistic models. NNs face comparable limitations when employed for direct behavioral control, as their internal decision processes are difficult to interpret and verify.

In contrast, cognitive and behavioral architectures are well suited for safety-critical applications, particularly in long-term interactions, as they support structured control, transparency, and the integration of explicit reasoning mechanisms. Mathematical modeling approaches, when used to implement well-defined processes grounded in domain knowledge, share similar advantages in terms of interpretability and predictability that make them appropriate for safety-critical settings.

Moreover, some applications of HSRI involve protocol-like sessions, where interaction follows a predefined structure, such as therapeutic exercises or educational lessons. In these cases, rule-based decision structures are particularly effective, as they explicitly encode the required flow of behaviors and ensure consistent execution of the protocol. Although the limited adaptability of rule-based methods is less problematic in such constrained scenarios, these approaches should still be complemented by adaptive mechanisms that can account for changes in the environment or user states over time.

Finally, some applications require domain-specific constraints or knowledge. In such cases, control approaches that allow explicit formulation of domain knowledge are particularly suitable. For example, in gaze control systems, mathematical modeling approaches are often well suited, as they encode well-established principles of social attention and coordination in a transparent and interpretable manner. Similar considerations apply to other specialized domains where robot behavior is governed by known interaction rules or dynamics.

5.5. Conclusions

This survey reviews control approaches used for behavioral control in Social Robots (SRs) and systematically analyzes how their underlying assumptions affect suitability across interaction durations, architectural roles, and application contexts. The analysis demonstrates that the suitability of control approaches for SRs cannot be evaluated in isolation but must be understood in relation to interaction duration, functional role of the control approach within the overall control architecture of the robot, and application context. Short-term, task-oriented interactions can be effectively supported by simpler methods, such as rule-based and heuristic approaches. In contrast, long-term interactions demand more sophisticated approaches that enable adaptation, personalization, and, ultimately, sustained user engagement. For such extended interactions, cognitive and behavioral architectures should provide the structural foundation, within which specialized methods fulfill distinct roles: Neural Networks (NNs) and classification for perception, logic-based and probabilistic approaches for internal reasoning, and model-based or bio-inspired methods for integrating internal reasoning and decision-making.

A key finding of this review is that approaches commonly employed as standalone

behavioral controllers for SRs (particularly Reinforcement Learning (RL) and rule-based and heuristic approaches) are unsuitable for primary decision-making in long-term, real-world interactions, though they may be used for implementing supplementary functions within larger architectures. Furthermore, application-specific requirements, especially safety and interpretability in interactions with vulnerable populations, impose additional constraints that favor transparent, validatable methods over black-box learning approaches.

By systematically reviewing how control approaches are used in social robotics, and evaluating their suitability for long-term interactions based on functional role and application context, this survey highlights the need to move beyond isolated control solutions and adopt principled integration of complementary approaches. Accordingly, future work should focus on:

- Developing cognitive and behavioral architectures that integrate specialized methods according to functional role: using learning-based approaches for perception, logic-based and probabilistic methods to support internal reasoning, and model-based or bio-inspired approaches for integrated internal reasoning and decision-making;
- Deploying model-based control frameworks that incorporate validated models of user mental states, enabling predictive and goal-oriented interaction while maintaining interpretability;
- Selecting control mechanisms that guarantee safety, interpretability, and ethical behavior in long-term deployments of SRs, particularly in applications involving vulnerable populations.

Ultimately, advancing such integrated frameworks is essential for achieving SRs that can successfully support meaningful, sustained social interactions with humans in real-world environments.

6

Discussions and Conclusions

6.1. From Promise to Practice: Revisiting the Motivation and Research Gaps

The combined effects of global population growth and demographic aging are rapidly increasing the number of individuals who require continuous social and healthcare support. At the same time, the available human and material resources are insufficient to meet this rising demand, placing increasing pressure on existing care systems. Social Robots (SRs) have emerged as a promising solution that may help alleviate part of the burden on social and healthcare systems, offering affordable, personalized, and complementary support to individuals through social interaction.

Nevertheless, for SRs to be effective and accepted by users in real-world settings, they should interact with users over extended periods while maintaining meaningful and engaging interactions. However, sustaining long-term user engagement remains a persistent challenge for SRs. The ability to maintain user engagement over extended periods depends on the robot's ability to generate adaptive, personalized, and context-sensitive behaviors that evolve over time and avoid repetitive interaction patterns. These characteristics are closely linked to the behavioral control strategies that govern the decisions and actions of SRs. Unlike humans, who continuously infer, track, and respond to the evolving mental states of their peers during social interactions, most state-of-the-art control approaches for SRs do not explicitly model the mental states and behaviors of users, nor do they account for the dynamics governing the evolution of these mental states. This, in turn, limits the adaptability and long-term effectiveness of current SRs.

Consequently, there is a pressing need for behavioral control frameworks that enable SRs to systematically infer, represent, and act upon user mental states. Such a framework should meet various objectives of the interaction, such as supporting both the personal goals of users and the higher-level objectives of the interaction, while respecting physical and social constraints.

This chapter reflects on how this thesis has addressed these challenges and evaluates the extent to which the proposed behavioral control framework advances the state-of-the-art. To this end, Section 6.2 revisits the research objective and research questions of this thesis, introduced in Section 1.2, and subsequently elaborates on the answers to research questions based on the findings of this thesis. Section 6.3 discusses the limitations of the proposed framework, outlines promising strategies to address them, and identifies relevant directions for future research. Finally, Section 6.4 provides concluding remarks that summarize the principal contributions of this work and reflect on its implications for advancing socially and cognition-aware SRs toward safe and sustained real-world deployment.

6.2. Revisiting the Research Objective and Questions

The previous section highlighted the challenge of sustaining user engagement in long-term Human-Social-Robot Interactions (HSRIs) and emphasized the need for SRs capable of representing user mental states and their dynamics, and systematically acting in accordance with those states. This challenge has been the central focus of this thesis and, as such, is articulated into a single overarching research objective.

Research Objective

Develop a context-independent behavioral control framework for Social Robots that models and leverages key mental states and behavioral elements of users to enable autonomous generation of meaningful, socially-aware, and interpretable robot behavior that sustains user engagement over the long term.

Achieving this objective required a sequence of interconnected steps: developing a model of human perception, cognition, and decision-making that can represent user behavior in HSRI, developing control and estimation mechanisms grounded in the model, and evaluating the complete framework in realistic interaction settings. To structure these methodological steps, six research questions were formulated.

6.2.1. Research Questions

The research questions, which were first presented in Chapter 1.2, are revisited here. Each question is presented in turn, together with a discussion of how the work carried out in this thesis addresses it.

The first research question concerned the identification of the theoretical foundations that informed the development of the model of human behavior.

Research Question 1

Which theoretical frameworks and behavioral or cognitive elements can provide a foundation for developing a comprehensive and context-appropriate model of human behavior in social interactions?

To address this research question, several theoretical frameworks were considered and some of them were adopted based on their relevance to social interactions, explanatory capacity for cognitive processes and mental state relationships, and feasibility for computational and mathematical formalization.

Theory of Mind (ToM) emerged as the most relevant theoretical foundation for the model proposed in this thesis, since it directly reflects the underlying principle that should guide decision-making by SRs in interactions with humans: be aware of, and reason about, the mental states of users. ToM emphasizes that other agents may hold beliefs that differ both from reality and from our own beliefs. This distinction provides the basis for modeling the perception and cognition of the observed agent as distinct from that of the observer agent. Since beliefs depend on an agent's perceptual access to real-life data, this distinction further provides theoretical grounding for explicitly modeling the perception process that underlies belief formation. Additionally, ToM establishes three essential requirements for SRs in social interactions: estimating mental states of users in real time, predicting how these states evolve and shape future behavior, and adapting the actions of the SR accordingly.

Furthermore, the Belief-Desire-Intention (BDI) framework provides an essential theoretical foundation for structuring the relationship between internal mental states and external actions. According to this framework, rational agents (including humans)

act by forming intentions based on what they believe about the world and what they desire to achieve. Within the model proposed in this thesis, the BDI framework complements ToM by providing a theoretical structure for decision-making, specifically by structuring the causal chain from beliefs and desires (or goals) to intentions, and finally to actions.

Other theories and frameworks, such as Theory of Planned Behavior (TPB) and Common Model of Mind (CMM), were also examined. However, they were not adopted to inform the development of the user model proposed in this thesis for the following reasons. TPB is predominantly employed in health psychology and intervention-driven settings to predict adherence to specific, goal-directed behaviors (e.g., smoking cessation or exercise adoption) and, therefore, offers limited suitability for modeling general-purpose social interactions. CMM is a comparatively novel framework with limited implementations to date and, as such, remains largely conceptual. Furthermore, its primary objective is to provide guidelines to integrate multiple abstraction levels (i.e., biological, cognitive, and motor) rather than guiding the modeling of each level, whereas the proposed model focuses specifically on the cognitive level.

In addition, the model proposed in this thesis incorporates a principle of rationality, which is often coupled with ToM. The principle assumes that, given their beliefs, rational agents select actions that are reasonable means to achieving their goals. In practice, this principle provides a baseline for modeling perception and decision-making as generally rational processes, while allowing perceptual constraints, emotions, biases, and individual traits to be represented as systematic deviations within the model.

While the aforementioned theories offer general-purpose principles for modeling human behavior, they are not tailored to social and assistive interactions. Thus, the model of human behavior proposed in this thesis was extended with emotions, biases, and personal characteristics, essential elements in such contexts. Emotions and biases, although not typically modeled in classical ToM and BDI, play a critical role in social interaction and complement the principle of rationality by accounting for deviations from purely rational behavior. Thus, they were included in the model. Finally, the model included slow-dynamics state variables, i.e., variables that capture personal characteristics that evolve slowly. These variables enable personalization and gradual improvement of the model as interactions occur and more information becomes available. The benefits of these variables are supported by the results presented in Chapter 2, which show that augmenting the model with emotions, biases, and a slow-dynamics state variable improves estimation accuracy of intentions and actions compared to a baseline model without these elements.

The identified theories and elements then served as the foundation for developing the user model, which is addressed in the second research question.

Research Question 2

How should the identified modeling frameworks, along with the behavioral and cognitive elements, be integrated and mathematically formulated to develop a model that captures the dynamic evolution of human behavior in social interactions?

To formalize the user model, human behavior is structured into three main modules: perception, cognition, and decision-making. This modularization improves the transparency, interpretability, and computational and mathematical tractability of the model by separating processes that play conceptually distinct roles. Following the same logic, perception and decision-making are modeled as a pipeline of subprocesses. In the perception module, real-world data is first filtered through perceptual access, then interpreted through rational reasoning, resulting in rationally perceived knowledge. Rationally perceived knowledge is the input to the cognition module, which updates beliefs, goals, and emotions of the user. Decision-making, inspired by the BDI framework, generates intentions as a consequence of the updated beliefs and goals, and then selects concrete actions based on those intentions.

The variables of the cognition module are categorized into fast-dynamics, slow-dynamics, and auxiliary state variables, based on their role in the cognition process. Fast-dynamics state variables (i.e., beliefs, goals, and emotions) are central, since they directly shape the user's behavior. Slow-dynamics variables (e.g., general preferences, personality traits) primarily personalize the model and modulate how fast-dynamics variables evolve. Auxiliary variables describe intermediate processes: bias is included to capture the systematic ways in which emotions and goals shape the formation of new beliefs from rationally perceived knowledge, thereby reflecting systematic deviations from purely rational reasoning. All the variables are then represented as a network of influences, which makes dependencies explicit and allows describing how variables evolve in response to each other.

The formalization also distinguishes between internal (invisible) and external (visible) variables and processes, consistent with ToM. As such, the model explicitly represents which variables and processes of the user need to be inferred from the user's behavior.

To facilitate the use of the model for SR behavioral control, the model was mathematically formulated in a state-space form. This representation models the dynamic evolution of the state variables (which belong to the cognition module) based on inputs (real-life data), and describes how they contribute to generating the output (actions). As a result, the model can be directly integrated with classic control tools such as model-based state estimators and model-based controllers. In turn, this enables SRs to infer invisible mental states, reason about the future consequences of their actions, and systematically adjust their behavior accordingly, while maintaining a white-box representation of the underlying cognitive processes of users.

Furthermore, the interdependencies between mental states of the cognition module are mathematically formulated via Fuzzy Cognitive Maps (FCMs). This representation supports cyclic influences between mental states and expresses them as state variables whose values represent degrees of activation, reflecting the graded and imprecise nature of psychological concepts. This representation also provides a formal route for converting the cognitive network into a discrete-time state-space formulation, as shown in Chapter 2.

Finally, personalizable parameters are introduced in the model with the objective of capturing differences between users, and are associated both with slow-dynamics variables and with the strengths of the interdependencies between mental states. The parameters can be identified based on data collected from users over time, enabling the

model to be personalized without altering its structure or sacrificing its transparency.

The resulting user model enables the representation of the dynamics of user mental states. Translating these dynamics into robot actions, however, requires a suitable control approach, which motivated the next research question.

Research Question 3

Which control approaches are most suitable for enabling autonomous, socially-aware, and interpretable behavior generation in SRs, based on the developed user model?

This research question is addressed in Chapter 3. The proposed solution is a model-based behavioral control framework that embeds the user model inside the control loop of the robot, both for estimating current mental states and for predicting their future evolution. The controller operates as a predictive controller in closed loop: it evaluates how candidate robot actions will influence the future mental states of users and chooses actions that meet predefined objectives and constraints.

This approach offers several key advantages that directly address the research question. First, it enables **autonomous** operation, as the controller selects interactive actions in real time without requiring human intervention. Second, it promotes **socially-aware behavior** by enabling the robot to adapt its actions to the current context and to personalize its responses to each user, as discussed in Chapter 1.3. By estimating the mental states of users and predicting how the actions of the SR affect these states, the SR is able to adapt its behavior to the current context and condition of the user. In turn, personalization is achieved because the model relies on user-specific parameter estimates, which lead to different predictions and, consequently, different behaviors for different users. Third, the controller is **interpretable**, since the internal states (beliefs, goals, emotions) are explicitly modeled rather than implicitly incorporated within a black box, and decisions are made by optimizing a transparent cost function based on these mental states.

Finally, the control-theoretic approach used in the thesis admits formal guarantees on constraint satisfaction, such as ensuring user comfort. This overcomes limitations of traditional model-free controllers such as machine learning or rule-based systems, as discussed in detail in Chapter 5. Such guarantees are particularly important in the context of autonomous operation because they ensure that the robot can make autonomous decisions while respecting safety, ethical, and interaction limits that should not be violated.

Building on the chosen control approach, practical adaptations were implemented to enable reliable operation in real human–robot settings.

Research Question 4

What adaptations are required when integrating the user model into the control framework to ensure the effective operation of the controller in real HSRIs, given the requirements and limitations of these interactions?

This research question is addressed in Chapter 3. Three main adaptations were required to integrate the user model into the control framework and ensure effective operation in real-world HSRIs.

The user model proposed in Chapter 2 mathematically defined most of its subprocesses through abstract functions, without specifying concrete formulations that allowed direct implementation in a control system. To address this, Chapter 3 proposed concrete, computationally tractable formulations as specific instances of the original model. For example, the perceptual access subprocess was specified as a simple memory filter that retains the last observed value when new data is not captured. The rational reasoning subprocess was specified as a set of independent, parameterized mappings from each input (i.e., each element of the perceived data) to each output (i.e., each element of the rationally perceived knowledge), making identification more computationally tractable. The inter-dynamics between mental states in the cognition module were modeled using piecewise-constant weights that depend on the current values of both the influencing and influenced variables, replacing abstract generic functions. Finally, the rational intention selection subprocess was formulated as a weighted sum of beliefs and goals with identifiable parameters per user. These mathematical specifications transform the general yet abstract user model from Chapter 3 into a computationally tractable model that can be identified, implemented, and deployed in real-world HSRIs, while preserving interpretability and transparency.

Given the extensive parameterization of the model and the limited data available from real-world HSRIs, initial attempts at model identification produced parameters that were not unequivocally identifiable. This, in turn, led to inaccurate estimation of user mental states and, consequently, to poor controller performance. To address this, a structured identification protocol was proposed to avoid overfitting and ensure meaningful personalization by combining redundancy evaluation with staged identification in an iterative way. Parameter redundancy was assessed by examining consistency across multiple optimization runs with different initializations. Until all parameters are uniquely identifiable, the model is simplified, re-identified, and parameter redundancy is reassessed. As for the identification procedure itself, rather than identifying all parameters simultaneously, two multi-stage procedures were proposed that decouple, pre-identify, and then jointly optimize the modules using warm start. The procedures improved the percentage of unequivocally identified parameters from 74% (conventional method) to 96% or 98%, depending on the procedure used. By ensuring that only meaningful parameters remain, the proposed protocol promotes model interpretability and prevents overfitting, which in turn maintains predictive accuracy.

