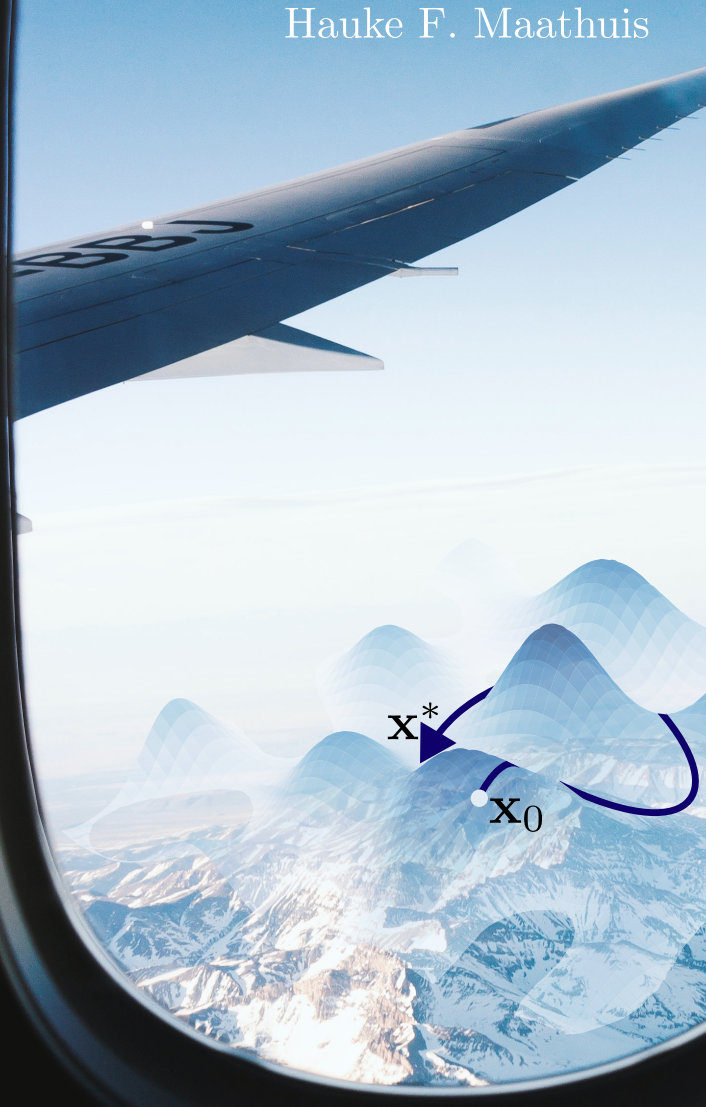
Constrained Bayesian Optimisation in High-Dimensional Spaces

Hauke F. Maathuis



TU Delft

Constrained Bayesian Optimisation in High-Dimensional Spaces

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

Wednesday, 1 April 2026 at 10:00 o'clock

by

Hauke Felix MAATHUIS

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

Composition of the doctoral committee:

Rector Magnificus	Chairperson
Prof.dr.ir. R. De Breuker	Delft University of Technology, promotor
Dr.ir. S. Giovani Pereira Castro	Delft University of Technology, copromotor
Independent members:	
Prof.dr.ir. R. Benedictus	Delft University of Technology
Prof. Dr. J.C. Chassaing	Sorbonne University, France
Prof. Dr. J. Morlier	ISAE-SUPAERO, France
Dr. E. Raponi	Leiden University
Non-Independent members:	
P. Higino Cabral	Embraer SA, Brazil
Reserve member:	
Prof.dr.ir. R.C. Alderliesten	Delft University of Technology

Keywords: Bayesian optimisation, black-box constraints, probabilistic surrogate models, Gaussian Processes, high-dimensional spaces, dimensionality reduction, multi-source information

Front & Back: Cover based on photo taken by Yuri Catalano (`https://www.pexels.com/@yuricatalano/`)

ISBN 978-94-6518-271-1

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

Für Katharina, Kaja, Mama & Papa.

Acknowledgements

This thesis marks the end of a four-year journey that, in hindsight, felt far too short. It allowed me to immerse myself in a subject that genuinely fascinates me and to grow, both academically and personally. I am aware that not every PhD experience unfolds this way, and I therefore feel especially fortunate. What truly shaped this time, however, were the wonderful people who accompanied me along the way. I would therefore like to use this space to express my gratitude to them.

First and foremost, I would like to thank my two supervisors, *Roeland* and *Saullo*, without whom this PhD would not have been possible. *Roeland*, van harte dank ik je voor het immense vertrouwen dat je me hebt gegeven, en voor al die vrijheid die ik heb gekregen om mijn passie te volgen. Ook dank voor al die verrijkende discussies, niet alleen over technische dingen, maar ook over alledaagse onderwerpen. En natuurlijk onze eindeloze "Friends"-grappen. Maar “There is nothing to tell.” Sommige verhalen beginnen precies zo, en hoewel deze thesis het einde van mijn promotie betekent, is het vooral het begin van een nieuw hoofdstuk. *Saullo*, for all your guidance and help throughout this journey I would like to say: muito obrigado! The insightful discussions we had, as well as the freedom you gave me, helped shape this thesis into what it is today. Beyond the technical side, you supported me immensely in navigating the academic world and acted as a mentor, especially during moments when I was unsure how to proceed.

Secondly, I would like to extend my gratitude to *Mike Osborne*, who kindly invited me to visit his research group. Thank you for the valuable discussions and for allowing me to dive deeper and deeper into the Bayesian world.

Many thanks to Embraer, especially *Alex* and *Pedro*, for their continuous support and for the fruitful discussions we had within the ATED project. I greatly appreciated the opportunity to connect academic research with real engineering challenges, and I am grateful for your valuable insights and collaboration throughout this work.

Furthermore, I would like to express my sincere gratitude to the members of my thesis committee for their invaluable time and dedication to this work. I am grateful to *Prof.dr.ir. R. Benedictus*, *Prof. Dr. J.C. Chassaing*, *Prof. Dr. J. Morlier*, and *Dr. E. Raponi* for their expert insights and thorough review. Thank you all for your careful reading of the manuscript and for the constructive feedback that helped bring this thesis to completion.

What is life without friends? And how wonderful it is when the colleagues you work with every day become friends. People with whom one can discuss not only research, but also day-to-day life. Immense thanks to *Alfy*, for all the coffees we shared and for the many conversations that went far beyond research. Speaking with you about life, different perspectives, and our experiences has broadened my horizon in ways I truly value. Your openness and honesty have meant a great deal to me. It has also been a joy to see you build such a loving family, who will always remain close to my heart. Muchas gracias por todo, *Xavi*, for your kindness and for always being there to listen. And for helping me navigate the jungle of legacy code. Beyond that, thank you for becoming a great friend and for the many enriching discussions and unforgettable moments we shared. A big thank you to *Srikanth* and *Siddarth* for welcoming me into your office and for being such great colleagues and friends. Also, *Srikanth*, thank you for sharing so many of my interests, from scientific computing to running marathons. Someday, I will catch up with your running pace! Thanks to *Arne* and *Kevin* for the amazing bike rides through Zuid-Holland. Also many thanks to all my other countless friends and colleagues who have supported me on my path.

Mehmet, thank you for being such a good friend from the very beginning of my time in Delft. Starting as flatmates when I arrived to the many evenings ending with a drink and good music, I am deeply grateful for your friendship.

Finally, the most important people in my life: My family.

Mama, Papa, Kaja. Euch verdanke ich unendlich viel. Ihr habt mich stets ermutigt, meiner Leidenschaft zu folgen, und mir den Rücken freigehalten, wann immer es nötig war. Eure Liebe und eure selbstverständliche Unterstützung haben all das erst möglich gemacht.

Auch meinen baldigen Schwiegereltern und der ganzen Familie danke ich von ganzem Herzen. Danke für eure Unterstützung, eure Herzlichkeit und dafür, dass ihr mich nun schon seit 15 Jahren „aushaltet“.

Und schließlich *Katharina.* Du bist diesen Weg mit mir gegangen und hast einen großen Anteil an dieser Arbeit. Deine Ruhe, dein Vertrauen, dein Humor und deine Liebe haben mir besonders in schwierigen Phasen Halt gegeben. Du glaubst an mich, gibst mir Zuversicht und erinnerst mich daran, was wirklich zählt. Ich freue mich auf unsere gemeinsame Zukunft!

Contents

Summary

Optimisation is at the heart of modern engineering. From reducing aircraft emissions to designing safer cars or tailoring drugs for specific diseases, the goal is to find the best solution among countless possibilities. Yet real-world systems are complex, and every design must meet strict constraints related to safety, performance, and physical laws. A design that performs well but violates just one constraint, such as structural failure during flight or non-compliance, is not desirable.

To assess a design's performance, engineers rely on complex computer simulations that capture physical processes like drag or structural deformation of an aircraft. These simulations often behave like black boxes, as they are expensive to run and the relationship between inputs and outputs is typically non-linear and opaque. This makes exhaustive search of the design space impossible and necessitates data-efficient optimisation strategies.

Bayesian Optimisation (BO) has emerged as a state-of-the-art method for optimising expensive black-box functions, offering a principled way to make the most of limited data. It builds a probabilistic model of the system to guide evaluations efficiently, balancing exploration of uncertain regions with exploitation of promising designs. Although BO has been widely adopted across scientific and engineering domains, it continues to face significant challenges in scenarios that involve both high-dimensional input spaces and complex feasibility constraints. These settings form the primary focus of this thesis.

The first contribution of this work is to show why techniques that work in unconstrained settings, such as random subspace embeddings or simple model priors, often fail under constraints. To address this, the thesis introduces supervised subspace methods and revisits dimensionality-scaled priors that improve both robustness and feasibility discovery in constrained problems.

Second, it proposes scalable strategies to model thousands of constraints, which arise, for example, in structural or aerospace design. Rather than modelling each constraint separately, the thesis uses dimensionality reduction to reduce input and output dimensionality, making constrained optimisation tractable at scale.

Finally, it develops methods for multi-source optimisation, where both accurate and approximate models are available. A modelling framework captures their discrepancies and a novel acquisition strategy balances information gain, cost, and constraint

satisfaction, accelerating convergence under tight budgets.

Together, these contributions extend the reach of BO to realistic, simulation-based engineering problems. The resulting tools are broadly applicable and help bridge the gap between theoretical advances in optimisation and the practical demands of high-stakes engineering design.

Samenvatting

Optimalisatie vormt het hart van de moderne engineering. Of het nu gaat om het verminderen van de uitstoot van vliegtuigen, het ontwerpen van veiligere auto's of het ontwikkelen van geneesmiddelen voor specifieke aandoeningen, het doel is steeds om de beste oplossing te vinden uit talloze mogelijkheden. Toch zijn systemen in de praktijk complex, en moet elk ontwerp voldoen aan strikte randvoorwaarden op het gebied van veiligheid, prestaties en natuurwetten. Een ontwerp dat goed presteert maar niet voldoet aan één enkele randvoorwaarde, bijvoorbeeld door structureel te falen tijdens de vlucht of door niet aan regelgeving te voldoen, is niet wenselijk.

Om de prestaties van een ontwerp te beoordelen, maken ingenieurs gebruik van complexe computersimulaties die natuurkundige processen modelleren, zoals luchtweerstand of structurele vervorming van een vliegtuig. Deze simulaties gedragen zich vaak als black-boxes: ze vereisen grote rekenkracht en de relatie tussen invoer en uitvoer is doorgaans niet-lineair en niet transparant. Daardoor is een volledig onderzoek van de ontwerpruimte onhaalbaar en zijn intelligente, data-efficiënte strategieën noodzakelijk.

Bayesiaanse optimalisatie (BO) is een geavanceerde methode gebleken voor het optimaliseren van dure black-box functies, en biedt een onderbouwde manier om optimaal gebruik te maken van beperkte gegevens. BO bouwt een probabilistisch model van het systeem om evaluaties efficiënt te sturen, met een evenwicht tussen exploratie van onbekende regio's en exploitatie van veelbelovende ontwerpen. Hoewel BO breed wordt toegepast in wetenschappelijke en technische domeinen, blijft het een aanzienlijke uitdaging om situaties met zowel hoge-dimensionale ruimtes als complexe randvoorwaarden te behandelen. Deze uitdagingen vormen de kern van dit proefschrift.

De eerste bijdrage van dit werk is het aantonen waarom technieken die goed presteren in situaties zonder randvoorwaarden, vaak falen onder randvoorwaarden. Om dit te ondervangen introduceert het proefschrift gesuperviseerde subruimtemethoden en herziet geschaalde priors, die zowel de robuustheid als het vinden van realistische oplossingen verbeteren in randvoorwaardeproblemen. Ten tweede worden schaalbare strategieën voorgesteld om duizenden randvoorwaarden te modelleren, die traditioneel voorkomen in structurele of luchtvaartkundige ontwerpen. In plaats van elke randvoorwaarde afzonderlijk te modelleren, past dit werk dimensiereductie toe op zowel de invoer- als uitvoerruimte, waardoor randvoorwaardeoptimalisatie op grote schaal haalbaar wordt.

Tot slot ontwikkelt het proefschrift methoden voor optimalisatie met meerdere modellen, zowel met meer als minder fysische fenomenen. Een modelleringskader lost de discrepanties tussen deze modellen op, en een nieuwe acquisitiestrategie weegt resultaat, rekenkracht en het voldoen aan randvoorwaarden af, wat de convergentie versnelt bij beperkte rekenkracht. Deze bijdragen breiden het toepassingsgebied van BO uit naar realistische, simuleerbare engineeringproblemen. De resulterende methoden zijn breed inzetbaar en helpen de kloof te dichten tussen theoretische vooruitgang in optimalisatie en de praktische eisen van typische ontwerptrajecten.

Acronyms

AERO-BO	Autoencoder-Enhanced Joint Dimensionality Reduction for Constrained Bayesian Optimisation. 71–73, 76, 77, 82–88, 90, 120
ALEBO	Adaptive Linear Embedding Bayesian Optimisation. 24, 36, 68, 75
ARD	Automatic Relevance Determination. 23
BAxUS	Bayesian Optimisation in Adaptively Expanding Subspaces. 26, 27, 37, 40–42, 76, 77
BNN	Bayesian Neural Networks. 16
BO	Bayesian Optimisation. 2–5, 7–9, 16–18, 21–25, 27–29, 33, 34, 41, 45, 47–55, 57, 59, 68, 69, 71–73, 75–77, 80, 85, 86, 88, 93–96, 103, 110, 114, 117–120
BOOSTRE	Bayesian Optimisation Over Supervised Trust Region Embeddings. 40, 42–44, 120
cBAxUS	Constrained Bayesian Optimisation in Adaptively Expanding Subspaces. 41, 43
CBO	Constrained Bayesian Optimisation. 3, 5, 29–32, 36, 40, 43, 45, 68, 74, 76, 93–96, 117–119
CDF	Cumulative Distribution Function. 18, 109
CEI	Constrained Expected Improvement. 32, 33, 41, 52, 74, 76, 85, 95, 112
CMA-ES	Covariance Matrix Adaptation Evolution Strategy. 65, 66, 82, 83
CMES	Constrained Max-value Entropy Search. 34, 41, 44, 95
CMES-IBO	Constrained MES via Information Lower Bound. 103, 104, 112
CMFBO	Constrained Multi-Fidelity Bayesian Optimisation. 103, 104
COBYLA	Constrained Optimisation by Linear Approximation. 12, 82, 83
CTS	Constrained Thompson Sampling. 33, 35, 36, 40, 41, 52, 74, 76
DoE	Design of Experiments. 50, 52, 55, 64–66

DSP	Dimensionality-Scaled Prior. 27, 28, 40, 41, 44, 118
EI	Expected Improvement. 18, 32, 52
ELBO	Evidence Lower Bound. 14
FITBO	Fast Information-Theoretic Bayesian Optimisation. 95
FuRBO	Feasibility-Driven Trust Region Bayesian Optimisation. 35, 41, 44, 103, 112
GIBBON	General-purpose Information-Based Bayesian Optimisation. 95
GP	Gaussian Process. 5, 9, 10, 12–18, 23, 25–27, 30–32, 35, 37, 39–41, 45, 49–62, 65–69, 72–80, 82–86, 95–99, 101–104, 109, 110, 118–120
HeSBO	Hashing-enhanced Subspace Bayesian Optimisation. 24, 26, 27, 36, 75
HOGP	High-Order Gaussian Processes. 59, 76
ICM	Intrinsic Co-regionalisation Model. 15, 31, 58, 76
KOH	Kennedy–O'Hagan. 16, 101, 103, 110, 111
kPCA	Kernel Principal Component Analysis. 55, 57–60, 65–67, 76
KS	Kreisselmeier–Steinhauser. 66
L-BFGS-B	Limited-memory Broyden-Fletcher-Goldfarb-Shanno. 12, 20
LHS	Latin Hypercube Sampling. 65
LMC	Linear Model of Coregionalisation. 15, 31, 58, 76
LogCEI	Log Constrained Expected Improvement. 41, 74, 76, 85, 103
LogEI	Log Expected Improvement. 18
MC	Monte Carlo. 41
MCMC	Markov Chain Monte Carlo. 16, 28
MDO	Multi-disciplinary Design Optimisation. 48, 62
MES	Max-value Entropy Search. 19, 20, 34, 95, 98
MISO	Multi-Information Source Optimisation. 16, 95, 97, 100, 101, 103, 109, 112, 113
MS-CMES	Multi-Source Constrained Max-value Entropy Search. 98, 101, 103–106, 112, 113, 120
MTGP	Multi-Task Gaussian Process. 15, 16, 25, 58, 59, 76, 101, 103, 110, 111
PCA	Principal Component Analysis. 23, 24, 37–39, 42–44, 54–61, 65–69, 76, 78, 82, 83, 87, 120
PDE	Partial Differential Equation. 1, 31, 57

PDF	Probability Distribution Function. 18
PES	Predictive Entropy Search. 19, 20, 33, 95
PESC	Predictive Entropy Search with Constraints. 33, 42, 74, 95, 103
PFN	Prior-Fitted Networks. 17, 120
PI	Probability of Improvement. 19
PLS	Partial Least Squares. 12, 37
RAASP	Random Axis-Aligned Subspace Perturbation. 26, 35, 41
RBF	Radial Basis Function. 11, 74, 111
ReLU	Rectified Linear Unit. 78
REMBO	Random Embedding Bayesian Optimisation. 24, 36, 68, 75
RQ	Research Question. 4
SAASBO	Sparse Axis-Aligned Subspace Bayesian Optimisation. 28, 75, 77
SCBO	Scalable Constrained Bayesian Optimisation. 35, 36, 41, 42, 44, 54, 58–61, 65–68, 72, 74, 76, 79, 82–87, 91, 103, 104, 112, 120
SKI	Structured Kernel Interpolation. 109
SVGP	Sparse Variational Gaussian Processes. 14
TR	Trust Region. 3, 25–27, 35–37, 40, 44, 45, 54, 62, 66, 68, 76, 77, 79–81, 84, 87, 90, 95, 101, 105, 106, 108, 109, 112, 115, 117
TS	Thompson Sampling. 19, 26, 27, 52, 54, 99
TuRBO	Trust Region Bayesian Optimisation. 25–27, 35, 54, 76
UCB	Upper Confidence Bound. 19
VBO	Vanilla Bayesian Optimisation. 27, 41, 44, 76, 82, 84, 85, 103, 104

1

Introduction

1.1. Motivation

In real-world engineering and scientific applications, optimisation problems are typically constrained. These constraints define the domain of physically meaningful, operationally viable, and safe solutions. Without them, optimisation may yield designs that are lightweight yet structurally unsound, fast yet dynamically unstable, or efficient yet fundamentally unsafe. In short, neglecting constraints therefore leads to solutions that are either impractical or unsafe.

In engineering design, constraints often stem from physical laws, safety regulations, certification requirements, and operational limitations. For instance, an aircraft must not only be aerodynamically efficient and lightweight, but also structurally sound, controllable under turbulence, capable of safe emergency manoeuvres, and compliant with stringent certification standards. Such constraints often extend beyond simple box bounds or linear relations. Instead, they are frequently governed by complex physical models described by Partial Differential Equations (PDEs). These include the Navier–Stokes equations for fluid dynamics (Jameson, 2003), elasticity and plasticity equations for structural mechanics (Amir, 2017), Maxwell's equations for electromagnetics (Yousept, 2012), reaction–diffusion systems in chemical and biological contexts (Christiansen, 2019), and the Schrödinger equation in quantum mechanics (Lazin et al., 2023).

Solving such PDEs is often computationally expensive, particularly when they are nonlinear, defined on complex geometries, or involve coupling between multiple physical domains. While analytical solutions may exist for simplified cases, practical engineering scenarios typically require numerical solvers, such as finite element or finite volume methods, to evaluate the system response. Moreover, the mapping from design variables such as geometry or material properties to quantities of interest

such as drag, lift, stress, or temperature is generally non-analytic and must be computed numerically. As a result, gradients with respect to design variables are often unavailable, expensive to compute, or unreliable due to solver instabilities, discontinuities, or numerical noise (Cranmer et al., 2020).

These challenges are pervasive across disciplines, ranging from optimising the design of an aircraft wing or a biomedical stent (Jameson, 2003, Li et al., 2017), to controlling the reaction-diffusion in chemical reactions (Christiansen, 2019), or the development of novel drugs (Guo and Schwaller, 2024). In all these cases, the true constraint is a complex simulation based on numerical solvers, rendering optimisation in such settings slow, expensive, and data-scarce. However, advances in computational modelling and optimisation now enable a more integrated, simulation-driven approach to system design, allowing engineers to revisit and rethink even long-established solutions. For instance, the adoption of emerging propulsion technologies such as hydrogen combustion or electric hybrid engines introduces new constraints related to safety, volume, cooling, and system integration. Even seemingly minor design changes, like relocating propulsion units, can shift the centre of gravity and cascade through interdependent subsystems, affecting everything from structural loads to cabin layout. The result is a tightly coupled, high-dimensional, and deeply constrained optimisation problem, where traditional incremental design strategies often fall short. In many real-world applications, particularly in aircraft design and structural topology optimisation, the number of constraints can be extremely high. Large-scale engineering problems may involve thousands of constraints, often defined by pointwise evaluations of field quantities such as stress, temperature, displacement, or flow velocity over critical regions, load cases, or operating conditions. Each constraint encodes a safety, performance, or regulatory requirement, and violating even a single one renders a design infeasible. This scale and complexity of constraint modelling substantially intensifies both the computational cost and the optimisation challenge.

When gradient information is available and reliable, gradient-based methods offer efficient local optimisation (Werter, 2017) for these types of problems. However, for many real-world problems, particularly those involving black-box solvers, legacy codes, discontinuous multi-physics interactions (Anand et al., 2025) or chaotic system responses where small design changes may yield large, unpredictable effects (Blonigan and Wang, 2018), gradient-based methods are either inapplicable or fundamentally limited by their reliance on local information, neglecting the global design space and hindering the discovery of fundamentally novel solutions.

In such settings, gradient-free, global optimisation methods are required. Among these, Bayesian Optimisation (BO) (Kushner, 1962, 1964, Frazier, 2018) has emerged as a powerful and data-efficient approach. Rather than relying on derivatives, BO builds probabilistic surrogate models of the objective and constraint functions. An

acquisition function, informed by these models, selects the next point to evaluate by balancing exploration to reduce uncertainty and exploitation by optimising promising regions. This makes BO well-suited for global search in expensive, black-box, and low-data regimes. However, this flexibility comes at a cost: the surrogate models must be accurate enough to guide the search effectively, despite being trained on very limited data. The curse of dimensionality becomes a central bottleneck. In high-dimensional spaces, the number of samples required to accurately model the entire design space grows exponentially, which is infeasible when evaluations are expensive. Constraints further exacerbate the difficulty. A feasible solution, one that satisfies all constraints, may be rare, making it hard to even find a first feasible design to begin with. Moreover, BO must now model not only the objective, but also each constraint, adding further modelling burden. This shifts the modelling focus in BO: efficient optimisation hinges not only on capturing objective trends but also on effectively learning where the constraints are active or violated, especially in high-dimensional design spaces.

This thesis focuses precisely on this intersection: the development of scalable Constrained Bayesian Optimisation (CBO) methods for high-dimensional engineering design problems. The overarching aim is to enable data-efficient global optimisation under the practical constraints of physical modelling, where the evaluation of both the objective and constraint functions are simulation or PDE-based.

1.2. Research Questions

Recent research in BO has made significant progress in addressing scalability to high-dimensional design spaces, particularly in unconstrained settings, e.g. in Wang et al. (2016), Letham et al. (2020), Eriksson et al. (2019), Ziomek and Bou-Ammar (2023). However, constrained high-dimensional BO remains relatively underexplored, despite its relevance to many real-world engineering applications, where feasibility is critical and constraint evaluations are expensive. Trust Region (TR) methods have recently emerged as state-of-the-art techniques for CBO (Eriksson et al., 2019), offering robust convergence under feasibility-aware local modelling assumptions. In parallel, subspace-based methods, particularly those leveraging random embeddings, have shown strong empirical performance in unconstrained BO when combined with TR strategies (Papenmeier et al., 2023). Nevertheless, their potential for constrained problems is still not well understood.

In a different direction, recent works (Hvarfner et al., 2024, Xu et al., 2025, Papenmeier et al., 2025) have shown that some of the failure modes of BO in high-dimensional, unconstrained scenarios can be mitigated by a simple, dimensionality-aware length-scale initialisation, improving the performance of even standard BO algorithms. While promising, this approach has not yet been systematically evaluated in constrained settings, particularly many constraints, where it may offer an efficient means to enhance existing algorithms in engineering design.

Importantly, modelling each constraint independently, as is commonly done in current CBO frameworks, can become computationally prohibitive in engineering problems involving thousands of black-box constraints. This motivates the development of more scalable modelling approaches that can handle many constraints without incurring prohibitive memory or runtime costs.

At the same time, many engineering workflows provide access to multiple models of the same physical system, each with different levels of accuracy, resolution, and computational cost. For example, an aircraft wing might be simulated using both a detailed 3D finite element model and a simpler beam theory-based approximation (Maathuis et al., 2024). These models may differ in accuracy and formulation but can provide complementary information that, if effectively combined, can significantly improve optimisation performance. Leveraging such multi-source information is especially valuable in data-scarce regimes where evaluations of the target model are expensive or limited. These multi-source formulations offer a promising opportunity to leverage cheaper or simpler models to guide exploration of the target design space, but they are rarely integrated systematically into constrained, high-dimensional BO frameworks.

Taken together, these observations motivate the following Research Questions (RQs):

RQ 1 *How can techniques from unconstrained Bayesian Optimisation be extended to efficiently solve high-dimensional, constrained engineering design problems under tight evaluation budgets?*

- (a) How can input space embeddings be leveraged to improve trust region Bayesian Optimisation in constrained, high-dimensional settings, building on their effectiveness in unconstrained problems with random embeddings?
- (b) How effective are dimensionality-scaled length-scale priors for Gaussian Process modelling in constrained, high-dimensional Bayesian Optimisation, compared to their performance in unconstrained settings?

RQ 2 *How can scalable methods be developed for Bayesian Optimisation in high-dimensional design spaces including large-scale constraints, while maintaining computational and memory efficiency?*

- (a) How can joint input-output dimensionality reduction using autoencoders support efficient Bayesian Optimisation in high-dimensional, constrained design problems?
- (b) How does the proposed methodology compare to classical penalty methods and constraint aggregation techniques in addressing feasibility and scalability in constrained Bayesian optimisation?

RQ 3 *How can information from multiple, potentially weakly correlated data sources be effectively integrated into Bayesian optimisation to solve high-dimensional, constrained design problems with a limited budget for expensive model evaluations?*

(a) How can statistical models capture the dependencies and discrepancies across information sources to support efficient optimisation?

(b) How can data source selection during acquisition be formalised to balance information gain, cost, and constraint satisfaction?