With the controller operating in real HSRIs, the next challenge concerned maintaining accurate estimates of user mental states throughout the interaction.

Research Question 5

How can the accuracy of model-based estimates of user mental states be maintained throughout interaction with the SR, in order to mitigate state drift and preserve controller performance?

As explained in Chapters 3 and 4, accurately estimating the mental states of users during interaction is challenging when these states are obtained solely by propagation of the user model. This limitation is not a deficiency of the model itself, but a general consequence of open-loop predictions: in the absence of measurement feedback, even small mismatches between model and reality can accumulate over time, leading to progressive estimation drift and, in turn, a deterioration of closed-loop controller performance.

To address this, Chapter 4 proposed embedding the user model into a Kalman-Filter (specifically, an Extended Kalman Filter (EKF) to accommodate non-linear dynamics). Within this formulation, the user model serves as the process model in the prediction step, while available measurements are integrated during the correction step. Therefore, the state estimator complements model propagation with measurement feedback, attenuating the accumulation of prediction error over time.

However, since Kalman filters are designed for engineering and physical systems with well-defined quantities, implementing an EKF to estimate user mental states in HSRIs is not trivial and requires domain-specific adaptations. For example, conventional techniques for defining the process and measurement noise covariances (crucial parameters of the EKF) cannot be used for HSRIs, because they rely on physics-based assumptions, objective sensor information, and large datasets — none of which are available in this domain. Thus, Chapter 4 proposed a domain-specific approach to construct and personalize these matrices per participant based on model prediction errors and measurement variance under approximately steady-state conditions. Furthermore, since traditional EKFs assume frequent measurements, whereas measurements are sparse in HSRIs, three variants were developed in Chapter 4: an intermittent EKF that applies corrections only when measurements are available, a forward-fill EKF that reuses the last measurement, and a forward-fill-evolving EKF that reuses the last measurement and adjusts measurement confidence over time.

Extensive experimental validation using data collected from multi-session HSRIs with ten participants demonstrated that the forward-fill EKF variants reduced average prediction error by 3 percentage points (from approximately 12% to 9%) compared to open-loop estimation, with improvements up to 13 percentage points for individual participants. These improvements are particularly relevant in sparse-measurement HSRIs settings, where limiting estimation drift is essential for the performance of the model-based controller of the SR.

After presenting and integrating the user model, behavioral controller, and state estimator, the final research question concerned the assessment and validation of the proposed control framework in realistic and relevant scenarios.

Research Question 6

To what extent does the proposed control framework enhance the awareness of SRs regarding user mental states and improve the quality of HSRIs, and what implications does this have for future behavioral control approaches of SRs?

In Chapter 2, the accuracy of the proposed user model in estimating user mental

states and actions was assessed. Twenty-one participants controlled virtual agents through sixteen emergency evacuation scenarios, deciding whether to rescue entities (dependents, colleagues, belongings) or exit directly. Twelve scenarios were used to personalize the model to each participant, while the remaining four scenarios were reserved for assessment. In each scenario, the user's intention concerned the entity saved, and the actions corresponded to the step-by-step trajectory. The proposed model estimated user intentions with 81.55% accuracy and predicted actions with 66.06% accuracy. The decrease in accuracy between estimating intention and predicting actions occurred because multiple action sequences satisfied the same intention, and small deviations in path selection or intermediate movements lead to mismatches in action prediction, even when the intention was correctly predicted. Nevertheless, this does not limit the model's utility for SRs, as these robots require awareness of users' mental states and social actions rather than exact path prediction. Thus, these results demonstrate the model's capability to provide SRs with explicit awareness of user mental states. Moreover, compared to a baseline model excluding emotions and biases, the model achieved 6.25% and 2.45% higher accuracy in estimating intentions and actions, respectively, confirming that these elements enhance SR awareness of user mental states.

In Chapter 3, a case study assessed the proposed model-based control framework, which leveraged the user model developed and validated in Chapter 2. In the case study, ten participants solved chess puzzles with a Nao robot across three sessions (45 to 90 minutes each). Data from the first two interaction sessions per participant were used to identify the model parameters, while the third session validated the controller's performance. The identified model achieved high accuracy in estimating the beliefs, goals, and emotions of participants, with an average of Mean Squared Error (MSE)= 0.067 across these three mental states on the test data, thereby promoting SRs awareness of user mental states. Compared to a rule-based controller, the model-based approach increased user engagement by 16% ($p = 0.009$) and reduced the desire to quit by 8% ($p = 0.032$). Post-interaction questionnaires confirmed that participants perceived the robot whose behavior was steered by the model-based predictive controller as more aware, engaging, and personalized to their needs. Taken together, these results show the potential of model-based behavioral control to improve the quality of HSRIs.

Chapter 4 built on the case study introduced in Chapter 2 by using the resulting dataset to evaluate improvements in mental-state estimation obtained with the EKFs proposed in that chapter. In particular, the data from the first two sessions was used to personalize the estimator parameters, while the data from the third session was used to assess the EKFs. The results demonstrated that the forward-fill EKF variants reduced average mental state prediction error by 3% compared to open-loop model-based estimation, with individual improvements up to 13%. These results provide empirical evidence that properly adapted state estimators can enhance and maintain SR awareness of user mental states throughout interactions, mitigating estimation drift, which is critical for maintaining controller performance and interaction quality.

Finally, Chapter 5 positions the proposed control framework within the broader landscape of SR behavioral control. It demonstrates that most existing approaches (i.e.,

rule-based, reinforcement learning, or reactive methods) struggle to sustain long-term engagement and adaptive interactions. The analysis suggests that model-based control integrated with dynamic cognitive architectures represents an emerging and promising direction. By showing that approaches that model and influence user mental states are rare but may be critical for engagement, the survey concludes that the proposed framework addresses a significant gap in enabling mentally-aware, adaptive SR behavior and may inform future adaptive control approaches for SR.

6.3. Limitations and Recommendations for Future Work

This thesis laid the foundation for socially and cognition aware SRs. By proposing the first systems and control-theoretic framework to steer the behavior of SRs based on a model of the user behavior, it opened the path for a new generation of SRs. This section discusses the limitations of the proposed behavioral control framework, promising strategies to address them, and relevant directions for future research. This discussion is separated into theoretical and mathematical assumptions, modeling and system design choices, and experimental and validation limitations.

6.3.1. Theoretical and Mathematical Assumptions

This section examines the theoretical and mathematical assumptions underlying the proposed behavioral control framework and discusses their implications for safe operation and reliable robot behavior in HSRIs.

As discussed in Chapter 3, all state variables of the user model are bounded to the interval $[-1, 1]$ to ensure mathematical stability. Importantly, this standardized scale does not compromise personalization because the mapping between the intensity of state variables and their numerical values implicitly results from the individual cognitive dynamics identified during training and, therefore, is user-specific (e.g., 0.4 may represent moderate frustration for one user, while representing mild frustration for another user). However, this boundedness may limit the ability of the personalized model to capture extreme behaviors that exceed the interval of intensities observed during training. This stems from the fact that the model is identified with data reflecting representative cognitive patterns rather than rare outliers. Nevertheless, if extreme responses become relevant to an application, the model can be re-identified with data that includes such dynamics.

To facilitate the practical deployment of the user model, Chapter 3 introduced piecewise-linear approximations, rather than fully non-linear formulations, to represent the dynamic relationships between the state variables of the cognition module. In particular, this choice promotes computational tractability both during identification and real-time execution, and simplifies the deployment of the behavioral control framework by eventual practitioners, by reducing the manual effort required to specify the relationships between variables. However, such a simplification may limit the ability of the model to capture highly complex or abrupt shifts in mental states. The impact of this limitation can be mitigated by increasing the number of piecewise regions that define each linkage to better capture nonlinearity, though this creates a trade-off between improved accuracy and higher computational demand.

Furthermore, the EKFs proposed in Chapter 4 assume Gaussian process and measurement noise, which, despite constituting a reasonable and tractable initial approximation, may not fully capture certain noise characteristics relevant in HSRIs: since state variables are bounded to $[-1, 1]$, noise near these bounds is unlikely to be perfectly symmetric and unbounded, as a Gaussian distribution assumes; additionally, self-report measurements may occasionally include larger errors (e.g., participants misjudging their mental state) in addition to the smaller deviations primarily captured by a Gaussian distribution. Thus, future work could:

Recommendation for future work

Explore state estimators that do not rely on Gaussian noise assumptions (e.g., particle filters) to further improve the robustness of the state estimates.

Finally, while the current formulation of the user model promotes practical stability through parameter constraints and bounded state variables, rigorous mathematical proofs of stability under model uncertainty and disturbances are needed to strengthen theoretical foundations and provide formal guarantees for safety-critical applications in SRs. This is particularly relevant given that SRs are often used in contexts with vulnerable users, such as older adults or users with cognitive impairments. Since formal stability guarantees are typically established for well-defined classes of admissible dynamics, this requires explicitly characterizing the classes of functions that formulate the processes of the model of human behavior proposed in this thesis. In light of these considerations, future work should:

Recommendation for future work

Identify relevant classes of functions for representing the processes and establish formal stability and robustness guarantees for these classes of functions for the proposed user model and control framework.

6.3.2. Modeling and System Design Choices

This section examines how modeling and system design choices influenced the performance and practical deployment of the proposed cognition-aware control framework.

Several limitations identified in earlier chapters were partially or fully addressed in subsequent chapters of this thesis. In Chapter 2, the cognition module was formulated using an extended FCM that represented each linkage (i.e., influence between variables) as generic functions that depended on the variables involved in the connection. In the case study, these functions were instantiated using Fuzzy Inference Systems (FISs), leading to two main limitations. First, this approach required manually defining membership functions and rule bases, increasing manual intervention and potentially introducing design-related errors. Second, the computational overhead of FISs led to long optimization times, even with a limited number of parameters. These limitations motivated the subsequent shift, in Chapter 3, toward explicit and parameterized

mathematical formulations. In this formulation, the influences between variables are defined through piece-wise functions with real-valued parameters, substantially reducing training time and increasing the level of automation in model deployment.

Similarly, in Chapter 3, the mental states of users were estimated by propagating the user model based on previous values. This led to the accumulation of state estimation errors when measurements were sparse, which ultimately may have harmed the controller action selection, since the predictive controller requires an accurate estimation of the current state to select an appropriate robot action. Consequently, Chapter 4 mitigated this limitation by embedding the model within a state estimator, allowing predictions to be periodically corrected using available measurements and thereby maintaining estimation accuracy over extended interactions.

Beyond these addressed limitations, some challenges related to the proposed control framework remain open and were not explicitly tackled within the scope of this thesis.

The user model presented in Chapter 2 was developed based on cognitive theories representing neurotypical users. Consequently, its underlying structure implicitly assumes neurotypical patterns of perception, cognition, and decision-making, which may not hold for neurodivergent individuals or users with cognitive impairments. In such populations, these processes may follow different dynamics or behave differently than what the current model represents. Therefore, the following recommendation for future work is proposed:

Recommendation for future work

Testing the user model for neurodivergent and cognitively impaired users and, where necessary, refining or expanding the model structure to accommodate distinct cognitive profiles.

In Chapter 2, the modular structure of the model enabled a multi-stage identification strategy, in which perception and cognition modules were temporarily decoupled to identify parameters in a more meaningful and interpretable manner. This required introducing simplifying assumptions that effectively excluded the bias variables during early stages, which may skew warm-start results toward local optima. This risk was addressed by a final fully coupled optimization and by performing multiple independent runs with randomized initializations.

Another modeling consideration affecting system performance is the potential for model mismatch, which may lead to underperformance of the model-based controller. Such a mismatch can occur when data for parameter identification is scarce, insufficiently diverse, or affected by unreliable self-reports, and may limit accurate personalization of the model. Model mismatch may also arise for users who exhibit highly unpredictable or atypical cognitive dynamics that are not well captured by the model structure, as discussed earlier in this section. To mitigate these effects, future work should rely less on self-reports during interactions and should regularly re-identify model parameters during long-term deployments. To mitigate these effects, future work should:

Recommendation for future work

Reduce reliance on self-reports during online interactions and regularly re-identify model parameters during long-term deployments. Reduced reliance on self-reports may be achieved by incorporating personality assessments prior to interaction, triggering self-reports based on estimation uncertainty, and integrating additional sensing modes (e.g., emotion recognition) to provide complementary measurements of mental states.

Facilitating real-world deployment of the control framework proposed in this thesis will require further automation of the modeling and identification pipeline. While the current approach demonstrates feasibility and effectiveness, several steps still require expert intervention, including model initialization, parameter identification, and configuration of estimation and control components. Reducing this dependence on expert knowledge is essential for enabling broader adoption of the proposed behavioral control framework in practical SRs applications, particularly in real-world settings where end users or practitioners may lack technical expertise, and where complex real-world scenarios may require a larger number of cognitive variables, increasing model complexity. Thus, future work should:

Recommendation for future work

Develop software tools that automate model initialization, parameter identification, and deployment of the proposed framework, to enable its use by non-expert practitioners in real-world socially assistive robotics applications.

Finally, some SRs applications may require modeling more complex cognitive phenomena, such as nested reasoning about others' beliefs or hierarchical goals. However, such constructs come at the cost of increased model complexity, which may compromise identifiability and computational tractability, and should be integrated in a way that preserves the model's mathematical formulation. Thus, future work should:

Recommendation for future work

Where required by the application, extend the model with richer cognitive constructs, while preserving the mathematical formulation of the model and introducing those constructs incrementally and only as needed.

6.3.3. Experimental and Validation Limitations

This section reflects on the limitations of the experimental validation and outlines directions for broader and more representative evaluation of the proposed behavioral control framework.

The experimental validation of the controller and the state estimator, presented in Chapters 3 and 4, involved a limited number of participants due to the intensive multi-session experimental protocol necessary for parameter identification and for validating the framework in long-term interaction settings. While this sample

size was appropriate for demonstrating feasibility and validating the methodological approach in this early-stage research, broader validation is recommended to establish generalizability. Therefore, future work should:

Recommendation for future work

Extend validation to larger participant pools across more diverse real-life contexts, user demographics, and application domains.

In the experimental case study of Chapter 3, the novelty effect may have biased data collected during early sessions, as the initial excitement about interacting with the robot could have influenced user responses. This may result in unrepresentative training data and, consequently, inaccurate model identification, leading the controller to make suboptimal decisions. Addressing this limitation, which mainly affects long-term interactions, is actually feasible for such interactions because they provide sufficient opportunities to capture user responses beyond initial novelty-induced behaviors. These observations motivate future work to:

Recommendation for future work

Develop mechanisms for quantifying the novelty effect and identification strategies that account for it, such as re-identifying model parameters once the effect has subsided, or incorporating temporal weighting schemes that discount early-session data.

Finally, while Chapter 4 demonstrated improved mental state estimation accuracy, the evaluation was conducted offline using data collected in the case study presented in Chapter 3. To address these limitations, future work should:

Recommendation for future work

Include the state estimators proposed in Chapter 4 during online HSRIs to assess how reduced prediction errors translate into improved controller performance and enhanced interaction quality.

6.4. Final Remarks

This thesis introduced a systems and control-theoretic framework for SRs that explicitly models and leverages user mental states to guide robot behavior. By integrating a dynamic user model with model-based control and state estimation, the framework enables interpretable, adaptive, and personalized interaction over time. Experimental results demonstrated the effectiveness of this approach in realistic human–SR interactions and highlighted its potential to improve engagement and behavioral appropriateness. While some limitations remain, they highlight clear and promising directions for future research. Overall, this work establishes a principled foundation for advancing SRs toward safe, reliable, and sustained real-world deployment.

A

Examples that illustrate the importance of the elements of the perception, cognition, and decision-making model

This appendix represents a number of examples according to real-life scenarios, where these examples motivate the elements and/or linkages that are introduced in the cognitive model proposed in Chapter 2.

Role of slow-dynamics state variables: The following four examples show the influence of the slow-dynamics state variables on the dynamic evolution of fast-dynamics state variables. In these examples, the observer agent (described by first-person pronouns) makes inferences about the mental state of the observed agent (referred to by a specific name in the examples) based on their observed actions (*inverse inference*).

Example 1 *Ana and I are both in the library at 5:30 PM. Ana picks up her wallet and walks toward the door (action of the observed agent noticed by the observer agent). The coffee house nearby has a late opening hour until 6:00 PM, but I do not know whether or not Ana knows about this (no access to the general world knowledge of the observed agent). I guess Ana knows about the opening hour of the coffee house and believes that it is still open (guessing the general world knowledge and inferring about the belief of the observed agent). I infer that Ana is going to buy a cup of coffee (goal of the observed agent inferred by the observer agent).*

In this example, since the observer agent does not have access to the general world knowledge of the observed agent, the inference involves an intermediate procedure, i.e., inference about the belief of the observed agent based on a guess rather than facts.

Suppose that, according to Ana's general world knowledge, the opening hour of the nearby coffee house is until 5:00 PM. Thus Ana does not go at 5:30 PM to the coffee

house, which she supposes to be closed. In case I had access to Ana's general world knowledge, then I would not conclude that Ana is going to grab a coffee. Therefore, my inference about Ana's goal was more reliable.

Example 2 *Consider the scenario of Example 1, but this time I have heard from Ana before that she knows that the coffee house nearby is open until 6:00 PM (access to the general world knowledge of the observed agent). This time I infer with more certainty that Ana is going to buy a cup of coffee.*

Although this example shows that when the general world knowledge of the observed agent is known by the observer agent, an inverse inference about the goals of the observed agent are less prone to uncertainties, the next example shows that knowing the general world knowledge of the observed agent alone may not suffice to make precise inverse inferences.

Example 3 *Suppose that in Example 2, in addition to being aware of Ana's general world knowledge, I know that she needs to drink coffee when studying late (access to the general preference of the observed agent). Then I infer that Ana is going to buy coffee with a much higher certainty than in Example 2. On the contrary, if I know that Ana never drinks coffee in the afternoon (general preference of the observed agent) I will not infer that her beliefs and goals are related to grabbing a coffee.*

This example shows that the access of an observer agent to the general world knowledge and general preferences of an observed agent significantly improves the reliability and level of certainty of the inferred fast-dynamics state variables. Moreover, having access to only one of these slow-dynamics state variables may still result in inaccurate or erroneous inferences. In particular, having access to the general preferences of the observed agent in addition to the general world knowledge in Example 3 supports the certainty of the inferences or prevents the observer agent from making erroneous inferences.

Example 4 *Now consider Example 3, where I am aware that Ana never drinks coffee in the afternoon. While the combination of Ana's general world knowledge and general preference prevents me from making a wrong inference, they do not provide me with a chance either to make an inference about what Ana's beliefs and goals at the moment are. Now suppose that I know Ana for long enough to be aware that she is an introvert (i.e., she needs some personal time after long interactions) with higher levels of neuroticism[1] (i.e., she often feels worried). These personality traits of Ana together with her actions help me to infer that Ana believes she needs some personal time (thus she leaves me to be by herself). I also infer that Ana believes that, if she leaves her wallet unattended, someone may steal it (thus takes her wallet with her).*

This example shows the importance of incorporating the specific personality traits of rational agents in cognitive models for achieving more precision and reliability with the resulting inferences.

[1]We use the terms introversion and neuroticism according to the categorization introduced by the Big Five Personality Traits O. John et al. (2008).

Elements that influence emotions: The following three examples illustrate the role of various state variables in triggering the emotions of an observed agent (referred to by a specific name).

Example 5 *While walking on the street, Elisa's wallet falls out of her purse (real-life data). Later on in a shop, Elisa reaches for her wallet and realizes that it is not in her purse (perceptual access). She reasons that she has lost the wallet (rationally perceived knowledge). She then supposes that she has lost her wallet (inference of a belief based on the rationally perceived knowledge). This belief makes her anxious (stimulation of emotions).*

In the given example, before Elisa notices that her wallet is missing (i.e., *without perceptual access*) and before she reasons that she has lost the wallet (i.e., *without rational reasoning*), she was not anxious (*no stimulation of emotions*). In a different situation, for the same perceptual access that causes the same perceived data, i.e., a missing wallet, Elisa may reason and believe that she has left her wallet on the dining table at home (*different rational reasoning and hence different rationally perceived knowledge*). Therefore, Elisa will not be anxious (*no stimulation of emotions*). In summary, independent of what the real-life data is (e.g., the wallet has fallen on the street or is at home) the emotions of a rational agent may be moderated by the perceptual access of the agent to that data and by the reasoning the agent applies to the perceived data. In other words, the emotions of a rational agent depend on its beliefs rather than on real-life data directly.

Example 6 *Frank is exploring a new city for the first time and wants to buy an ice cream (goal). While walking, he notices a few people across the street who are eating ice cream (perceived data). Correspondingly, he reasons and believes that there should be an ice cream shop close by (rationally perceived knowledge transformed into a belief), which makes him feel satisfied (stimulated emotions).*

This example shows a case where a belief by itself does not stimulate emotions, but the belief together with a goal does. In other words, if Frank did not want to eat an ice cream, the belief that an ice cream shop is nearby would not influence his emotional status.

Example 7 *Grace is afraid of dogs (general preference). While walking in a park, she notices the footprints of a dog (perceptual access) and correspondingly reasons and believes that there should be a dog nearby (rationally perceived knowledge transformed into a belief). This belief makes her anxious (stimulated emotion).*

This example illustrates how general preferences alongside beliefs may directly stimulate emotions in rational agents. Note that in this case, if Grace was not afraid of dogs, the belief that a dog is nearby by itself would not stimulate particular emotions in her.

Elements that are influenced by emotions: The following example illustrates how emotions of rational agents may influence their goals.

Example 8 *Hailey has planned to go to a party tonight (original goal). In the afternoon, Hailey receives some bad news that make her deeply sad (emotion). As a consequence, she decides not to go to the party anymore (change in the goal due to the emotions).*

This example shows that emotions may change the goals of rational agents. Although Hailey's general preference may be to participate in such parties, and she may in general be an extrovert with high levels of conscientiousness (i.e., self-discipline and tendency to follow her schedules), due to her sadness she may make a goal (i.e., skipping the party) that contradicts her initial goal that corresponded to her general preferences, personality traits, etc.

Importance of personalizing the perceptual access and the rational reasoning processes: The first example below illustrates the importance of personalizing the perception procedure of various rational agents. In the given example, the tourist is the observed agent and the tour leader and the tourist's close friend are the observer agents. The second example below demonstrates the importance of decomposing the process that yields the rationally perceived knowledge into the subprocesses that are explained in subsubsection 2.2.2.

Example 9 *Suppose that a tourist tells her tour leader that she has already been to the historical city center of the city they are visiting. The tour leader may assume that the tourist has a perfect knowledge of the real-life data, including the location of the church, the old building of the City Hall, and all the souvenir shops (*general world knowledge of the observed agent according to the observer agent*), while in her previous visit, the tourist has overlooked the old building of the City Hall. Then the general world knowledge considered by the tour leader for the tourist is inaccurate. Now suppose that the tourist tells her close friend (*an observer agent who is aware of the personalized perception of the observed agent*) that she has once been to the historical city center. The friend assumes - knowing the personalized perception procedure of the tourist - that she might have overlooked the old building of the City Hall.*

Example 10 *Brian, Charlie, and Diana are inside a shopping mall. Although they cannot see the outside, before they entered the shopping mall, it was sunny. Someone soaked in water enters the shopping mall. Brian does not notice this person (*no updated perceptual access*) and thus, keeps the* belief *that outside is sunny. Charlie and Diana notice this person (updated perceptual access). Charlie reasons and accordingly* believes *that it must be raining now, while Diana reasons that this person has fallen into a ditch (different [personalized] rational reasoning) and keeps the* belief *that it is still sunny outside.*

Note that in the above example, if an observer agent infers about the beliefs of Brian, Charlie, and Diana (all as observed agents), if their personalized perceptual access and rational reasoning processes are excluded, the observer agent may infer that all three observed agents hold the same belief that, e.g., it is now raining outside.

Elements that bias the generation of beliefs: The two following examples show how the generation of beliefs for rational agents may be biased by their emotions and goals, respectively. Igor and Jane (in the first example), and Kevin and his father (in the second example) are the observed agents.

Example 11 *Igor and Jane are having a walk together. While walking, they both see a dog (real-life data, which after perceptual access and rational reasoning, for both observed agents results in the belief that there is a dog nearby). Since Igor is afraid of dogs (general preference), he feels afraid (emotion resulting from emotion trigger 1, see Remark 3), and starts to believe that the dog might harm him (belief biased by emotion). Jane, however, does not feel any fear and thus believes there is no threat from the dog.*

In the above example, although both observed agents initially had the same belief (i.e., there is a dog nearby) one of them develops a biased new belief about the dog because of his triggered emotions.

Example 12 *Kevin is a football fan (general preference). The team he supports is currently in the second place in the championship. When all evidences are studied by an objective analyst, they conclude that - although not impossible yet - the chances that Kevin's favorite team wins the championship are very small (unbiased belief). Since Kevin wants his team to win (Kevin's desire or goal), he believes that his team will win (belief biased by a goal). Now suppose that Kevin's father has the same general preference and goal as Kevin, while he has a much lower level of conscientiousness (a personality trait that implies Kevin's father is generally less stubborn and more flexible regarding various situations). Consequently, Kevin's father develops a much less strong belief than his son about their team winning the championship.*

The above example illustrates that goals can bias the beliefs of rational agents, whereas personality traits - while not generating a bias by themselves - may regulate (boost or hinder) this influence.

Inverse inference of emotions from actions: The following two examples illustrate the process of inverse inference of emotions based on the observed actions of rational agents, which is based on updating the belief and the goal of the observed agent (due to the influence of emotions), respectively. The observer agent is specified by the first-person pronouns, and the other person in the examples is the observed agent.

Example 13 *I see Anabel and smile at her. Anabel and I are friends, so I expect Anabel to smile back (expected action). Anabel, however, looks unfriendly instead and turns her back at me (observed action). Thus, I conclude that Anabel may feel negative emotions about me at the moment (inferred emotion).*

In the above example, the observer agent initially estimates the belief of the observed agent to be "we are good to each other". The observed action of the observed agent, however, implies that this belief is wrong. Therefore, the observer agent updates the belief of the observed agent to "Anabel is not good to me", and deduces that she holds negative emotions. Now, in order for the observer agent to infer precisely about the emotion of the observed agent (e.g., whether Anabel is angry or sad), the observer agent should be aware of the belief(s) and goal(s) of the observed agent in their previous interaction(s).

Suppose that this morning Anabel showed me a picture of a dress. Her goal was to wear it for her sister's wedding (Anabel's goal in the previous interaction). I said

the dress may not look nice on her. Anabel believed that I was being mean to her (Anabel's belief in the previous interaction). This made her feel **angry** at me, and thus in our next interaction she updated her belief from "we are good to each other" to "we are not friends anymore".

Example 14 *Last week, I had a good discussion with Lewis about his projects. While walking on the campus, I come across Lewis. I know he has a vacancy for his new project. I tell to Lewis that I am seeking a new project to join. I suppose that Lewis has developed the goal of establishing a collaboration with me after our previous talk, so I expect him to invite me to join his new project (expected action). Instead, Lewis wishes me good luck and leaves (observed action). Clearly my inference about Lewis' goal was wrong.*

In this example, the observer agent expects a particular action from the observed agent based on inferring the observed agent's belief to be "the observer agent has very high qualifications". Although this belief is correctly deduced, the goal is wrongly inferred, because the observer agent has not considered the beliefs and goals of the observer agent during their last interaction, i.e., their talk (Lewis believed that he should always be better than his employees and while talking his goal was to show he knows the best), and thus the emotions that were triggered (Lewis feels threatened by the qualifications of the observer agent and thus develops the goal to avoid collaborating with him).

B

Supplementary Material for the Nao Chess Interaction Case Study

This appendix gathers the supplementary material related to the case study presented in Chapter 3, where the model-based control framework proposed in this chapter was experimentally validated through chess-based HSRIs sessions with a Nao robot. The appendix includes the design and scheduling of puzzle difficulty levels, the graphical representations and structure of the Mathematical Model of Mind (MMM) modules used in the experiment, the reward movements performed by the robot, the questions posed to participants during the interactions to assess user mental states, and the questionnaires administered in the final session.

B.1. Difficulty levels of the chess puzzles

This appendix includes information on how the difficulty of the puzzles was selected throughout the interactions with the Nao robot. Table B.1.1 describes the conversion between the ratings attributed to the puzzles in the *lichess.org* database Lichess.org (2024), where the puzzles were retrieved from, and the categorization in six difficulty levels adopted by the controller of Nao. Moreover, Table B.1.2 shows the difficulty level of each puzzle proposed to the participants in the first two interactions.

Table B.1.1: Difficulty levels of the chess puzzles and the corresponding minimum and maximum ratings attributed in the *lichess.org* database Lichess.org (2024).

Puzzle difficulty level	Minimum Rating	Maximum Rating
0	600	800
1	920	1120
2	1240	1440
3	1560	1760
4	1880	2080
5	2200	2400

Table B.1.2: Difficulty levels of the chess puzzles in the first two experimental sessions with participants.

Order number of the puzzle offered to the participant	Difficulty level in session 1	Difficulty level in session 2
1, 2, 3	0	5
4, 5, 6	2	3
7, 8, 9	4	1
10, 11, 12	4	1
13, 14, 15	2	3
16, 17, 18[1]	0	5
19, 20, 21	0 (as in the 1st row)	5 (as in the 1st row)
21, 23, 24	2 (as in the 2nd row)	3 (as in the 2nd row)
...	...	...

B.2. Structure of the modules of the MMM study

This appendix includes graphical representations of part of the perception module (Figure B.2.1), the cognition module (Figure B.2.3 and Figure B.2.2), and part of the decision-making module (see Figure B.2.4). These figures showcase the variables that are part of each module, as well as which variables influence each other (these influences are represented by arrows).

Furthermore, two structures are displayed for the cognitive module, Figure B.2.3 and Figure B.2.2. While Figure B.2.3 represents the model used in the first two sessions with the participants, Figure B.2.2 shows the model used in the last session for the integration within the model-based controller. For clarity, the influencers of the variables in the cognition module are also given in Table B.2.3.

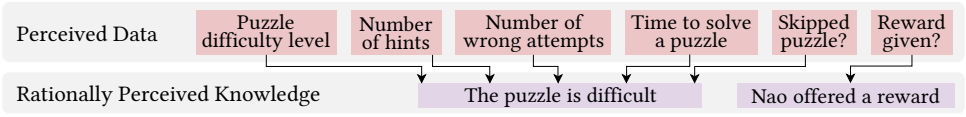


Figure B.2.1: Rational reasoning subprocess of the perception module of the ToM model (MMM) used in the case study.

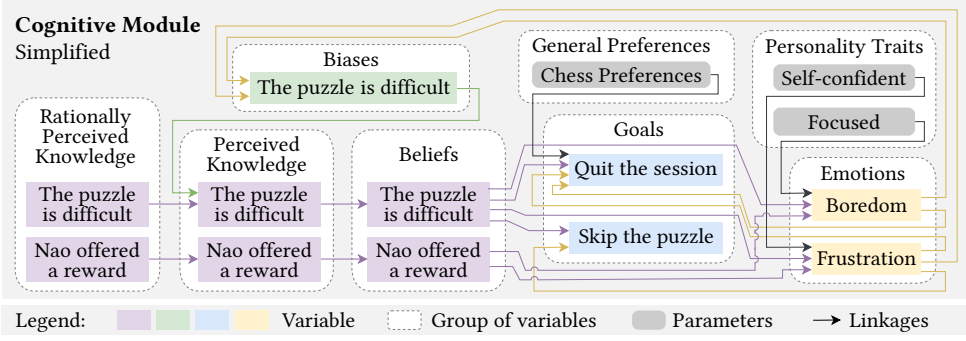


Figure B.2.2: Variables and linkages of the simplified cognitive module of the ToM model (MMM) used in the case study.

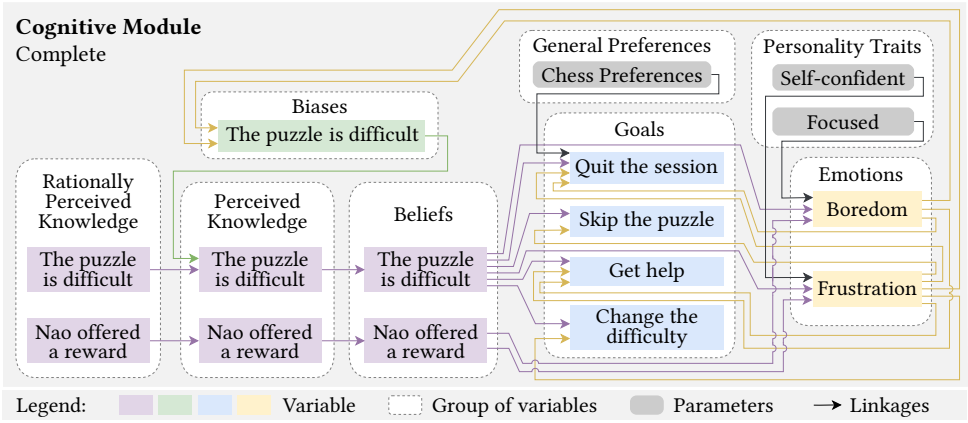


Figure B.2.3: Variables and linkages of the cognitive module of the ToM model (MMM) used in the case study.