1.3. Thesis Outline

The thesis begins by laying the theoretical foundation of BO in Chapter 2, introducing its core principles and explaining how probabilistic surrogate models, particularly Gaussian Processes (GPs), are constructed to model black-box functions. This includes a discussion of widely used acquisition functions that guide the exploration–exploitation trade-off. Chapter 2 also introduces methods for modelling multiple outputs, outlines scalability challenges, and sets the stage for the more specific difficulties arising in high-dimensional settings. It then formalises the constrained optimisation problem, surveys state-of-the-art techniques for constraint handling, and highlights their limitations in high-dimensional spaces.

Chapter 3 then bridges the methodological gap between unconstrained and constrained BO, showing why techniques successful in unconstrained settings, such as random embeddings, do not trivially extend to constrained problems. It proposes modifications to enable their use under constraints, thereby addressing RQ 1. Moreover, the previously introduced state-of-the-art methods are compared on a set of benchmark problems to assess their performance.

To further improve the scalability of CBO for complex engineering design problems involving potentially thousands of constraints, Chapter 4 presents an approach for output-space dimensionality reduction. Modelling each constraint independently becomes computationally infeasible at this scale. To address this challenge, the chapter proposes projecting the constraint training data onto a low-dimensional subspace, thereby reducing the number of surrogate models required. This substantially lowers computational cost and renders large-scale constrained optimisation tractable, addressing RQ 2.

Building on this idea, Chapter 5 explores joint dimensionality reduction of the input space of design variables and the output space of constraints. The proposed method leverages a pair of interconnected autoencoders to learn a shared latent space in which the optimisation is conducted. This improves optimisation efficiency in problems where both the input and output spaces are prohibitively high-dimensional, thereby answering RQ $2_{(a)}$. In addition, the presented approaches are compared

against a classical penalty approach and constraint aggregation in Chapters 4 and 5, respectively, providing a comprehensive evaluation that supports the answer to RQ $2_{(b)}$.

The final methodological chapter, Chapter 6, explores a complementary direction by considering settings where multiple data sources are available for the same optimisation problem. Engineers may have access to auxiliary models in addition to a target model, such as simplified physics-based approximations, coarse numerical simulations, or generative models trained on historical data. This chapter investigates multi-source GPs models to capture dependencies and discrepancies across sources, and proposes a novel information-theoretic acquisition function that balances information gain, cost, and constraint satisfaction, addressing RQ 3.

The thesis concludes with a summary of key findings and contributions, followed by a discussion of potential avenues for future research in Chapter 7.

1.4. Scientific Contributions

This thesis makes the following scientific contributions to the field of CBO, with a particular focus on high-dimensional and simulation-based engineering design problems:

- In Chapter 3, random subspace embeddings are critically evaluated in constrained settings, revealing limitations due to poor feasibility preservation. To overcome this, a supervised embedding strategy based on weighted Principal Component Analysis (PCA) is introduced. Furthermore, the effect of scaling GP length-scale priors according to input dimensionality is investigated for constrained optimisation. The proposed methods are benchmarked against state-of-the-art CBO approaches.
- In Chapter 4, a scalable CBO framework is introduced for problems involving thousands of constraints. The method projects constraint outputs into a low-dimensional subspace, thereby avoiding the need to train one GP per constraint. Scalability is preserved through a TR-based acquisition strategy.
- In Chapter 5, a joint input–output dimensionality reduction approach is developed. A dual-autoencoder architecture is used to learn a shared latent space that compresses both the input and constraint spaces. GPs are trained in this latent space to enable efficient optimisation in settings with high-dimensional designs and large numbers of constraints.
- In Chapter 6, a multi-source GP model is extended to constrained optimisation problems with weakly correlated information sources. A novel information-theoretic acquisition function is proposed to balance cost, information gain, and feasibility, accelerating convergence under tight evaluation budgets.

Bibliography

O. Amir. Stress-constrained continuum topology optimization: a new approach based on elasto-plasticity. *Structural and Multidisciplinary Optimization*, 55(5):1797–1818, May 2017. ISSN 1615-147X, 1615-1488. doi: 10.1007/s00158-016-1618-8. URL http://link.springer.com/10.1007/s00158-016-1618-8.

S. Anand, N. Dighe, P. Gupta, R. Alderliesten, and S. G. Castro. Failure modes and energy absorption in Glass Reinforced aluminum (GLARE) hybrid laminates subjected to three-point bending. *Composites Part C: Open Access*, 18:100651, Oct. 2025. ISSN 26666820. doi: 10.1016/j.jcomc.2025.100651. URL https://linkinghub.elsevier.com/retrieve/pii/S2666682025000933.

P. J. Blonigan and Q. Wang. Multiple shooting shadowing for sensitivity analysis of chaotic dynamical systems. *Journal of Computational Physics*, 354:447–475, Feb. 2018. ISSN 00219991. doi: 10.1016/j.jcp.2017.10.032. URL https://linkinghub.elsevier.com/retrieve/pii/S002199911730791X.

L. H. Christiansen. *Optimal Control of PDE-constrained Systems*. PhD thesis, 2019.

K. Cranmer, J. Brehmer, and G. Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, Dec. 2020. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1912789117. URL https://pnas.org/doi/full/10.1073/pnas.1912789117.

D. Eriksson, M. Pearce, J. Gardner, R. D. Turner, and M. Poloczek. Scalable Global Optimization via Local Bayesian Optimization. 2019.

P. I. Frazier. A Tutorial on Bayesian Optimization, July 2018. URL http://arxiv.org/abs/1807.02811. arXiv:1807.02811 [cs, math, stat].

J. Guo and P. Schwaller. It Takes Two to Tango: Directly Optimizing for Constrained Synthesizability in Generative Molecular Design, Oct. 2024. URL http://arxiv.org/abs/2410.11527. arXiv:2410.11527 [q-bio].

C. Hvarfner, E. O. Hellsten, and L. Nardi. Vanilla Bayesian Optimization Performs Great in High Dimensions, Dec. 2024. URL http://arxiv.org/abs/2402.02229. arXiv:2402.02229 [cs].

A. Jameson. Aerodynamic shape optimization using the adjoint method. 2003. URL https://api.semanticscholar.org/CorpusID:2129299.

H. J. Kushner. A versatile stochastic model of a function of unknown and time varying form. *Journal of Mathematical Analysis and Applications*, 5(1):150–167, Aug. 1962. ISSN 0022247X. doi: 10.1016/0022-247X(62)90011-2. URL https://linkinghub.elsevier.com/retrieve/pii/0022247X62900112.

H. J. Kushner. A New Method of Locating the Maximum Point of an Arbitrary Multipeak Curve in the Presence of Noise. *Journal of Basic Engineering*, 86(1):97–106, Mar. 1964. ISSN 0021-9223. doi: 10.1115/1.3653121. URL https://asmedigitalcollection.asme.org/fluidsengineering/article/86/1/97/392213/A-New-Method-of-Locating-the-Maximum-Point-of-an.

M. F. Lazin, C. R. Shelton, S. N. Sandhofer, and B. M. Wong. High-dimensional multi-fidelity Bayesian optimization for quantum control. *Machine Learning: Science and Technology*, 4(4):045014, Dec. 2023. ISSN 2632-2153. doi: 10.1088/2632-2153/ad0100. URL https://iopscience.iop.org/article/10.1088/2632-2153/ad0100.

B. Letham, R. Calandra, A. Rai, and E. Bakshy. Re-Examining Linear Embeddings for High-Dimensional Bayesian Optimization, Oct. 2020. URL http://arxiv.org/abs/2001.11659. arXiv:2001.11659 [cs, stat].

H. Li, T. Liu, M. Wang, D. Zhao, A. Qiao, X. Wang, J. Gu, Z. Li, and B. Zhu. Design optimization of stent and its dilatation balloon using kriging surrogate model. *BioMedical Engineering OnLine*, 16(1):13, Dec. 2017. ISSN 1475-925X. doi: 10.1186/s12938-016-0307-6. URL http://biomedical-engineering-online.biomedcentral.com/articles/10.1186/s12938-016-0307-6.

H. Maathuis, S. G. P. Castro, and R. D. Breuker. Exploring Multi-Fidelity Aeroelastic Tailoring: Prospect and Model Assessment, Nov. 2024. URL http://arxiv.org/abs/2411.03247. arXiv:2411.03247 [cs].

L. Papenmeier, L. Nardi, and M. Poloczek. Increasing the Scope as You Learn: Adaptive Bayesian Optimization in Nested Subspaces, Apr. 2023. URL http://arxiv.org/abs/2304.11468. arXiv:2304.11468 [cs].

L. Papenmeier, M. Poloczek, and L. Nardi. Understanding High-Dimensional Bayesian Optimization, June 2025. URL http://arxiv.org/abs/2502.09198. arXiv:2502.09198 [cs].

Z. Wang, F. Hutter, M. Zoghi, D. Matheson, and N. de Freitas. Bayesian Optimization in a Billion Dimensions via Random Embeddings, Jan. 2016. URL http://arxiv.org/abs/1301.1942. arXiv:1301.1942 [cs, stat].

N. P. Werter. *Aeroelastic Modelling and Design of Aeroelastically Tailored and Morphing Wings*. PhD thesis, Delft University of Technology, 2017. URL http://resolver.tudelft.nl/uuid:74925f40-1efc-469f-88ee-e871c720047e.

Z. Xu, H. Wang, J. M. Phillips, and S. Zhe. Standard Gaussian Process is All You Need for High-Dimensional Bayesian Optimization, Mar. 2025. URL http://arxiv.org/abs/2402.02746. arXiv:2402.02746 [cs].

I. Yousept. Optimal control of Maxwell's equations with regularized state constraints. *Computational Optimization and Applications*, 52(2):559–581, June 2012. ISSN 0926-6003, 1573-2894. doi: 10.1007/s10589-011-9422-2. URL http://link.springer.com/10.1007/s10589-011-9422-2.

J. Ziomek and H. Bou-Ammar. Are Random Decompositions all we need in High Dimensional Bayesian Optimisation?, Jan. 2023. URL http://arxiv.org/abs/2301.12844. arXiv:2301.12844 [cs, stat].

2

Foundations of Bayesian Optimisation: From Unconstrained to Constrained Settings

To begin with, we consider the problem of optimising a parameter-dependent objective function f. Let $f : \mathcal{X} \to \mathbb{R}$ represent that function over a bounded D-dimensional domain $\mathcal{X} \subseteq \mathbb{R}^D$. The aim is to find $\mathbf{x}^* \in \mathcal{X}$ such that

$$\mathbf{x}^* = \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmin}} f(\mathbf{x}). \tag{2.1}$$

To solve these problems, a multitude of different approaches and methods exist. Especially when the aforementioned objective is expensive-to-evaluate, sample efficient methods need to be employed to keep the problem tractable. BO has been proven to be a state-of-the-art method for these types of problems (Kushner, 1962, 1964).

2.1. The Bayesian Approach to Optimisation

BO usually considers the objective function f as a black-box, meaning no other information about the problem is known, except for the objective function value f at a given point $\mathbf{x} \in \mathcal{X}$. Operating in rounds, in each iteration, a probabilistic surrogate model is constructed or updated, which is subsequently leveraged within an acquisition function to find the next point $\mathbf{x}_+$, the objective is then evaluated on. Both parts are briefly introduced in the following.

Probabilistic Surrogate Model Since f can only be evaluated pointwise, its global structure remains unknown. To address this, BO constructs a probabilistic surrogate model $\hat{f}(\mathbf{x}) \sim f(\mathbf{x})$ that approximates the true objective. This surrogate provides both:

1. a point estimate $\mu(\mathbf{x}) = \mathbb{E}[\hat{f}(\mathbf{x})]$ where $\mu : \mathcal{X} \to \mathbb{R}$, which represents the model's prediction of $f(\mathbf{x})$, and

2. an uncertainty estimate $\sigma^2(\mathbf{x}) = \mathrm{Var}[\hat{f}(\mathbf{x})]$ where $\sigma : \mathcal{X} \to \mathbb{R}$, quantifying how uncertain the model is about f at a certain $\mathbf{x}$.

Thus, for any $\mathbf{x} \in \mathcal{X}$, the surrogate model defines a posterior distribution over $f(\mathbf{x})$, typically as

$$f(\mathbf{x}) \sim \hat{f}(\mathbf{x}) = \mathcal{N}\left(\mu(\mathbf{x}), \sigma^2(\mathbf{x})\right). \tag{2.2}$$

$\mathcal{N}$ being a normal distribution with mean μ and variance σ^2. The idea of BO is now to leverage these probabilistic capabilities of the surrogate model $\hat{f}(\mathbf{x})$, to guide the optimisation through the design space (Frazier, 2018).

Acquisition Function To determine where to evaluate the objective function next, an acquisition function is used to balance exploration and exploitation of the design space where exploration refers to the exploration of the design space, especially regions with high uncertainty. Since BO is a global optimisation algorithm, it is aimed to explore the entire design space $\mathcal{X}$ to obtain the global optimum $\mathbf{x}^*$ of the function. However, the amount of data required to train an accurate surrogate model $\hat{f}(\mathbf{x})$ typically grows exponentially with the input dimensionality D, which can adversely affect model accuracy in higher dimensional settings (Bellman, 1957). Thus, in addition to global exploration, it is essential to exploit the gathered information, particularly in regions that exhibit promising objective values. By carefully balancing this trade-off by the use of the acquisition function $\alpha(\mathbf{x}) : \mathcal{X} \to \mathbb{R}$ which maps a point $\mathbf{x} \in \mathcal{X}$ to a utility measure in $\mathbb{R}$, the optimisation process ensures efficient use of limited evaluations to both discover new potential optima and improve confidence in previously identified promising regions. Since BO acts sequentially, let us consider the optimisation at iteration t where we define $\mathcal{D}_t = \{(\mathbf{x}_i, f(\mathbf{x}_i))\}_{i=1}^{N_t}$ as the set of N_t observed data points. The acquisition function $\alpha(\mathbf{x})$ serves as a utility metric to determine where to evaluate f next. It is constructed to balance two competing goals:

1. Exploration: Evaluate points $\mathbf{x}$ with high uncertainty $\sigma(\mathbf{x})$.

2. Exploitation: Evaluate points $\mathbf{x}$ with optimal predicted function values $\mu(\mathbf{x})$.

Mathematically, the next evaluation point $\mathbf{x}_+$ is chosen by maximising this acquisition function:

$$\mathbf{x}_+ = \arg\max_{\mathbf{x} \in \mathcal{X}} \alpha(\mathbf{x}; \mathcal{D}_t), \tag{2.3}$$

given the observed data $\mathcal{D}_t$ so far. Different acquisition functions define the trade-off between exploration and exploitation in various ways. More details will be given in Chapter 2.3.

The Algorithm To sum up this brief introduction, BO proceeds iteratively, summarised in Algorithm 1.

Algorithm 1 Bayesian Optimisation

while $t < t_{max}$ **do**
  Fit surrogate $\hat{f}(\mathbf{x}; \mathcal{D}_t)$
  Solve $\alpha(\mathbf{x}; \hat{f}, \mathcal{D}_t)$ for $\mathbf{x}_+$ ▷ Optimise Equation (2.3)
  Evaluate $f(\mathbf{x}_+)$ ▷ Evaluate the true function
  $\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{\mathbf{x}_+, f(\mathbf{x}_+)\}$ ▷ Add information to the set of data points
  $t \leftarrow t + 1$
end while
Choose $\mathbf{x}^* \in \arg\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}; \mathcal{D}_t)$

2.2. Gaussian Processes: A Probabilistic Surrogate Model

The choice of the probabilistic surrogate model used in BO has a major impact on the performance of the method. GPs as non-parametric, probabilistic models are a popular choice since they are known to be highly flexible and capable of modelling a wide range of functions. Additionally, their predictive mean and variance as well as the marginal likelihood can be computed analytically, naturally quantifying uncertainty (Rasmussen and Williams, 2006).
To keep the formulation general, we consider the case where only noisy observations of the true function are available. Each observation is modelled as

$$y_i = f(\mathbf{x}_i) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2), \tag{2.4}$$

where $f(\mathbf{x}_i)$ is the true (latent) function value, and ϵ represents independent Gaussian noise. Let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top$ denote the input design points, and $\mathbf{y} = [y_1, \ldots, y_N]^\top$ the corresponding noisy outputs. We place a Gaussian Process prior on the latent function:

$$f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')), \tag{2.5}$$

defined by a mean $\mu(\mathbf{x})$ and covariance $k(\mathbf{x}, \mathbf{x}')$ function which encodes prior beliefs about the smoothness and structure of the underlying function. The goal is to infer the unobservable latent function values $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]^\top$. Further details are provided in the following section.

2.2.1. A Bayesian Perspective

Bayes' theorem is a cornerstone of probabilistic inference, allowing to update the belief about an unknown quantity based on observed data. In its general form, Bayes' theorem states:

$$p(\text{latent} \mid \text{observed}) \propto p(\text{observed} \mid \text{latent})p(\text{latent}). \tag{2.6}$$

That is, the posterior probability is proportional to the likelihood times the prior. In the context of GPs, the unknown quantity of interest is the latent function $f : \mathcal{X} \to \mathbb{R}$, and the observed data consist of noisy evaluations of this function at a finite set of inputs. When a set of observations $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ is available, we wish to infer the posterior distribution over the function values $\mathbf{f}_+ = f(\mathbf{x}_+)$ at a new input $\mathbf{x}_+$. This posterior, denoted $p(\mathbf{f}_+ \mid \mathbf{X}_+, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta})$, is obtained by conditioning on the observed data and employing Bayes' rule:

$$p(\mathbf{f} \mid \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}) = \frac{p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \boldsymbol{\theta})p(\mathbf{f} \mid \mathbf{X}, \boldsymbol{\theta})}{p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta})}. \tag{2.7}$$

The interpretation of each component is as follows:

- $p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \boldsymbol{\theta})$ is the ***likelihood*** of the observations given the latent function values,
- $p(\mathbf{f} \mid \mathbf{X}, \boldsymbol{\theta})$ is the ***prior*** distribution over the latent function values at the training inputs,
- $p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta})$ is the ***marginal likelihood***, also known as the model evidence,
- $p(\mathbf{f} \mid \mathbf{y}, \mathbf{X}, \boldsymbol{\theta})$ is the ***posterior*** distribution over the latent function values.

All probabilities are conditioned on a set of hyperparameters $\boldsymbol{\theta}$ that define the probabilistic model. The four central terms are outlined below:

Prior We begin by specifying a GP prior over the latent function:

$$f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \tag{2.8}$$

This implies that any finite collection of function values $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)]^\top \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K})$ follows a multivariate Gaussian distribution:

$$p(\mathbf{f} \mid \mathbf{X}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}), \qquad \text{where} \qquad \mu_i = \mu(\mathbf{x}_i),\ K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j). \tag{2.9}$$

The mean function $\mu : \mathcal{X} \to \mathbb{R}$ is often set to zero, $\boldsymbol{\mu} \equiv 0$ with $\boldsymbol{\mu} \in \mathbb{R}^N$. The kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ with $\mathbf{K} \in \mathbb{R}^{N \times N}$ encodes assumptions about the structure of f, such as smoothness or periodicity (Frazier, 2018). The kernel function must be chosen such that $\mathbf{K}$ is symmetric positive definite, ensuring it is invertible. This property is guaranteed if and only if the kernel function itself is positive definite, as

established by Schoenberg (1938). Its form is determined by a set of hyperparameters $\boldsymbol{\theta}$, which are discussed in the training section.

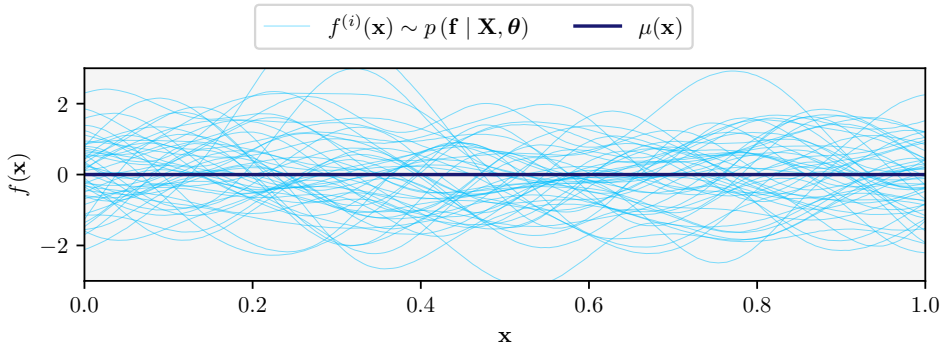


Figure 2.1: Drawing 50 functions $\{f^{(i)}\}_{i=1}^{50}$ from the prior with its mean $\mu(\mathbf{x}) \equiv 0$

For example, the commonly used Radial Basis Function (RBF) kernel takes the form:

$$k_{\text{RBF}}(\mathbf{x}, \mathbf{x}') = \exp\left(-\tfrac{1}{2}r^2\right), \tag{2.10}$$

where the distance metric r is defined as

$$r = \sqrt{(\mathbf{x} - \mathbf{x}')^\top \boldsymbol{\Lambda}^{-1} (\mathbf{x} - \mathbf{x}')}, \tag{2.11}$$

and with $\boldsymbol{\Lambda} = \text{diag}(l_1^2, ..., l_D^2)$ as a diagonal matrix containing the squared length scales which act as tunable hyperparameters. Another example is the Matérn kernel with smoothness parameter $\nu = \frac{5}{2}$, which takes the form (Genton, 2002):

$$k_{\text{Mat-5/2}}(\mathbf{x}, \mathbf{x}') = \left(1 + \sqrt{5}r + \frac{5}{3}r^2\right) \exp\left(-\sqrt{5}r\right). \tag{2.12}$$

In Figure 2.1, we sample from the prior $p(\mathbf{f} \mid \mathbf{X}, \boldsymbol{\theta})$ and plot all samples together with their specified mean $\boldsymbol{\mu}$ which is set to zero. Furthermore, Figure 2.2 depicts the function values of $k(\mathbf{x}_0, \mathbf{x})$ over varying length scales l.

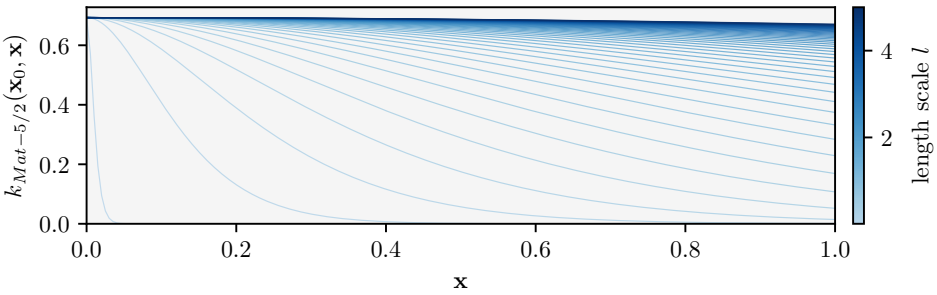


Figure 2.2: Matérn-5/2 kernel with different length scales.

Likelihood As defined in Equation 2.4, the observations $\mathbf{y} \in \mathbb{R}^N$ are assumed to be noisy evaluations of the latent function values $\mathbf{f}$. The conditional distribution over the observed data is:

$$p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{y} \mid \mathbf{f}, \sigma^2 \mathbf{I}). \tag{2.13}$$

This reflects the assumption that each observation is normally distributed around its corresponding latent function value, with independent Gaussian noise. In other words, even though we cannot observe the true function values $\mathbf{f}$ directly, the likelihood defines how likely the noisy observations $\mathbf{y}$ are, given any assumed values of $\mathbf{f}$. Although the true value of $\mathbf{f}$ is unknown, specifying a noise model encodes the belief that $\mathbf{f}$ varies smoothly, as assumed in the prior, and that the true function value can be inferred from noisy measurements.

Marginal Likelihood The marginal likelihood is obtained by integrating out the latent function values:

$$p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta}) = \int p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \boldsymbol{\theta}) p(\mathbf{f} \mid \mathbf{X}, \boldsymbol{\theta}) d\mathbf{f}. \tag{2.14}$$

Because the GP prior and the likelihood are both Gaussian, this integral is analytically tractable, thus yielding:

$$\begin{aligned} p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta}) &= \mathcal{N}(\mathbf{y} \mid \boldsymbol{\mu}, \mathbf{K} + \sigma^2 \mathbf{I}), \\ &= -\tfrac{1}{2} \mathbf{y}^\top (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{y} + \tfrac{1}{2} \log |\mathbf{K} + \sigma^2 \mathbf{I}| + \tfrac{n}{2} \log(2\pi). \end{aligned} \tag{2.15}$$

In other words, the marginal likelihood measures how well the model explains the observed data $\mathbf{y}$, integrating over all possible latent functions consistent with the prior. This expression is central for learning hyperparameters from data, as discussed next.

Training The kernel function $k(\mathbf{x}, \mathbf{x}')$ is governed by the hyperparameters $\boldsymbol{\theta} = \{l_1, ..., l_D, \sigma\} \in \mathbb{R}^{D+1}$, e.g. length scales and signal variance, which must be selected appropriately. These are not inferred directly in the GP regression equations, but instead learned by maximising the log marginal likelihood derived from Equation (2.14). Hence, we define the optimal hyperparameters $\boldsymbol{\theta}$ as

$$\boldsymbol{\theta} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta}), \tag{2.16}$$

where the hyperparameter dependent marginal likelihood $p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta})$ is defined as

$$\log p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta}) = -\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^\top (\mathbf{K}+\sigma^2\mathbf{I})^{-1}(\mathbf{y}-\boldsymbol{\mu}) - \frac{1}{2}\log|\mathbf{K}+\sigma^2\mathbf{I}| - \frac{n}{2}\log 2\pi. \tag{2.17}$$

Computing the log of the marginal likelihood increases numerical stability. By computing the partial derivative of Equation (2.17) with respect to each θ_j:

$$\frac{\partial}{\partial \theta_j}\log p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta}) = \frac{1}{2}\mathbf{y}^\top \mathbf{K}^{-1}\frac{\partial \mathbf{K}}{\partial \theta_j}\mathbf{K}^{-1}\mathbf{y} - \frac{1}{2}\operatorname{tr}\left(\mathbf{K}^{-1}\frac{\partial \mathbf{K}}{\partial \theta_j}\right), \tag{2.18}$$

gradient-based optimisation as e.g. Adam (Kingma and Ba, 2017) can be employed for model training.

In practice, toolboxes differ in how the hyperparameters are optimised. `BoTorch` (Balandat et al., 2020) employs a gradient-based optimiser, specifically the L-BFGS-B algorithm with bound constraints, which is well-suited to high-dimensional problems due to its efficient use of gradient information and gradient computation via back propagation. In contrast, other toolkits such as `SMT` (Saves et al., 2024) use gradient-free methods like Constrained Optimisation by Linear Approximation (COBYLA) (Powell, 1994, Zhang, 2023), which can struggle in high-dimensional settings (Saves et al., 2024). To address this, `SMT` applies dimensionality reduction techniques such as Partial Least Squares (PLS) to reduce the number of hyperparameters and improve optimisation tractability (Amine Bouhlel et al., 2018).