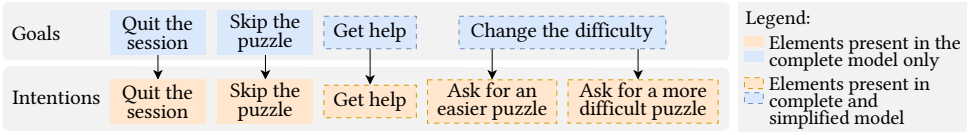


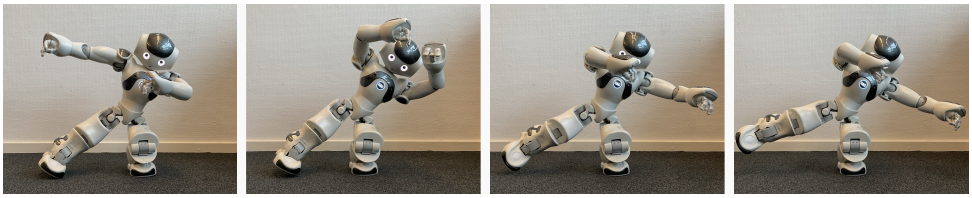
Figure B.2.4: Variables and linkages of the rational intention selection of the decision-making module of the ToM model (MMM) used in the case study.

Table B.2.3: The influencer variables of each one of the variables of the cognition module.

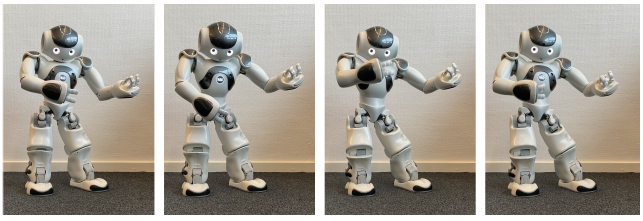
Variable and concept	Influencers (and concepts)
$x_1^{\mathrm{b}}(k)$ (The puzzle is difficult)	$x_1^{\mathrm{PK}}(k)$ (The puzzle is difficult)
$x_2^{\mathrm{b}}(k)$ (Nao offered a reward)	$x_2^{\mathrm{PK}}(k)$ (Nao offered a reward)
$x_1^{\mathrm{g}}(k)$ (Quit the session)	$x_1^{\mathrm{b}}(k)$ (The puzzle is difficult), $x_1^{\mathrm{e}}(k)$ (Boredom), and $x_2^{\mathrm{e}}(k)$ (Frustration)
$x_2^{\mathrm{g}}(k)$ (Skip the puzzle)	$x_1^{\mathrm{b}}(k)$ (The puzzle is difficult) and $x_2^{\mathrm{e}}(k)$ (Frustration)
$x_3^{\mathrm{g}}(k)$ (Get help)	$x_1^{\mathrm{b}}(k)$ (The puzzle is difficult), $x_1^{\mathrm{e}}(k)$ (Boredom), and $x_2^{\mathrm{e}}(k)$ (Frustration)
$x_4^{\mathrm{g}}(k)$ (Change difficulty)	$x_1^{\mathrm{b}}(k)$ (The puzzle is difficult) and $x_2^{\mathrm{e}}(k)$ (Frustration)
$x_1^{\mathrm{e}}(k)$ (Boredom)	$x_1^{\mathrm{b}}(k)$ (The puzzle is difficult) and $x_2^{\mathrm{b}}(k)$ (Nao offered a reward)
$x_2^{\mathrm{e}}(k)$ (Frustration)	$x_1^{\mathrm{b}}(k)$ (The puzzle is difficult) and $x_2^{\mathrm{b}}(k)$ (Nao offered a reward)
$x^{\mathrm{bias}}(k)$ (The puzzle is difficult)	$x_1^{\mathrm{e}}(k)$ (Boredom) and $x_2^{\mathrm{e}}(k)$ (Frustration)
$x_1^{\mathrm{PK}}(k)$ (The puzzle is difficult)	$y_1^{\mathrm{RR}}(k)$ (The puzzle is difficult) and $x^{\mathrm{bias}}(k)$ (The puzzle is difficult)
$x_2^{\mathrm{PK}}(k)$ (Nao offered a reward)	$y_2^{\mathrm{RR}}(k)$ (Nao offered a reward)

B.3. Reward movements performed by Nao

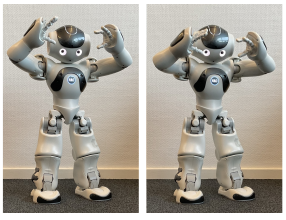
This appendix contains a visual representation of the five entertaining movements performed by the Nao during experimental sessions of the case study. These movements were performed by the Nao robot to reward the participants and keep them engagement in the interaction. These five movements included performing tai chi (Figure B.3.5a), playing the guitar (Figure B.3.5b), pretending to take a photo (Figure B.3.5c), dancing (Figure B.3.5d), and pretending to be an elephant (Figure B.3.5e).



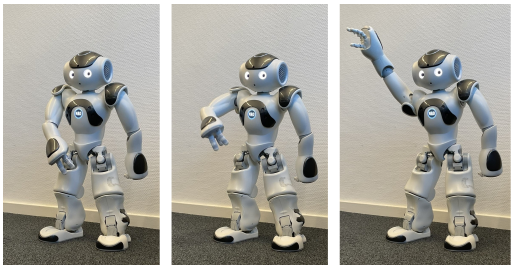
(a) Performing Tai Chi



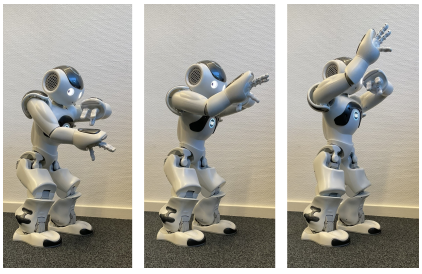
(b) Playing the guitar



(c) Taking a photo



(d) Dancing

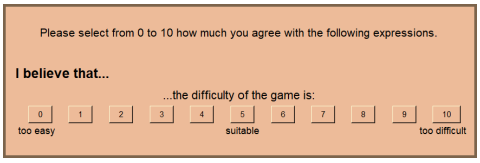


(e) Mimicking an elephant

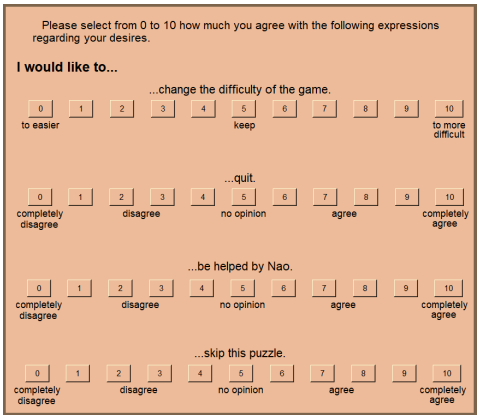
Figure B.3.5: The five reward movements performed by the Nao robot in the experimental sessions with the participants.

B.4. Questions posed to the participants during the interactions with Nao

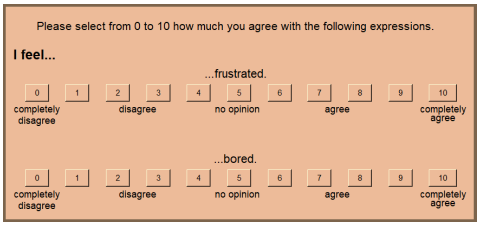
This appendix includes the questions asked to the participants during their interactions with the Nao robot. The questions were posed to the participants to gather information about their beliefs (Figure B.4.6a), goals (Figure B.4.6b), and emotions (Figure B.4.6c). Furthermore, at the end of each puzzle, participants were presented with the questions shown in Figure B.4.6d, allowing them to request an easier or more difficult puzzle. However, while participants could make this request, Nao did not fulfill it. Instead, the request was used to gather information about the participant's actions.



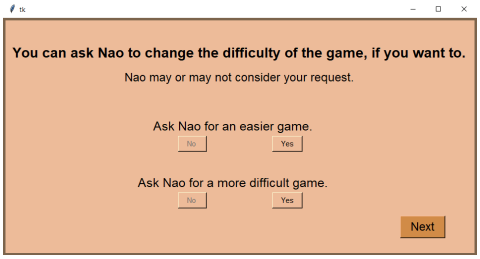
(a) Beliefs



(b) Goals



(c) Emotions

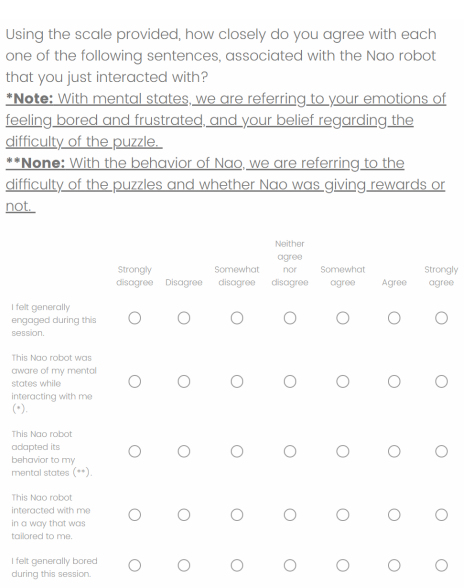


(d) Actions

Figure B.4.6: Periodic questions asked to the participants during the interaction with the Nao robot. In the first and second sessions, they enabled the collection of the data used to identify the MMM per participant. In the last session, they were used to assess the interaction and to update the state variables of the MMM that was integrated in the controller of Nao.

B.5. Questionnaire

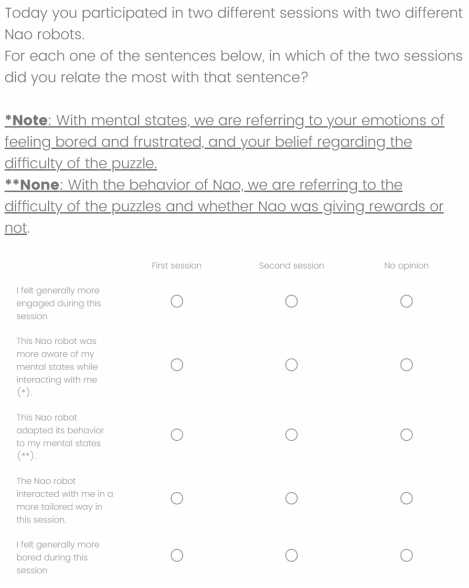
This appendix provides the questionnaires that were given to the participants in session 3.



Using the scale provided, how closely do you agree with each one of the following sentences, associated with the Nao robot that you just interacted with?
***Note:** With mental states, we are referring to your emotions of feeling bored and frustrated, and your belief regarding the difficulty of the puzzle.
****None:** With the behavior of Nao, we are referring to the difficulty of the puzzles and whether Nao was giving rewards or not.

	Strongly disagree	Disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Agree	Strongly agree
I felt generally engaged during this session.	○	○	○	○	○	○	○
This Nao robot was aware of my mental states while interacting with me (*).	○	○	○	○	○	○	○
This Nao robot adapted its behavior to my mental states (**).	○	○	○	○	○	○	○
This Nao robot interacted with me in a way that was tailored to me.	○	○	○	○	○	○	○
I felt generally bored during this session.	○	○	○	○	○	○	○

(a) Absolute questionnaire given to the participants after each one of the two interactions of the third session.



Today you participated in two different sessions with two different Nao robots.
For each one of the sentences below, in which of the two sessions did you relate the most with that sentence?

***Note**: With mental states, we are referring to your emotions of feeling bored and frustrated, and your belief regarding the difficulty of the puzzle.
****None**: With the behavior of Nao, we are referring to the difficulty of the puzzles and whether Nao was giving rewards or not.

	First session	Second session	No opinion
I felt generally more engaged during this session	○	○	○
This Nao robot was more aware of my mental states while interacting with me (*).	○	○	○
This Nao robot adapted its behavior to my mental states (**).	○	○	○
The Nao robot interacted with me in a more tailored way in this session.	○	○	○
I felt generally more bored during this session	○	○	○

(b) Comparative questionnaire given to the participants at the end of the third session, comparing the two interactions.

Figure B.5.7: Questionnaires given to the participants during the third session.

C

Methodology and Overview of the Literature Review presented in Chapter 5

This appendix provides complementary material supporting the literature survey presented in Chapter 5 of this thesis. It details the systematic search protocol adopted to identify relevant publications and presents a structured overview of the reviewed studies.

Together, these sections ensure methodological transparency and offer a consolidated reference of the analyzed works, including their application domains, robotic platforms, and underlying control approaches.

C.1. Literature Search Strategy and Source Overview

This appendix details the literature search strategy adopted for this survey. Table C.1.1 summarizes the bibliographic sources consulted, the number of papers retrieved from each source, the selected publications, the applied search queries, and any additional filtering criteria. The combination of structured database searches and backward snowballing ensured broad coverage while maintaining relevance to the scope of this work.

Table C.1.1: Overview of the sources used to collect relevant papers for this survey, including the papers retrieved from each source, the number of papers per source, and the corresponding search terms. For both Scopus and Google Scholar, only the papers from the last 10 years were considered.

Source	Number of papers	Selected papers	Keywords
Scopus	21	Aliasghari et al. (2020); Benedictis et al. (2023); Bettosi et al. (2024); Cao, Beir et al. (2017); Cao et al. (2019); Céspedes et al. (2020); Dell'Anna and Jamshidnejad (2024); Dziergwa et al. (2018); Kubota et al. (2020); Lazzeri et al. (2018); Magyar and Vircikova (2015); Maroto-Gómez, Malfaz, Castro-González and Salichs (2023); Mazzei et al. (2014); Mishra and Skantze (2022); Páez et al. (2020); Sequeira et al. (2016); Shiarlis et al. (2017); Tahir et al. (2020); Zaraki et al. (2014); Zhu et al. (2024) and work proposed in Chapter 2 (published in Morão Patrício and Jamshidnejad (2023a))	robot AND (social OR assistive) AND (behaviour OR behavior) AND control AND NOT trajectory AND NOT manipulation AND NOT navigation AND NOT obstacle avoidance AND NOT review
Google Scholar	13	Bagheri et al. (2021); Clabaugh et al. (2019); Dissing and Bolander (2020); Gao, Barendregt et al. (2018); Kidd and Breazeal (2008); J. J. Lee et al. (2019); M. H. Lee et al. (2023); Patacchiola and Cangelosi (2020); Rodríguez-Lera et al. (2018); Rossi et al. (2020); Scassellati et al. (2018); Tani et al. (2020); Tapus and Mataric (2008)	Three independent searches: 1) social robot AND behaviour AND control; 2) social robot AND behaviour AND interaction; 3) social robot AND theory of mind
Snowballing	11	Céspedes et al. (2021); Filippini et al. (2021); Gray and Breazeal (2014); Hussain et al. (2022); Magyar et al. (2016); Mitchinson and Prescott (2016); Munster and Jamshidnejad (2025); Schicchi and Pilato (2018); Schrum et al. (2019); Vossen et al. (2018) and work proposed in Chapter 3 (published in Morão Patrício and Jamshidnejad (2026b))	Does not apply

C.2. Comparative Overview of Reviewed Literature

This appendix presents a comparative overview of the literature reviewed in Chapter 5, categorizing each study by application context, robot platform used, and the underlying control strategies, thereby illustrating trends and methodological diversity across approaches.

Table C.2.2: Overview table

Paper	Application	Robot	Control Approaches			
			Symbolic and rule-based approaches	Learning-based and data-driven approaches	Modeling approaches	Probabilistic approaches
Tapus and Mataric (2008)	Post-stroke rehabilitation	N/A	-	RL	-	-
Kidd and Breazeal (2008)	Weight loss coaching	Autom	Heuristic and hard-coded rule-based systems (rule set)	-	Mathematical modeling	-
Mazzei et al. (2014)	General social interaction	FACE	Rule-based decision structure	-	Cognitive and Behavioral Architectures; Bio-inspired approaches	-
Zaraki et al. (2014)	Gaze control system	FACE	Heuristic and hard-coded rule-based systems	Classification Algorithms	Mathematical modeling; Bio-inspired approaches	-
Gray and Breazeal (2014)	Mental state manipulation	N/A	-	-	Model-based control	-

Table C.2.2: Overview table

Paper	Application	Robot	Control Approaches			
			Symbolic and rule-based approaches	Learning-based and data-driven approaches	Modeling approaches	Probabilistic approaches
Magyar and Vircikova (2015)	General social interaction (Meta-framework for behavior design)	N/A	-	RL	-	-
Mitchinson and Prescott (2016)	General companionship	Miro	-	-	Cognitive and Behavioral Architectures; Bio-inspired approaches	-
Sequeira et al. (2016)	General social interaction (Meta-framework for behavior design)	Nao				
Cao, Beir et al. (2017)	Social skills training for children	Nao	-	-	Homeostatic control	-

Table C.2.2: Overview table

Paper	Application	Robot	Control Approaches			
			Symbolic and rule-based approaches	Learning-based and data-driven approaches	Modeling approaches	Probabilistic approaches
Magyar et al. (2016)	General social interaction (Meta-framework for behavior design)	N/A	-	RL	-	-
Shiarlis et al. (2017)	Telepresence robot	TERESA	-	NN	-	-
Scassellati et al. (2018)	Personalized ASD therapy	Jibo robot	Rule-based decision structure (Decision tree)	-	-	-
Rodríguez-Lera et al. (2018)	General assistive interactions	N/A	Rule-based decision structure (FSM), Logic systems (Planning Domain Definition Language (PDDL))	-	Cognitive and Behavioral Architectures; Mathematical modeling; Bio-inspired approaches	-
Gao, Barendregt et al. (2018)	Personalized robotic tutoring	Pepper	Rule-based decision structure (Decision tree),	RL (Multi-Armed Bandit (MAB))	-	-

Table C.2.2: Overview table

Paper	Application	Robot	Control Approaches			
			Symbolic and rule-based approaches	Learning-based and data-driven approaches	Modeling approaches	Probabilistic approaches
Schicchi and Pilato (2018)	Vocabulary teaching for children	Pepper	Heuristic and hard-coded rule-based systems (RiveScript)	-	Mathematical modeling (for technical task)	-
Lazzeri et al. (2018)	General social interaction	FACE	Rule-based decision structure	-	Bio-inspired approaches	-
Dziergwa et al. (2018)	Home companionship	EMYS	Rule-based decision structure (FSM)	Classification Algorithms	Cognitive and Behavioral Architectures	-
Vossen et al. (2018)	General social interaction	Pepper	Logic systems	NN	Cognitive and Behavioral Architectures	-
Clabaugh et al. (2019)	Personalized ASD tutoring	Sprite	Rule-based decision structure	RL	-	-
Schrum et al. (2019)	Dementia exercise therapy	Pepper	Heuristic and hard-coded rule-based systems	-	-	-
Cao et al. (2019)	General assistive interactions	Nao/Pepper	-	-	Cognitive and behavioral architectures	-

Table C.2.2: Overview table

Paper	Application	Robot	Control Approaches			
			Symbolic and rule-based approaches	Learning-based and data-driven approaches	Modeling approaches	Probabilistic approaches
J. J. Lee et al. (2019)	Story telling	Tobii	-	-	Model-based control	POMDP and DBN
Rossi et al. (2020)	Pediatric vaccination	Nao	Heuristic and hard-coded rule-based systems (rule set)	-	-	-
Kubota et al. (2020)	End-user SR programming	Kuri	Rule-based decision structure (FSM); Logic system(Linear Temporal Logic (LTL))	-	-	-
Céspedes et al. (2020)	Gait rehabilitation therapy	Nao	Heuristic and hard-coded rule-based systems (rule set)	-	-	-
Páez et al. (2020)	Collaborative learning support	Baxter	Fuzzy logic	Classification Algorithms	Cognitive and Behavioral Architectures	
Aliasghari et al. (2020)	Gaze control system	RASA	Heuristic and hard-coded rule-based systems	-	Mathematical modeling; Bio-inspired approaches	-

Table C.2.2: Overview table

Paper	Application	Robot	Control Approaches			
			Symbolic and rule-based approaches	Learning-based and data-driven approaches	Modeling approaches	Probabilistic approaches
Tani et al. (2020)	General social interaction	Nao	Heuristic and hard-coded rule-based systems	NN	Cognitive and Behavioral Architectures; Bio-inspired approaches; Homeostatic Control	-
Patacchiola and Cangelosi (2020)	General social interaction	iCub	-	RL; Classification Algorithms	Bio-inspired approaches	Bayesian Network
Dissing and Bolander (2020)	General social interaction	Pepper	Logic systems; Heuristic and hard-coded rule-based systems (rule set)	NN	-	-
Tahir et al. (2020)	Social mediation in conversations	Nao	-	Classification Algorithms	-	-
Bagheri et al. (2021)	General assistive interactions	Pepper	-	NN (stack auto encoder), RL (Contextual Bandit Algorithm)	-	-

Table C.2.2: Overview table

Paper	Application	Robot	Control Approaches			
			Symbolic and rule-based approaches	Learning-based and data-driven approaches	Modeling approaches	Probabilistic approaches
Filippini et al. (2021)	General social interaction with children	Mio Amico Robot	Heuristic and hard-coded rule-based systems (rule set)	NN (Multi-layer perceptron)	-	-
Céspedes et al. (2021)	Gait rehabilitation therapy	Nao	Heuristic and hard-coded rule-based systems (rule set)	-	-	-
Hussain et al. (2022)	General social interaction	Furhat	-	RL	-	-
Mishra and Skantze (2022)	Gaze control system	Furhat	Heuristic and hard-coded rule-based systems	-	Mathematical modeling; Bio-inspired approaches	-
M. H. Lee et al. (2023)	Post-stroke rehabilitation	NAO	Rule-based decision structure (FSM)	NN	Mathematical modeling (weighted average)	Gaussian probability density functions
Benedictis et al. (2023)	General elderly assistance	N/A	Logic systems; Heuristic and hard-coded rule-based systems (rule set)	NN	Cognitive and Behavioral Architectures; mathematical modeling	-

Table C.2.2: Overview table

Paper	Application	Robot	Control Approaches			
			Symbolic and rule-based approaches	Learning-based and data-driven approaches	Modeling approaches	Probabilistic approaches
Maroto-Gómez, Malfaz, Castro-González and Salichs (2023)	General social interaction	Mini	-	-	Mathematical modeling; Bio-inspired approaches; Homeostatic control	-
Work proposed in Chapter 2	General social interaction	-	Fuzzy logic	-	Cognitive and Behavioral Architectures; Model-based control	
Bettosi et al. (2024)	General social interaction (Meta-framework for behavior design)	Pepper	Rule-based decision structure	Classification Algorithms	-	Transition matrices
Dell'Anna and Jamshidnejad (2024)	General social interaction	Nao	Fuzzy logic	NN	Cognitive and Behavioral Architectures	-

Table C.2.2: Overview table

Paper	Application	Robot	Control Approaches			
			Symbolic and rule-based approaches	Learning-based and data-driven approaches	Modeling approaches	Probabilistic approaches
Zhu et al. (2024)	Gaze control system	Xiaopang	-	-	Mathematical modeling; Bio-inspired approaches; Model-based control	Gaussian Process observer
Munster and Jamshidnejad (2025)	General social interaction	-	Fuzzy logic	RL	-	-
Work proposed in Chapter 3	General social interaction	Nao	-	-	Cognitive and Behavioral Architectures; Model-based control	-

References

Abdi, H., Valentin, D., & Edelman, B. (1999). *Neural networks* (1st). Sage. https://doi.org/10.4135/9781412985277

Ajzen, I. (1991). The theory of planned behavior. *Organizational Behavior and Human Decision Processes*, *50*(2), 179–211. https://doi.org/10.1016/0749-5978(91)90020-T

Akalin, N., Loutfi, A., Schmitz, A., & Distante, C. (2021). Reinforcement learning approaches in social robotics. *Sensors*, *21*(4), 1–37. https://doi.org/10.3390/S21041292

Al Moubayed, S., Skantze, G., & Beskow, J. (2013). The furhat back-projected humanoid head-lip reading, gaze and multi-party interaction. *International Journal of Humanoid Robotics*, *10*. https://doi.org/10.1142/S0219843613500059

Alhmiedat, T., & Alotaibi, M. (2022). Design and evaluation of a personal robot playing a self-management for children with obesity. *Electronics*, *11*(23), 1–15. https://doi.org/10.3390/electronics11234000

Aliasghari, P., Taheri, A., Meghdari, A., & Maghsoodi, E. (2020). Implementing a gaze control system on a social robot in multi-person interactions. *SN Applied Sciences*, *2*, 1–13. https://doi.org/10.1007/S42452-020-2911-0

Al-Qaderi, M. K., & Rad, A. B. (2018). A brain-inspired multi-modal perceptual system for social robots: An experimental realization. *IEEE Access*, *6*, 35402–35424. https://doi.org/10.1109/ACCESS.2018.2851841

Andrade, E. B., & Ariely, D. (2009). The enduring impact of transient emotions on decision making. *Organizational Behavior and Human Decision Processes*, *109*(1), 1–8. https://doi.org/10.1016/j.obhdp.2009.02.003

Arshad, N. I., Hashim, A. S., Mohd Ariffin, M., Mohd Aszemi, N., Low, H. M., & Norman, A. A. (2020). Robots as assistive technology tools to enhance cognitive abilities and foster valuable learning experiences among young children with autism spectrum disorder. *IEEE Access*, *8*, 116279–116291. https://doi.org/10.1109/ACCESS.2020.3001629

Ascensão, T., & Jamshidnejad, A. (2022). Autonomous socially assistive drones performing personalized dance movement therapy: An adaptive fuzzy-logic-based control approach for interaction with humans. *IEEE Access*, *10*, 15746–15770. https://doi.org/10.1109/ACCESS.2022.3143992

Åström, K. J. (1965). Optimal Control of Markov Processes with Incomplete State Information I. *Journal of Mathematical Analysis and Applications*, *10*(1), 174–205. https://doi.org/10.1016/0022-247X(65)90154-X

Bagheri, E., Bagheri, A., Esteban, P. G., & Vanderborgth, B. (2020). A novel model for emotion detection from facial muscles activity. *Robot 2019: Fourth Iberian Robotics Conference*, 237–249. https://doi.org/10.1007/978-3-030-36150-1_20

Bagheri, E., Roesler, O., Cao, H. L., & Vanderborght, B. (2021). A reinforcement learning based cognitive empathy framework for social robots. *International Journal of Social Robotics*, *13*, 1079–1093. https://doi.org/10.1007/S12369-020-00683-4

Baker, C. L. (2012). *Bayesian Theory of Mind : modeling human reasoning about beliefs, desires, goals, and social relations* [Doctoral dissertation, Massachusetts Institute of Technology]. https://dspace.mit.edu/handle/1721.1/73768

Baker, C. L., Goodman, N. D., & Tenenbaum, J. B. (2008). Theory-based social goal inference. https://cocolab.stanford.edu/papers/BakerEtAl2008-Cogsci.pdf

Baker, C. L., Jara-Ettinger, J., Saxe, R., & Tenenbaum, J. B. (2017). Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, *1*(4), 1–10. https://doi.org/10.1038/s41562-017-0064

Baker, C. L., Saxe, R. R., & Tenenbaum, J. B. (2011). Bayesian theory of mind: modeling joint belief-desire attribution. *Proceedings of the Annual Meeting of the Cognitive Science Society 33*, 2469–2474. https://escholarship.org/uc/item/5rk7z59q

Baker, C. L., Tenenbaum, J. B., & Saxe, R. R. (2007). Goal inference as inverse planning. *Proceedings of the Annual Meeting of the Cognitive Science Society 29*, 779–784. https://escholarship.org/uc/item/5v06n97q

Baron-Cohen, S., Leslie, A. M., & Frith, U. (1985). Does the autistic child have a "theory of mind" ? *Cognition*, *21*(1), 37–46. https://doi.org/10.1016/0010-0277(85)90022-8

Battini Sönmez, E., Han, H., Karadeniz, O., Dalyan, T., & Sarıoğlu, B. (2022). Emres: A new emotional respondent robot. *IEEE Transactions on Cognitive and Developmental Systems*, *14*(2), 772–780. https://doi.org/10.1109/TCDS.2021.3120562

Benedetto, A., Morrone, M. C., & Tomassini, A. (2020). The common rhythm of action and perception. *Journal of Cognitive Neuroscience*, *32*, 187–200. https://doi.org/10.1162/jocn_a_01436

Benedictis, R. D., Umbrico, A., Fracasso, F., Cortellessa, G., Orlandini, A., & Cesta, A. (2023). A dichotomic approach to adaptive interaction for socially assistive robots. *User Modeling and User-Adapted Interaction*, *33*, 293–331. https://doi.org/10.1007/s11257-022-09347-6

Bettosi, C., Baillie, L., Ross, M. K., & Broz, F. (2024). A systematic approach to modeling structured behavior in social robots. *International Symposium on Technological Advances in Human-Robot Interaction*, 29–37. https://doi.org/10.1145/3648536.3648540

Bishop, C. M., & Bishop, H. (2024). Gradient descent. In *Deep learning: Foundations and concepts* (1st, pp. 209–232). Springer, Cham. https://doi.org/10.1007/978-3-031-45468-4_7

Blackburn, P., De Rijke, M., & Venema, Y. (2001). *Modal logic: Graph. darst* (1st, Vol. 53). Cambridge University Press. https://philpapers.org/rec/BLAML-2

Block, J., & Funder, D. C. (1986). Social roles and social perception. Individual differences in attribution and error. *Journal of Personality and Social Psychology*, *51*(6), 1200–1207. https://doi.org/10.1037/0022-3514.51.6.1200

Bonnabel, S., & Slotine, J.-J. (2015). A contraction theory-based analysis of the stability of the deterministic extended kalman filter. *IEEE Transactions on Automatic Control*, *60*, 565–569. https://doi.org/10.1109/TAC.2014.2336991

Bono, J. E., & Vey, M. A. (2007). Personality and emotional performance: Extraversion, neuroticism, and self-monitoring. *Journal of Occupational Health Psychology*, *12*(2), 177–192. https://doi.org/10.1037/1076-8998.12.2.177

Bosse, T., Memon, Z. A., & Treur, J. (2011). A recursive BDI agent model for theory of mind and its applications. *Applied Artificial Intelligence*, *25*(1), 1–44. https://doi.org/10.1080/08839514.2010.529259

Boutayeb, M., Rafaralahy, H., & Darouach, M. (1997). Convergence analysis of the extended kalman filter used as an observer for nonlinear deterministic discrete-time systems. *Automatic Control, IEEE Transactions on*, *42*, 581–586. https://doi.org/10.1109/9.566674

Bradwell, H. L., Noury, G. E. A., Edwards, K. J., Winnington, R., Thill, S., & Jones, R. B. (2021). Design recommendations for socially assistive robots for health and social care based on a large scale analysis of stakeholder positions: Social robot design recommendations. *Health Policy and Technology*, *10*(3), 100544. https://doi.org/10.1016/J.HLPT.2021.100544

Bratman, M. (1987). *Intention, Plans, and Practical Reason* (1st). Harvard University Press. https://web.stanford.edu/group/cslipublications/cslipublications/site/1575861925.shtml

Broersen, J., Dastani, M., Hulstijn, J., Huang, Z., & Van Der Torre, L. (2001). The BOID architecture conflicts between beliefs, obligations, intentions and desires. *Proceedings of the fifth international conference on Autonomous agents*, 9–16. https://doi.org/10.1145/375735

Buzsáki, G., & Draguhn, A. (2004). Neuronal oscillations in cortical networks. *Science*, *304*, 1926–1929. https://doi.org/10.1126/science.1099745

Canli, T., Zhao, Z., Desmond, J. E., Kang, E., Gross, J., & Gabrieli, J. D. (2001). An fMRI study of personality influences on brain reactivity to emotional stimuli. *Behavioral Neuroscience*, *115*(1), 33–42. https://doi.org/10.1037/0735-7044.115.1.33

Cao, H. L., Beir, A. D., Esteban, P. G., Simut, R., Perre, G. V. D., Lefeber, D., & Vanderborght, B. (2017). An end-user interface to generate homeostatic behavior for nao robot in robot-assisted social therapies. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10306*, 609–619. https://doi.org/10.1007/978-3-319-59147-6_52

Cao, H. L., Esteban, P. G., Beir, A. D., Simut, R., Perre, G. V. D., Lefeber, D., & Vanderborght, B. (2017). A survey on behavior control architectures for social robots in healthcare interventions. *International Journal of Humanoid Robotics*, *14*(4), 1–24. https://doi.org/10.1142/S0219843617500219

Cao, H. L., Perre, G. V. D., Kennedy, J., Senft, E., Esteban, P. G., Beir, A. D., Simut, R., Belpaeme, T., Lefeber, D., & Vanderborght, B. (2019). A personalized and platform-independent behavior control system for social robots in therapy: Development and applications. *IEEE Transactions on Cognitive and Developmental Systems*, *11*(3), 334–346. https://doi.org/10.1109/TCDS.2018.2795343

Carvalho, J. P., & Tomé, J. A. (2001). Rule based fuzzy cognitive maps - expressing time in qualitative system dynamics. *IEEE International Conference on Fuzzy Systems*, 280–283. https://doi.org/10.1109/FUZZ.2001.1007303

Céspedes, N., Múnera, M., Gómez, C., & Cifuentes, C. A. (2020). Social human-robot interaction for gait rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *28*(6), 1299–1307. https://doi.org/10.1109/TNSRE.2020.2987428