Inference After determining the hyperparameters $\boldsymbol{\theta}$, to make predictions at new input locations $\mathbf{X}_+ \in \mathbb{R}^{N_+ \times D}$, we consider the joint prior over both training and test function values:

$$\begin{bmatrix}\mathbf{f} \\ \mathbf{f}_+\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\boldsymbol{\mu} \\ \boldsymbol{\mu}_+\end{bmatrix}, \begin{bmatrix}\mathbf{K} & \mathbf{K}_+^\top \\ \mathbf{K}_+ & \mathbf{K}_{++}\end{bmatrix}\right), \tag{2.19}$$

where $\mathbf{K}_+ = k(\mathbf{X}_+, \mathbf{X}) \in \mathbb{R}^{N_+ \times N}$, $\mathbf{K}_+^\top = k(\mathbf{X}, \mathbf{X}_+) \in \mathbb{R}^{N \times N_+}$ and $\mathbf{K}_{++} = k(\mathbf{X}_+, \mathbf{X}_+) \in \mathbb{R}^{N_+ \times N_+}$. By conditioning on the observed outputs $\mathbf{y}$, the posterior predictive distribution is:

$$p(\mathbf{f}_+ \mid \mathbf{X}_+, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+), \tag{2.20}$$

with:

$$\boldsymbol{\mu}_+ = \boldsymbol{\mu} + \mathbf{K}_+(\mathbf{K} + \sigma^2\mathbf{I})^{-1}(\mathbf{y} - \boldsymbol{\mu}), \tag{2.21}$$

$$\boldsymbol{\Sigma}_+ = \mathbf{K} - \mathbf{K}_+(\mathbf{K} + \sigma^2\mathbf{I})^{-1}\mathbf{K}_+^\top, \tag{2.22}$$

which result form the standard conditioning of a Gaussian distribution (von Mises, 1964, Rasmussen and Williams, 2006). In the noiseless case, where $\sigma^2 = 0$, the predictive variance simplifies to

$$\boldsymbol{\mu}_+ = \boldsymbol{\mu} + \mathbf{K}_+\mathbf{K}^{-1}(\mathbf{y} - \boldsymbol{\mu}), \tag{2.23}$$

$$\boldsymbol{\Sigma}_+ = \mathbf{K} - \mathbf{K}_+\mathbf{K}^{-1}\mathbf{K}_+^\top. \tag{2.24}$$

In Figure 2.3, 50 samples are drawn from a posterior distribution $p(\mathbf{f}_* \mid \mathbf{X}_+, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta})$ conditioned on two data points, see Equation (2.20). Additionally, the predicted mean is plotted. Additional derivation details can be found in Rasmussen and Williams (2006) and Frazier (2018).

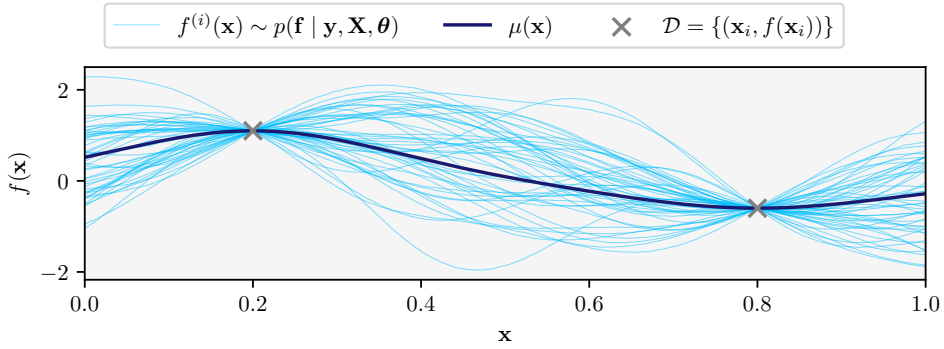


Figure 2.3: Drawing 50 functions $\{f^{(i)}\}_{i=1}^{50}$ from the posterior, Equation (2.20), with its mean $\mu(\mathbf{x})$, conditioned on two data points.

2.2.2. Scalability

While being a powerful tool for regression and classification tasks, a critical aspect of GPs is their scalability in terms of data points. Defined by a mean function and a symmetric positive definite covariance matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$. Inference and training in GPs require the inversion of the covariance matrix $\mathbf{K}$, as seen in Equation (2.21). This operation entails a computational complexity of $\mathcal{O}(N^3)$ while memory costs are $\mathcal{O}(N^2)$, rendering standard GPs impractical for large N. To mitigate this issue, several approximation techniques have been proposed:

- Sparse GPs, which reduce computational costs by selecting a small set of inducing points at selected input locations, enabling a low-rank approximation of the full covariance matrix (Silverman, 1985, Smola and Bartlett, 2000, Quiñonero-Candela and Rasmussen, 2005),

- Iterative linear solves to iteratively approximate $(\mathbf{K} + \sigma\mathbf{I})^{-1}$ as in Gardner et al. (2021),
- Structured kernel approximation methods, such as those proposed in Wilson and Nickisch (2015) to improve scalability.

These approaches provide feasible alternatives for deploying GPs in large-scale settings while preserving their expressive power.

Sparse Variational Gaussian Processes A well established method of sparse GPs was proposed by Hensman et al. (2014) named Sparse Variational Gaussian Processes (SVGP) which introduces a set of $m \ll N$ inducing variables to construct a computationally efficient approximation of the full GP, implemented in Gardner et al. (2021).

Let $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_m]^\top \in \mathbb{R}^{m \times D}$ denote a set of inducing inputs, and let $\mathbf{u} = f(\mathbf{Z}) \in \mathbb{R}^m$ be the corresponding latent function values. The joint prior over the training function values $\mathbf{f} = f(\mathbf{X})$ and inducing values $\mathbf{u}$ is Gaussian:

$$p(\mathbf{f}, \mathbf{u}) = \mathcal{N}\left(\begin{bmatrix}\mathbf{f} \\ \mathbf{u}\end{bmatrix} \middle| \mathbf{0}, \begin{bmatrix}\mathbf{K_{XX}} & \mathbf{K_{XZ}} \\ \mathbf{K_{ZX}} & \mathbf{K_{ZZ}}\end{bmatrix}\right), \tag{2.25}$$

with $\mathbf{K_{XX}} \in \mathbb{R}^{N \times N}$, $\mathbf{K_{XZ}} \in \mathbb{R}^{N \times m}$, $\mathbf{K_{ZX}} \in \mathbb{R}^{m \times N}$ and $\mathbf{K_{ZZ}} \in \mathbb{R}^{m \times m}$. Moreover, all kernel matrices are defined via the same kernel function $k(\cdot, \cdot)$ with hyperparameters $\boldsymbol{\theta}$. A variational approximation is introduced by positing a tractable distribution $q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f} \mid \mathbf{u})q(\mathbf{u})$, where $q(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mathbf{m}, \mathbf{S})$ is a free-form Gaussian. The objective is to maximise the Evidence Lower Bound (ELBO) $\mathcal{L}_{\text{ELBO}}$:

$$\mathcal{L}_{\text{ELBO}} = \sum_{i=1}^{N} \mathbb{E}_{q(f_i)}\left[\log p(y_i \mid f_i)\right] - D_{\text{KL}}\left[q(\mathbf{u}) \| p(\mathbf{u})\right], \tag{2.26}$$

where the first term encourages data fit and the second penalises deviation from the prior.

The computational efficiency arises from the fact that inference is performed through the lower-dimensional inducing variables $\mathbf{u}$, rather than directly over $\mathbf{f}$. Specifically, the conditional $p(\mathbf{f} \mid \mathbf{u})$ is Gaussian with:

$$p(\mathbf{f} \mid \mathbf{u}) = \mathcal{N}\left(\mathbf{K_{XZ}}\mathbf{K_{ZZ}^{-1}}\mathbf{u},\ \mathbf{K_{XX}} - \mathbf{K_{XZ}}\mathbf{K_{ZZ}^{-1}}\mathbf{K_{ZX}}\right). \tag{2.27}$$

This approach reduces the computational cost by requiring inversion of only the $m \times m$ kernel matrix $\mathbf{K_{ZZ}}$ defined by the number of inducing points, instead of the full $N \times N$ matrix. This reduces the time complexity from $\mathcal{O}(N^3)$ to $\mathcal{O}(Nm^2)$, and the memory complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(Nm)$. Furthermore, SVGP models can be trained using stochastic gradient descent and mini-batches, making them scalable to large datasets.

2.2.3. Gaussian Processes for Correlated Outputs

In many real-world applications, multiple outputs or sources of information are available and potentially correlated. These include, for example, tasks in multi-objective optimisation, quantities computed using different solvers such as coarse and fine mesh simulations in computational fluid dynamics, heterogeneous data modalities such as simulations and physical experiments, or auxiliary outputs like black-box constraints in constrained optimisation. In the following a distinction is made between two key scenarios:

- **Parallel outputs**, where one aims to model multiple outputs simultaneously, such as two objectives like weight and drag.
- **Hierarchical or multi-source settings**, where one output can be obtained from multiple sources, for example, drag estimated from both a simulation and an experiment.

Independent GPs for Multiple Outputs For L parallel outputs, the simplest approach is to model each $f_\ell(\mathbf{x})$ independently using a separate GP:

$$f_t(\mathbf{x}) \sim \mathcal{GP}(\mu_0, k_t(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}_t)), \quad \ell = 1, \ldots, L. \tag{2.28}$$

While simple and computationally efficient, this approach ignores any potential correlation between outputs and may therefore underperform.

Linear Model of Coregionalisation and Intrinsic Coregionalisation Model The Linear Model of Coregionalisation (LMC) (Goovaerts, 1997, Álvarez and Lawrence, 2011) provides a more expressive framework by modelling each output as a linear combination of shared latent functions:

$$f_\ell(\mathbf{x}) = \sum_{q=1}^{Q} a_{\ell q} u_q(\mathbf{x}), \quad u_q(\cdot) \sim \mathcal{GP}(0, k_q(\cdot, \cdot)). \tag{2.29}$$

This allows for flexible cross-task dependencies, with each latent process u_q capturing shared variation across outputs via task-specific weights $a_{\ell q}$. The resulting cross-covariance is given by:

$$\mathrm{Cov}[f_\ell(\mathbf{x}), f_{\ell'}(\mathbf{x}')] = \sum_{q=1}^{Q} a_{\ell q} a_{\ell' q} k_q(\mathbf{x}, \mathbf{x}'). \tag{2.30}$$

A more restrictive variant is the Intrinsic Co-regionalisation Model (ICM) (Bonilla et al., 2007), in which all latent processes share the same kernel $k(\cdot, \cdot)$. In this case, the covariance between tasks is captured by a fixed co-regionalisation matrix $\mathbf{B} \in \mathbb{R}^{L \times L}$, yielding a separable kernel:

$$k_{\mathrm{ICM}}((\ell, \mathbf{x}), (\ell', \mathbf{x}')) = B_{\ell \ell'} \cdot k(\mathbf{x}, \mathbf{x}'), \tag{2.31}$$

with resulting covariance structure $\mathbf{K} = \mathbf{B} \otimes k(\mathbf{X}, \mathbf{X}')$, where $\otimes$ denotes the Kronecker product. While less flexible than LMC, ICM is computationally efficient due to its Kronecker structure, and widely used in practical Multi-Task Gaussian Process (MTGP) models (Alvarez et al., 2012).

Hierarchical Models for Multi-Source Outputs In multi-fidelity or multi-source settings, different outputs represent observations of the same underlying function at varying levels of approximation and cost. For instance, in simulation-based design, one might combine data from fast but inaccurate coarse simulations and accurate but expensive experiments. In such cases, hierarchical models are employed to explicitly model the relationships between outputs. A popular example is the autoregressive model by Kennedy–O'Hagan (KOH) (Kennedy and O'Hagan, 2000, Forrester et al., 2007), which assumes:

$$f_L(\mathbf{x}) = \rho f_\ell(\mathbf{x}) + \Delta(\mathbf{x}), \tag{2.32}$$

where f_ℓ is the low-fidelity function, f_L the high-fidelity target, $\rho \in \mathbb{R}$ a scaling factor, and $\Delta(\mathbf{x}) \sim \mathcal{GP}(0, k_\Delta)$ a discrepancy function. More recent approaches such as the Multi-Information Source Optimisation (MISO) framework (Poloczek et al., 2016) generalises this idea by allowing shared latent components and discrepancies between sources, defined as:

$$f_\ell(\mathbf{x}) = f_L(\mathbf{x}) + \Delta(\mathbf{x}). \tag{2.33}$$

An alternative is to use a MTGP for multiple sources, which however does not introduce a hierarchy between the sources.

2.2.4. Alternative Approaches: A Short Excursion

While this thesis focuses on GPs as the primary surrogate model for BO, it is important to briefly consider alternative modelling paradigms that have gained traction over the past decade. These approaches offer promising scalability and flexibility, especially in high-data settings, but also come with notable trade-offs that justify the continued use of GPs in this work.

Bayesian Neural Networks Bayesian Neural Networks (BNN) extend standard neural networks by placing probabilistic priors over weights, yielding a posterior distribution over functions. They offer scalability to larger datasets and non-stationary modelling capacity (Snoek et al., 2015). However, in practice, BNNs require complex approximate inference schemes, as for example variational Bayes or Markov Chain Monte Carlo (MCMC), and often suffer from poor uncertainty calibration (Binois and Wycoff, 2022). Despite their expressive power, BNNs typically require substantial training data and can be sensitive to architectural and hyperparameter choices. Moreover, posterior inference is rarely analytic, complicating acquisition function computation in BO.

Deep Ensembles Ensemble-based approaches (Lakshminarayanan et al., 2017) approximate uncertainty by aggregating predictions from multiple independently trained neural networks. They are easy to implement and scale well with data, but they do not provide a coherent probabilistic model of uncertainty. This lack of a formal posterior makes integration with acquisition functions, such as entropy-based criteria, less principled and often heuristic. For instance, ensemble variance is used as a proxy for epistemic uncertainty, but this quantity is not guaranteed to be calibrated or consistent. To scale BO to large data-sets, the authors in Om et al. (2025) use ensembles in combination with normalising flows (Kobyzev et al., 2021).

Transformer-based Models Recent works have explored the use of transformer architectures for surrogate modelling (Müller et al., 2024). Prior-Fitted Networkss (PFNs) aim to learn an implicit prior over functions by training transformer-based models on a large set of synthetic tasks. At test time, they perform inference without gradient updates, effectively amortising the learning process. PFNs can be interpreted as meta-learned function predictors that bypass the need for task-specific training. However, their integration with BO pipelines remains a topic of ongoing research rather than established methodology (Müller et al., 2023).

Why Gaussian Processes? Despite the appeal of these alternatives, GPs remain the standard surrogate model in BO for several key reasons:

- *Closed-form posterior inference:* GPs offer tractable and exact posterior updates, allowing efficient uncertainty quantification.
- *Principled uncertainty estimates:* Unlike heuristics such as ensemble variance, GPs provide well-calibrated predictive distributions under minimal assumptions.
- *Analytic acquisition functions:* The combination of Gaussian priors and likelihoods allows for closed-form expressions in many acquisition strategies.
- *Strong performance in low-data regimes:* GPs are particularly effective when evaluation budgets are limited, a common setting in engineering design problems.

Of course, this comes at a cost: GPs scale poorly with the number of data points, as discussed in Section 2.2.2. However, given the simulation-driven nature of the problems considered in this thesis, characterised by expensive evaluations and limited data, GPs offer the best balance between tractability, uncertainty-awareness, and integration with the broader BO framework. Accordingly, all subsequent chapters build upon GP-based surrogates, extending them to handle constraints, high-dimensional inputs, and multiple fidelities.

2.3. Acquisition Strategies

After introducing GPs as a probabilistic surrogate modelling technique, this section dives into the aforementioned acquisition strategies which play a central role in guiding the search for the optimal solution. To do so, a function $\alpha(\mathbf{x}) : \mathcal{X} \to \mathbb{R}$ maps each candidate point $\mathbf{x} \in \mathcal{X}$ in the search space to a real-valued utility score, which reflects its potential value. Key properties of acquisition functions include:

- Exploration-exploitation trade-off: $\alpha(\mathbf{x})$ tries to balance exploration, meaning exploring regions with high uncertainties, and exploitation, refining the search in regions in $\mathcal{X}$ where promising values are expected due to already available data.
- Computational efficiency: Evaluating $\alpha(\mathbf{x})$ should be significantly cheaper than evaluating the objective function.
- Differentiability: In many cases, smooth and differentiable acquisition functions allow for efficient gradient-based optimisation to find the new query point $\mathbf{x}_+ \in \mathcal{X}$.

The next point is then chosen such that $\mathbf{x}_+ \in \arg\max_{\mathbf{x}\in\mathcal{X}} \alpha_{(\cdot)}(\mathbf{x})$. In the following, the most prominent acquisition function are briefly introduced:

Expected Improvement Expected Improvement (EI) (Jones et al., 1998) is a foundational acquisition function in BO that balances exploration and exploitation by quantifying the expected gain over the current best observation $f^\dagger \in \mathcal{D}_t$. It is defined as:

$$\alpha_{\text{EI}}(\mathbf{x}) = \mathbb{E}_{f(\mathbf{x})}\left[\max(f(\mathbf{x}) - f^\dagger, 0)\right]. \tag{2.34}$$

Under a GP model, $\alpha_{\text{EI}}(\mathbf{x})$ has a closed-form expression:

$$\alpha_{\text{EI}}(\mathbf{x}) = \int_{-\infty}^{\infty} \left(\max(f(\mathbf{x}) - f^\dagger, 0)\right) \phi(Z) dZ \tag{2.35}$$

$$= (\mu(\mathbf{x}) - f^\dagger)\Phi(z) + \sigma(\mathbf{x})\phi(z), \tag{2.36}$$

where $z = \frac{\mu(\mathbf{x}) - f^\dagger}{\sigma(\mathbf{x})}$, and $\Phi(\cdot)$, $\phi(\cdot)$ denote the standard normal Cumulative Distribution Function (CDF) and Probability Distribution Function (PDF), respectively. A similar form of EI is used in Amine Bouhlel et al. (2018) and Sasena et al. (2002). This formulation suffers from numerical instabilities when $z \approx 0$, i.e., when the model predicts low variance or marginal improvement, which can lead to vanishing gradients and poor optimisation performance. To address this, Ament et al. (2023) proposed Log Expected Improvement (LogEI), a numerically stable variant of EI based on computing $\log \alpha(\mathbf{x})$, defined as:

$$\log \alpha_{\text{EI}}(\mathbf{x}) = \log\_\text{h}(z) + \log \sigma(\mathbf{x}), \tag{2.37}$$

where the function log_h(z) is a numerically stable approximation of Equation (2.36) and is computed piecewise depending on the value of z. Specifically:

$$\log\_\mathrm{h}(z) = \begin{cases} \log\left(\phi(z) + z\Phi(z)\right) & \text{if } z > -1, \\ -\frac{z^2}{2} - c_1 + \log c_0 & \text{if } -1/\sqrt{\varepsilon} < z \leq -1, \\ -\frac{z^2}{2} - c_1 - 2\log(|z|) & \text{if } z \leq -1/\sqrt{\varepsilon}, \end{cases} \tag{2.38}$$

where $c_0 = \log1\mathrm{mexp}\left(\log\left(\mathrm{erfcx}(-z\sqrt{2})|z|\right) + c_2\right)$, $c_1 = \log(2\pi)/2$, $c_2 = \log(\pi/2)/2$, and ε is the numerical precision. Functions like log1mexp and erfcx are numerically stable implementations of $\log(1 - \exp(z))$ and $\exp(z^2)\mathrm{erfc}(z)$, respectively. This logarithmic form preserves the smooth behaviour of EI while avoiding flat regions in the acquisition landscape, making gradient-based optimisation significantly more robust. Empirically, LogEI outperforms standard EI in both convergence speed and numerical stability, particularly in low-noise or near-deterministic settings where standard EI gradients tend to vanish (Ament et al., 2023, Papenmeier, 2025).

Probability of Improvement The Probability of Improvement (PI) acquisition function (Jones, 2001) prioritises regions where the likelihood of improving the best observed value is high. It is defined as:

$$\alpha_{\mathrm{PI}}(\mathbf{x}) = \mathbb{P}(f(\mathbf{x}) < f^\dagger), \tag{2.39}$$

where $f^\dagger$ denotes the current best observed function value, as defined earlier. Assuming a Gaussian surrogate model, PI can be expressed as:

$$\alpha_{\mathrm{PI}}(\mathbf{x}) = \Phi\left(\frac{\mu(\mathbf{x}) - f^\dagger}{\sigma(\mathbf{x})}\right). \tag{2.40}$$

PI is simple and easy to interpret. However, it is noted in the literature that it can be overly greedy, thus leading to premature convergence.

Upper Confidence Bound Upper Confidence Bound (UCB) (Srinivas et al., 2010) explicitly balances exploration and exploitation by defining an optimistic bound on the function:

$$\alpha_{\mathrm{UCB}}(\mathbf{x}) = \mu(\mathbf{x}) + \kappa\sigma(\mathbf{x}), \tag{2.41}$$

where $\kappa > 0$ is a parameter controlling the trade-off between exploration and exploitation. A larger κ favours exploration, while a smaller κ emphasizes exploitation. Consequently, PI is sensitive to the choice of κ.

Thompson Sampling Thompson Sampling (TS) (Thompson, 1933) selects the next evaluation point by drawing a sample from the posterior distribution:

$$\tilde{f} \sim p(f \mid \mathcal{D}_t), \tag{2.42}$$

and choosing the maximum. The TS acquisition function is then:

$$\mathbf{x}_+ = \underset{\mathbf{x}\in\mathcal{X}}{\operatorname{argmax}}\, \alpha_{\mathrm{TS}}(\mathbf{x}) = \underset{\mathbf{x}\in\mathcal{X}}{\operatorname{argmax}}\, \tilde{f}(\mathbf{x}) \quad \text{with} \quad \alpha_{\mathrm{TS}}(\mathbf{x}) := \tilde{f}(\mathbf{x}), \tag{2.43}$$

where $\tilde{f}$ is fixed for the duration of the acquisition optimisation. TS is computationally efficient as it is often used in combination with grid search techniques, explained later. Alternatively, gradient-based or evolutionary strategies can be employed.

Information-Theoretic Acquisition Functions Information-theoretic acquisition functions aim to reduce uncertainty about the global optimum, either in terms of its location $\mathbf{x}^* = \operatorname{argmax}_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$ or its corresponding value $f^* = f(\mathbf{x}^*)$. These methods select points that maximise the expected information gain about the optimisation target. Let $H[p(\mathbf{x})]$ denote the differential entropy (Cover and Thomas, 2005) of the unknown maximum value:

$$H[p(\mathbf{x})] := -\int p(\mathbf{x}) \log p(\mathbf{x})\, d\mathbf{x}. \tag{2.44}$$

The acquisition function selects the point that is expected to most reduce this entropy. Two notable instances are Predictive Entropy Search (PES) (Hernández-Lobato et al., 2014) and Max-value Entropy Search (MES) (Wang and Jegelka, 2018). PES seeks to maximise the expected reduction in entropy of the global optimum location $\mathbf{x}^*$:

$$\alpha_{\mathrm{PES}}(\mathbf{x}) = H[p(f \mid \mathbf{x}, \mathcal{D}_t)] - \mathbb{E}_{\mathbf{x}^*}\left[H[p(f \mid \mathbf{x}, \mathbf{x}^*, \mathcal{D}_t)]\right]. \tag{2.45}$$

PES involves sampling potential optima $\mathbf{x}^*$ and estimating how observing f at $\mathbf{x}$ would change the conditional entropy of the observations. It is typically approximated using expectation propagation or variational methods due to the intractability of exact computations. MES instead focuses on the maximum value $f^* = f(\mathbf{x}^*)$, and selects points that are expected to reduce the uncertainty about this scalar value:

$$\alpha_{\mathrm{MES}}(\mathbf{x}) = H[p(f \mid \mathbf{x}, \mathcal{D}_t)] - \mathbb{E}_{f^*}\left[H[p(y \mid \mathbf{x}, f^*, \mathcal{D}_n)]\right]. \tag{2.46}$$

This objective is simpler to approximate than PES, as it involves conditioning on a scalar f^* rather than the entire function. In both Equations (2.45) and (2.46) the first term can be usually computed analytically as $H[p(f \mid \mathbf{x}, \mathcal{D}_t)] = 0.5 \log[2\pi e \sigma_n(\mathbf{x})]$, whereas the second term must be approximated. As a result, MES has become a popular alternative due to its tractability and ease of implementation. Both PES and MES embody the principle of exploring regions that are most informative about the optimum, rather than simply where the surrogate model predicts high values. This leads to robust performance in challenging optimisation problems, particularly under uncertainty.

2.4. Acquisition Function Optimisation

After defining different types of acquisition function $\alpha_{(\cdot)}$, this section briefly reviews how to solve the optimisation problem:

$$\mathbf{x}_+ \in \operatorname*{argmax}_{\mathbf{x}\in\mathcal{X}} \alpha(\mathbf{x}; \mathcal{D}_t), \tag{2.47}$$

to obtain the next point $\mathbf{x}_+ \in \mathcal{X}$

Gradient-based Optimisation When the gradients of the acquisition function α w.r.t. the input variables $\mathbf{x}$, $\nabla_{\mathbf{x}}\alpha(\mathbf{x})$, are available, which are in many frameworks computable via automatic differentiation, local optimisation algorithms such as Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS-B) or Adam (Snoek et al., 2015) can be used to perform gradient ascent:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \eta\nabla_{\mathbf{x}}\alpha(\mathbf{x}^{(k)}), \tag{2.48}$$

with step size $\eta > 0$ and iteration index k. To mitigate convergence to local maxima, multi-start strategies are employed:

$$\mathbf{x}_+ = \operatorname*{argmax}_{j\in\{1,\ldots,R\}} \alpha(\mathbf{x}_j^\star), \quad \text{where} \quad \mathbf{x}_j^\star = \operatorname*{argmax}_{\mathbf{x}\in\mathcal{X}} \alpha(\mathbf{x}) \text{ with } \mathbf{x}_0^{(j)} \sim \mathcal{X}. \tag{2.49}$$

Initial points $\{\mathbf{x}_0^{(j)}\}_{j=1}^R$ are typically sampled from a space-filling design, such as Sobol (Sobol', 1967) or Latin Hypercube Sampling (LHS) (McKay et al., 1979). This approach is most effective when $\mathcal{X} \subset \mathbb{R}^D$ is box-constrained.

Batch Construction via Fantasy Models In batched BO, where q new points $\{\mathbf{x}_+^{(j)}\}_{j=1}^q$ are selected per iteration, directly optimising a joint acquisition function over $\mathcal{X}^q$ becomes intractable as q increases. A common alternative is greedy batch construction using fantasy models. Given the current dataset $\mathcal{D}_t$, candidate points are selected sequentially. At step $j \in \{1, \ldots, q\}$, a fantasy model is constructed by augmenting $\mathcal{D}_t$ with the previously selected points $\{\mathbf{x}_+^{(i)}\}_{i=1}^{j-1}$ and their fantasised objective values $\{\tilde{f}^{(i)}\}_{i=1}^{j-1}$ (e.g. the predicted posterior mean $\mu(\mathbf{x}_+^{(i)})$ of the current model $\tilde{f}^{(i)}$). Then:

$$\tilde{f}^{(j)} \sim p\left(f \mid \mathcal{D}_t \cup \{(\mathbf{x}_+^{(i)}, \tilde{f}^{(i)})\}_{i=1}^{j-1}\right), \tag{2.50}$$

and the next point is selected by maximising the acquisition function under this updated model:

$$\mathbf{x}_+^{(j)} = \arg\max_{\mathbf{x}\in\mathcal{X}} \alpha\left(\mathbf{x} \mid \mathcal{D}_t \cup \{(\mathbf{x}_+^{(i)}, \tilde{f}^{(i)}(\mathbf{x}_+^{(i)}))\}_{i=1}^{j-1}\right). \tag{2.51}$$

This iterative construction encourages diversity by conditioning each selection on the previously fantasised observations. The resulting batch is given by $\{\mathbf{x}_+^{(1)}, \ldots, \mathbf{x}_+^{(q)}\}$.

Sampling-based Search When the acquisition function is non-differentiable or the gradients $\nabla_{\mathbf{x}}\alpha$ are not available, standard gradient-based optimisation cannot be applied. This situation can arise due to noise, discrete variables, or mixed input domains, in which case a simple yet robust alternative is to use sampling-based or grid search methods. However, sampling is not only a fallback mechanism, but is also used deliberately in certain acquisition strategies, such as TS, which rely on stochastic sampling rather than explicit maximisation. Given a candidate set $\mathbf{X}_c = \{\mathbf{x}_j\}_{j=1}^{N_c} \in \mathcal{X}$, the next query point is selected via:

$$\mathbf{x}_+ = \underset{j\in\{1,\dots,N_c\}}{\operatorname{argmax}} \alpha(\mathbf{x}_j). \tag{2.52}$$

Candidates are typically drawn from quasi-random sequences such as Sobol to ensure good space coverage. This method is parallelisable, simple to implement, and robust to rugged acquisition landscapes, though it may require a large N_c to reliably locate narrow optima.

Hybrid Strategy A common practice is to use a two-stage procedure:

1. Global search via quasi-random sampling to generate a large number of candidates.

2. Local refinement via gradient-based optimisation, using the best few samples as starting points.

This hybrid strategy combines the global coverage of Sobol sequences with the precision of gradient-based optimisation and is widely used in practice, e.g. in `BoTorch` and other modern BO libraries.

2.5. Unconstrained Bayesian Optimisation in High Dimensions

Early empirical studies and practical applications of BO suggested that it performs reliably only in problems of relatively low dimensionality (typically $D \leq 20$), and that scaling the algorithm to higher dimensional problems leads to issues that need to be mitigated (Binois and Wycoff, 2022). This section briefly discusses the origin of these issues by introducing the curse of dimensionality, before an overview of the most well known methods for unconstrained high-dimensional Bayesian optimisation is given.

2.5.1. The Curse of Dimensionality

An increasing dimension of the input poses severe issues for methods like BO due to the so-called curse of dimensionality and its effects. Four main reasons can be named (Binois and Wycoff, 2022):

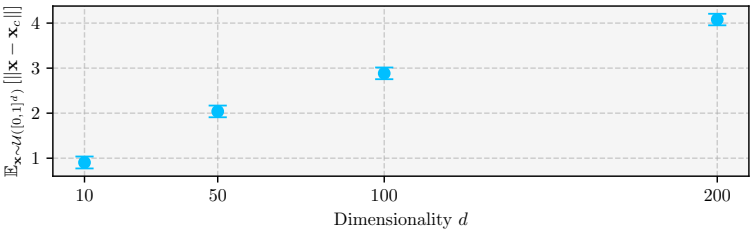


Figure 2.4: Visualising the curse of dimensionality. Computing the expected distance $\mathbb{E}_{\mathbf{x}\sim\mathcal{U}([0,1]^D)}[\|\mathbf{x}-\mathbf{x}_c\|]$ between N randomly sampled points $\mathbf{x} \sim \mathcal{U}([0,1]^d)$ and the centre $\mathbf{x}_c = [0.5, ..., 0.5]$, visualising that with increasing dimensions, the distance increases further and further, thus points populate the boundary of the hypercube.

- With increasing number of dimensions, two points lie relatively far away in this high-dimensional space, see Figure 2.4. This leads to sparser data coverage, increasing the uncertainty of the surrogate model within the space $\mathcal{X}$.
- As seen in Equation (2.14), the number of hyperparameters that need to be tuned to train the surrogate model depends on the number of dimensions. Maximising the possibly non-convex likelihood can become prohibitive, even for gradient-based optimisation techniques.
- Similarly, the acquisition function must be optimised over the high-dimensional input space $\mathcal{X}$. If a gradient-based optimiser is employed, additional difficulties may arise due to poor gradient quality or local optima. This highlights one of the reasons why sampling-based approaches can be advantageous in high-dimensional settings.

To mitigate the aforementioned difficulties, many strategies have been proposed for unconstrained optimisation problems, which are briefly introduced in the next subsection to provide an overview over the most recent literature.

2.5.2. Variable Selection or Screening

A straightforward solution, where possible, is to reduce the number of variables based on expert knowledge or engineering insight. However, in many black-box optimisation settings, such prior knowledge is limited or unavailable, requiring data-driven alternatives. A common assumption in such approaches is that only a small subset of the variables significantly influences the objective function. This implies that the full function $f(\mathbf{x})$ can be well-approximated by a lower-dimensional function $g(\tilde{\mathbf{x}})$, where $\tilde{\mathbf{x}} \in \mathbf{x}$. Identifying the relevant variables can be approached in several ways. One such method is Automatic Relevance Determination (ARD), which uses the inverse length-scale hyperparameters of a GP kernel to infer input relevance (MacKay, 1996, Neal, 1996). Variables associated with very long length scales, i.e. whose influence on the output is weak, can be considered unimportant. Alternatively, Ulmasov et al.

(2015) propose using PCA on the design matrix $\mathbf{X}$ to identify directions of high input variance, assuming these directions are more likely to influence the objective.

While variable selection can reduce dimensionality and improve model interpretability, it has limitations. Methods like ARD may be unreliable in small-data regimes or when input features are highly correlated. Moreover, variable importance may change dynamically over the course of optimisation, suggesting a need for adaptive strategies or integration with subspace modelling techniques (Binois and Wycoff, 2022).

2.5.3. Additive Models

Another path to scale BO are additive models (Kandasamy et al., 2016) which assume that the objective function $f : \mathbb{R}^D \to \mathbb{R}$ decomposes into a sum of M functions defined over low-dimensional subspaces:

$$f(\mathbf{x}) = \sum_{m=1}^{M} f_m(\mathbf{x}^{(m)}), \quad \text{where } \mathbf{x}^{(m)} \in \mathbf{x}. \tag{2.53}$$

Each component f_m depends on a (typically disjoint or partially overlapping) subset of the input variables and is modelled using an independent GP. The resulting additive GP prior corresponds to a kernel k_{add} of the form:

$$k_{\text{add}}(\mathbf{x}, \mathbf{x}') = \sum_{m=1}^{M} k_m(\mathbf{x}^{(m)}, \mathbf{x}'^{(m)}), \tag{2.54}$$

where each k_m is a kernel defined on the corresponding low-dimensional subspace. This structure enables acquisition functions to be optimised separately over each subspace, significantly reducing the effective dimensionality of the optimisation problem and improving scalability.

Additive models are particularly effective when the true objective exhibits low-order interactions, i.e. when only small groups of variables interact significantly. However, a key challenge is determining a suitable decomposition. In practice, this decomposition is often unknown, and learning it from data is non-trivial. To address this, Ziomek and Bou-Ammar (2023) propose using random additive decompositions, which avoid the need for explicit structure learning by averaging over many random partitions. While this sacrifices interpretability, it has been shown to perform well empirically, particularly in settings where no prior knowledge on input structure is available.

2.5.4. Subspace Methods

Beyond variable selection and additive models, a major strategy for addressing high dimensionality in BO involves exploiting a low-dimensional structure within the

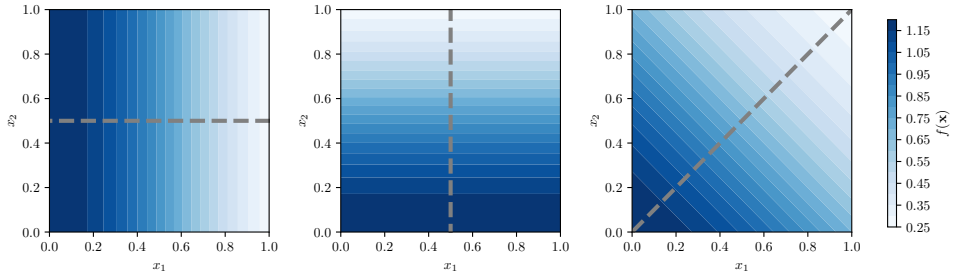


Figure 2.5: Examples of intrinsic subspaces. The first two are axis-aligned and solvable via e.g. variable selection. The third is diagonally aligned and requires linear or nonlinear subspace methods.

high-dimensional input space. The underlying assumption is that the function of interest varies primarily along a subspace $\tilde{\mathcal{X}} \subset \mathbb{R}^d$, with $d \ll D$, such that a mapping $\mathcal{P} : \mathbb{R}^D \to \mathbb{R}^d$ exists and optimisation can be performed in $\tilde{\mathcal{X}}$. Or conversely, the original input is recovered via a inverse mapping $\mathcal{P}^{-1}$ such that $\mathbf{x} = \mathcal{P}^{-1}(\tilde{\mathbf{x}})$.

Linear Embeddings One of the first approaches to exploiting low-dimensional structure in BO is Random Embedding Bayesian Optimisation (REMBO) (Wang et al., 2016), which introduces a linear embedding via a random projection matrix $\mathbf{A} \in \mathbb{R}^{D \times d}$. An inverse mapping is defined as $\mathcal{P}^{-1}(\tilde{\mathbf{x}}) = \mathbf{A}\tilde{\mathbf{x}}$, and optimisation is performed in the subspace $\tilde{\mathcal{X}}$. However, this mapping can produce $\mathbf{x} = \mathbf{A}\tilde{\mathbf{x}}$ that lies outside of the feasible domain $\mathcal{X}$. To address this, a projection onto the feasible domain is applied:

$$p_{\mathcal{X}}(\mathbf{A}\tilde{\mathbf{x}}) = \operatorname*{argmin}_{\mathbf{z} \in \mathcal{X}} \|\mathbf{z} - \mathbf{A}\tilde{\mathbf{x}}\|_2, \tag{2.55}$$

though this may lead to over-exploration near the boundary. Adaptive Linear Embedding Bayesian Optimisation (ALEBO) (Letham et al., 2020) improves this by constraining the acquisition function optimisation:

$$\max_{\tilde{\mathbf{x}} \in \tilde{\mathcal{X}}} \alpha(\tilde{\mathbf{x}}) \quad \text{s.t.} \quad \mathbf{A}\tilde{\mathbf{x}} \in \mathcal{X}, \tag{2.56}$$

thus ensuring feasibility without post hoc projection. Similarly, Hashing-enhanced Subspace Bayesian Optimisation (HeSBO) (Nayebi et al., 2019) uses a structured random matrix $\mathbf{A}$ with elements in $\{-1, 0, 1\}$, avoiding the back-projection issue entirely. These methods are all based on random, unsupervised projections. In contrast, Raponi et al. (2020) incorporate task information by constructing the subspace via output-weighted PCA, guiding the search towards more promising regions. Optimisation is then performed in the learned subspace using a penalised acquisition function.

Nonlinear Embeddings When the objective varies along a nonlinear manifold, linear projections may be insufficient. In such cases, autoencoder-based methods

learn nonlinear mappings:

$$\mathbf{x} = \phi(\tilde{\mathbf{x}}), \quad \tilde{\mathbf{x}} = \psi(\mathbf{x}), \tag{2.57}$$

where ϕ and ψ are decoder and encoder functions, respectively. GPs are then trained in the latent space $\tilde{\mathcal{X}}$, leveraging its reduced dimensionality while maintaining a mapping to the original input space. Gómez-Bombarelli et al. (2018) demonstrated this approach in molecular design, ensuring that the latent space remains smooth and meaningful. Follow-up work extended this idea to BO settings: Moriconi et al. (2020) use MTGPs for reconstructing the objective from latent inputs, Grosnit et al. (2021) and Tripp et al. (2020) combine variational autoencoders with deep metric learning to shape latent spaces that reflect function behaviour. Maus et al. (2024) apply joint latent embeddings to composite high-dimensional functions. While nonlinear embeddings increase modelling flexibility, they introduce additional complexity. Training robust autoencoder depends on principled architectural choices, such as latent dimensionality and regularisation, to avoid degenerate or uninformative embeddings. Additionally, acquisition optimisation must account for the geometry of the latent space. Moreover, ensuring that the latent space supports smooth backmapping to valid, interpretable high-dimensional inputs remains a key challenge in practical applications.

2.5.5. Trust-Region Bayesian Optimisation

In high-dimensional BO, maximising the acquisition function over the entire input domain $\mathcal{X}$ can become inefficient and unstable due to the curse of dimensionality and the challenge of surrogate model reliability far away from observed data (Papenmeier et al., 2025). TR methods address this challenge by restricting the optimisation to a local region $\mathcal{T} \subset \mathcal{X}$, typically centred around the best solution found so far $\mathbf{x}^\dagger \in \mathcal{X}$. Eriksson et al. (2019) propose the Trust Region Bayesian Optimisation (TuRBO) algorithm, which follows this principle by dynamically adapting a TR during the optimisation process. At iteration t, the TR is defined as a hypercube of side length r centred at the current incumbent solution, given by:

$$\mathbf{x}^\dagger = \underset{\mathbf{x}_i \in \mathcal{D}_t}{\operatorname{argmax}}\, f(\mathbf{x}_i). \tag{2.58}$$

The TR is then defined as:

$$\mathcal{T}(\mathbf{x}^\dagger, r) = \left\{ \mathbf{x} \in \mathcal{X} \,\middle|\, x_z^\dagger - \tfrac{r}{2} \leq x_z \leq x_z^\dagger + \tfrac{r}{2}, \quad \forall z = 1, \ldots, D \right\}, \tag{2.59}$$

and is clipped to remain within the bounds of the global input domain $\mathcal{X}$, typically normalised to $[0,1]^D$. The next evaluation point $\mathbf{x}_+$ is chosen by maximising the acquisition function within the TR:

$$\mathbf{x}_+ = \underset{\mathbf{x} \in \mathcal{T}(\mathbf{x}^\dagger, r)}{\operatorname{argmax}}\, \alpha_{(\cdot)}(\mathbf{x}; \mathcal{D}_t), \tag{2.60}$$

This heuristic promotes local exploitation while retaining the potential for global exploration through adaptive expansion and restarts and is visualised in Figure 2.6.

In Eriksson et al. (2019), the authors use TS in combination with a dynamically adaptive side length r, based on the optimiser's progress. Let τ_s and τ_f denote thresholds for consecutive successes and failures, respectively. A step is considered a success if the new point improves upon the incumbent such that $f(\mathbf{x}_+) > f(\mathbf{x}^\dagger)$ and a failure otherwise. After τ_s consecutive successes, the region expands as $r \leftarrow \min(2r, r_{\max})$, while it contracts after τ_f failures: $r \leftarrow r/2$. If $r < r_{\min}$, the TR is restarted around a new candidate point. To incorporate the anisotropy of the surrogate model, Eriksson et al. (2019) further scale the side lengths according to the GP length scales $\{l_i\}_{i=1}^D$:

$$r_i = \frac{l_i r}{\left(\prod_{j=1}^D l_j\right)^{1/D}}, \tag{2.61}$$

ensuring that the TR adapts to the local geometry implied by the surrogate.

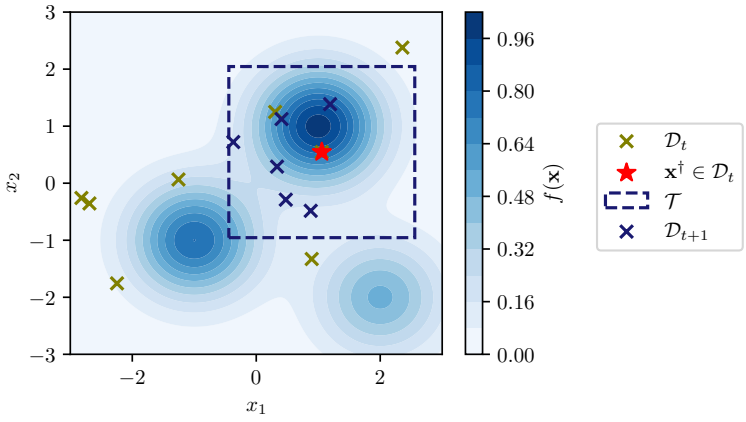


Figure 2.6: Illustration of a trust region $\mathcal{T}(\mathbf{x}^\dagger, r)$ centred around the best solution $\mathbf{x}^\dagger$. Candidate points are sampled within the region, which expands or contracts based on optimisation progress.

Notably, in Eriksson et al. (2019), candidate points $\mathbf{X}_c$ are sampled not via Sobol sequences, but by perturbing a subset of dimensions of $\mathbf{x}^\dagger$. Each dimension is perturbed independently with probability $\min(1, 20/D)$, enabling efficient sampling in high dimensions. This heuristic, also referred to as Random Axis-Aligned Subspace Perturbation (RAASP), is further investigated in Ament et al. (2023) and Rashidi et al. (2024). Latter propose an extension called cylindrical TS, which improves sampling efficiency and overall performance in high-dimensional settings.

Trust Region–Subspace Hybridisation Bayesian Optimisation in Adaptively Expanding Subspaces (BAxUS) (Papenmeier et al., 2023) extends the TuRBO strategy by combining TR optimisation with a gradually expanding, HeSBO-style subspace

embedding. This is motivated by the observation that many high-dimensional functions possess an intrinsic low-dimensional active subspace, of effective dimension $d \ll D$, where most of the variation in the objective occurs. At each iteration, the optimisation is performed in a randomly projected subspace:

$$f(\mathbf{x}) = f(\mathbf{A}\tilde{\mathbf{x}}), \quad \tilde{\mathbf{x}} \in \mathbb{R}^d, \quad \mathbf{A} \in \{-1, 0, 1\}^{D \times d}, \tag{2.62}$$

where $\mathbf{A} : \mathbb{R}^d \to \mathbb{R}^D$ defines a sparse embedding matrix. Each input dimension contributes to exactly one subspace coordinate, similar to the hashing scheme used in HeSBO (Nayebi et al., 2019). However, BAxUS improves on this by balancing bin sizes and introducing an embedding growth mechanism via nested subspaces. Let d_0 denote the initial embedding dimension. After each restart or failure to make progress, e.g. when the TR size r falls below a minimum threshold $r_{\min}$, BAxUS increases the embedding dimensionality according to:

$$d_{n+1} = \min\left(d_n(b+1), D\right), \tag{2.63}$$

where b denotes the number of new bins per dimension split. This yields a sequence of nested embeddings $\tilde{\mathcal{X}}_1 \subset \tilde{\mathcal{X}}_2 \subset \ldots$ of increasing dimensionality, allowing the optimiser to gradually explore more complex regions of the input space without full restarts. Importantly, the GP surrogate is trained and the acquisition is performed in the subspace $\tilde{\mathcal{X}} \subset \mathbb{R}^d$ using TS.

As in TuRBO, a TR is maintained around the current incumbent $\tilde{\mathbf{x}}^\dagger$ in subspace coordinates, and the acquisition function is maximised within this region. The side lengths are adapted based on GP length scales and the optimiser's recent progress. However, rather than performing a random restart upon stagnation and generating a new random subspace, BAxUS expands the embedding dimension, preserving past observations through projection consistency. This hybrid approach leverages the benefits of both TR-based local search and subspace-based dimensionality reduction. The authors demonstrate that combining these strategies improves sample efficiency and stability in high-dimensional Bayesian optimisation tasks over full-dimensional TR methods.

2.5.6. Length-scale Initialisation and Learning

While many high-dimensional BO methods rely on structural assumptions such as the existence of a low-dimensional active subspace, additive structure, or local TR constraints, recent work (Hvarfner et al., 2024, Xu et al., 2025, Papenmeier et al., 2025) has focused on identifying the core failure mechanisms of Vanilla Bayesian Optimisation (VBO) in high dimensions. Rather than introducing additional structure, these approaches examine how standard GP models behave under high-dimensional settings, and how they can be corrected with minimal modification.

Hvarfner et al. (2024) identify degeneracies in standard GP priors as a key failure mode. In particular, when the length-scale prior is not adapted to the ambient

dimension D, the implied function class becomes too flexible, leading to overfitting. They propose a dimension-scaled log-normal prior, in the following also referred to as Dimensionality-Scaled Prior (DSP):

$$l \sim \text{LogNormal}(\mu, \sigma^2), \quad \text{with} \quad \mu \propto D, \tag{2.64}$$

which regularises the model towards smoother functions as dimensionality increases. The underlying assumption is that every black-box function is simple enough to be modelled globally, independent of the dimensionality. This correction improves model stability and posterior calibration in high dimensions. Empirically, this simple prior adjustment enables standard BO (without subspace modelling) to match or outperform state-of-the-art methods across benchmark tasks.

In parallel, Xu et al. (2025) examine a related failure mode: vanishing gradients during hyperparameter learning in high-dimensional spaces. They propose an even simpler fix by initialising the length scale according to:

$$l_0 = c\sqrt{D}, \quad c > 0, \tag{2.65}$$

which improves the conditioning of the marginal likelihood and prevents gradient collapse early in training. This leads to faster convergence and improved robustness, even without structural assumptions.

Building on these insights, Papenmeier et al. (2025) analyse why standard BO methods sometimes work surprisingly well in high dimensions. They show that the combination of appropriate length-scale initialisation and random axis-aligned perturbations, e.g. as in Rashidi et al. (2024), Ament et al. (2023), enables surprisingly effective exploration. They argue that apparent high-dimensional performance arises from local behaviour and posterior variance scaling, and recommend dimension-aware length-scale heuristics to maintain model stability.

Collectively, these studies demonstrate that with DSP or initialisations, standard BO approaches, without explicit subspaces or architectural assumptions, can remain competitive in high-dimensional, unconstrained problems.

Another approach is taken by Eriksson and Jankowiak (2021), who propose a sparsity-inducing prior to reflect the belief that only a subset of input dimensions are relevant. Their method, Sparse Axis-Aligned Subspace Bayesian Optimisation (SAASBO), imposes a global shrinkage prior via:

$$\nu \sim \mathcal{HC}(\beta), \quad \text{and} \quad l_i \sim \mathcal{HC}(\nu), \tag{2.66}$$

where $\mathcal{HC}(\cdot)$ denotes the half-Cauchy distribution, ν a global shrinkage parameter and β controls the level of shrinkage. This hierarchical prior encourages long length scales in irrelevant dimensions and shorter length scales only where the objective is sensitive.

As optimisation progresses, the model automatically identifies a low-dimensional axis-aligned subspace. While SAASBO has demonstrated strong empirical performance, it incurs substantial computational overhead, particularly due to the hierarchical sampling and MCMC inference steps, making it less scalable to larger datasets.

2.6. Bayesian Optimisation with Unknown Constraints

This section extends the theory of unconstrained BO to the setting where the feasible region is implicitly defined by unknown, black-box constraints. Such constraints are typically expensive to evaluate, non-differentiable, and lack analytical form, ruling out classical optimisation techniques. Their unknown nature can necessitate the construction of separate surrogate models to estimate feasibility, and introduces additional uncertainty into the optimisation process. These features make BO particularly well-suited, as it naturally balances the trade-off between exploring feasible regions and exploiting high-performing areas under uncertainty.

The section begins by discussing different modelling and acquisition strategies developed for CBO, followed by a review of techniques aimed at scaling CBO to high-dimensional input spaces. The general constrained optimisation problem considered in this work is formulated as:

$$\begin{aligned} \min_{\mathbf{x}\in\mathcal{X}\subset\mathbb{R}^D} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & c_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, G, \end{aligned} \tag{2.67}$$

where $f : \mathcal{X} \to \mathbb{R}$ is the objective function, $\mathcal{X} \subset \mathbb{R}^D$ denotes the bounded design domain and G is the number of constraints. The constraint functions $\mathbf{c} : \mathcal{X} \to \mathbb{R}^G$ define a feasible region:

$$\mathcal{X}_f = \{\mathbf{x} \in \mathcal{X} \,|\, c_i(\mathbf{x}) \leq 0, \forall i = 1, \dots, G\}, \tag{2.68}$$

and the goal is to find the optimal solution $\mathbf{x}^* \in \mathcal{X}_f$ minimising the objective within this feasible set. Furthermore, both the objective f and the constraint functions c_i $\forall\{1, ..., G\}$ are assumed to be expensive-to-evaluate black-box functions. As a result, their functional forms are unknown, gradients are unavailable, and evaluations are only accessible pointwise. Consequently, all objective and constraints must be modelled jointly using probabilistic surrogate models. Thus, at iteration t, the dataset consists of observations of both the objective and the constraints:

$$\mathcal{D}_t = \{(\mathbf{x}_j, f_j, \mathbf{c}_j)\}_{j=1}^{N_t}, \tag{2.69}$$

where $f_j = f(\mathbf{x}_j)$ and $\mathbf{c}_j = \{c_i(\mathbf{x}_j)\}_{i=1}^{G}$. The surrogate models are then used to guide the selection of future query points via acquisition functions that incorporate both performance and feasibility considerations. A conceptual illustration of this constrained optimisation setup is provided in Figure 2.7.

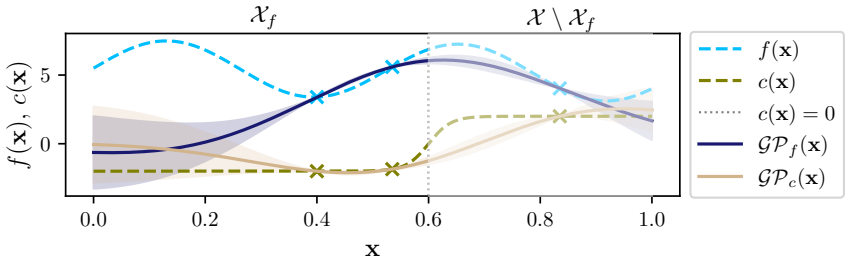


Figure 2.7: Illustrative example of CBO in one dimension. The true objective function $f(x)$ and constraint function $c(x)$ are modelled using GP ($\mathcal{GP}_f$, $\mathcal{GP}_c$) with predictive means and confidence intervals. The vertical dotted line denotes the feasibility threshold $c(x) = 0$, separating the feasible region $\mathcal{X}_f \subset \mathcal{X}$. Observed data points are shown as crosses. The plot highlights the need to balance objective optimisation with constraint satisfaction.

2.6.1. Challenges in Constrained Bayesian Optimisation

CBO seeks to optimise black-box objective functions under unknown constraints and tight evaluation budgets. This introduces several additional challenges beyond those already presented in Section 2.5.1, making the problem significantly more difficult, particularly in simulation-based settings:

- *Unknown Feasible Region*: The constraint functions $c_i(\mathbf{x})$ are typically black-box, making the feasible set $\mathcal{X}_f$ unknown a priori. It must be learned from data, requiring joint modelling of objective and constraints, and a trade-off between exploring feasibility and exploiting known feasible areas.
- *Infeasible Observations*: Much of the design space may be infeasible, making random sampling inefficient, yielding only infeasible points. Without feasible observations, model training and guided exploration are impaired.
- *Conflicting Acquisition Objectives*: Acquisition functions must balance feasibility and objective improvement. High-objective regions may be infeasible, while feasible ones may not improve performance. This tension complicates acquisition function design and optimisation.
- *Expensive Constraint Evaluations*: Constraint evaluations may require costly simulations, e.g. PDE solvers for structural or thermal analyses, potentially exceeding the cost of objective evaluations.
- *Rare Feasibility*: Feasible regions may be sparse or disconnected, making it difficult to find even a single feasible point. This might be an essential prerequisite for many CBO strategies to operate effectively.

2.6.2. Surrogate Modelling of Constraints

The choice of modelling strategy for the objective and constraints plays a critical role in the efficiency and accuracy of the optimisation process. Two broad approaches

can be distinguished: independent modelling and joint modelling.

Independent Modelling A straightforward strategy is to model each constraint independently using a separate GP. This is commonly done when there is no prior knowledge of relationships between constraints or between the constraints and the objective function. Independent modelling simplifies inference and allows for modular design of the acquisition function. However, it may ignore useful inductive biases or correlations that could improve sample efficiency.

Joint Modelling of Objective and Constraints When correlations exist between constraints, or between constraints and the objective function, it can be beneficial to model them jointly. This can be achieved using multi-output GP models, such as the ICM or LMC, see Section 2.2.3. These models capture shared latent structure across outputs, allowing information to be transferred between tasks. Joint modelling can be particularly effective when constraints are physically or functionally related to the objective or to each other. This is often the case in structural design, where stress limits and performance metrics depend on the same input features. It is also advantageous when some outputs are expensive or sparsely sampled, while others are cheaper and informative, allowing for transfer learning across tasks. Moreover, when the feasible region is small or fragmented, data-efficient constraint modelling becomes essential.

Considerations and Limitations While joint models offer potential gains in data efficiency, they require additional assumptions about task correlation and typically introduce more hyperparameters, which may lead to overfitting if not carefully regularised. Furthermore, joint modelling may not always lead to improved performance if the outputs are weakly correlated or if the added model complexity is not justified by the available data. However, while computational complexity for modelling functions jointly is considerably higher than for modelling independently due to the populated off-diagonals of the covariance matrix $\mathbf{K}$. In problems where the number of constraints is high, the computational costs may be too expensive.