Céspedes, N., Raigoso, D., Múnera, M., & Cifuentes, C. A. (2021). Long-term social human-robot interaction for neurorehabilitation: Robots as a tool to support gait therapy in the pandemic. *Frontiers in Neurorobotics*, *15*, 1–12. https://doi.org/10.3389/FNBOT.2021.612034

Chella, A., & Pipitone, A. (2020). A cognitive architecture for inner speech. *Cognitive Systems Research*, *59*, 287–292. https://doi.org/10.1016/j.cogsys.2019.09.010

Clabaugh, C., Mahajan, K., Jain, S., Pakkar, R., Becerra, D., Shi, Z., Deng, E., Lee, R., Ragusa, G., & Matarić, M. (2019). Long-term personalization of an in-home socially assistive robot for children with autism spectrum disorders. *Frontiers in Robotics and AI*, *6*(110), 1–18. https://doi.org/10.3389/frobt.2019.00110

Collins, E. C., Prescott, T. J., Mitchinson, B., & Conran, S. (2015). Miro: A versatile biomimetic edutainment robot. *Proceedings of the 12th International Conference on Advances in Computer Entertainment Technology*, 1–4. https://doi.org/10.1145/2832932.2832978

Cooper, S., Di Fava, A., Vivas, C., Marchionni, L., & Ferro, F. (2020). Ari: The social assistive robot and companion. *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 745–751. https://doi.org/10.1109/RO-MAN47096.2020.9223470

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, *13*(1), 21–27. https://doi.org/10.1109/TIT.1967.1053964

Cuijpers, R., & Knops, M. (2015). Motions of robots matter! the social effects of idle and meaningful motions. *7th International Conference, ICSR 2015*, 174–183. https://doi.org/10.1007/978-3-319-25554-5_18

Dell'Anna, D., & Jamshidnejad, A. (2022). Evolving fuzzy logic systems for creative personalized socially assistive robots. *Engineering Applications of Artificial Intelligence*, *114*, 105064. https://doi.org/10.1016/j.engappai.2022.105064

Dell'Anna, D., & Jamshidnejad, A. (2024). Sonar: An adaptive control architecture for social norm aware robots. *International Journal of Social Robotics*, *16*, 1–32. https://doi.org/10.1007/s12369-024-01172-8

Dennett, D. C. (1987). *The Intentional Stance* (6th). MIT Press. https://psycnet.apa.org/record/1987-98612-000

Dissing, L., & Bolander, T. (2020). Implementing theory of mind on a robot using dynamic epistemic logic. *Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)*, 1615–1621. https://doi.org/10.24963/ijcai.2020/224

Duchetto, F. D., Baxter, P., & Hanheide, M. (2020). Are you still with me? Continuous engagement assessment from a robot's point of view. *Frontiers in Robotics and AI*, *7*, 116. https://doi.org/10.3389/FROBT.2020.00116/BIBTEX

Dziergwa, M., Kaczmarek, M., Kaczmarek, P., Kedzierski, J., & Wadas-Szydłowska, K. (2018). Long-term cohabitation with a social robot: A case study of the influence of human attachment patterns. *International Journal of Social Robotics*, *10*, 163–176. https://doi.org/10.1007/S12369-017-0439-2

Ekman, P., & Cordaro, D. (2011). What is meant by calling emotions basic. *Emotion Review*, *3*(4), 364–370. https://doi.org/10.1177/1754073911410740

Farrell, S., & Lewandowsky, S. (2011). *Computational modeling in cognition: Principles and practice* (1st). SAGE Publications. https://doi.org/10.4135/9781483349428

Farrell, S., & Lewandowsky, S. (2018). *Computational modeling of cognition and behavior* (1st). Cambridge University Press. https://doi.org/10.1017/CBO9781316272503

Feil-Seifer, D., & Matarić, M. J. (2005). Defining socially assistive robotics. *Proceedings of the 9th International Conference on Rehabilitation Robotics*, 465–468. https://doi.org/10.1109/ICORR.2005.1501143

Feingold Polak, R., & Tzedek, S. L. (2020). Social robot for rehabilitation: Expert clinicians and post-stroke patients' evaluation following a long-term intervention. *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, 151–160. https://doi.org/10.1145/3319502.3374797

Feng, B., Fu, M., Ma, H., Xia, Y., & Wang, B. (2014). Kalman filter with recursive covariance estimation—sequentially estimating process noise covariance. *IEEE Transactions on Industrial Electronics*, *61*(11), 6253–6263. https://doi.org/10.1109/TIE.2014.2301756

Filippini, C., Spadolini, E., Cardone, D., Bianchi, D., Preziuso, M., Sciarretta, C., del Cimmuto, V., Lisciani, D., & Merla, A. (2021). Facilitating the child–robot interaction by endowing the robot with the capability of understanding the child engagement: The case of mio amico robot. *International Journal of Social Robotics*, *13*, 677–689. https://doi.org/10.1007/S12369-020-00661-W

Fumero, A., & Ahmetoglu, G. (2008). The effect of personality on reasoning. *Nature Precedings*, *3*, 1–9. https://doi.org/10.1038/NPRE.2008.2099.1

Gao, Y., Barendregt, W., Obaid, M., & Castellano, G. (2018). When robot personalisation does not help: Insights from a robot-supported learning study. *RO-MAN 2018 - 27th IEEE International Symposium on Robot and Human Interactive Communication*, 705–712. https://doi.org/10.1109/ROMAN.2018.8525832

Gao, Y., Yang, F., Frisk, M., Hernández, D., Peters, C. E., & Castellano, G. (2018). Social behavior learning with realistic reward shaping. *ArXiv*. abs/1810.06979

George, J. M., & Dane, E. (2016). Affect, emotion, and decision making. *Organizational Behavior and Human Decision Processes*, *136*, 47–55. https://doi.org/10.1016/j.obhdp.2016.06.004

Giorgi, I., Tirotto, F. A., Hagen, O., Aider, F., Gianni, M., Palomino, M., & Masala, G. L. (2022). Friendly but faulty: A pilot study on the perceived trust of older adults in a social robot. *IEEE Access*, *10*, 92084–92096. https://doi.org/10.1109/ACCESS.2022.3202942

Gordon, G., Spaulding, S., Westlund, J., Lee, J., Plummer, L., Martinez, M., Das, M., & Breazeal, C. (2016). Affective personalization of a social robot tutor for children's second language skills. *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 3951–3957. https://doi.org/10.1609/aaai.v30i1.9914

Gray, J., & Breazeal, C. (2014). Manipulating mental states through physical action. *International Journal of Social Robotics*, *6*, 315–327. https://doi.org/10.1007/s12369-014-0234-2

Grewal, M. S. (2008). *Kalman filtering: Theory and practice with matlab* (3rd). Wiley. https://doi.org/10.1002/9780470377819

Guest, O., & Martin, A. E. (2021). How Computational Modeling Can Force Theory Building in Psychological Science. *Perspectives on Psychological Science*, *16*(4), 789–802. https://doi.org/10.1177/1745691620970585

Hansen, L. D., & Bolander, T. (2020). Implementing theory of mind on a robot using dynamic epistemic logic. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)*, 1615–1621. https://doi.org/10.24963/ijcai.2020/224

Heckerman, D. (2008). A tutorial on learning with bayesian networks. In *Innovations in bayesian networks: Theory and applications* (1st ed., pp. 33–82, Vol. 156). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-85066-3_3

Heredia, J., Lopes-Silva, E., Cardinale, Y., Diaz-Amado, J., Dongo, I., Graterol, W., & Aguilera, A. (2022). Adaptive multimodal emotion detection architecture for social robots. *IEEE Access*, *10*, 20727–20744. https://doi.org/10.1109/ACCESS.2022.3149214

Hussain, N., Erzin, E., Sezgin, T. M., & Yemez, Y. (2022). Training socially engaging robots: Modeling backchannel behaviors with batch reinforcement learning. *IEEE Transactions on Affective Computing*, *13*, 1840–1853. https://doi.org/10.1109/TAFFC.2022.3190233

Jara-Ettinger, J. (2019). Theory of mind as inverse reinforcement learning. *Current Opinion in Behavioral Sciences*, *29*, 105–110. https://doi.org/10.1016/J.COBEHA.2019.04.010

Jara-Ettinger, J., Gweon, H., Schulz, L. E., & Tenenbaum, J. B. (2016). The Naïve Utility Calculus: Computational Principles Underlying Commonsense Psychology. *Trends in Cognitive Sciences*, *20*(8), 589–604. https://doi.org/10.1016/j.tics.2016.05.011

John, G. H., & Langley, P. (1995). Estimating continuous distributions in bayesian classifiers. *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, 338–345. https://dl.acm.org/doi/10.5555/2074158.2074196

John, O., Naumann, L. P., & Soto, C. J. (2008). Paradigm Shift to the Integrative Big-Five Trait Taxonomy: History, Measurement, and Conceptual Issues. In *Handbook of personality: Theory and research* (3rd, pp. 114–158). Guilford Press. https://psycnet.apa.org/record/2008-11667-004

Kedzierski, J., Muszynski, R., Zoll, C., Oleksy, A., & Frontkiewicz, M. (2013). Emys-emotive head of a social robot. *International Journal of Social Robotics*, *5*. https://doi.org/10.1007/s12369-013-0183-1

Khosla, R., Chu, M.-T., Khaksar, S. M. S., Nguyen, K., & Nishida, T. (2019). Engagement and experience of older people with socially assistive robots in home care. *Assistive Technology*, *33*, 1–15. https://doi.org/10.1080/10400435.2019.1588805

Kidd, C. D., & Breazeal, C. (2008). Robots at home: Understanding long-term human-robot interaction. *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 3230–3235. https://doi.org/10.1109/IROS.2008.4651113

Kim, Y., & Bang, H. (2018). Introduction to kalman filter and its applications. In *Introduction and implementations of the kalman filter* (1st, pp. 7–22). IntechOpen. https://doi.org/10.5772/intechopen.80600

Ko, W.-R., Lee, J., Jang, M., & Kim, J. (2020). End-to-end learning of social behaviors for humanoid robots. *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 1200–1205. https://doi.org/10.1109/SMC42975.2020.9283177

Kosko, B. (1986). Fuzzy cognitive maps. *International Journal of Man-Machine Studies*, *24*(1), 65–75. https://doi.org/10.1016/S0020-7373(86)80040-2

Kramer, O. (2017). *Genetic algorithms essentials* (1st). Springer, Cham. https://doi.org/10.1007/978-3-319-52156-5

Kubota, A., Peterson, E. I., Rajendren, V., Kress-Gazit, H., & Riek, L. D. (2020). Jessie: Synthesizing social robot behaviors for personalized neurorehabilitation and beyond. *ACM/IEEE International Conference on Human-Robot Interaction*, 121–129. https://doi.org/10.1145/3319502.3374836

Kwon, D.-S., Chung, M. J., Park, J. C., Yoo, C. D., Jee, E.-S., Park, K.-S., Kim, Y.-M., Kim, H.-R., Park, J.-C., Min, H.-J., Park, J. W., Yun, S., & Lee, K.-W. (2008). Emotional Exchange of a Socially Interactive Robot. *17th IFAC World Congress*, 4330–4335. https://doi.org/10.3182/20080706-5-kr-1001.00729

Laird, J. E., Lebiere, C., & Rosenbloom, P. S. (2017). A standard model of the mind: toward a common computational framework across artificial intelligence, cognitive science, neuroscience, and robotics. *AI Magazine*, *38*(4), 13–26. https://doi.org/10.1609/AIMAG.V38I4.2744

Lazzeri, N., Mazzei, D., Cominelli, L., Cisternino, A., & Rossi, D. E. D. (2018). Designing the mind of a social robot. *Applied Sciences*, *8*(2), 302. https://doi.org/10.3390/app8020302

Lee, J. J., Sha, F., & Breazeal, C. (2019). A bayesian theory of mind approach to nonverbal communication. *ACM/IEEE International Conference on Human-Robot Interaction*, 487–496. https://doi.org/10.1109/HRI.2019.8673023

Lee, M. H., Siewiorek, D. P., Smailagic, A., Bernardino, A., & i Badia, S. B. (2023). Design, development, and evaluation of an interactive personalized social robot to monitor and coach post-stroke rehabilitation exercises. *User Modeling and User-Adapted Interaction*, *33*, 545–569. https://doi.org/10.1007/S11257-022-09348-5

Leite, I., Martinho, C., & Paiva, A. (2013). Social Robots for Long-Term Interaction: A Survey. *International Journal of Social Robotics*, *5*(2), 291–308. https://doi.org/10.1007/s12369-013-0178-y

Lerner, J. S., Li, Y., Valdesolo, P., & Kassam, K. S. (2015). Emotion and decision making. *Annual Review of Psychology*, *66*, 799–823. https://doi.org/10.1146/annurev-psych-010213-115043

Li, M., Guo, F., Wang, X., Chen, J., & Ham, J. (2023). Effects of robot gaze and voice human-likeness on users' subjective perception, visual attention, and cerebral activity in voice conversations. *Computers in Human Behavior*, *141*, 107645. https://doi.org/10.1016/j.chb.2022.107645

Lichess.org. (2024, February). Lichess.org open database [Original dataset of chess puzzles]. https://database.lichess.org/#puzzles

Luperto, M., Monroy, J., Moreno, F. A., Lunardini, F., Renoux, J., Krpic, A., Galindo, C., Ferrante, S., Basilico, N., Gonzalez-Jimenez, J., & Borghese, N. A. (2023). Seeking at-home long-term autonomy of assistive mobile robots through the integration with an iot-based monitoring system. *Robotics and Autonomous Systems*, *161*, 104346. https://doi.org/10.1016/J.ROBOT.2022.104346

Luperto, M., Romeo, M., Lunardini, F., Monroy, J., García, D. H., Abbate, C., Cangelosi, A., Ferrante, S., Gonzalez-Jimenez, J., Basilico, N., & Borghese, N. A. (2024). Exploring the viability of socially assistive robots for at-home cognitive monitoring: Potential and limitations. *International Journal of Social Robotics*, *17*, 823–841. https://doi.org/10.1007/s12369-024-01158-6

Macalupu, V., Miller, E., Martin, L., & Caldwell, G. (2025). Human–robot interactions and experiences of staff and service robots in aged care. *Scientific Reports*, *15*, 1–15. https://doi.org/10.1038/S41598-025-86255-W

Magyar, G., & Vircikova, M. (2015). Proposal of a cloud-based agent for social human-robot interaction that learns from the human experimenters. *IEEE 19th International Conference on Intelligent Engineering Systems 2015*, 107–111. https://doi.org/10.1109/INES.2015.7329648

Magyar, G., Vircikova, M., & Sincak, P. (2016). Interactive q-learning for social robots that learn from the wizard: A pilot study. *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 3683–3687. https://doi.org/10.1109/SMC.2016.7844806

Mahmoud, M. S., & Singh, M. G. (2012). *Discrete systems: Analysis, control and optimization* (1st). Springer, Cham. https://doi.org/10.1007/978-3-642-82327-5

Mareschal, D., & Thomas, M. (2007). Computational Modeling in Developmental Psychology. *IEEE Transactions on Evolutionary Computation*, *11*(2), 137–150. https://doi.org/10.1109/TEVC.2006.890232

Maroto-Gómez, M., Alonso-Martín, F., Malfaz, M., Castro-González, Á., Castillo, J. C., & Salichs, M. Á. (2023). A systematic literature review of decision-making and control systems for autonomous and social robots. *International Journal of Social Robotics*, *15*, 745–789. https://doi.org/10.1007/S12369-023-00977-3

Maroto-Gómez, M., Malfaz, M., Castro-González, Á., Álvarez Arias, S., & Ángel Salichs, M. (2024). Deep reinforcement learning for the biologically inspired social behaviour of autonomous robots acting in dynamic environments. *IEEE Access*, *12*, 180146–180160. https://doi.org/10.1109/ACCESS.2024.3508139

Maroto-Gómez, M., Malfaz, M., Castro-González, Á., & Salichs, M. Á. (2023). A motivational model based on artificial biological functions for the intelligent decision-making of social robots. *Memetic Computing*, *15*, 237–257. https://doi.org/10.1007/s12293-023-00390-3

Martín, C. A., Rivera, D. E., Riley, W. T., Hekler, E. B., Buman, M. P., Adams, M. A., & King, A. C. (2014). A dynamical systems model of social cognitive theory. *2014 American Control Conference*, 2407–2412. https://doi.org/10.1109/ACC.2014.6859463

Martinez-Martin, E., & Cazorla, M. (2019). A socially assistive robot for elderly exercise promotion. *IEEE Access*, *7*, 75515–75529. https://doi.org/10.1109/ACCESS.2019.2921257

Mascaro, E. V., & Lee, D. (2025). Robot behavior generation for social human-robot interaction. *International Journal of Social Robotics 2025*, *11*, 1–20. https://doi.org/10.1007/S12369-025-01333-3