2.6.3. Acquisition Functions for Constrained Problems

In CBO, a standard modelling approach is to represent the objective and constraint functions using independent GPs. The objective function is typically modelled as:

$$f(\mathbf{x}) \mid \mathcal{D}_t \sim \mathcal{GP}(\mu_f(\mathbf{x}), k_f(\mathbf{x}, \mathbf{x}')). \tag{2.70}$$

Since correlations between the objective and constraint functions are generally unknown, each constraint is modelled independently:

$$c_j(\mathbf{x}) \mid \mathcal{D}_t \sim \mathcal{GP}(\mu_j(\mathbf{x}), k_j(\mathbf{x}, \mathbf{x}')), \quad \forall j = 1, \ldots, G. \tag{2.71}$$

The acquisition function $\alpha(\mathbf{x}; \mathcal{D}_t)$ quantifies the utility of evaluating a new point $\mathbf{x}$. In constrained settings, this utility must reflect both the potential for objective

improvement and the likelihood of satisfying the constraints. The most widely used acquisition strategies for constrained problems are introduced in the following.

Soft Penalties A classical and straightforward approach to constrained optimisation reformulates the problem as an unconstrained one by adding a penalty term for constraint violation to the objective (or acquisition) function:

$$f_p(\mathbf{x}) = f(\mathbf{x}) + \lambda \sum_{i=1}^{G} \max(0, c_i(\mathbf{x}))^p, \tag{2.72}$$

where $\lambda > 0$ is a penalty weight and $p \geq 1$ determines the severity of the penalisation. Common choices include linear ($p = 1$) and quadratic ($p = 2$) penalties. While simple and widely applicable, this method is sensitive to the choice of λ, and can suffer from numerical instability or poor convergence behaviour, particularly when the penalty term overwhelms the objective or distorts the feasible region. After reformulation, standard unconstrained acquisition strategies can be directly applied.

Constrained Expected Improvement Constrained Expected Improvement (CEI) (Gelbart et al., 2014, Gardner et al., 2014) extends the classical EI acquisition function (see Section 2.3) by incorporating the probability of feasibility. Given the best observed feasible value $f^\dagger \in \mathcal{D}_t$, the CEI at a candidate point $\mathbf{x}$ is defined as:

$$\alpha_{\text{CEI}}(\mathbf{x}) = \alpha_{\text{EI}}(\mathbf{x}) \cdot \prod_{j=1}^{G} \Pr(c_j(\mathbf{x}) \leq 0) \tag{2.73}$$

$$= \mathbb{E}_{f(\mathbf{x})}[\max(f^\dagger - f(\mathbf{x}), 0)] \cdot \prod_{j=1}^{G} \Phi_j(\mathbf{x}), \tag{2.74}$$

where $\Phi_j(\mathbf{x}) = \Pr(c_j(\mathbf{x}) \leq 0)$ denotes the probability of satisfying constraint j under its GP posterior. As in the unconstrained case, the expected improvement term admits a closed-form expression for Gaussian surrogates. This acquisition function encourages selection of points that are both likely to satisfy all constraints and to improve upon the best feasible objective. To improve numerical robustness, particularly in high-dimensional or near-feasible settings, Ament et al. (2023) propose a logarithmic variant of CEI, defined as:

$$\alpha_{\text{LogCEI}}(\mathbf{x}) = \alpha_{\text{LogEI}}(\mathbf{x}) + \sum_{j=1}^{G} \log\left(\Pr(c_j(\mathbf{x}) \leq 0)\right), \tag{2.75}$$

where α_{LogEI} is the logarithmic expected improvement introduced for unconstrained BO. This log-formulation reduces numerical underflow in the acquisition landscape and promotes stability during optimisation.

Constrained Thompson Sampling Constrained Thompson Sampling (CTS) (Eriksson and Poloczek, 2021) samples candidate functions $\tilde{f}, \tilde{c}_1, ..., \tilde{c}_G$ from the posterior distributions of both the objective and constraints. At iteration t, one draws:

$$\tilde{f} \sim \mathcal{GP}(f \mid \mathcal{D}_t), \tag{2.76}$$

$$\tilde{c}_j \sim \mathcal{GP}(c_j \mid \mathcal{D}_t), \quad j = 1, \ldots, G. \tag{2.77}$$

A new point is selected as the feasible minimiser of the sampled surrogate functions:

$$\mathbf{x}_+ = \begin{cases} \operatorname{argmax}_{\mathbf{x}\in\mathcal{X}} \tilde{f}(\mathbf{x}) & \text{if } \exists \mathbf{x} \in \mathcal{X} \text{ s.t. } \tilde{c}_j(\mathbf{x}) \leq 0 \ \forall j, \\ \operatorname{argmin}_{\mathbf{x}\in\mathcal{X}} \sum_j \max(0, \tilde{c}_j(\mathbf{x})) & \text{otherwise.} \end{cases} \tag{2.78}$$

In practice, this is often done via random sampling over a candidate set $\mathbf{X}_c \subset \mathcal{X}$, followed by filtering infeasible samples and selecting the one with the lowest $\tilde{f}(\mathbf{x})$.

Algorithm 2 Constrained Thompson Sampling

Given data $\mathcal{D}_t$, construct candidate set $\mathbf{X}_c \subset \mathcal{X}$
Sample functions $\tilde{f} \sim p(f \mid \mathcal{D}_t)$ and $\tilde{c}_j \sim p(c_j \mid \mathcal{D}_t)$ for all constraints $j = 1, \ldots, G$
Evaluate $\tilde{f}(\mathbf{x})$ and $\tilde{c}_j(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{X}_c$
Identify feasible subset $\mathcal{X}_f = \{\mathbf{x} \in \mathcal{X}_c : \tilde{c}_j(\mathbf{x}) \leq 0, \forall j\}$
if $\mathcal{X}_f \neq \emptyset$ **then**
    Select $\mathbf{x}_+ = \arg\max_{\mathbf{x}\in\mathcal{X}_f} \tilde{f}(\mathbf{x})$
else
    Select $\mathbf{x}_+ = \arg\min_{\mathbf{x}\in\mathcal{X}} \sum_{j=1}^{G} \max(0, \tilde{c}_j(\mathbf{x}))$
end if
Evaluate true $f(\mathbf{x}_+)$ and constraints $c_j(\mathbf{x}_+)$
Update dataset $\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{(\mathbf{x}_+, f(\mathbf{x}_+), c_j(\mathbf{x}_+))\}$

Predictive Entropy Search with Constraints Predictive Entropy Search with Constraints (PESC) (Hernández-Lobato et al., 2016) is an extension of PES to constrained problems. It selects points that maximise the expected information gain about the location of the constrained global optimum $\mathbf{x}^*$. This optimum is defined as:

$$\mathbf{x}^* = \operatorname*{argmax}_{\mathbf{x}\in\mathcal{X}_f} f(\mathbf{x}) \quad \text{with} \quad \mathcal{X}_f = \{\mathbf{x} \in \mathcal{X} \mid c_j(\mathbf{x}) \leq 0 \ \forall j = 1, \ldots, G\}. \tag{2.79}$$

The corresponding acquisition function maximises the mutual information between the next query and the unknown optimum:

$$\alpha_{\text{PESC}}(\mathbf{x}) = H[p(\mathbf{x}^* \mid \mathcal{D}_t)] - \mathbb{E}_{\mathbf{u}|\mathbf{x},\mathcal{D}_t}\left[H[p(\mathbf{x}^* \mid \mathcal{D}_t \cup (\mathbf{x}, \mathbf{u}))]\right], \tag{2.80}$$

where $\mathbf{u} = [f(\mathbf{x}), c_1(\mathbf{x}), \ldots, c_G(\mathbf{x})]$ is the joint observation at $\mathbf{x}$, and $H[\cdot]$ denotes again differential entropy. Here again, under a Bayesian model the first term $H[p(\mathbf{x}^* \mid \mathcal{D}_t)]$ can be computed exactly, whereas the second has to be approximated.

Constrained Max-value Entropy Search Constrained Max-value Entropy Search (CMES) (Perrone et al., 2019, Takeno et al., 2022) extends MES to constrained settings by selecting evaluation points that are expected to yield the greatest reduction in uncertainty about the constrained global maximum:

$$f^* = \max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \quad \text{s.t.} \quad c_j(\mathbf{x}) \leq 0, \ \forall j \in \{1, \dots, G\}. \tag{2.81}$$

The CMES acquisition function quantifies the mutual information between a candidate point $\mathbf{x}$ and the unknown optimal value f^*:

$$\alpha_{\text{CMES}}(\mathbf{x}) = H(p(f^* \mid \mathcal{D}_t)) - \mathbb{E}_{\mathbf{u}}\left[H(p(f^* \mid \mathcal{D}_t \cup \{(\mathbf{x}, \mathbf{u})\}))\right], \tag{2.82}$$

where $\mathbf{u} = [f(\mathbf{x}), c_1(\mathbf{x}), \dots, c_G(\mathbf{x})]$ is the vector of outcomes at $\mathbf{x}$, and the expectation is taken over their joint posterior. The first entropy term can often be computed analytically or approximated using samples of feasible maxima. The second term, however, requires integrating over the posterior of all model outputs at $\mathbf{x}$, which is intractable in general. Perrone et al. (2019) proposed an approach for the special case of a single constraint, whereas Takeno et al. (2022) introduced a general variational lower bound formulation for arbitrary numbers of constraints.

2.6.4. Scaling Constrained Bayesian Optimisation with Trust Regions

Scaling BO to high-dimensional spaces presents significant challenges due to the curse of dimensionality: surrogate models and consequently acquisition functions become less informative as dimensionality increases. This difficulty is further compounded in constrained settings, where the optimiser must not only locate high-performing regions but also identify and remain within the feasible set. To address these issues, recent work has focused on restricting the search to local subspaces or TRs where the optimisation is more tractable.

Scalable Constrained Bayesian Optimisation Scalable Constrained Bayesian Optimisation (SCBO) (Eriksson and Poloczek, 2021) extends the TuRBO framework, see Section 2.5.5, to the constrained setting. SCBO constructs $G + 1$ independent probabilistic surrogate models, one for the objective function f and one for each constraint c_j. CTS is used as acquisition function. To scale this process to high dimensions, SCBO operates within a local TR centred at the best point found so far:

$$\mathbf{x}^\dagger = \begin{cases} \text{argmax}_{\mathbf{x}\in\mathcal{X}} \, f(\mathbf{x}) & \text{if } \exists \mathbf{x} \text{ such that } c_j(\mathbf{x}) \leq 0 \ \forall j, \\ \text{argmin}_{\mathbf{x}\in\mathcal{X}} \sum_{j=1}^{G} \max(0, c_j(\mathbf{x})) & \text{otherwise.} \end{cases} \tag{2.83}$$

using a side length r as in TuRBO, initialised as $r = r_{\text{init}}$. As before, the TR tracks the number of successes and failures according to the progress of the optimisation. A success occurs when a new point improves over the current best point, a failure occurs

when no such improvement is observed within the batch. The TR is always centred around the best point found so far. If a certain threshold is met, the TR is either expanded $r \leftarrow \min(2r, r_{\min})$ or shrunken $r \leftarrow r/2$. If $r < r_{\min}$ the TR is restarted. Within this TR, CTS is employed to find the next point to evaluate. Candidate points $\mathbf{X}_c$ are generated via RAASP of this incumbent solution, forming a batch that lies within a small subspace. A binary perturbation mask controls which input dimensions are modified, ensuring that search remains local. Specifically, for each batch element, a subset of dimensions is selected independently with a fixed probability, and perturbations are drawn from a standardised distribution, e.g. uniform or Gaussian. This is all similar to the unconstrained TuRBO algorithm, except for the constraint handling. The mechanism balances exploration (via expansion) and exploitation (via contraction), while always preferring feasibility over optimality, allowing SCBO to progressively identify feasible regions and refine solutions within them.

Feasibility-Driven Trust Region Bayesian Optimisation In SCBO, the TR is a hypercube with side length r, centred around the current best point $\mathbf{x}^\dagger$, and updated heuristically based on the number of successes or failures in the batch. In contrast, Ascia et al. (2025) propose a constraint-driven TR, named Feasibility-Driven Trust Region Bayesian Optimisation (FuRBO), where the shape and size of the region are inferred from the constraint models themselves. Specifically, instead of relying on a fixed side-length and success/failure counters, the TR is adaptively defined by "constraint inspectors", metrics derived from the GP posteriors over the constraints. Around the selected centre $\mathbf{x}^\dagger$, a set of local inspector points is generated within a radius r. These are perturbed versions of $\mathbf{x}^\dagger$, e.g.:

$$\mathbf{x}^{(i)} = \mathbf{x}^\dagger + \boldsymbol{\delta}^{(i)}, \quad \|\boldsymbol{\delta}^{(i)}\|_\infty \leq r/2. \tag{2.84}$$

The inspector points are ranked using the same feasibility-first criterion as above. The top $P\%$ of inspectors according to this ranking are retained and define the TR $\mathcal{T}$ as:

$$\mathcal{T} = \left\{ \mathbf{x} \in \mathbb{R}^D \;\middle|\; \ell_d \leq x_d \leq u_d, \quad \forall d \in \{1, \ldots, D\} \right\} \tag{2.85}$$

$$\text{with} \quad a_d = \min_{1 \leq i \leq P} x_{id}, \quad b_d = \max_{1 \leq i \leq P} x_{id}. \tag{2.86}$$

In this new defined TR CTS is applied to find the new point. Numerical experiments show that this method outperforms or is at least on par with SCBO due to its more flexible, constrained-driven TR formulation.

BIBLIOGRAPHY

M. A. Álvarez and N. D. Lawrence. Computationally efficient convolved multiple output gaussian processes. *Journal of Machine Learning Research*, 12(41):1459–1500, 2011. URL http://jmlr.org/papers/v12/ez11a.html.

M. A. Alvarez, L. Rosasco, and N. D. Lawrence. Kernels for Vector-Valued Functions: a Review, Apr. 2012. URL http://arxiv.org/abs/1106.6251. arXiv:1106.6251 [cs, math, stat].

S. Ament, S. Daulton, D. Eriksson, M. Balandat, and E. Bakshy. Unexpected Improvements to Expected Improvement for Bayesian Optimization, Jan. 2023. URL http://arxiv.org/abs/2310.20708. arXiv:2310.20708 [cs].

M. Amine Bouhlel, N. Bartoli, R. G. Regis, A. Otsmane, and J. Morlier. Efficient global optimization for high-dimensional constrained problems by using the Kriging models combined with the partial least squares method. *Engineering Optimization*, 50(12):2038–2053, Dec. 2018. ISSN 0305-215X, 1029-0273. doi: 10.1080/0305215X.2017.1419344. URL https://www.tandfonline.com/doi/full/10.1080/0305215X.2017.1419344.

P. Ascia, E. Raponi, T. Bäck, and F. Duddeck. Feasibility-Driven Trust Region Bayesian Optimization, June 2025. URL http://arxiv.org/abs/2506.14619. arXiv:2506.14619 [cs].

M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*, 2020. URL http://arxiv.org/abs/1910.06403.

R. Bellman. *Dynamic programming*. Princeton Univ. Pr, Princeton, NJ, 1957. ISBN 978-0-691-07951-6.

M. Binois and N. Wycoff. A Survey on High-dimensional Gaussian Process Modeling with Application to Bayesian Optimization. *ACM Transactions on Evolutionary Learning and Optimization*, 2(2):1–26, June 2022. ISSN 2688-299X, 2688-3007. doi: 10.1145/3545611. URL https://dl.acm.org/doi/10.1145/3545611.

E. V. Bonilla, K. Chai, and C. Williams. Multi-task gaussian process prediction. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/66368270ffd51418ec58bd793f2d9b1b-Paper.pdf.

T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1 edition, Sept. 2005. ISBN 978-0-471-24195-9 978-0-471-74882-3. doi: 10.1002/047174882X. URL https://onlinelibrary.wiley.com/doi/book/10.1002/047174882X.

D. Eriksson and M. Jankowiak. High-Dimensional Bayesian Optimization with Sparse Axis-Aligned Subspaces, June 2021. URL http://arxiv.org/abs/2103.00349. arXiv:2103.00349 [cs].

D. Eriksson and M. Poloczek. Scalable Constrained Bayesian Optimization, Feb. 2021. URL http://arxiv.org/abs/2002.08526. arXiv:2002.08526 [cs, stat].

D. Eriksson, M. Pearce, J. Gardner, R. D. Turner, and M. Poloczek. Scalable Global Optimization via Local Bayesian Optimization. 2019.

A. I. Forrester, A. Sóbester, and A. J. Keane. Multi-fidelity optimization via surrogate modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 463(2088):3251–3269, Dec. 2007. ISSN 1364-5021, 1471-2946. doi: 10.1098/rspa.2007.1900.

P. I. Frazier. A Tutorial on Bayesian Optimization, July 2018. URL http://arxiv.org/abs/1807.02811. arXiv:1807.02811 [cs, math, stat].

J. R. Gardner, M. J. Kusner, Z. Xu, K. Q. Weinberger, and J. P. Cunningham. Bayesian optimization with inequality constraints. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, page II–937–II–945. JMLR.org, 2014.

J. R. Gardner, G. Pleiss, D. Bindel, K. Q. Weinberger, and A. G. Wilson. GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration, June 2021. URL http://arxiv.org/abs/1809.11165. arXiv:1809.11165 [cs].

M. A. Gelbart, J. Snoek, and R. P. Adams. Bayesian optimization with unknown constraints. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, UAI'14, page 250–259, Arlington, Virginia, USA, 2014. AUAI Press. ISBN 9780974903910.

M. G. Genton. Classes of kernels for machine learning: a statistics perspective. *J. Mach. Learn. Res.*, 2:299–312, Mar. 2002. ISSN 1532-4435.

P. Goovaerts. *Geostatistics for Natural Resources Evaluation*. Oxford University PressNew York, NY, Sept. 1997. ISBN 978-0-19-511538-3 978-0-19-770959-7. doi: 10.1093/oso/9780195115383.001.0001. URL https://academic.oup.com/book/53785.

A. Grosnit, R. Tutunov, A. M. Maraval, R.-R. Griffiths, A. I. Cowen-Rivers, L. Yang, L. Zhu, W. Lyu, Z. Chen, J. Wang, J. Peters, and H. Bou-Ammar. High-Dimensional Bayesian Optimisation with Variational Autoencoders and Deep Metric Learning, Nov. 2021. URL http://arxiv.org/abs/2106.03609. arXiv:2106.03609 [cs].

R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Central Science*, 4(2):268–276, Feb. 2018. ISSN 2374-7943, 2374-7951. doi: 10.1021/acscentsci.7b00572. URL https://pubs.acs.org/doi/10.1021/acscentsci.7b00572.

J. Hensman, A. Matthews, and Z. Ghahramani. Scalable Variational Gaussian Process Classification, Nov. 2014. URL http://arxiv.org/abs/1411.2005. arXiv:1411.2005 [stat].

J. M. Hernández-Lobato, M. W. Hoffman, and Z. Ghahramani. Predictive Entropy Search for Efficient Global Optimization of Black-box Functions, June 2014. URL http://arxiv.org/abs/1406.2541. arXiv:1406.2541 [stat].

J. M. Hernández-Lobato, M. A. Gelbart, R. P. Adams, M. W. Hoffman, and Z. Ghahramani. A General Framework for Constrained Bayesian Optimization using Information-based Search, Sept. 2016. URL http://arxiv.org/abs/1511.09422. arXiv:1511.09422 [stat].

C. Hvarfner, E. O. Hellsten, and L. Nardi. Vanilla Bayesian Optimization Performs Great in High Dimensions, Dec. 2024. URL http://arxiv.org/abs/2402.02229. arXiv:2402.02229 [cs].

D. R. Jones. A Taxonomy of Global Optimization Methods Based on Response Surfaces. *Journal of Global Optimization*, 21(4):345–383, Dec. 2001. ISSN 0925-5001, 1573-2916. doi: 10.1023/A:1012771025575. URL https://link.springer.com/10.1023/A:1012771025575.

D. R. Jones, M. Schonlau, and W. J. Welch. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization*, 13(4):455–492, Dec. 1998. ISSN 0925-5001, 1573-2916. doi: 10.1023/A:1008306431147. URL https://link.springer.com/10.1023/A:1008306431147.

K. Kandasamy, J. Schneider, and B. Poczos. High Dimensional Bayesian Optimisation and Bandits via Additive Models, May 2016. URL http://arxiv.org/abs/1503.01673. arXiv:1503.01673 [cs, stat].

M. C. Kennedy and A. O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000. ISSN 00063444, 14643510.

D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization, Jan. 2017. URL http://arxiv.org/abs/1412.6980. arXiv:1412.6980 [cs].

I. Kobyzev, S. J. D. Prince, and M. A. Brubaker. Normalizing Flows: An Introduction and Review of Current Methods. *IEEE Transactions on Pattern Analysis and*

Machine Intelligence, 43(11):3964–3979, Nov. 2021. ISSN 0162-8828, 2160-9292, 1939-3539. doi: 10.1109/TPAMI.2020.2992934. URL http://arxiv.org/abs/1908.09257. arXiv:1908.09257 [stat].

H. J. Kushner. A versatile stochastic model of a function of unknown and time varying form. *Journal of Mathematical Analysis and Applications*, 5(1):150–167, Aug. 1962. ISSN 0022247X. doi: 10.1016/0022-247X(62)90011-2. URL https://linkinghub.elsevier.com/retrieve/pii/0022247X62900112.

H. J. Kushner. A New Method of Locating the Maximum Point of an Arbitrary Multipeak Curve in the Presence of Noise. *Journal of Basic Engineering*, 86(1):97–106, Mar. 1964. ISSN 0021-9223. doi: 10.1115/1.3653121. URL https://asmedigitalcollection.asme.org/fluidsengineering/article/86/1/97/392213/A-New-Method-of-Locating-the-Maximum-Point-of-an.

B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles, Nov. 2017. URL http://arxiv.org/abs/1612.01474. arXiv:1612.01474 [stat].

B. Letham, R. Calandra, A. Rai, and E. Bakshy. Re-Examining Linear Embeddings for High-Dimensional Bayesian Optimization, Oct. 2020. URL http://arxiv.org/abs/2001.11659. arXiv:2001.11659 [cs, stat].

D. J. C. MacKay. Bayesian Methods for Backpropagation Networks. In *Models of Neural Networks III*, pages 211–254. Springer New York, New York, NY, 1996. ISBN 978-1-4612-6882-6 978-1-4612-0723-8. doi: 10.1007/978-1-4612-0723-8_6. URL http://link.springer.com/10.1007/978-1-4612-0723-8_6. Series Title: Physics of Neural Networks.

N. Maus, Z. J. Lin, M. Balandat, and E. Bakshy. Joint Composite Latent Space Bayesian Optimization, July 2024. URL http://arxiv.org/abs/2311.02213. arXiv:2311.02213 [cs].

M. D. McKay, R. J. Beckman, and W. J. Conover. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics*, 21(2):239, May 1979. ISSN 00401706. doi: 10.2307/1268522. URL https://www.jstor.org/stable/1268522?origin=crossref.

R. Moriconi, M. P. Deisenroth, and K. S. S. Kumar. High-dimensional Bayesian optimization using low-dimensional feature spaces, Sept. 2020. URL http://arxiv.org/abs/1902.10675. arXiv:1902.10675 [stat].

S. Müller, M. Feurer, N. Hollmann, and F. Hutter. PFNs4BO: In-Context Learning for Bayesian Optimization, July 2023. URL http://arxiv.org/abs/2305.17535. arXiv:2305.17535 [cs].

S. Müller, N. Hollmann, S. P. Arango, J. Grabocka, and F. Hutter. Transformers Can Do Bayesian Inference, Aug. 2024. URL http://arxiv.org/abs/2112.10510. arXiv:2112.10510 [cs].

A. Nayebi, A. Munteanu, and M. Poloczek. A framework for Bayesian optimization in embedded subspaces. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4752–4761. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/nayebi19a.html.

R. M. Neal. *Bayesian Learning for Neural Networks*, volume 118 of *Lecture Notes in Statistics*. Springer New York, New York, NY, 1996. ISBN 978-0-387-94724-2 978-1-4612-0745-0. doi: 10.1007/978-1-4612-0745-0. URL http://link.springer.com/10.1007/978-1-4612-0745-0.

K. Om, K. Sim, T. Yun, H. Kang, and J. Park. Posterior Inference in Latent Space for Scalable Constrained Black-box Optimization, July 2025. URL http://arxiv.org/abs/2507.00480. arXiv:2507.00480 [cs].

L. Papenmeier. *Bayesian Optimization in High Dimensions: A Journey Through Subspaces and Challenges*. Computer Science, Lund University, Lund, 2025. ISBN 9789181045482. Medium: Elektronisk resurs.

L. Papenmeier, L. Nardi, and M. Poloczek. Increasing the Scope as You Learn: Adaptive Bayesian Optimization in Nested Subspaces, Apr. 2023. URL http://arxiv.org/abs/2304.11468. arXiv:2304.11468 [cs].

L. Papenmeier, M. Poloczek, and L. Nardi. Understanding High-Dimensional Bayesian Optimization, June 2025. URL http://arxiv.org/abs/2502.09198. arXiv:2502.09198 [cs].

V. Perrone, I. Shcherbatyi, R. Jenatton, C. Archambeau, and M. Seeger. Constrained Bayesian Optimization with Max-Value Entropy Search, Oct. 2019. URL http://arxiv.org/abs/1910.07003. arXiv:1910.07003 [stat].

M. Poloczek, J. Wang, and P. I. Frazier. Multi-Information Source Optimization, Nov. 2016. URL http://arxiv.org/abs/1603.00389. arXiv:1603.00389 [stat].

M. J. D. Powell. A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation. In S. Gomez and J.-P. Hennart, editors, *Advances in Optimization and Numerical Analysis*, pages 51–67. Springer Netherlands, Dordrecht, 1994. ISBN 978-90-481-4358-0 978-94-015-8330-5. doi: 10.1007/978-94-015-8330-5_4. URL http://link.springer.com/10.1007/978-94-015-8330-5_4.

J. Quiñonero-Candela and C. E. Rasmussen. A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 6(65):1939–1959, 2005. URL http://jmlr.org/papers/v6/quinonero-candela05a.html.

E. Raponi, H. Wang, M. Bujny, S. Boria, and C. Doerr. High Dimensional Bayesian Optimization Assisted by Principal Component Analysis, July 2020. URL http://arxiv.org/abs/2007.00925. arXiv:2007.00925 [cs].

B. Rashidi, K. Johnstonbaugh, and C. Gao. Cylindrical Thompson sampling for high-dimensional Bayesian optimization. In S. Dasgupta, S. Mandt, and Y. Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 3502–3510. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/rashidi24a.html.

C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass, 2006. ISBN 978-0-262-18253-9. OCLC: ocm61285753.

M. J. Sasena, P. Papalambros, and P. Goovaerts. Exploration of Metamodeling Sampling Criteria for Constrained Global Optimization. *Engineering Optimization*, 34 (3):263–278, Jan. 2002. ISSN 0305-215X, 1029-0273. doi: 10.1080/03052150211751. URL http://www.tandfonline.com/doi/abs/10.1080/03052150211751.

P. Saves, R. Lafage, N. Bartoli, Y. Diouane, J. Bussemaker, T. Lefebvre, J. T. Hwang, J. Morlier, and J. R. R. A. Martins. SMT 2.0: A surrogate modeling toolbox with a focus on hierarchical and mixed variables gaussian processes. *Advances in Engineering Sofware*, 188:103571, 2024. doi: https://doi.org/10.1016/j.advengsoft.2023.103571.

I. J. Schoenberg. Metric spaces and positive definite functions. *Transactions of the American Mathematical Society*, 44(3):522–536, 1938. ISSN 0002-9947, 1088-6850. doi: 10.1090/S0002-9947-1938-1501980-0. URL https://www.ams.org/tran/1938-044-03/S0002-9947-1938-1501980-0/.

B. W. Silverman. Some Aspects of the Spline Smoothing Approach to Non-Parametric Regression Curve Fitting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 47(1):1–21, Sept. 1985. ISSN 1369-7412, 1467-9868. doi: 10.1111/j.2517-6161.1985.tb01327.x. URL https://academic.oup.com/jrsssb/article/47/1/1/7028165.

A. J. Smola and P. Bartlett. Sparse greedy gaussian process regression. In *Proceedings of the 14th International Conference on Neural Information Processing Systems*, NIPS'00, page 598–604, Cambridge, MA, USA, 2000. MIT Press.

J. Snoek, O. Rippel, K. Swersky, R. Kiros, N. Satish, N. Sundaram, M. M. A. Patwary, Prabhat, and R. P. Adams. Scalable Bayesian Optimization Using Deep Neural Networks, July 2015. URL http://arxiv.org/abs/1502.05700. arXiv:1502.05700 [stat].

I. Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4): 86–112, Jan. 1967. ISSN 00415553. doi: 10.1016/0041-5553(67)90144-9. URL https://linkinghub.elsevier.com/retrieve/pii/0041555367901449.

N. Srinivas, A. Krause, S. Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: no regret and experimental design. ICML'10, page 1015–1022, Madison, WI, USA, 2010. Omnipress. ISBN 9781605589077.

S. Takeno, T. Tamura, K. Shitara, and M. Karasuyama. Sequential and parallel constrained max-value entropy search via information lower bound. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 20960–20986. PMLR, 17–23 Jul 2022.

W. R. Thompson. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika*, 25(3/4):285, Dec. 1933. ISSN 00063444. doi: 10.2307/2332286. URL https://www.jstor.org/stable/2332286?origin=crossref.

A. Tripp, E. Daxberger, and J. M. Hernández-Lobato. Sample-Efficient Optimization in the Latent Space of Deep Generative Models via Weighted Retraining, Oct. 2020. URL http://arxiv.org/abs/2006.09191. arXiv:2006.09191 [cs].

D. Ulmasov, C. Baroukh, B. Chachuat, M. P. Deisenroth, and R. Misener. Bayesian Optimization with Dimension Scheduling: Application to Biological Systems, 2015. URL https://arxiv.org/abs/1511.05385. Version Number: 1.

R. von Mises. *Mathematical Theory of Probability and Statistics*. Elsevier, 1964. ISBN 978-1-4832-3213-3. doi: 10.1016/C2013-0-12460-9. URL https://linkinghub.elsevier.com/retrieve/pii/C20130124609.

Z. Wang and S. Jegelka. Max-value Entropy Search for Efficient Bayesian Optimization, Jan. 2018. URL http://arxiv.org/abs/1703.01968. arXiv:1703.01968 [stat].

Z. Wang, F. Hutter, M. Zoghi, D. Matheson, and N. de Freitas. Bayesian Optimization in a Billion Dimensions via Random Embeddings, Jan. 2016. URL http://arxiv.org/abs/1301.1942. arXiv:1301.1942 [cs, stat].

A. G. Wilson and H. Nickisch. Kernel Interpolation for Scalable Structured Gaussian Processes (KISS-GP), Mar. 2015. URL http://arxiv.org/abs/1503.01057. arXiv:1503.01057 [cs].

Z. Xu, H. Wang, J. M. Phillips, and S. Zhe. Standard Gaussian Process is All You Need for High-Dimensional Bayesian Optimization, Mar. 2025. URL http://arxiv.org/abs/2402.02746. arXiv:2402.02746 [cs].

Z. Zhang. PRIMA: Reference Implementation for Powell's Methods with Modernization and Amelioration. available at http://www.libprima.net, DOI: 10.5281/zenodo.8052654, 2023.

J. Ziomek and H. Bou-Ammar. Are Random Decompositions all we need in High Dimensional Bayesian Optimisation?, Jan. 2023. URL http://arxiv.org/abs/2301.12844. arXiv:2301.12844 [cs, stat].

3

Why Unconstrained Strategies May Fail in Constrained Bayesian Optimisation

Recent advances in unconstrained high-dimensional BO have demonstrated the effectiveness of random subspace embeddings and TRs (Papenmeier et al., 2023) and DSP for GP kernels (Hvarfner et al., 2024). These methods significantly enhance scalability and optimisation efficiency. However, their applicability to constrained problems, where feasibility must be actively discovered and preserved, remains unclear.

3.1. Introduction

This chapter investigates whether these unconstrained techniques can be successfully extended to constrained, high-dimensional BO. Two complementary strategies are explored. Firstly, input space embeddings that reduce the search dimensionality, and secondly kernel hyperparameter priors adapted to the ambient dimension via DSP. The analysis begins with a critical evaluation of random embeddings under constraints, followed by a constraint-aware supervised alternative. The second part examines dimensionality-scaled priors and their impact on surrogate modelling in constrained problems. All methods are benchmarked against state-of-the-art CBO approaches across a range of test cases.

3.2. Random Embeddings in Constrained Scenarios

It has been shown that, for unconstrained problems, the use of random subspaces can be advantageous in high-dimensional optimisation tasks, see Chapter 2.5.4. In particular, the work of Papenmeier et al. (2023) demonstrated promising results by introducing a hybrid trust-region approach that employs a continuously expanding linear random embedding. However, such strategies are not directly applicable to constrained optimisation problems. In the constrained setting, the subspace must not only contain an optimal solution but also preserve feasibility. In other words, the subspace must include at least one point that satisfies all constraints. Methods that rely on purely random subspaces, such as REMBO, ALEBO, and HeSBO, may therefore suffer from convergence issues, as they often fail to identify any feasible point. This limitation arises because the probability that a randomly chosen subspace simultaneously contains both feasible and optimal regions is already low in moderately-sized problems, and it diminishes further as the dimensionality and the number of constraints increases (Priem, 2020). This phenomenon is illustrated in Figure 3.1, which highlights the challenge of random subspace selection in constrained optimisation.

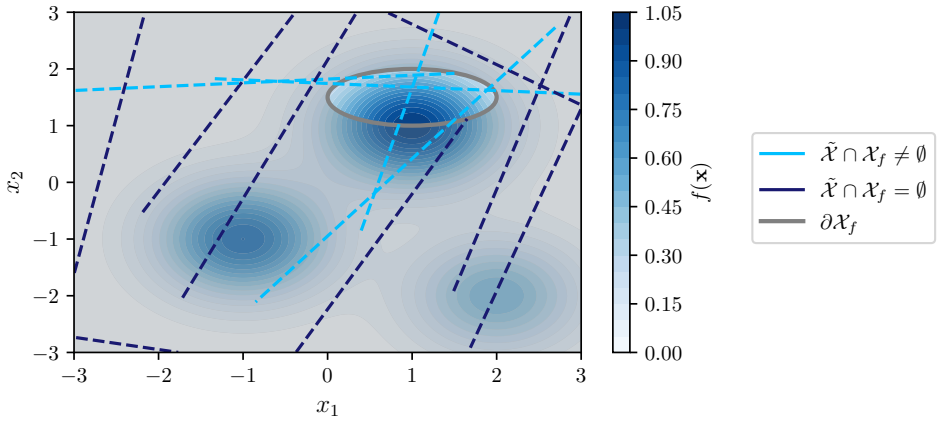


Figure 3.1: Illustration of random subspace projections in a constrained optimisation problem. $\mathcal{X}_f$ is the feasible domain, bounded by the contour $\partial\mathcal{X}_f$. Thin coloured, dashed lines represent random one-dimensional subspaces $\tilde{\mathcal{X}} \subset \mathcal{X}$. Lines that intersect the feasible region belong to $\tilde{\mathcal{X}} \cap \mathcal{X}_f \neq \emptyset$, while those missing it entirely belong to $\tilde{\mathcal{X}} \cap \mathcal{X}_f = \emptyset$. This highlights a potential failure mode of random embeddings in constrained settings, where the feasible region is small or narrow relative to the ambient space.

This probability depends on the hypervolume of the feasible region in contrast to the full search space, as well as the shape of the lower-dimensional manifold and thus the manner how the subspace is constructed.

3.3. Constrained Bayesian Optimisation via Supervised Embeddings

Rather than relying on randomly constructed subspaces, which may fail to contain feasible or optimal regions, supervised embeddings offer a principled alternative by incorporating output information into the subspace construction. This idea is motivated by the work of Amine Bouhlel et al. (2018), who employed PLS to extract an output-aware subspace within GP models. In their setting, however, the embedding was used solely to reduce the number of hyperparameters during model training, while acquisition optimisation was still conducted in the full input space $\mathcal{X}$. Moreover, distinct subspaces were constructed per surrogate model.

In contrast, subspace-based optimisation methods, see Chapter 2.5.4, perform the search directly within a lower-dimensional embedding. Papenmeier et al. (2023) further demonstrated that combining such embeddings with TR strategies can significantly enhance scalability. To address this, we propose replacing the random embedding matrix in BAxUS with a supervised embedding derived from a weighted PCA, inspired by Raponi et al. (2020). The key idea is to extract input directions most informative for identifying feasible and optimal regions. Each point $\{\mathbf{x}_i, f_i, \mathbf{c}_i\} \in \mathcal{D}_t \ \forall i = \{1, ..., N_t\}$ is assigned a scalar weight w_i that depends jointly on its objective value and total constraint values. Points that are both feasible and near-optimal receive large weights and thus exert greater influence on the embedding. Conversely, clearly infeasible points receive near-zero weights, effectively excluding them from the subspace construction. As a result, the principal components are "pulled" toward promising regions of the input space, while uninformative directions are naturally suppressed.

This supervised subspace yields a low-dimensional representation that captures the joint structure of both objective and constraint landscapes, offering a more targeted and robust basis for CBO in high dimensions.

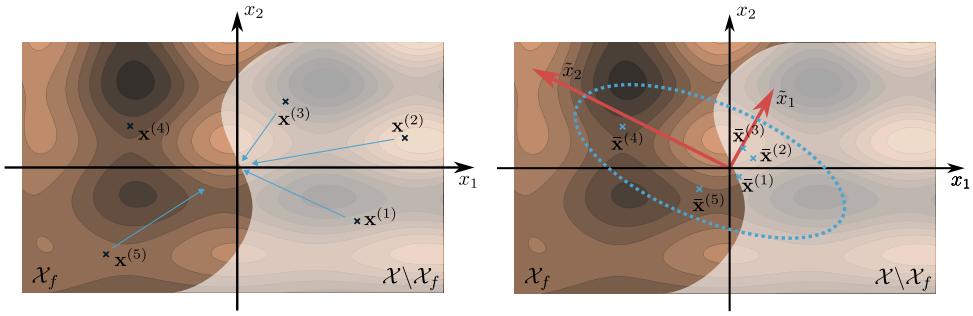


Figure 3.2: The idea of the weighted PCA. The points $\mathbf{x}^{(i)}$ are scaled to the centre according to the corresponding objective and constraint values. PCA is then applied on the weighted points to obtain the projection $\mathcal{P}$.

Weighting Scheme To favour feasible and optimal solutions, we define a constraint-aware weighting scheme that ranks all samples based on feasibility and objective value. Feasible points are given higher preference than infeasible ones, and within each group, samples are ranked by objective value or constraint violation, respectively. Let $\mathcal{D}_t = \{(\mathbf{x}_i, f_i, \mathbf{c}_i)\}_{i=1}^{N}$ denote a dataset of input–output pairs. We define the normalised constraint violations for each constraint dimension:

$$\bar{\mathbf{c}}_i = \max\left(0, \frac{\mathbf{c}_i}{\max|\mathbf{c}_i|}\right), \tag{3.1}$$

so that all constraint violations are scaled to $[0, 1]$, and satisfied constraints yield zero. The total violation score is $v_i = \sum_{j=1}^{g} \bar{c}_{ij}$. We then define the feasible set $\mathcal{F} := \{i \mid v_i = 0\}$ and its complement $\mathcal{I} := \{1, \ldots, n\} \setminus \mathcal{F}$ denoting all infeasible samples. Samples in $\mathcal{F}$ are ranked by objective value, while those in $\mathcal{I}$ are ranked by total violation:

$$r_i = \text{rank}(f_i), \qquad \text{if } i \in \mathcal{F}, \tag{3.2}$$

$$r_i = \text{rank}(v_i) + |\mathcal{F}|, \qquad \text{if } i \in \mathcal{I}, \tag{3.3}$$

where ranks are assigned from 1 to N, and infeasible ranks are offset by $|\mathcal{F}|$ to ensure all feasible samples are ranked higher. To compute weights, we apply a logarithmic decay over the ranks:

$$w_i = \log N - \log r_i, \tag{3.4}$$

which smoothly reduces the influence of lower-ranked points, as depicted in Figure 3.3. Finally, weights are normalised:

$$w_i \leftarrow \frac{w_i}{\sum_{j=1}^{N} w_j}, \tag{3.5}$$

and optionally assembled into a diagonal weighting matrix $\mathbf{W} = \text{diag}(\mathbf{w}) \in \mathbb{R}^{N \times N}$, suitable for use in weighted loss functions or regularisation schemes.

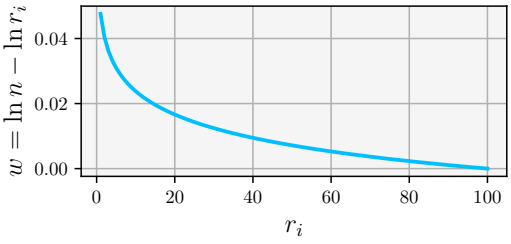


Figure 3.3: Weighting function with $\sum_i w_i = 1$.

Weighted Principal Component Analysis Further, let $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_N] \in \mathbb{R}^{N \times D}$ denote a set of N input design points in a D-dimensional space, with corresponding objective values $f_i \in \mathbb{R}$ and constraint values $\mathbf{c}_i \in \mathbb{R}^G$. The design domain is assumed to be $\mathcal{X} = [0, 1]^D$. The aim of Weighted PCA is to construct a linear embedding that emphasises regions of high objective performance and low constraint violation. To this end, the samples in $\mathbf{X} \in \mathcal{D}_t$ are first centred with $\bar{\mathbf{X}} = \mathbf{X} - \boldsymbol{\mu}$ by subtracting the mean $\boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{X}_i$, such that they lie within $[-1, 1]^D$. Subsequently, using the weighting matrix $\mathbf{W} = \text{diag}(w_i) \in \mathbb{R}^{N \times N}$ with w_i from Equation (3.4), we multiply every sample $\mathbf{x}_i$ with its corresponding weight w_i:

$$\mathbf{X}' = \mathbf{W}\bar{\mathbf{X}} = \left[w_1\bar{\mathbf{x}}_1, w_2\bar{\mathbf{x}}_2, ..., w_n\bar{\mathbf{x}}_n\right], \tag{3.6}$$

where $\bar{\mathbf{X}} = [\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2, ..., \bar{\mathbf{x}}_n]$. The weighted samples are centred again $\bar{\mathbf{X}}' = \mathbf{X}' - \boldsymbol{\mu}'$ with $\boldsymbol{\mu}' = \frac{1}{n}\sum_{i=1}^{n} \mathbf{X}'_i$. As in PCA, the covariance matrix is computed via $\boldsymbol{\Sigma} = \frac{1}{N}(\bar{\mathbf{X}}')^\top \bar{\mathbf{X}}' \in \mathbb{R}^{D \times D}$ and its eigendecomposition is computed via $\boldsymbol{\Sigma}\mathbf{A} = \mathbf{A}\boldsymbol{\Lambda}$ where $\mathbf{A} \in \mathbb{R}^{D \times D}$ is the matrix of eigenvectors and $\boldsymbol{\Lambda} \in \mathbb{R}^{D \times D}$ the diagonal matrix of eigenvalues. The matrix $\mathbf{A}$ is then truncated based on the d most important eigenvalues, obtaining $\mathbf{A}_t$. This matrix can then be used to construct a mapping $\mathcal{P} : \mathcal{X} \subset \mathbb{R}^D \to \tilde{\mathcal{X}} \subset \mathbb{R}^d$ with $d \ll D$:

$$\mathcal{P}_t(\mathbf{x}) = (\mathbf{x} - \boldsymbol{\mu} - \boldsymbol{\mu}')\mathbf{A}_t. \tag{3.7}$$

Note, that we train the GPs in every iteration from scratch and so we recompute the projection $\mathcal{P}_t$.

Trust Region Heuristic In terms of the TR heuristic, we adopt a hybrid strategy inspired by BAxUS (Papenmeier et al., 2023), using CTS in the subspace $\tilde{\mathcal{X}}$. The algorithm maintains a TR of side length r around the current centre in the latent subspace $\tilde{\mathcal{X}}$, and adapts its size based on the observed success or failure of candidate evaluations. A success is registered if a newly evaluated point improves over the current best feasible value (or reduces total constraint violation if no feasible point has been found). Otherwise, a failure is recorded. Once a predefined threshold of consecutive successes (τ_s) or failures (τ_f) is reached, the trust region is updated as follows:

$$r \leftarrow \begin{cases} \min(2r, r_{\max}) & \text{after } \tau_s \text{ successes}, \\ r/2 & \text{after } \tau_f \text{ failures}. \end{cases} \tag{3.8}$$

When the TR becomes too small, i.e., $r < r_{\min}$, a restart is triggered and the search subspace is expanded by increasing the embedding dimensionality. The subspace is always initialised with $d_{\text{init}} > 2$, and follow the same dimensional expansion schedule as in BAxUS. The full procedure is outlined in Algorithm 3. In the following we will refer to this method as Bayesian Optimisation Over Supervised Trust Region Embeddings (BOOSTRE).

Algorithm 3 BOOSTRE

Input: Input space $\mathcal{X}$, Number of candidates N_c, batch size q_c, number of initial samples N_i, TR hyperparameters,
$t = 0$
$\mathcal{D}_t = \{\mathbf{x}_i, f(\mathbf{x}_i), \mathbf{c}(\mathbf{x}_i)\}_{i=1:N_i}$
while Computational budget is not exhausted **do**
  $\mathbf{W}_t \leftarrow$ getRankMatrix$(\mathcal{D}_t)$
  $\mathcal{P}_t \leftarrow$ computeProjection$(\mathcal{D}_t, \mathbf{W}_t)$
  Project $\mathbf{X}$ onto lower subspace $\tilde{\mathbf{X}} = \mathcal{P}_t(\mathbf{X})$
  Fit models $\hat{f}(\tilde{\mathbf{x}})$ and $\hat{c}_i(\tilde{\mathbf{x}})$ $\forall i \in \{1, ..., G\}$
  Project candidate points $\mathbf{X}_c \in \mathbb{R}^{N_c \times D}$ to subspace $\tilde{\mathbf{X}}_c = \mathcal{P}_t(\mathbf{X}_c)$
  Solve acquisition function $\mathbf{x}_+ = \mathbf{X}_c[\text{argmax}_{\mathbf{x} \in \tilde{\mathbf{X}}_c} \alpha_{CTS}]$ via index mapping
  Evaluate $\mathbf{x}_+$ and observe $f(\mathbf{x}_+)$, $\mathbf{c}(\mathbf{x}_+)$
  Update TR state
  $\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{\mathbf{x}_+, f(\mathbf{x}_+), \mathbf{c}(\mathbf{x}_+)\}$
  $t \leftarrow t + 1$
end while

3.4. Vanilla Bayesian Optimisation in Constrained Scenarios

Another approach that has not yet been explored in scaling CBO is the use of DSPs, which have proven effective in the unconstrained case, see Chapter 2.5.6. Scaling the prior mean or initial values of the GP kernel length scales with the input dimensionality D has been shown to significantly improve performance, and in some high-dimensional benchmark problems, even surpass state-of-the-art methods. This approach is referred to here as VBO, and combines: (i) the dimensionality-scaled log-normal prior on the length scales, as introduced by Hvarfner et al. (2024), and (ii) the log-Transformed CEI acquisition function. To assess whether this setup can also perform competitively in the constrained setting, we evaluate VBO in combination with RAASP sampling, as demonstrated in Papenmeier et al. (2025) and Rashidi et al. (2024). While the authors in Ament et al. (2023) report promising results in low-dimensional constrained scenarios up to $D = 7$ using Log Constrained Expected Improvement (LogCEI), it remains an open question whether this prior-based approach scales well to higher dimensions. This experiment thus provides insights into the transferability of recent advances in unconstrained BO to the constrained setting, highlighting whether a carefully designed GP prior alone can guide optimisation effectively under feasibility constraints.

3.5. Numerical Experiments

In what follows, the previously discussed approaches are benchmarked against each other, to obtain an overview about the current state-of-the-art. Therefore, a wide

range of constrained benchmark problems is used, ranging from low- ($D = 4$) to high-dimensional ($D = 100$) problems with up to 11 constraints.

Baseline Methods We compare the following methods:

- SCBO, using the implementation from BoTorch Balandat et al. (2020), and adopting the hyperparameters from (Eriksson and Poloczek, 2021).
- FuRBO and adpoting the hyperparameters from Ascia et al. (2025).[1]
- VBO w/ DSP from Hvarfner et al. (2024) with LogCEI from Ament et al. (2023) while using `BoTorch`'s RAASP implementation, using 64 Monte Carlo (MC) samples, 256 initial samples and 3 restarts for optimising the acquisition function α_{logCEI} in `BoTorch`.
- VBO w/o DSP baseline with logCEI but without a dimensionality-scaled length-scale prior to investigate its influence, using 64 MC samples, 256 initial samples, 3 restarts for optimising the acquisition function α_{logCEI} in `BoTorch`.
- CMES (Takeno et al., 2022), using our own implementation with $K = 32$ MC samples to sample f^*, 256 initial samples and 3 restarts for optimising the acquisition function α_{CMES} in `BoTorch`.[2]
- BAxUS as a constrained version (Constrained Bayesian Optimisation in Adaptively Expanding Subspaces (cBAxUS)) where we use CTS and construct the objective and each constraint in a shared random subspace, defined by the BAxUS heuristic. We use the same hyperparameters as in Papenmeier et al. (2023).
- BOOSTRE using the BAxUS implementation[3] and replacing the embedding heuristic with the weighted PCA as explained in 3.3. Additionally, we ensure that the initial dimensionality $d_{init} > 2$.[4]

We do not compare against PESC and Soft Penalties because they have been outperformed by SCBO in (Eriksson and Poloczek, 2021).

Benchmark problems To compare the performance of the aforementioned baseline methods, this work employs six physics-based and synthetic benchmarks. The following list briefly introduces these problems and defines the number of initial samples N_0 and batch size q which are used throughout all baseline methods:

- *Pressure Vessel* ($D = 4$, $G = 4$): Minimise the total cost of a pressure vessel, including shell and head thickness, radius, and length. Originally proposed by Coello Coello and Mezura Montes (2002). We use $N_0 = 10$ and $q = 1$.

[1] We use the implementation from `https://anonymous.4open.science/r/FuRBO`.
[2] The code is available at `https://github.com/haukemaa/cmesp/`.
[3] `https://botorch.org/docs/tutorials/baxus/`
[4] The code can be found in `https://github.com/haukemaa/boostre/`.

- *Speed Reducer* ($D = 7$, $G = 11$): Optimise the weight of a mechanical speed reducer, with variables including shaft lengths, gear dimensions, and tooth module. From Lemonge et al. (2010), using $N_0 = 10$ and $q = 1$.
- *Ackley (constrained)* ($D = 10$, $G = 2$): A constrained variant of the classical multimodal Ackley function as defined in Eriksson and Poloczek (2021), with $N_0 = 20$ and $q = 5$.
- *Different Powers* and *Rastrigin* ($D = 40$, $G = 9$): From the BBOB-constrained suite (Dufossé et al., 2022), combining transformed base functions with complex constraints. We use $N_0 = 100$ and $q = 5$ for both.
- *Rosenbrock (constrained)* ($D = 100$, $G = 2$): A high-dimensional Rosenbrock objective (Rosenbrock, 1960) with two nonlinear constraints from Eriksson and Poloczek (2021), testing performance in narrow feasible regions. Here, we use $N_0 = 100$, $q = 5$.

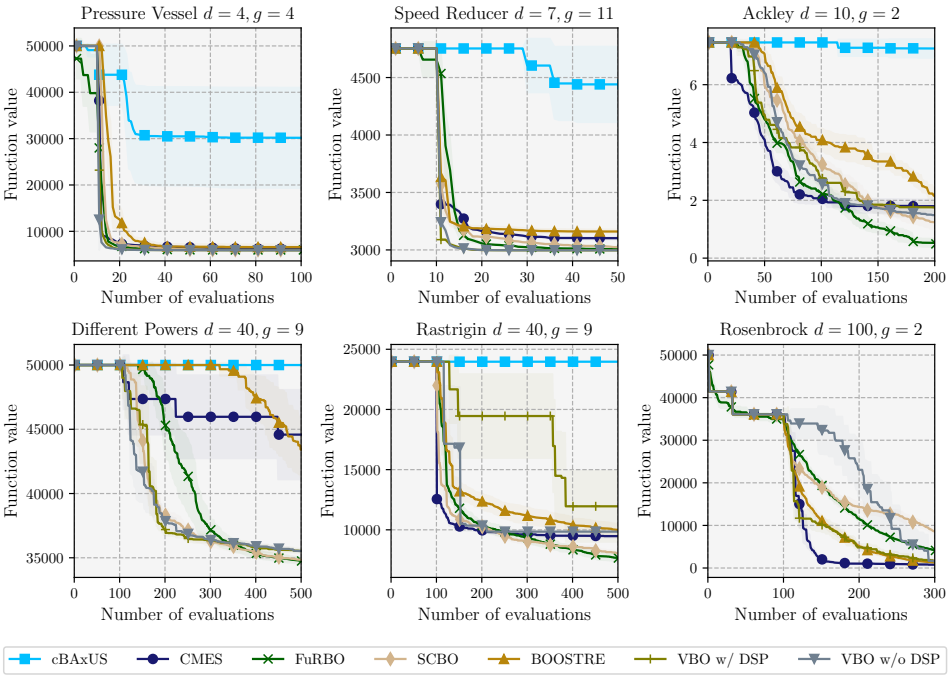


Figure 3.4: Performance comparison of CBO methods across six benchmark problems. Each subplot shows the best (i.e., lowest) objective value found over the course of the optimisation, plotted against the number of function evaluations, while infeasible samples are set to the highest feasible value found so far over 10 seeds. Shaded regions indicate standard deviations across repeated runs. Lower curves indicate faster and more effective optimisation performance.

Figure 3.4 presents a comparative performance analysis of the aforementioned baseline methods across a range of benchmark problems, highlighting their relative performance in terms of best-found objective value over time.

3.6. Discussion

As expected, cBAxUS, which relies on random linear embeddings, consistently underperforms across all benchmarks. Its inability to align the subspace with feasible regions leads to frequent failures in identifying feasible solutions. This reinforces the known limitation of random subspaces in constrained settings, where feasibility is often confined to narrow regions poorly represented by random projections.

In contrast, the supervised extension of cBAxUS proposed in this work, denoted here as BOOSTRE, demonstrates a clear performance improvement. By learning a constraint-aware subspace through a supervised weighted PCA method, it significantly outperforms cBAxUS and often matches or exceeds the performance of established CBO methods. In particular, it shows strong results on the *Pressure Vessel*, *Rastrigin*, and *Rosenbrock* problems, even though it is generally outperformed by methods that model the objective and constraints directly in the full-dimensional space. Especially in *Ackley* and *Different Powers*, the performance of BOOSTRE is more mixed. These problems likely feature objectives and constraints that lie in different or only partially overlapping subspaces, undermining the effectiveness of a single shared embedding. When such alignment is poor, the optimisation may be biased toward infeasible or non-optimal regions, especially if the subspace fails to preserve important constraint information.

Both variants of VBO, with and without DSP, show competitive results across most problems. Interestingly, their relative performance differs between benchmarks: on *Rastrigin*, the variant without DSP achieves better results, while on *Rosenbrock*, the version with DSP clearly outperforms its counterpart. This behaviour can likely be attributed to the underlying characteristics of the functions being modelled. In problems where the objective or constraints vary strongly at small scales, e.g. *Rastrigin*, overly large length scales induced by DSP may lead to over-smoothing, missing critical local detail. Conversely, for smoother problems such as *Rosenbrock*, DSP appears to regularise the model and promote more stable convergence. This highlights the importance of adapting prior assumptions to the nature of the function landscape, a non-trivial challenge in black-box optimisation.