Matarić, M. J., & Scassellati, B. (2016). Socially Assistive Robotics. In *Springer handbook of robotics* (1st, pp. 1973–1993). Springer. https://dblp.uni-trier.de/rec/reference/robo/MataricS16.html

Matheus, K., Ramnauth, R., Scassellati, B., & Salomons, N. (2025). Long-term interactions with social robots: Trends, insights, and recommendations. *ACM Transactions on Human-Robot Interaction*, *14*(3), 1–42. https://doi.org/10.1145/3729539

Matisko, P., & Havlena, V. (2013). Noise covariance estimation for kalman filter tuning using bayesian approach and monte carlo. *International Journal of Adaptive Control and Signal Processing*, *27*, 957–973. https://doi.org/10.1002/acs.2369

Matz, J., Birouche, A., Mourllion, B., Bouziani, F., & Basset, M. (2020). Parameter identification for nonlinear models from a state-space approach. *IFAC-PapersOnLine*, *53*(2), 13910–13915. https://doi.org/10.1016/j.ifacol.2020.12.905

Mazzei, D., Cominelli, L., Lazzeri, N., Zaraki, A., & Rossi, D. D. (2014). I-clips brain: A hybrid cognitive system for social robots. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *8608 LNAI*, 213–224. https://doi.org/10.1007/978-3-319-09435-9_19

Mehra, R. K. (1970). On the identification of variances and adaptive kalman filtering. *IEEE Transactions on Automatic Control*, *15*(2), 494–505. https://doi.org/10.1109/TAC.1970.1099422

Mele, A. R. (1997). Real self-deception. *Behavioral and Brain Sciences*, *20*(1), 91–136. https://doi.org/10.1017/S0140525X97420034

Merino-Fidalgo, S., Sánchez-Girón, C., Zalama, E., Gómez-García-Bermejo, J., & Duque-Domingo, J. (2025). Behavior tree generation and adaptation for a social robot control with llms. *Robotics and Autonomous Systems*, *194*, 105165. https://doi.org/10.1016/j.robot.2025.105165

Mishra, C., & Skantze, G. (2022). Knowing where to look: A planning-based architecture to automate the gaze behavior of social robots. *31st IEEE International Conference on Robot and Human Interactive Communication: Social, Asocial, and Antisocial Robots, RO-MAN 2022*, 1201–1208. https://doi.org/10.1109/RO-MAN53752.2022.9900740

Mitchinson, B., & Prescott, T. J. (2016). Miro: A robot "mammal" with a biomimetic brain-based control system. *Biomimetic and Biohybrid Systems 5th International Conference*, 179–191. https://doi.org/10.1007/978-3-319-42417-0_17

Morão Patrício, M. L., & Jamshidnejad, A. (2023a). Dynamic mathematical models of theory of mind for socially assistive robots. *IEEE Access*, *11*, 103956–103975. https://doi.org/10.1109/ACCESS.2023.3316603

Morão Patrício, M. L., & Jamshidnejad, A. (2023b). Supplementary data to the paper: Dynamic mathematical models of theory of mind for socially assistive robots. https://doi.org/10.4121/0cc4d150-1c9d-4b54-9e75-fbe7dd3efacf.v1

Morão Patrício, M. L., & Jamshidnejad, A. (2025a). Data from experiments of "leveraging systems and control theory for social robotics: A model-based behavioral control approach to human-robot interaction". https://doi.org/10.4121/ccadc914-9502-46d6-9ba5-fef581f2933f

Morão Patrício, M. L., & Jamshidnejad, A. (2025b). A systems-theoretic approach to mental state estimation for theory-of-mind-aware social robots. *IEEE Access*, *13*, 158467–158482. https://doi.org/10.1109/ACCESS.2025.3607165

Morão Patrício, M. L., & Jamshidnejad, A. (2026a). Behavioral control approaches in social robotics: A functional role synthesis and roadmap for sustained human-robot interaction. *ACM Transactions on Human-Robot Interaction*.

Morão Patrício, M. L., & Jamshidnejad, A. (2026b). Leveraging systems and control theory for social robotics: A model-based behavioral control approach to human-robot interaction. *Applied Intelligence*, *56*, 141. https://doi.org/10.1007/s10489-026-07100-9

Morari, M., & H. Lee, J. (1999). Model predictive control: Past, present and future. *Computers & Chemical Engineering*, *23*(4), 667–682. https://doi.org/10.1016/S0098-1354(98)00301-9

Mourhir, A., Rachidi, T., Papageorgiou, E. I., Karim, M., & Alaoui, F. S. (2016). A cognitive map framework to support integrated environmental assessment. *Environmental Modelling & Software*, *77*, 81–94. https://doi.org/10.1016/J.ENVSOFT.2015.11.018

Munster, M., & Jamshidnejad, A. (2025). Personalized human-robot cognitive interaction via a novel fuzzy logic control and learning-based paradigm. *IEEE Access*, *13*, 112568–112593. https://doi.org/10.1109/ACCESS.2025.3584194

Nakamura, M. (1982). Relationship between steady state kalman filter gain and noise variances. *International Journal of Systems Science*, *13*, 1153–1163. https://doi.org/10.1080/00207728208926419

Navarro-Barrientos, J. E., Rivera, D. E., & Collins, L. M. (2011). A dynamical model for describing behavioural interventions for weight loss and body composition change. *Mathematical and Computer Modelling of Dynamical Systems*, *17*(2), 183–203. https://doi.org/10.1080/13873954.2010.520409

Nocentini, O., Fiorini, L., Acerbi, G., Sorrentino, A., Mancioppi, G., & Cavallo, F. (2019). A survey of behavioral models for social robots. *Robotics 2019, Vol. 8, Page 54*, *8*(3), 54. https://doi.org/10.3390/ROBOTICS8030054

Odelson, B. J., Rajamani, M. R., & Rawlings, J. B. (2006). A new autocovariance least-squares method for estimating noise covariances. *Automatica*, *42*(2), 303–308. https://doi.org/10.1016/J.AUTOMATICA.2005.09.006

Ogata, K. (1990). *Modern control engineering* (4th). Prentice Hall. https://dl.acm.org/doi/10.5555/533046

Ong, D. C., Zaki, J., & Goodman, N. D. (2020). Computational models of emotion inference in theory of mind: A review and roadmap. *Topics in Cognitive Science*, *11*(2), 338–357. https://doi.org/10.1111/tops.12371

Páez, J., González, E., & Impedovo, M. (2020). Hrs-edu: Architecture to control social robots in education. *Advances in Intelligent Systems and Computing*, *1241*, 117–126. https://doi.org/10.1007/978-3-030-52538-5_13

Pandey, A. K., & Gelin, R. (2018). A mass-produced sociable humanoid robot: Pepper: The first machine of its kind. *IEEE Robotics & Automation Magazine*, *PP*, 1–1. https://doi.org/10.1109/MRA.2018.2833157

Pandey, D., Subedi, A., & Mishra, D. (2022). Improving language skills and encouraging reading habits in primary education: A pilot study using nao robot. *2022 IEEE/SICE International Symposium on System Integration (SII)*, 827–832. https://doi.org/10.1109/SII52469.2022.9708843

Parent, X., & van Der Torre, L. (2013). *Handbook of deontic logic and normative systems* (1st). College Publications. https://philpapers.org/rec/GABHOD-4

Patacchiola, M., & Cangelosi, A. (2020). A developmental cognitive architecture for trust and theory of mind in humanoid robots. *IEEE Transactions on Cybernetics*, *52*(3), 1–13. https://doi.org/10.1109/TCYB.2020.3002892

Pioggia, G., Ahluwalia, A., Carpi, F., Marchetti, A., Ferro, M., Rocchia, W., & De Rossi, D. (2004). Face: Facial automaton for conveying emotions. *Appl. Bionics Biomechanics*, *1*(2), 91–100. https://doi.org/10.1533/abib.2004.1.2.91

Popper, K. (1994). *The myth of the framework* (1st). Routledge. https://philomatica.org/wp-content/uploads/2023/05/myth_of_fw_popper.pdf

Premack, D., & Woodruff, G. (1978). Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, *1*(4), 515–526. https://doi.org/10.1017/S0140525X00076512

Rabinowitz, N. C., Perbet, F., Song, H. F., Zhang, C., Eslami, S. M., & Botvinick, M. (2018). Machine theory of mind. *Proceedings of the 35th International Conference on Machine Learning*, 4218–4227. https://doi.org/10.48550/arXiv.1802.07740

Raghunathan, R., & Pham, M. T. (1999). All negative moods are not equal: Motivational influences of anxiety and sadness on decision making. *Organizational Behavior and Human Decision Processes*, *79*(1), 56–77. https://doi.org/10.1006/obhd.1999.2838

Rao, A. S. (1995). Decision procedures for prepositional linear-time belief-desire-intention logics. *International Joint Conferences on Artificial Intelligence: Agent Theories, Architectures, and Languages Workshop*, 33–48. https://doi.org/10.1007/3540608052_57

Riley, W., Rivera, D., Atienza, A., Nilsen, W., Allison, S., & Mermelstein, R. (2011). Health behavior models in the age of mobile interventions: Are our theories up to the task? *Translational Behavioral Medicine*, *1*, 53–71. https://doi.org/10.1007/s13142-011-0021-7

Robaczewski, A., Bouchard, J., Bouchard, K., & Gaboury, S. (2021). Socially assistive robots: The specific case of the nao. *International Journal of Social Robotics*, *13*. https://doi.org/10.1007/s12369-020-00664-7

Rodríguez-Lera, F. J., Matellán-Olivera, V., Conde-González, M. Á., & Martín-Rico, F. (2018). Himop: A three-component architecture to create more human-acceptable social-assistive robots. *Cognitive Processing*, *19*, 233–244. https://doi.org/10.1007/s10339-017-0850-5

Rossi, S., Larafa, M., & Ruocco, M. (2020). Emotional and behavioural distraction by a social robot for children anxiety reduction during vaccination. *International Journal of Social Robotics*, *12*, 765–777. https://doi.org/10.1007/S12369-019-00616-W

Russell, J. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, *39*, 1161–1178. https://doi.org/10.1037/h0077714

Saxe, R., & Houlihan, S. D. (2017). Formalizing emotion concepts within a Bayesian model of theory of mind. *Current Opinion in Psychology*, *17*, 15–21. https://doi.org/10.1016/j.copsyc.2017.04.019

Scassellati, B. (2002). Theory of mind for a humanoid robot. *Autonomous Robots*, *12*(1), 13–24. https://doi.org/10.1023/A:1013298507114

Scassellati, B., Boccanfuso, L., Huang, C. M., Mademtzi, M., Qin, M., Salomons, N., Ventola, P., & Shic, F. (2018). Improving social skills in children with asd using a long-term, in-home social robot. *Science Robotics*, *3*(21), 1–9. https://doi.org/10.1126/scirobotics.aat7544

Scherer, K. R. (2009). The dynamic architecture of emotion: Evidence for the component process model. *Cognition and Emotion*, *23*(7), 1307–1351. https://doi.org/10.1080/02699930902928969

Schicchi, D., & Pilato, G. (2018). A social humanoid robot as a playfellow for vocabulary enhancement. *Proceedings - 2nd IEEE International Conference on Robotic Computing, IRC 2018*, *2018-January*, 205–208. https://doi.org/10.1109/IRC.2018.00044

Schrum, M., Park, C. H., & Howard, A. (2019). Humanoid therapy robot for encouraging exercise in dementia patients. *ACM/IEEE International Conference on Human-Robot Interaction*, *2019-March*, 564–565. https://doi.org/10.1109/HRI.2019.8673155

Sequeira, P., Alves-Oliveira, P., Ribeiro, T., Tullio, E. D., Petisca, S., Melo, F. S., Castellano, G., & Paiva, A. (2016). Discovering social interaction strategies for robots from restricted-perception wizard-of-oz studies. *ACM/IEEE International Conference on Human-Robot Interaction*, *2016*, 197–204. https://doi.org/10.1109/HRI.2016.7451752

Shiarlis, K., Messias, J., & Whiteson, S. (2017). Acquiring social interaction behaviours for telepresence robots via deep learning from demonstration. *IEEE International Conference on Intelligent Robots and Systems*, *2017*, 37–42. https://doi.org/10.1109/IROS.2017.8202135

Short, E., Short, D., Fu, Y., & Matarić, M. (2020). Sprite: Stewart platform robot for interactive tabletop engagement. https://doi.org/10.48550/arXiv.2011.05786

Simon, D. (2006). *Optimal state estimation: Kalman, h∞, and nonlinear approaches* (1st). Wiley Blackwell. https://doi.org/10.1002/0470045345

Skiena, S. S. (2008). *The algorithm design manual* (3rd, Vol. 2). Springer. https://doi.org/10.1007/978-3-030-54256-6

Smaldino, P. (2023). *Modeling social behavior: Mathematical and agent-based models of social dynamics and cultural evolution* (1st). Princeton University Press. https://doi.org/10.1257/jel.62.4.1682.r7

Smith, N. J. (2012). *Logic: The laws of truth*. Princeton University Press. https://www.academia.edu/39985799/The_Laws_of_Truth_Smith_Nicholas_J_J

Snyder, M., & Ickes, W. (1985). Personality and social behavior. In *Handbook of social psychology* (3rd, pp. 883–947, Vol. 2). Random House. https://www.researchgate.net/publication/273123749_Personality_and_social_behavior

SoftBank Robotics. (n.d.). NAO Robot [Accessed: 2025-04-14]. https://www.softbankrobotics.com/emea/en/nao

Stigler, S. M. (1982). Thomas bayes's bayesian inference. *Journal of the Royal Statistical Society. Series A (General)*, *145*(2), 250–258. https://doi.org/10.2307/2981538

Stylios, C. D., & Groumpos, P. P. (2004). Modeling complex systems using fuzzy cognitive maps. *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, *34*(1), 155–162. https://doi.org/10.1109/TSMCA.2003.818878

Sueki, K., Nishizawa, S., Yamaura, T., & Tomita, H. (2022). Precision and convergence speed of the ensemble kalman filter-based parameter estimation: Setting parameter uncertainty for reliable and efficient estimation. *Progress in Earth and Planetary Science*, *9*. https://doi.org/10.1186/s40645-022-00504-4

Sutton, R. S., Barto, A. G., et al. (1998). *Reinforcement learning: An introduction* (2nd, Vol. 1). MIT press. https://mitpress.mit.edu/9780262039246/reinforcement-learning

Tahir, Y., Dauwels, J., Thalmann, D., & Thalmann, N. M. (2020). A user study of a humanoid robot as a social mediator for two-person conversations. *International Journal of Social Robotics*, *12*, 1031–1044. https://doi.org/10.1007/s12369-018-0478-3

Tani, J., Mao, X., Zeng, Y., Zhao, Y., Zhang, T., Zhao, D., Zhao, F., & Lu, E. (2020). A brain-inspired model of theory of mind. *Frontiers in Neurorobotics*, *14*, 1–17. https://doi.org/10.3389/fnbot.2020.00060

Tapus, A., Mataric, M., & Scassellati, B. (2007). The grand challenges in socially assistive robotics. *IEEE Robotics and Automation Magazine*, *14*(1), 35–42. https://doi.org/10.1109/MRA.2007.339605

Tapus, A., & Mataric, M. J. (2008). Socially assistive robots: The link between personality, empathy, physiological signals, and task performance. *AAAI Spring Symposium*, 133–140. https://aaai.org/papers/0021-SS08-04-021-socially-assistive-robots-the-link-between-personality-empathyphysiological-signals-and-task-performance

Van Benthem, J. (1983). *The logic of time: A model-theoretic investigation into the varieties of temporal ontology and temporal discourse* (2nd, Vol. 156). Springer Science & Business Media. https://doi.org/10.1007/978-94-010-9868-7

Vossen, P., Baez, S., Bajcetić, L., & Kraaijeveld, B. (2018). Leolani: A reference machine with a theory of mind for social communication. *International Conference on Text, Speech, and Dialogue*, 15–25. https://doi.org/10.1007/978-3-030-00794-2_2

Welch, G., & Bishop, G. (1995). *An introduction to the kalman filter* (tech. rep.). USA, University of North Carolina at Chapel Hill. https://dl.acm.org/doi/book/10.5555/897831

Wellman, H. M., Cross, D., & Watson, J. (2001). Meta-analysis of theory-of-mind development: The truth about false belief. *Child Development*, *72*(3), 655–684. https://doi.org/10.1111/1467-8624.00304

Witkin, H. A. (1949). The nature and importance of individual differences in perception. *Journal of Personality*, *18*(2), 145–170. https://doi.org/10.1111/J.1467-6494.1949.TB01237.x

Wu, F., Luo, H., Jia, H., Zhao, F., Xiao, Y., & Gao, X. (2021). Predicting the noise covariance with a multitask learning model for kalman filter-based gnss/ins integrated navigation. *IEEE Transactions on Instrumentation and Measurement*, *70*, 1–13. https://doi.org/10.1109/TIM.2020.3024357

Yannakakis, G. N., & Togelius, J. (2018). *Artificial intelligence and games* (2nd, Vol. 2). Springer. https://doi.org/10.1007/978-3-031-83347-2

Yu, C., & Pei, H. (2021). Face recognition framework based on effective computing and adversarial neural network and its implementation in machine vision for social robots. *Computers and Electrical Engineering*, *92*, 107128. https://doi.org/10.1016/j.compeleceng.2021.107128

Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning—I. *Information Sciences*, *8*(3), 199–249. https://doi.org/10.1016/0020-0255(75)90036-5

Zadeh, L. A. (1988). Fuzzy logic. *Computer*, *21*(4), 83–93. https://doi.org/10.1109/2.53

Zak, S. H., et al. (2003). *Systems and control* (1st, Vol. 198). Oxford University Press. https://global.oup.com/academic/product/systems-and-control-9780195150117?cc=nl&lang=en&

Zaraki, A., Mazzei, D., Giuliani, M., & Rossi, D. D. (2014). Designing and evaluating a social gaze-control system for a humanoid robot. *IEEE Transactions on Human-Machine Systems*, *44*(2), 157–168. https://doi.org/10.1109/THMS.2014.2303083

Zelenski, J. (2007). The role of personality in emotion, judgment, and decision making. In *Do emotions help or hurt decision making? a hedgefoxian perspective* (1st, pp. 117–132). Russell Sage Foundation. https://www.researchgate.net/publication/275019980_The_role_of_personality_in_emotion_judgment_and_decision_making

Zhu, H., Liu, X., Su, H., Wang, W., & Yu, J. (2024). The methodology of quantitative social intention evaluation and robot gaze behavior control in multi-objects scenario. *IEEE Transactions on Cognitive and Developmental Systems*, *17*(2), 400–409. https://doi.org/10.1109/TCDS.2024.3461335

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Anahita, for our many interesting scientific discussions and for constantly challenging me not to settle, but to always strive to improve. Her guidance has shaped both this work and my growth as a researcher. I would also like to thank my promotor, Hans, for his words of wisdom and for always helping me keep sight of the bigger picture. I am deeply grateful for the time, insight, and support both of you have given me throughout this journey.