CMES shows consistently strong performance, particularly in the early stages of optimisation on *Ackley*, *Rastrigin*, and *Rosenbrock*, where it quickly identifies good feasible solutions. However, on several problems, a performance plateau is observed in later stages, suggesting that CMES may be prone to premature convergence or local exploration bias. On *Different Powers*, CMES struggles to find competitive solutions.

The TR-based methods, SCBO and FuRBO, also perform robustly across most benchmarks. FuRBO, in particular, performs well on *Ackley*, *Different Powers*, and *Rastrigin*, while showing competitive performance on *Pressure Vessel* and *Speed Reducer*. These results underscore the strength of local TR mechanisms in navigating complex feasible landscapes, especially when combined with constraint-aware modelling.

Reflecting on this, the approach of constructing surrogate models for objective and constraints independently but within a shared subspace introduces a clear trade-off: it improves modelling efficiency and reduces dimensionality, but may compromise accuracy when objective and constraint structure diverge. Since the supervised PCA used here is a linear method, it effectively applies a global rotation and rescaling of the design space. While this is computationally efficient and captures dominant variation directions, it lacks the flexibility to model nonlinear, local, or disjoint feasible regions, which are common in real-world engineering problems where constraints originate from distinct physical models or design disciplines.

3.7. Conclusion

This chapter has investigated the transferability of recent advances in high-dimensional BO to constrained scenarios, addressing RQ 1. It introduced a supervised subspace approach called BOOSTRE, that significantly improves feasibility attainment over random embeddings, and evaluated its performance against state-of-the-art constrained methods. Additionally, the adaptation of dimensionality-scaled priors from the unconstrained setting to constrained BO was shown to offer competitive results in specific scenarios, highlighting the potential of prior-based inductive biases.

While the presented methods provide strong baselines for high-dimensional CBO, several open challenges remain. In particular, the assumption of a shared subspace for objective and constraints may limit performance when their underlying structures diverge. This motivates future work on flexible, non-linear, or function-specific embeddings. Moreover, the sensitivity of methods such as VBO to prior mis-specification indicates a need for adaptive or meta-learned priors in constrained black-box settings.

While these methods provide a strong foundation, important challenges remain, particularly in the context of high-dimensional, simulation-based engineering design problems with large numbers of constraints and expensive evaluations or how multiple sources of information can be leveraged. The following three chapters address these remaining challenges, focusing on RQ 2 and RQ 3, introduced earlier.

Bibliography

S. Ament, S. Daulton, D. Eriksson, M. Balandat, and E. Bakshy. Unexpected Improvements to Expected Improvement for Bayesian Optimization, Jan. 2023. URL http://arxiv.org/abs/2310.20708. arXiv:2310.20708 [cs].

M. Amine Bouhlel, N. Bartoli, R. G. Regis, A. Otsmane, and J. Morlier. Efficient global optimization for high-dimensional constrained problems by using the Kriging models combined with the partial least squares method. *Engineering Optimization*, 50(12):2038–2053, Dec. 2018. ISSN 0305-215X, 1029-0273. doi: 10.1080/0305215X.2017.1419344. URL https://www.tandfonline.com/doi/full/10.1080/0305215X.2017.1419344.

P. Ascia, E. Raponi, T. Bäck, and F. Duddeck. Feasibility-Driven Trust Region Bayesian Optimization, June 2025. URL http://arxiv.org/abs/2506.14619. arXiv:2506.14619 [cs].

M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*, 2020. URL http://arxiv.org/abs/1910.06403.

C. A. Coello Coello and E. Mezura Montes. Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. *Advanced Engineering Informatics*, 16(3):193–203, July 2002. ISSN 14740346. doi: 10.1016/S1474-0346(02)00011-3. URL https://linkinghub.elsevier.com/retrieve/pii/S1474034602000113.

P. Dufossé, N. Hansen, D. Brockhoff, P. R. Sampaio, A. Atamna, and A. Auger. Building scalable test problems for benchmarking constrained optimizers. Technical report, Technical Report. http://numbbo. github. io/coco-doc/bbob-constrained/To be . . . , 2022.

D. Eriksson and M. Poloczek. Scalable Constrained Bayesian Optimization, Feb. 2021. URL http://arxiv.org/abs/2002.08526. arXiv:2002.08526 [cs, stat].

J. R. Gardner, M. J. Kusner, Z. Xu, K. Q. Weinberger, and J. P. Cunningham. Bayesian optimization with inequality constraints. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, page II–937–II–945. JMLR.org, 2014.

M. A. Gelbart, J. Snoek, and R. P. Adams. Bayesian optimization with unknown constraints. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, UAI'14, page 250–259, Arlington, Virginia, USA, 2014. AUAI Press. ISBN 9780974903910.

J. M. Hernández-Lobato, M. A. Gelbart, R. P. Adams, M. W. Hoffman, and Z. Ghahramani. A General Framework for Constrained Bayesian Optimization

using Information-based Search, Sept. 2016. URL http://arxiv.org/abs/1511.09422. arXiv:1511.09422 [stat].

C. Hvarfner, E. O. Hellsten, and L. Nardi. Vanilla Bayesian Optimization Performs Great in High Dimensions, Dec. 2024. URL http://arxiv.org/abs/2402.02229. arXiv:2402.02229 [cs].

A. C. C. Lemonge, H. J. C. Barbosa, C. C. H. Borges, and F. B. dos Santos Silva. Constrained optimization problems in mechanical engineering design using a real-coded steady-state genetic algorithm. 2010. URL https://api.semanticscholar.org/CorpusID:54994542.

L. Papenmeier, L. Nardi, and M. Poloczek. Increasing the Scope as You Learn: Adaptive Bayesian Optimization in Nested Subspaces, Apr. 2023. URL http://arxiv.org/abs/2304.11468. arXiv:2304.11468 [cs].

L. Papenmeier, M. Poloczek, and L. Nardi. Understanding High-Dimensional Bayesian Optimization, June 2025. URL http://arxiv.org/abs/2502.09198. arXiv:2502.09198 [cs].

V. Perrone, I. Shcherbatyi, R. Jenatton, C. Archambeau, and M. Seeger. Constrained Bayesian Optimization with Max-Value Entropy Search, Oct. 2019. URL http://arxiv.org/abs/1910.07003. arXiv:1910.07003 [stat].

R. Priem. *Optimisation bayésienne sous contraintes et en grande dimension appliquée à la conception avion avant projet*. PhD thesis, 2020.

E. Raponi, H. Wang, M. Bujny, S. Boria, and C. Doerr. High Dimensional Bayesian Optimization Assisted by Principal Component Analysis, July 2020. URL http://arxiv.org/abs/2007.00925. arXiv:2007.00925 [cs].

B. Rashidi, K. Johnstonbaugh, and C. Gao. Cylindrical Thompson sampling for high-dimensional Bayesian optimization. In S. Dasgupta, S. Mandt, and Y. Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 3502–3510. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/rashidi24a.html.

H. H. Rosenbrock. An Automatic Method for Finding the Greatest or Least Value of a Function. *The Computer Journal*, 3(3):175–184, Mar. 1960. ISSN 0010-4620, 1460-2067. doi: 10.1093/comjnl/3.3.175.

S. Takeno, T. Tamura, K. Shitara, and M. Karasuyama. Sequential and parallel constrained max-value entropy search via information lower bound. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 20960–20986. PMLR, 17–23 Jul 2022.

Z. Xu, H. Wang, J. M. Phillips, and S. Zhe. Standard Gaussian Process is All You Need for High-Dimensional Bayesian Optimization, Mar. 2025. URL http://arxiv.org/abs/2402.02746. arXiv:2402.02746 [cs].

4

Scaling Bayesian Optimisation for High-Dimensional and Large-Scale Constrained Spaces

This chapter is based on the following publication and has been reproduced with minor adjustments to notation and formatting for consistency within the thesis.
Maathuis, H., & De Breuker, R. & Castro, S.G.P. (2025); Scaling Bayesian Optimisation for High-Dimensional and Large-Scale Constrained Spaces. AIAA Journal, DOI: 10.2514/1.J065252.

Abstract Design optimisation offers the potential to develop lightweight aircraft structures with reduced environmental impact. Due to the high number of design variables and constraints, these challenges are typically addressed using gradient-based optimisation methods to maintain efficiency, however overlooking the global design space. Moreover, gradients are frequently unavailable. BO presents a promising gradient-free alternative, enabling sample-efficient global optimisation through probabilistic surrogate models. Although BO has shown its effectiveness for problems with a small number of design variables, it struggles to scale to high-dimensional problems, particularly when incorporating large-scale constraints. This challenge is especially pronounced in aeroelastic tailoring, where directional stiffness properties are integrated into the structural design to manage aeroelastic deformations and

enhance both aerodynamic and structural performance. Ensuring the safe operation of the system requires simultaneously addressing constraints from various analysis disciplines, making global design space exploration even more complex. This study seeks to address this issue by employing high-dimensional Bayesian optimisation combined with dimensionality reduction to tackle the optimisation challenges in aeroelastic tailoring. The proposed approach is validated through experiments on a well-known benchmark case, as well as its application to the aeroelastic tailoring problem, demonstrating the feasibility of BO for high-dimensional problems with large-scale constraints.

4.1. Introduction

The design of modern aircraft with enhanced efficiency is crucial for enabling more sustainable aviation. Achieving this involves optimising structural designs to reduce energy consumption. Aeroelastic tailoring emerges as a key technique that has the potential to reduce the weight of aeroelastically efficient high aspect ratio wings. Pioneered by Shirk et al. (1986), aeroelastic tailoring incorporates directional stiffness properties to effectively carry and control the aeroelastic deformations. Performing aeroelastic tailoring is a Multi-disciplinary Design Optimisation (MDO) effort, involving aerodynamics for the outer-mould shape definition and calculation of the loads acting over the wing, structural design that usually defines the layout of the main structural components of the wingbox, structural analysis to define and evaluate the relevant failure modes that should be considered as constraints, aeroelasticity that couples the aerodynamic loads with the inertial and elastic properties of the wing in order to characterise the flutter behaviour, and optimisation, to properly explore the design space. Other disciplines are also involved, such as manufacturing, typically resulting in additional constraints for the design variables.

Evaluating these complex aeroelastic models is computationally expensive, therefore necessitating efficient optimisation algorithms that require fewer analyses before finding an optimum solution. Due to the high number of design variables, describing the structural properties of the system, commonly gradient-based optimisation algorithms are used, leading to an efficient convergence towards the optimal solution. However, the computation of gradients is not always feasible, especially if the model's source code is unavailable. In such cases, the model must be treated as a black box, relying on methods like finite differences to obtain the design sensitivities, which can lead to prohibitively high computational costs that would ultimately motivate the use of gradient-free methods. Furthermore, many engineering problems, such as noisy responses or experimental results, possess inherent complexities that can render gradient-based approaches less effective or even impractical. Additionally, the response surface for feasible designs in aeroelastic tailoring is often multi-modal. This complexity can cause gradient-based methods to become trapped in local optima, overlooking the broader global design space and hindering the discovery of superior designs. Therefore, it is essential to develop methods that efficiently explore the

global design space, optimising structures to achieve lighter aircraft configurations.

The optimisation problem at hand can be formulated as follows:

$$\min_{\mathbf{x}\in\mathcal{X}\subset\mathbb{R}^D} f(\mathbf{x}) \text{ s.t.} \forall j \in \{1, ..., G\}, c_j(\mathbf{x}) \leq 0, \tag{4.1}$$

where $\mathcal{X} \subset \mathbb{R}^D$ is a D-dimensional space of potential designs, $f(\mathbf{x}) : \mathbf{x} \in \mathcal{X} \rightarrow \mathbb{R}$ the objective function and G constraints arising from the multi-disciplinary analyses. Overall, aeroelastic tailoring can be seen as an optimisation problem consisting of high-dimensional inputs and outputs, where the utilised models are able to map the vector of design variables to the objective function $f(\mathbf{x}) \in \mathbb{R}$ and all G constraints $\mathbf{c}(\mathbf{x}) \in \mathbb{R}^G$.

The simultaneous consideration of multiple disciplines can lead to large-scale constraints where $G \gg 10^3$, combining buckling, aeroelastic stability, maximum stress, maximum strain, and various others. In aeroelastic tailoring, the optimal stiffness distribution is achieved by means of a sizing optimisation that, in the case of laminated composite wings consists of finding the best set of lamination parameters and the optimum thickness for one or more composite regions (Werter, 2017). Lamination parameters allow a condensed and theoretical representation of the membrane, bending, and coupled stiffness terms of a laminate with continuous variables (Dillinger et al., 2013), making the sizing optimisation more convex and more adequate to established continuous optimisation techniques, where the design variables can be treated as continuous variables. Once this sizing optimisation is complete, a second discrete optimisation is performed to retrieve a manufacturable set of ply orientations. Yet, the presence of multiple design regions to maintain design freedom can still result in the number of design variables being in the order of hundreds or thousands.

Given the expensive nature of evaluating an aeroelastic model to obtain the objective function values and associated constraints, a sample-efficient optimisation algorithm is crucial. Compared to other gradient-free approaches like Random Search, Genetic Algorithms, and others, BO has proven to be a powerful method for complex and computationally costly problems (Mockus, 1989) and has been extensively applied across various domains, including aircraft design (Saves et al., 2022). BO addresses the challenge of expensive evaluations by using computationally inexpensive probabilistic surrogate models, such as GPs. These models replace the black-box functions representing the objective and constraints, significantly improving optimisation efficiency (Frazier, 2018). While many authors have shown that for lower dimensional problems, BO methods perform well, high-dimensional cases pose significant challenges due to the curse of dimensionality (Eriksson and Jankowiak, 2021, Priem, 2020), resulting from the fact that high dimensional search spaces are difficult to explore exhaustively. However, BO offers a probabilistic approach to efficiently search the design space to find promising regions and to reduce the computational burden. While these algorithms offer a variety of advantages, including the learning-from-data aspect,

uncertainty quantification, the lack of need for gradients, the ability to fuse data in a multi-fidelity context, and the capability to learn the correlation between simulation and experimental data, their scalability to high-dimensional problems with many constraints, as is often the case in engineering design, remains a significant challenge.

The present study focuses on employing high-dimensional BO algorithms for aeroelastic tailoring while considering large-scale constraints arising from the multidisciplinary analyses, as formulated in Equation (4.1). The novelty of this paper lies in the application of a high-dimensional BO method with a dimensionality reduction approach that significantly lowers the computational burden arising from the incorporation of a large number of constraints. Subsequently, the methodology is applied to the $7D$ speed reducer benchmark problem with 11 black box constraints Lemonge et al. (2010) before its application to aeroelastic tailoring is presented.

The structure of the paper is as follows. First, Section 4.2 introduces GPs as a probabilistic surrogate modelling technique, as well as BO for both unconstrained and constrained problems, highlighting scalability challenges. Section 4.3 then explores dimensionality reduction in the context of constrained BO, followed by a discussion of the numerical results in Section 4.4. Finally, the paper concludes with a discussion and directions for future work in Section 4.5.

4.2. High-Dimensional Constrained Bayesian Optimisation

This section briefly introduces BO within the context of high dimensionality and constraints. GPs are introduced as the herein employed surrogate modelling technique. Subsequently, GPs are linked to unconstrained BO, which is then expanded to address the constrained scenario, followed by an outline of the challenges encountered in this work.

4.2.1. Gaussian Processes

A GP in the context of BO serves as a probabilistic surrogate model that efficiently represents an unknown function $f(\mathbf{x})$. Recall that $\mathcal{X} \subset \mathbb{R}^D$ is a D-dimensional domain and the corresponding minimisation problem is presented in Equation (4.1). Beginning with a Design of Experiments (DoE) denoted by $\mathcal{D}_0 = \{\mathbf{x}_i, f(\mathbf{x}_i)\}_{i=1,\dots,N_0}$, where $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^D$ is the i-th of N samples and $f(\mathbf{x}_i) : \mathcal{X} \to \mathbb{R}$ the objective function, mapping from the design space to a scalar value. GPs are commonly employed within BO to construct a surrogate model $\hat{f}(\mathbf{x}) : \mathcal{X} \to \mathbb{R}$ of the objective function f from this given data set $\mathcal{D}$. Therefore, it is assumed that the objective function f follows a GP, which is also called a multivariate normal distribution $\mathcal{N}$. By defining the mean $m : \mathcal{X} \to \mathbb{R}$ and covariance $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, a noise-free surrogate can thus be denoted as:

$$f(\mathbf{x}) \mid \mathcal{D} \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right), \tag{4.2}$$

also known as the prior. The prior encapsulates the initial belief that observations are normally distributed. A common choice for the covariance function, also called kernel, is the squared exponential kernel $k(\mathbf{x}, \mathbf{x}')$ defined as

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f \exp\left(-\frac{1}{2}\sum_{i=1}^{D}\left(\frac{x_i - x_i'}{l_i}\right)^2\right), \tag{4.3}$$

which encodes the similarity between two chosen points $\mathbf{x}$ and $\mathbf{x}'$ (Rasmussen and Williams, 2006). The parameter l_i for $i = 1, ..., D$ is called the length scale and measures the distance for being correlated along x_i. Together with σ_f, often called the signal variance, the parameters form a set of so-called hyperparameters $\boldsymbol{\theta} = \{l_1, ..., l_D, \sigma_f\}$, in total $D + 1$ parameters, which need to be determined to train the model with respect to the target function. The kernel matrix is defined as $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{i,j=1,...,N} \in \mathbb{R}^{N\times N}$. The kernel needs to be defined such that $\mathbf{K}$ is symmetric positive definite to ensure its invertibility. The positive definite symmetry is guaranteed if and only if the used kernel is a positive definite function, as detailed in Schoenberg (1938). The values of the hyperparameters $\boldsymbol{\theta}$ are determined by maximising the marginal likelihood, written as

$$\log p(\mathbf{f} \mid \mathcal{D}, \boldsymbol{\theta}) = -\frac{1}{2}\mathbf{f}^\top \mathbf{K}^{-1}\mathbf{f} - \frac{1}{2}\log|\mathbf{K}| - \frac{N}{2}\log 2\pi. \tag{4.4}$$

Computing the partial derivative with respect to the hyperparameters $\boldsymbol{\theta}$ gives

$$\frac{\partial}{\partial\theta_j}\log p(\mathbf{f} \mid \mathcal{D}, \boldsymbol{\theta}) = \frac{1}{2}\mathbf{f}^\top \mathbf{K}^{-1}\frac{\partial \mathbf{K}}{\partial\theta_j}\mathbf{K}^{-1}\mathbf{f} - \frac{1}{2}\operatorname{tr}\left(\mathbf{K}^{-1}\frac{\partial \mathbf{K}}{\partial\theta_j}\right) \tag{4.5}$$

which can be used within a gradient-based optimisation for model selection or in other words, hyper-parameter tuning. More detailed information can be found in Rasmussen and Williams (2006).

Considering a new query point $\mathbf{x}_+ \in \mathcal{X}$, the stochastic process in Equation (4.2) can be used to predict the new query point

$$f(\mathbf{x}_+) \mid \mathcal{D} \sim \mathcal{N}\left(\mu(\mathbf{x}_+), k(\mathbf{x}_+, \mathbf{x}_+)\right). \tag{4.6}$$

The posterior mean $\mu(\bullet)$ and covariance function $\sigma(\bullet)$ are computed by

$$\mu(\mathbf{x}_+) = \mathbf{k}(\mathbf{x}_+, \mathbf{X})\mathbf{K}(\mathbf{X}, \mathbf{X})^{-1}\mathbf{f}, \tag{4.7}$$

$$\sigma(\mathbf{x}_+) = k(\mathbf{x}_+, \mathbf{x}_+) - \mathbf{k}(\mathbf{x}_+, \mathbf{X})\mathbf{K}(\mathbf{X}, \mathbf{X})^{-1}\mathbf{k}(\mathbf{X}, \mathbf{x}_+), \tag{4.8}$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N] \subset \mathcal{D}$ is the collection of samples and $\mathbf{f} = [f_1, f_2, ..., f_N] \subset \mathcal{D}$ of computed objective values in $\mathcal{D}$.

4.2.2. Unconstrained Bayesian Optimisation

Up to this stage, the GP has been computed using the initial samples contained in $\mathcal{D}_t$ with $t = 0$. BO now proceeds iteratively to enhance the accuracy of the surrogate model by enriching $\mathcal{D}_t$ while exploring the design space. Thus, leveraging the acquired data, the endeavour is to identify regions expected to yield optimal values. The problem at hand can be written as

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}). \tag{4.9}$$

An acquisition function $\alpha : \mathcal{X} \to \mathbb{R}$ is used to guide the optimisation through the design space while trading off exploration and exploitation based on the posterior mean and variance defined in Equation (4.7). The former describes the exploration of the whole design space, whereas the latter tries converging to an optimum based on the data observed. This can be written as

$$\mathbf{x}_+ \in \operatorname*{argmax}_{\mathbf{x}\in\mathcal{X}} \alpha(\mathbf{x} \mid \mathcal{D}_t). \tag{4.10}$$

Numerous acquisition functions exist, often making use of the predictive mean $\hat{\mu}(\mathbf{x})$ and variance $\hat{\sigma}(\mathbf{x})$. Popular choices for such an acquisition function are for example EI (Mockus, J. et al., 1978) or TS (Thompson, 1933).

4.2.3. Constrained Bayesian Optimisation

Most engineering design problems involve constraints, which can be integrated into the previously introduced BO method, discussed in e.g. Gardner et al. (2014), Gelbart et al. (2014), Hernández-Lobato et al. (2016). Assuming that the output of a model evaluation at design point $\mathbf{x}_i$ includes not only the objective function $f(\mathbf{x}_i)$, but also a mapping from the design space to a collection of G constraints $\mathbf{c}(\mathbf{x}_i) : \mathcal{X} \to \mathbb{R}^G$. Consequently, the DoE for this scenario is represented as $\mathcal{D}_t = \{\mathbf{x}_i, f(\mathbf{x}_i), \mathbf{c}(\mathbf{x}_i)\}_{i=1,..,N_t}$. The new design point found needs to lie in the feasible space $\mathcal{X}_f$, written as $\mathbf{x}_+ \in \mathcal{X}_f \subset \mathcal{X}$ where $\mathcal{X}_f := \{\mathbf{x} \in \mathcal{X} \mid c_j(\mathbf{x}) \leq 0,\ j = 1, ..., G\}$. Gardner et al. (2014) propose modelling each constraint $c_j(\mathbf{x}), j = 1, ..., G$ with an independent surrogate model, akin to how the objective function is modelled:

$$c_j(\mathbf{x}) \mid \mathcal{D} \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right), \tag{4.11}$$

leading to $G + 1$ GP models in total. Accordingly, these surrogate models can then be used within a constrained acquisition strategy, solving the optimisation problem formulated as

$$\mathbf{x}_+ \in \operatorname*{argmax}_{\mathbf{x}\in\mathcal{X}_f\subset\mathcal{X}} \alpha_c(\mathbf{x} \mid \mathcal{D}), \tag{4.12}$$

where α_c denotes a constrained acquisition function. This subsection serves to introduce the fundamental aspects of constrained BO concisely, emphasising that each constraint must be modelled via a separate GP model. Of course, a multitude of constrained acquisition functions exist. Among these approaches, for instance, is

the use of TS (Thompson, 1933) as an acquisition function (Hernández-Lobato et al., 2017), extended to the constrained setting in Eriksson and Poloczek (2021). A major advantage is its scalability to larger batch sizes. The latter study also demonstrates the superiority of this approach compared to CEI which is why CTS is employed in the course of this work and is explained in Algorithm 4. Therein, for each GP used for modelling the objective function and the G constraints, the posterior is computed. For a batch size of Q points, a sample is drawn to get realisations of the surrogate models. Then, N_c candidate points are evaluated on the GPs to obtain either a set of feasible points with optimal objective value or points with a minimum total constraint violation $\mathbf{X}_+$.

Algorithm 4 Constrained Thompson Sampling

Input: $\mathcal{D}_t$ of t-th iteration, Q batch size, $\mathbf{X}_c = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{N_c}]$ with N_c candidates
while Computational budget is not exhausted **do**
  $\mathbf{X}_+ = \{\}$
  Compute current posterior $\mathrm{p}(\boldsymbol{\theta}|\mathcal{D}_t)$ for $f, c_1, ..., c_G$
  for q = 1:Q **do**
    Sample $\boldsymbol{\theta}$ from $\mathrm{p}(\boldsymbol{\theta}|\mathcal{D}_t)$ to obtain realisations for $\hat{f}, \hat{c}_1, ..., \hat{c}_G$
    Evaluate $\{\mathbf{x}_i \mid i \in \mathbb{N},\ 1 \leq i \leq N_c\}$ on $\hat{f}(\mathbf{x}_i), \hat{c}_1(\mathbf{x}_i), ..., \hat{c}_G(\mathbf{x}_i)$
    Obtain $\hat{f}(\mathbf{x}_i), \hat{c}_1(\mathbf{x}_i), ..., \hat{c}_G(\mathbf{x}_i)$
    Choose $\mathbb{X}_f = \{\mathbf{x}_i | \hat{c}_l(\mathbf{x}_i) \leq 0 \text{ for } 1 \leq l \leq G\}$
    if $\mathbb{X}_f \neq \emptyset$ **then** $\mathbf{x}_+^q = \mathrm{argmax}_{\mathbf{x} \in \mathbb{X}_f} \hat{f}(\mathbf{x})$
    else Compute $\mathbf{x}_+^q = \mathrm{argmin}_{\mathbf{x} \in \mathbf{X}_c} \sum_{i=1:G} \max(\hat{c}_i(\mathbf{x}), 0)$
    end if
    $\mathbf{X}_+ = \mathbf{X}_+ \cup \{\mathbf{x}_+^q\}$
  end for
end while

4.2.4. High-Dimensional Bayesian Optimisation: Challenges and Advances

BO algorithms consist of two main components, namely the probabilistic surrogate model, GPs, which are based on Bayesian statistics (Rasmussen and Williams, 2006), and an acquisition function to guide the selection where to query the next point to converge towards the minimiser of the objective function. While these algorithms have been proven to be very efficient for lower-dimensional problems (Binois and Wycoff, 2022), scaling them to higher dimensions implies some difficulties:

- The curse of dimensionality dictates that as the number of dimensions increases, the size of the design space grows exponentially, making an exhaustive search impractical.
- With higher dimensions, there is an increase in the number of tunable hyper-parameters $\boldsymbol{\theta} \in \mathbb{R}^{D+1}$, resulting in a more cumbersome GP model learning,

possibly leading to increased uncertainty.

- Higher-dimensional problems necessitate more samples N to construct an accurate surrogate model. The inversion of the covariance matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$ becomes computationally intensive with a complexity for inference and learning of $\mathcal{O}(N^3)$ and $\mathcal{O}(N^2)$ for memory.
- Insufficient data collection results in sparse sampling across the D-dimensional hyperspace, causing samples to be widely dispersed from each other. This dispersion hinders effective correlation among the samples.
- Acquisition function optimisation faces increased uncertainty in high-dimensional settings, requiring more evaluations of the surrogate model (Binois and Wycoff, 2022).

Various strategies have been employed to address the challenge of high-dimensional input spaces in scenarios with few or no constraints. In Wang et al. (2016), random projections are utilised to reduce high-dimensional inputs to a lower-dimensional subspace, allowing for the construction of the GP model directly in this reduced space, thereby reducing the number of hyperparameters. Similarly, Raponi et al. (2020), Antonov et al. (2022) employ (kernel) PCA on the input space to identify a reduced set of dimensions based on evaluated samples, followed by training the surrogate model in this reduced dimensional space. In contrast, Eriksson and Jankowiak (2021) adopt a hierarchical Bayesian model that assumes varying importance among design variables, using a sparse axis-aligned prior on the length scale to discard dimensions unless supported by accumulated data. However, Santoni et al. (2023) demonstrates high computational overhead in this approach. Additionally, decomposition techniques, such as additive methods, are employed to partition the original space, as demonstrated in Kandasamy et al. (2016), Ziomek and Bou-Ammar (2023).