To Riccardo, I cannot thank you enough for everything you did for me over these past years. You have been my safe haven, my solid ground, and it is not a hyperbole to say that I would not have managed to complete this PhD if it were not for you. You taught me to be patient, trust myself, and to never give up. Thank you for always supporting me in my side projects (and even joining so many of them), for the travels we did together, and for building this house with me. You were by my side every step of the way, and for that, I could not be more thankful.

Thank you to my mom and dad, who, ever since I was a kid, encouraged me to always give my all in every task I set myself to achieve, to chase challenges and opportunities, and to fight for what I want. That shaped the researcher I became, and I wouldn't have reached this milestone otherwise. Thank you as well for the unconditional support you showed me when I decided to move abroad at the age of 20 to pursue my Master's degree. Despite not being your personal preference, you gave me everything I needed to take this leap, which in turn made this PhD possible. To my three little siblings, thank you for always making our house a joyful place, for the companionship, and for always making returning home feel so welcoming. In particular, thank you, Leonor, for filling my life with creativity and craziness, be it from the chaotic video calls at dinner time, to the creative conversations, and visits in the Netherlands. Thank you, Henrique, for being the sibling who thinks like I think, from the shared interests to the kind of understanding that's hard to put into words but impossible to miss. And thank you, Miguel, for always being there for me with your heart of gold, whether with a hug, kind words, or simply a joke. It amazes me how much you grew in these 4 years.

To my Aunts Sofia and Graça, thank you for making me look at life from a bright and hopeful perspective, for bringing fun and color to my world, and for teaching me that sometimes all I need is to take things easy. To my Grandma Leonor, thank you for being such a great inspiration and showing me from a young age that success has no gender, from being the first woman to get a driver's license in her entire hometown, to going to university and becoming a teacher, and for never giving up on her thoughts. You taught me to think critically, and to never be afraid to speak my mind. To my Grandma Julia and my Grandad Carlos, whom I lost since I started this journey, thank you for the love and kindness you always showed me. Thank you, Afonso, I really

cherish our long intellectual discussions and the nice times we spend during holidays. And thank you to Rosalba, Antonio, and Valeria, who became my second family in a different country, for always hosting me with such kindness and love, and for always helping us when we need it most.

To all the PhD candidates and graduates in C&S, thank you for the memories together over these past years. Thank you for the lunches (and lunch walks, of course), for the fun activities, from karaoke, bowling, dinners, beach volleyball, afternoons at Delftse Hout, and even a trip to Croatia to get the vitamin D we cannot collect in the Netherlands. To my paranymphs, Isabelle and Rowenna, thank you for being the people I can always count on, no matter what. To Isabelle, thank you for being my confidante, music partner, and my infinite source of goofiness. One day we will be touring the world together (or at least playing for five friends in my living room). Rowenna, when the faculty move happened, I was really sad; I was moved to a different floor than everyone I knew well. Little did I know that that would bring me close to the coolest, wisest PhD in C&S. Thank you for all your support, for helping me think straight when I freak out over a little detail, and for the company on the way home. I am so glad we took over the second floor together. To Jan, thank you for being the best office mate one could ask for. You knew exactly when to be fun and chaotic (although I am glad you can no longer throw rubber bands at me), but also were ready to give me a hug of support or a word of encouragement whenever I needed. You made these 4 years so much more fun, and it fills me with pride to have shared the PhD journey with you. To Giulia, thank you for always organizing the most fun and out-of-the-box plans, for getting enthusiastic about any DIY and craft project, and for always saying *yes* to any idea. Esther, thank you for making days at the office so much more fun, for being my Halloween costume partner, for making me laugh with your witty comebacks, and above all, for being such an inspiration of strength. Thank you to Jasper for the amazing cooking (including cheesecakes), Prashant for the crazy plans and pranks, Paul for the fancy cooking and dinner parties, and Malte for the mix of craziness and wisdom. Thanks to Liming and Filip for being such good office mates and for making the few days we were all together at the office so fun. A special thanks to Bob, Riccardo E., Sara, Sasha, Aidana, Fazlur, Alberto, Casper, Da Hwi, Nils, Quinten and Till for all the fun plans we did throughout the years. You made this journey so much more pleasant.

To my other PhD friends from the aerospace faculty: To Ata, thank you for your never-ending enthusiasm, for the wise words of advice, and for encouraging me to take a lighter look at things when they seem complicated. To Marina, thank you for the honest conversations and for always talking so openly to me about the struggles of living abroad and pursuing a PhD; our conversations often came at a time of need and gave me crucial perspective and comfort. To Kherlen, thank you for making my head hurt with all your paradoxical and mind-challenging questions, and for never letting us be low on mandarin stock.

Thank you to Manel, Carolina J., João N., João B., Carolina O., Leonor, Rubén, and Inês for being my small embassy of Portugal in Delft; you made me feel at home 2000km away. Thank you also to Manel for accepting the crazy challenge of joining our band (including putting up with my bossy attempts to bring order to our chaotic

rehearsals).

A big hug to all the teammates and coaches from Tutor, where I spent 3 years of my PhD. It was a pleasure training and playing by your side. A special thanks to my *bandidas*, Samantha and Marina, for distracting me with your Portuguese nonsense chats during practice and for our monthly brunches ever since, which brought me so much comfort and joy. An extra special thanks to Samantha for bridging the gap between science and art by beautifully reflecting my research on the cover of this thesis.

Thank you to João, Rajeeb, Aung, and all the other friends from our volleyball group, not only for all the amazing Sunday mornings playing volleyball for hours, but also for all the dinners, barbeques, practices, and crazy last-minute plans. It is incredible that a group that started as some near-strangers who like to prevent a ball from touching the ground would result in such great friendships and beautiful moments. You made these four years so much more worthwhile, and I hope our knees and ankles allow for many more to come. A special thanks to João, my first friend in the Netherlands, for the concerts, travels, cozy gatherings that felt just as good as any adventure, and for the honest conversations that reminded me how much we have in common.

Going further back in time, a big thanks to the two guys who started the aerospace engineering challenge with me in 2016, at Técnico, Diogo and Gonçalo; who, like me, decided to move abroad to pursue their dreams. The fact that our paths crossed and remain aligned after all these years is a great source of inspiration and comfort to me, and I am truly thankful for your friendship.

Finally, a big thank you to the friends who have been there since the beginning: Henrique C., Andreia, Francisa, Pedro, and Mafalda. Winnie the Pooh said, "How lucky I am to have something that makes saying goodbye so hard," and that is something that pretty much sums up how I felt when I left Portugal, in large part because of you. In particular, thank you to my oldest friend, Andreia, for proving to me that, even after these 25 years, friendships do last when roots are strong. And thank you to Henrique C., for being the friend who I can always turn to, for the long night drives that gave me so much perspective, for the heart-to-heart conversations, and above all, for showing me that distance is just a number.

Curriculum Vitæ

Maria Luís Morão Patrício

19-12-1998 Born in Lisbon, Portugal.

Education

2022–2026 PhD. Aerospace Engineering
Delft University of Technology
Thesis: Toward Socially Aware Robots: A systems-theoretic approach for behavioral control of social robots
Promotor: Prof. dr. ir. H. Hellendoorn
Copromotor: Dr. A. Jamshidnejad

2019–2021 MSc. Aerospace Engineering
Delft University of Technology
Thesis: Mathematical Modelling of Theory of Mind: Enabling Socially Assistive Robots to understand and predict humans in long-term interactions

2016–2019 BSc. Aerospace Engineering
Técnico, Universidade de Lisboa

Experience

2026-present Control Software Engineer at Allseas B.V.
Delft, the Netherlands

2020-2021 Internship at Nederlands Lucht- en Ruimtevaartcentrum (NLR)
Amsterdam, the Netherlands

Awards

2017/2018/2019	Honour Student Distinction Técnico, Universidade de Lisboa
2014	Third Place Portuguese National Physics Olympiads
2013	Third Place *Olimpíada Paulista de Matemática* (Portugal National Ranking)
2013	First Place Portuguese Regional Physics Olympiads
2012	Third Place, Ibero-American Math Competition

List of Publications

Journal Publications

1. Morão Patrício, M. L., & Jamshidnejad, A. (2023a). Dynamic mathematical models of theory of mind for socially assistive robots. *IEEE Access*, *11*, 103956–103975. https://doi.org/10.1109/ACCESS.2023.3316603
2. Morão Patrício, M. L., & Jamshidnejad, A. (2026b). Leveraging systems and control theory for social robotics: A model-based behavioral control approach to human-robot interaction. *Applied Intelligence*, *56*, 141. https://doi.org/10.1007/s10489-026-07100-9
3. Morão Patrício, M. L., & Jamshidnejad, A. (2025b). A systems-theoretic approach to mental state estimation for theory-of-mind-aware social robots. *IEEE Access*, *13*, 158467–158482. https://doi.org/10.1109/ACCESS.2025.3607165

Journal Publications, under review

4. Morão Patrício, M. L., & Jamshidnejad, A. (2026a). Behavioral control approaches in social robotics: A functional role synthesis and roadmap for sustained human-robot interaction. *ACM Transactions on Human-Robot Interaction* [*under review*]

Propositions

accompanying the dissertation

Toward Socially Aware Robots

A systems-theoretic approach for behavioral control of social robots

by

Maria Luís MORÃO PATRÍCIO

1. Social robots need to model and understand their users to achieve socially meaningful interactions *(Chapter 2)*.
2. For predictive models, increased detail does not necessarily guarantee improved accuracy *(Chapter 3)*.
3. Mental states that are not visible to humans may be inferable by a systems-theoretic state estimator *(Chapter 4)*.
4. No single method will ever be universally suited for social robot behavioral control *(Chapter 5)*.
5. The association of systems and control theory with physically grounded systems reflects convention rather than intrinsic limitation.
6. The application of machine learning methods in research without an understanding of their underlying mechanisms by the researcher fosters inefficiency.
7. Terminology constitutes a bigger barrier to collaboration between scientific fields than scientific content itself.
8. Excessive reliance on hypothesis-driven research may bias methodology and prevent paradigm changes.
9. Banning children from using social media protects, not harms, their freedom.
10. Sometimes the solution lies not in working an extra hour, but in taking an extra walk.

These propositions are regarded as opposable and defendable, and have been approved as such by the promotor prof. dr. ir. Hans Hellendoorn and the copromotor dr. A. Jamshidnejad.

Acronyms

BDI Belief-Desire-Intention. 20, 21, 30, 133, 159, 161, 162, 191–193

BN Bayesian Network. 176

CLIPS C Language Integrated Production System. 134

CMM Common Model of Mind. 22, 192

CNN Convolutional Neural Network. 148

DBN Dynamic Bayesian Network. 177

DBSoC Deep Behavioural Social Cloning. 147

DEL Dynamic Epistemic Logic. 139, 144, 148

EA Elicited attention. 133, 145, 165, 166, 170

EKF Extended Kalman Filter. 101–104, 107–110, 112, 113, 115–120, 124, 196, 197, 199

FCBT Fuzzy Cognitive Binary Tree. 141

FCM Fuzzy Cognitive Map. 18, 19, 31–34, 45, 141, 193, 199

FIS Fuzzy Inference System. 34, 38, 45, 199

FSM Finite State Machine. 134–139, 159, 160

hHRL Hierarchical Human-Robot Learning. 135, 136, 151

HiMop Hierarchy of needs, Motivational variables and Pool of FSM. 135, 138, 159, 164, 169

HRS-EDU Human-Robot Scaffolding architecture. 156, 161

HSRI Human-Social-Robot Interaction. 6–14, 50–52, 54, 56–59, 61–65, 70–75, 82, 84, 85, 87, 89, 91–96, 99, 101–106, 109, 112–116, 118, 120, 121, 123, 124, 126–129, 146, 147, 149, 157, 158, 162, 164, 170, 173–175, 178–183, 187, 190, 191, 194–199, 202, 209

JESSIE Just Express Specifications, Synthesize, and Interact. 136, 138

KF Kalman Filter. 102–105, 107, 111, 112, 123

LSTM Long short-term memory. 148

LTL Linear Temporal Logic. 136, 138, 140, 221

MAB Multi-Armed Bandit. 151, 219

MBC Model-Based Controller. 69, 70, 73, 77, 78, 81, 83–85, 88–90, 92, 93

ML Machine Learning. 4, 6, 10, 136, 143, 145, 147, 149, 150, 155, 156, 165

MMM Mathematical Model of Mind. 50, 51, 54–57, 59–64, 67–74, 76, 78, 81–85, 88, 93–95, 99–109, 112–114, 116, 123, 124, 209

MSE Mean Squared Error. 48, 51, 84, 85, 95, 105, 197

NN Neural Network. 21, 51, 52, 147–150, 156, 157, 169, 173, 182, 185, 187

PDDL Planning Domain Definition Language. 138, 140, 219

POMDP Partially Observable Markov Decision Process. 20, 21, 177

RBC Rule-Based Controller. 88–90

RDF Resource Description Framework. 138

RL Reinforcement Learning. 5, 51, 52, 128, 135, 136, 142, 148, 150–157, 176, 179, 183–186, 188

SAR Socially Assistive Robot. 2–6

SCT Social Cognitive Theory. 22

SOM Self-organizing Map. 156, 157, 170

SONAR SOcial Norm Aware Robots. 141, 142, 149, 162

SR Social Robot. 4–11, 13, 18, 19, 24, 26, 35, 48–57, 61, 62, 69, 70, 73, 93, 94, 96, 98–102, 123, 124, 126–129, 131–139, 141–153, 155–170, 172–176, 178–188, 190, 191, 193–199, 201, 202

ToM Theory of Mind. 4, 5, 18–24, 28, 31, 35–38, 40–45, 49, 50, 54, 55, 63, 65, 70, 94, 133, 148, 159, 161, 169, 173–175, 177, 183, 191–193, 210, 211

TPB Theory of Planned Behavior. 22, 192

WoZ Wizard-of-Oz. 143, 151, 156

The growth and aging of the world population is increasing the number of vulnerable individuals requiring continuous support, yet health and social care systems lack the resources to meet this demand. Social robots offer a promising solution, complementing existing care systems by supporting vulnerable individuals.

However, state-of-the-art social robots struggle to sustain long-term user engagement, which hinders their real-life applicability. In turn, user engagement depends on the behavioral control approaches used by social robots, which, unlike humans, typically fail to model or adapt to users' dynamic mental states.

To address this limitation, this thesis presents a systems and control-theoretic approach to steer the behavior of social robots. The framework models the dynamics of user mental states and leverages this to systematically generate robot behavior and to improve the robot's estimations of said states. In doing so, this thesis paves the way for social robots to meaningfully support individuals over sustained, real-world interactions.