The TuRBO algorithm, described in Eriksson et al. (2020), takes a different route where the design space is partitioned into multiple independent TR. Results from Eriksson et al. (2020) demonstrate promising outcomes for this approach, particularly in high-dimensional problems where gathering sufficient data to construct a globally accurate surrogate model is challenging due to the curse of dimensionality. Instead, surrogates are focused on these defined TR, which adjust in size during optimisation. TR are defined as hyper-rectangles of size $r \in \mathbb{R}$, centred at the best solution found so far and initialised with $r \leftarrow r_{init}$, a user-defined parameter. The size r_{TR} of each TR is determined using the length scale l_i of the GP, defined in Equation (4.3), and a base length scale r:

$$r_{TR} = \frac{l_i r}{\left(\prod_{j=1}^{D} l_j\right)^{1/D}}. \tag{4.13}$$

In each optimisation iteration, a batch of q samples are drawn within the TR. When the design space is normalised to $\mathcal{X} \in [-1, 1]$ and r spans the entire design space

with $r \to 2$ kept constant, the TR approach resembles a standard BO algorithm as outlined in Frazier (2018). The evolution of r significantly influences the convergence of this method, and specific hyperparameters governing its adaptation are detailed in Eriksson et al. (2020).

All the algorithms previously discussed focus exclusively on unconstrained optimisation problems. The TR approach, however, addresses constraints explicitly by adapting the batched TS method from Thompson (1933) as an acquisition function for constrained problems (Eriksson and Poloczek, 2021), detailed in Algorithm 4. This method, known as SCBO, employs separate GPs to model each constraint within the current TR. Scaling BO to high-dimensional problems necessitates addressing significant challenges through specific assumptions. While existing approaches demonstrate promising results, handling large-scale constraints, such as those encountered in aircraft design problems where $G > 10^3$, remains insufficiently addressed. This work adopts the constrained TuRBO algorithm SCBO for high-dimensional BO due to its explicit treatment of constraints. Next, an extension of this method is introduced to address the challenge posed by large-scale constraints.

4.3. Large-Scale Constrained Bayesian Optimisation via Latent Space Gaussian Processes

Recall the optimisation problem formulated in Equation (4.1). By using constrained BO methods, as shown earlier, each of the G constraints needs to be modelled with an independent GP, denoted as $\hat{c}_j(\mathbf{x})$. This work follows the idea of Higdon et al. (2008) to construct the surrogates on a lower dimensional, latent output space. Let $\mathcal{V} \subset \mathbb{R}^G$ denote a G-dimensional space. The objective of this work is to identify a latent space $\mathcal{V}' \subset \mathbb{R}^g$ such that $\mathcal{V}' \subset \mathcal{V}$, where $g \ll G$. This subspace may be found by using dimensionality reduction methods such as PCA (Jolliffe and Cadima, 2016) on the training data in $\mathcal{D}_t$. An extended nonlinear version of PCA is the Kernel Principal Component Analysis (kPCA), presented by Schölkopf et al. (1998).

During the DoE, alongside the samples $\mathbf{x}_i$ and their corresponding objective function values f_i, constraint values $\mathbf{c} : \mathcal{X} \to \mathbb{R}^G$ are also collected in $\mathcal{D}_t$. This enables the construction of a matrix $\mathbf{C}(\mathbf{x})$ given by:

$$\mathbf{C}(\mathbf{x}) = \begin{bmatrix} \mathbf{c}(\mathbf{x}_1)^\top \\ \mathbf{c}(\mathbf{x}_2)^\top \\ \vdots \\ \mathbf{c}(\mathbf{x}_N)^\top \end{bmatrix} = \begin{bmatrix} c_1(\mathbf{x}_1) & c_2(\mathbf{x}_1) & \dots & c_G(\mathbf{x}_1) \\ c_1(\mathbf{x}_2) & c_2(\mathbf{x}_2) & \dots & c_G(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ c_1(\mathbf{x}_N) & c_2(\mathbf{x}_N) & \dots & c_G(\mathbf{x}_N) \end{bmatrix} \in \mathbb{R}^{N \times G}. \tag{4.14}$$

Here, N_t represents the number of samples and G denotes the number of constraints.

4.3.1. Principal Component Analysis

Within PCA, a linear combination with maximum variance is sought, such that

$$\mathbf{C}\mathbf{v} = \lambda\mathbf{v} \tag{4.15}$$

where $\mathbf{v}$ is a vector of constants. These linear combinations are called the principal components of the data contained in $\mathbf{C}$. After centring the data with $\bar{\mathbf{C}} = \mathbf{C} - \mathbf{I}_N\mu$ with $\mu = \frac{1}{N}\sum_{i=1}^{N}\mathbf{c}_i$, a covariance matrix $\mathcal{C}$ is computed

$$\mathcal{C} = \frac{1}{N-1}\bar{\mathbf{C}}^\top\bar{\mathbf{C}} \in \mathbb{R}^{G\times G}. \tag{4.16}$$

Subsequently, PCA seeks the set of orthogonal vectors that capture the maximum variance in the data. This is achieved by performing an eigenvalue decomposition of $\mathcal{C}$, to obtain the corresponding eigenvalues λ and eigenvectors $\mathbf{v}$ such that

$$\mathcal{C}\mathbf{v}_i = \lambda_i\mathbf{v}_i, \ \forall i = 1, 2, ..., G \tag{4.17}$$

with $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_G \geq 0$. The eigendecomposition of $\mathcal{C}$ is then written as

$$\mathcal{C} = \mathbf{\Psi}\mathbf{\Lambda}\mathbf{\Psi}^{-1} \tag{4.18}$$

The matrix $\mathbf{\Psi} = [\Psi_1, ..., \Psi_G] \in \mathbb{R}^{G\times G}$ has orthonormal columns such that $\mathbf{\Psi}^\top\mathbf{\Psi} = \boldsymbol{I}_{\mathbf{\Psi}}$ and $\mathbf{\Lambda} = \text{diag}(\lambda_1, ..., \lambda_G) \in \mathbb{R}^{G\times G}$ is a diagonal matrix, containing the eigenvalues. By investigating the eigenvalues in $\mathbf{\Lambda}$, and choosing the ones with the g highest values, the truncated decomposition is obtained, consisting of the reduced basis containing g orthogonal basis vectors in $\mathbf{\Psi}_g \in \mathbb{R}^{G\times g}$ with $g \ll G$. The new basis vectors can subsequently be used as a projection $\mathbf{\Psi}_g^\top : \mathcal{V} \subset \mathbb{R}^G \rightarrow \mathcal{V}' \subset \mathbb{R}^g$ to project the matrix $\mathbf{C}$ onto the reduced subspace $\tilde{\mathbf{C}} \in \mathbb{R}^{N\times g}$, written as

$$\tilde{\mathbf{C}} = \mathbf{C}\mathbf{\Psi}_g. \tag{4.19}$$

Summarising, the G constraints $\mathbf{c}(\mathbf{x})$ can be represented on a reduced subspace through the mapping $\mathbf{\Psi}_g$ while the eigenvalues λ_i give an indication about the loss of information, potentially drastically lowering the number of constraints that need to be modelled. A graphical interpretation is depicted in Figure 4.1. For a more thorough derivation of this method, the reader is referred to Jolliffe and Cadima (2016).

4.3.2. Kernel Principal Component Analysis

While PCA can be seen as a linear dimensionality reduction technique, in Schölkopf et al. (1998) the authors present an extension, called kernel PCA, using a nonlinear projection step to depict nonlinearities in the data. Similarly to the PCA algorithm, the starting point is the (centred) samples $\mathbf{c}_i(\mathbf{x}_i) \in \mathcal{V} \subset \mathbb{R}^G \ \forall i \in \{1, ..., N\}$.
Let $\mathcal{F}$ be a dot product space (in the following, also called feature space) of arbitrary

large dimensionality. A nonlinear map $\boldsymbol{\phi}(\mathbf{x})$ is defined as $\boldsymbol{\phi} : \mathbb{R}^G \to \mathcal{F}$. This map is used to construct a covariance matrix $\mathcal{C}$, similar to PCA, defined as

$$\mathcal{C} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{\phi}(\mathbf{c}(\mathbf{x}_i))\boldsymbol{\phi}(\mathbf{c}(\mathbf{x}_i))^\top. \tag{4.20}$$

The corresponding eigenvalues and eigenvectors in $\mathcal{F}$ are computed by solving

$$\mathcal{C}\mathbf{v} = \lambda \mathbf{v}. \tag{4.21}$$

As stated earlier, since the function $\boldsymbol{\phi}$ maps possibly to a very high-dimensional space $\mathcal{F}$, solving the eigenvalue problem therein may be costly. A workaround is used to avoid computations in $\mathcal{F}$. Therefore, similar to the formulation of the GP models in Section 4.2.1, a kernel $k : \mathbb{R}^G \times \mathbb{R}^G \to \mathbb{R}$ is defined as

$$k(\mathbf{c}(\mathbf{x}_i), \mathbf{c}(\mathbf{x}_j)) = \left\langle \boldsymbol{\phi}(\mathbf{c}(\mathbf{x}_i)), \boldsymbol{\phi}(\mathbf{c}(\mathbf{x}_j)) \right\rangle = \boldsymbol{\phi}(\mathbf{c}(\mathbf{x}_i))^\top \boldsymbol{\phi}(\mathbf{c}(\mathbf{x}_j)) \tag{4.22}$$

and the corresponding kernel matrix $\mathbf{K}_{ij}$ as

$$\mathbf{K}_{ij} := (\boldsymbol{\phi}(\mathbf{c}(\mathbf{x}_j)), \boldsymbol{\phi}(\mathbf{c}(\mathbf{x}_j))) \in \mathbb{R}^{N \times N}. \tag{4.23}$$

By solving the eigenvalue problem for non-zero eigenvalues

$$\mathbf{K}\boldsymbol{\alpha}_i = \lambda_i \boldsymbol{\alpha}_i \tag{4.24}$$

the eigenvalues $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_N$ and eigenvectors $\boldsymbol{\alpha}^1, ..., \boldsymbol{\alpha}^N$ are obtained. This part can be seen as the linear PCA, as presented before, although in the space $\mathcal{F}$. To map a test point $\mathbf{c}_+(\mathbf{x})$ from the feature space $\mathcal{F}$ to the q-th principal component $\mathbf{v}^q$ of Equation (4.21), the following relationship is evaluated

$$\left((\mathbf{v}^q)^\top \boldsymbol{\phi}(\mathbf{c}_+(\mathbf{x}))\right) = \sum_{i=1}^{N} \boldsymbol{\alpha}_i^q (\boldsymbol{\phi}(\mathbf{c}(\mathbf{x}_i))^\top \boldsymbol{\phi}(\mathbf{c}_+(\mathbf{x}))) \equiv \tilde{\mathbf{c}}_+(\mathbf{x}_+). \tag{4.25}$$

A graphical interpretation can be found in Figure 4.1. The kernel function in Equation (4.22) can also be replaced by another a priori chosen kernel function.

4.3.3. Dimensionality Reduction for Large-Scale Constraints

When large-scale constraints are involved, the computational time as well as the needed storage scales drastically since one GP model has to be constructed and trained for each constraint. Therefore, describing the constraints on a latent space allows to significantly lower the computational burden. This idea is based on the work of Higdon et al. (2008), who project the simulation output onto a lower dimensional subspace where the GP models are constructed. Other works extended this method then by employing, among others, kPCA as well as manifold learning techniques to

account for nonlinearities (Xing et al., 2015, 2016). However, the aforementioned authors try to approximate PDE model simulations with high-dimensional outputs, whereas, to the best of the authors' knowledge, the combination of dimensionality reduction techniques for use in high-dimensional BO with large-scale constraints for design optimisation is novel. The methods herein presented are capable of

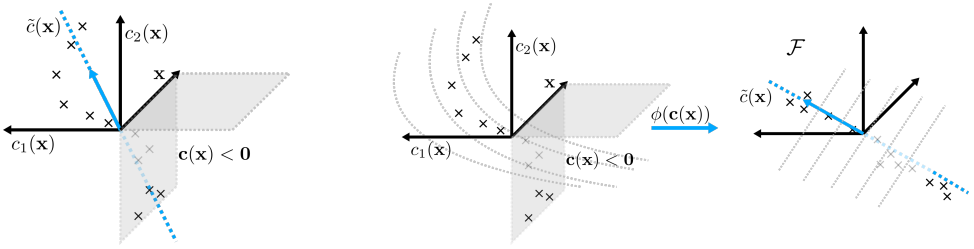


Figure 4.1: Graphical interpretation of dimensionality reduction for constraints. **(left)** Principal Component analysis. **(right)** Kernel Principal Component Analysis.

extracting the earlier introduced, most important principal components of available data, reducing the required amount of GP models to g instead of G, with $\mathbf{v}_j$ as the j-th orthogonal basis vector. After projecting the data onto the lower dimensional subspace by using either PCA as in Equations (4.19) or kPCA in Equation (4.25), GPs are constructed on the latent output space as independent batch GPs , formulated as

$$\tilde{\mathbf{c}}_i \sim \mathcal{GP}\left(m_i(\mathbf{x}), k_i(\mathbf{x}, \mathbf{x}')\right) \forall i \in \{1, ..., g\}. \tag{4.26}$$

These constraint surrogates on the latent space are then used to navigate through the design space to ultimately find a feasible and optimal design. A graphical interpretation is depicted in Figure 4.1, inspired by Schölkopf et al. (1998). In the following, the projection of the constraints onto the lower-dimensional subspace in the i-th iteration is denoted as $\mathcal{P}_i : \mathbb{R}^G \to \mathbb{R}^g$. A schematic illustration of the GP

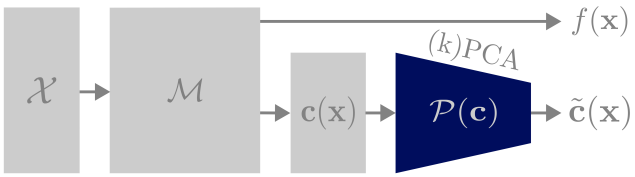


Figure 4.2: Schematic illustration of (k)PCA-GP.

construction is presented in Figure 4.2, where $\mathcal{M} : \mathcal{X} \to \mathbb{R}^{G+1}$ denotes the numerical model, mapping from the design space $\mathcal{X}$ to the objective $f : \mathcal{X} \to \mathbb{R}$ and constraints $\mathbf{c} : \mathcal{X} \to \mathbb{R}^G$ as outputs. The constraints are then projected via (k)PCA onto a lower-dimensional representation $\tilde{\mathbf{c}}$ where the independent GPs are constructed. It is important to emphasise that the validity of a feasible design, where no constraints

are violated, is checked in the original space rather than within the lower-dimensional subspace. This is made possible since in each iteration, a batch of q new samples is obtained and evaluated using the expensive-to-evaluate model. Hereinafter, the two methods are called PCA-GP SCBO and kPCA-GP SCBO and are summarised in Algorithm 5.

Algorithm 5 SCBO with Latent Space Gaussian Processes

Input: Input space $\mathcal{X}$, Number of candidates N_c, batch size q_c, number of initial samples N_0, SCBO hyperparameters, number of eigenvalues N_{ev} or tolerance τ_{ev}
Compute DoE $\mathcal{D}_0 = \{\mathbf{x}_i, f(\mathbf{x}_i), \mathbf{c}(\mathbf{x}_i)\}_{i=1:N_0}$
$t = 0$
while Computational budget is not exhausted **do**
  With $\mathbf{c}(\mathbf{x}) \subset \mathcal{D}_t$ compute projection $\mathcal{P}_t$
  Project constraints onto lower dimensional subspace $\tilde{\mathbf{c}}(\mathbf{x}) = \mathcal{P}_t(\mathbf{c}(\mathbf{x}))$
  Fit GP for $f(\mathbf{x}), \tilde{c}_1(\mathbf{x}), ..., \tilde{c}_g(\mathbf{x})$
  $\mathbf{x}_+ \leftarrow$ constrainedThomsponSampling (see Algorithm 4)
  Evaluate $\mathbf{x}_+$ and observe $f(\mathbf{x}_+),\ \mathbf{c}(\mathbf{x}_+)$
  Update TuRBO state
  $\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{\mathbf{x}_+, f(\mathbf{x}_+), \mathbf{c}(\mathbf{x}_+)\}$
  $t \leftarrow t + 1$
end while

4.3.4. Related Work and Complexity Considerations

To tackle the issue of many outputs, several works have been published. The ICM can be related to the LMC, presented in Alvarez et al. (2012) and are based on MTGP (Bonilla et al., 2007). However, due to taking into account inter-task correlation, the size of the covariance matrix increases drastically. While in independent GP models, inference and learning typically has a complexity of $\mathcal{O}((G+1)N^3)$ and $\mathcal{O}((G+1)N^2)$ for storage, the size of multi-task models extends due to their Kronecker structure to complexities of $\mathcal{O}(N^3(G+1)^3)$ for inference and learning, with $G+1$ denoting the number of constraints plus the objective. Similarly, the storage complexity also scales to $\mathcal{O}(N^2(G+1)^2)$, posing significant computational challenges when the number of tasks/constraints and/or data points becomes large. The benefit of (k)PCA-GPs now is the fact that by mapping the outputs/constraints onto a g-dimensional subspace while no inter-task correlations are respected, the computational costs for inference and learning only scale linearly to $\mathcal{O}((g+1)N^3 + G^3)$ where $\mathcal{O}(G^3)$ accounts for the eigendecomposition during (k)PCA and $\mathcal{O}((g+1)N^2)$ for storage, where $g \ll G$.

To address some of the issues, apart from Higdon et al. (2008), Zhe et al. (2019) present scalable High-Order Gaussian Processes (HOGP) and show that their method is superior to (k)PCA-GP in terms of accuracy. Since (k)PCA-GPs assume a linear structure of the outputs, meaning that the output is a linear combination of

bases vectors, HOGP does not impose this kind of structure, thus claiming to be more flexible. The authors in Maddox et al. (2021) then extend MTGPs and later HOGP for a large number of outputs by employing Mathoron's rule to alleviate the computational burden of sampling from the posterior. Additionally, Bruinsma et al. (2019) introduce a method that tackles the problem of needing a high number of linear basis vectors in PCA-GP, which still scale cubically in the dimensionality of the subspace when inter-task correlations are taken into account. They leverage the statistics of data to achieve linear scaling. However, all these works take into account inter-task correlation and thus scale poorly compared to (k)PCA-GP, as concluded by Zhe et al. (2019). Due to the fact that in engineering design problems the dimensionality and constraints can become very large, thus high values for N and G can be expected, this work uses batched, independent GPs in the reduced latent space as originally proposed by Higdon et al. (2008). Due to the use in BO and the continuous retraining of the surrogates, the approach employed here significantly accelerates computations while maintaining acceptable accuracy as presented in Zhe et al. (2019).

4.4. Numerical Experiments

In this section, the presented methodology is applied to a benchmark case before results for the aeroelastic tailoring optimisation problem are shown. For comparison purposes, we adopt the reasoning of Hernández-Lobato et al. (2016), where a feasible solution is always preferred over an infeasible one. Therefore, we use the maximum value from all feasible solutions as the default for all infeasible solutions. To leverage the capabilities of existing, well performing frameworks, this study employs `BoTorch` (Balandat et al., 2020) and `GPyTorch` (Gardner et al., 2018) to make use of their extensive capabilities.

4.4.1. 7D Speed Reducer Problem with 11 Black-Box Constraints

The $7D$ speed reducer problem from Lemonge et al. (2010) includes 11 black-box constraints. The known optimal value for this problem is $f^* = 2996.3482$. The results for all three evaluated methods (SCBO, PCA-GP SCBO and kPCA-GP SCBO) are shown in Figure 4.3. 20 experiments are performed, where the solid line represents the mean objective value over the 20 experiments and the shaded area the standard deviation. f^* denotes the known optimal value of this problem. The eigenvalues of the matrix $\mathbf{C}$ with $N = 10$ samples are plotted on the right. Additionally, the decay of the eigenvalues λ of the constraint matrix $\mathbf{C} \subset \mathcal{D}$ is depicted. In this example where $G = 11$, $g = 4$ principal components are chosen. The SCBO hyperparameters are defined according to Eriksson and Poloczek (2021). The batch size is defined as $q = 1$ and $N = 20$ initial samples. The results are compared in Table 4.1. All methods find a feasible and optimal design. It is obvious that the original SCBO method converges faster than the ones employing latent GPs. In SCBO, each constraint is modelled independently via batched GPs. However,

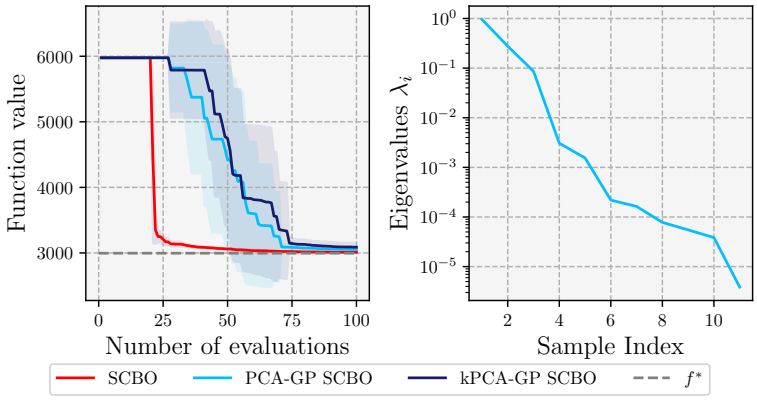


Figure 4.3: $7D$ Speed reducer problem with 11 black-box constraints from Lemonge et al. (2010).

.

Table 4.1: Computational time for speed reducer benchmark

Method	$\tilde{f}^*$	$(\tilde{f}^* - f^*)/f^*$	**Time**	**Time Saving**	**Succ.**
SCBO	3007.20	0.36%	501.38s	-	20/20
PCA-GP SCBO	3053.30	1.90%	201.38s	59.83%	20/20
kPCA-GP SCBO	3088.39	3.07%	216.96s	56.73%	20/20

besides the fact that the proposed methods are significantly faster, see Table 4.1, it is shown that both ultimately converging to an optimum very close to the one obtained via SCBO and the analytical solution f^*. kPCA-GP SCBO uses the Gaussian kernel, written as

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{-||\mathbf{x} - \mathbf{x}'||^2}{2\sigma^2}\right). \tag{4.27}$$

Here, PCA-GP SCBO converges slightly faster than kPCA-GP SCBO. It needs to be emphasised that this problem also does not show a fast decay of the eigenvalues, as can be seen in Figure 4.3 (left).

In addition, the influence of the number of principal components, g, is studied in Figure 4.4. It can be observed that g affects the convergence of the optimisation. Notably, when $g = 2$, although convergence is slower, the mean value found is close to the analytic value f^*. However, when $g = 1$ the subspace does not cover enough of the feasible design space, resulting in no feasible value being found. Lastly, the dimensionality of both the input and output spaces are examined. After demonstrating in this section that the method is generally effective on a lower dimensional problem, the benchmark is now extended to explore cases with either high-dimensional inputs or high-dimensional outputs. To achieve this, the benchmark is modified in two ways. First, it is embedded into a 100-dimensional input space. Second, artificial

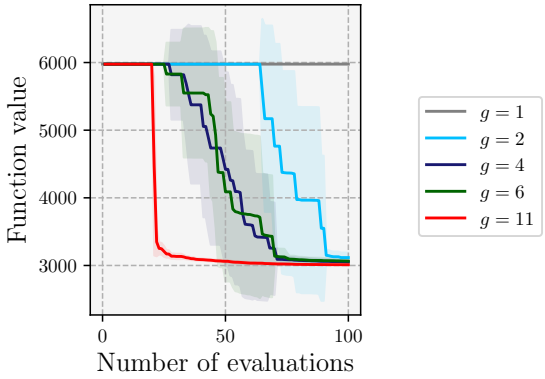


Figure 4.4: The influence of the number of principal components g on the result.

constraints are introduced, increasing the number of constraints to $G = 500$, while preserving the original optimisation problem's characteristics. This is ensured by adding non-violated constraints such that each additional constraint $c_k \leq 0$. These modifications allow for an in-depth investigation of different components. The corresponding results are presented in Figure 4.5, where:

- Case 1 represents the benchmark problem embedded in a high-dimensional input space with $D = 100$ and $G = 11$.
- Case 2 retains the original input space dimension with $D = 7$ but extends the number of constraints to $G = 500$.

(k)PCA-GP SCBO is applied to both cases, where the number of principal components is set to $g = 6$. Additionally, SCBO is included for Case 1, where constraints are directly modelled using independent GPs. However, Case 2 exceeds memory resources since SCBO tries to construct 501 GPs, one for the objective and 500 for the constraints. Moreover, $N = 20$ initial samples are used, with a batch size of $q = 3$ and an evaluation budget of 200 over 20 experiments. The experiments demonstrate that our proposed method successfully handles both high-dimensional input spaces (Case 1) and large-scale constraints (Case 2), whereas the original approach fail in the latter scenario due to the need for excessive surrogate model construction, exceeding memory resources. While both cases converged to similar function values, it can be observed that Case 1 converges at a slower rate than Case 2. This discrepancy can be attributed to the curse of dimensionality, which affects the efficiency of surrogate modelling and exploration in high-dimensional spaces. In contrast, while a large number of constraints increases computational cost, it does not fundamentally alter the exploration process when efficient dimensionality reduction is possible, allowing for faster convergence. These findings confirm the effectiveness of our method in tackling high-dimensional and large-scale constrained optimisation problems.

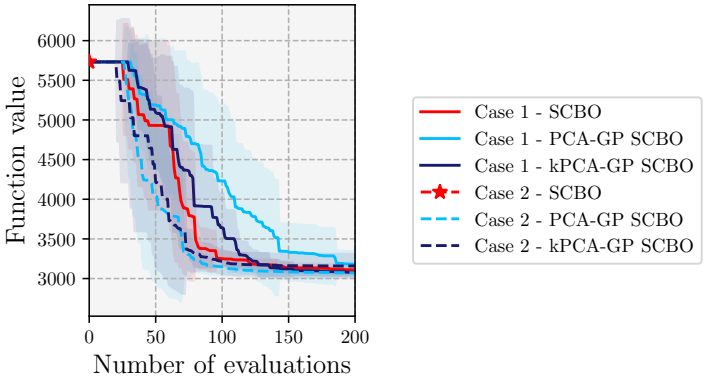


Figure 4.5: The influence of the input and output dimensionality.

Summarising, the lower dimensional subspace is constructed based on the constraint values in $\mathcal{D}_t$. Assuming that the global optimum lies on the boundary of the feasible space $\mathcal{X}_f$, the success of the method highly depends on how accurately the lower dimensional subspace captures the original space. That stresses the importance of computing the projection matrix $\mathcal{P}_t$ in every iteration. However, we find that for this specific case fixing $\mathcal{P}_t = \mathcal{P}_0$, the Algorithm 5 exhibits a better performance, presumably due to the rather low dimensionality and low number of constraints in combination with the use of the TR.

4.4.2. Aeroelastic Tailoring: An MDO Problem with 108D and 1786 Black-Box Constraints

The MDO problem of aeroelastic tailoring addressed in this work presents a high-dimensional problem with large-scale constraints, involving both high-dimensional inputs and outputs. Unlike the aforementioned benchmark problem where it is practical to construct a GP for each constraint, this is computationally infeasible here, where the number of constraints is $10^3 < G < 10^5$. Therefore, the methodology presented in this study facilitates the process by modelling these constraint GPs in a latent space. Figure 4.6 depicts the wing to be aeroelastically tailored by optimising the stiffness and thickness of the wingbox. The wingbox is span-wise discretised in three sections, where top skin, bottom skin, front spar and rear spar can take on different stiffness and thickness values. The wing span exhibits $b = 12.28\ m$, with a $c = 2.068\ m$ chord at the root and $c = 1.113\ m$ chord at the tip. The front and rear spar are located at $x_{fs} = 0.15c$ and $x_{rs} = 0.65c$, respectively. In total, $D = 108$ design variables are defined, consisting of the lamination parameters $\xi \in [-1, 1]$ and the thickness $t \in [0.002, 0.03]\ m$ of each panel, respectively. Each panel is described

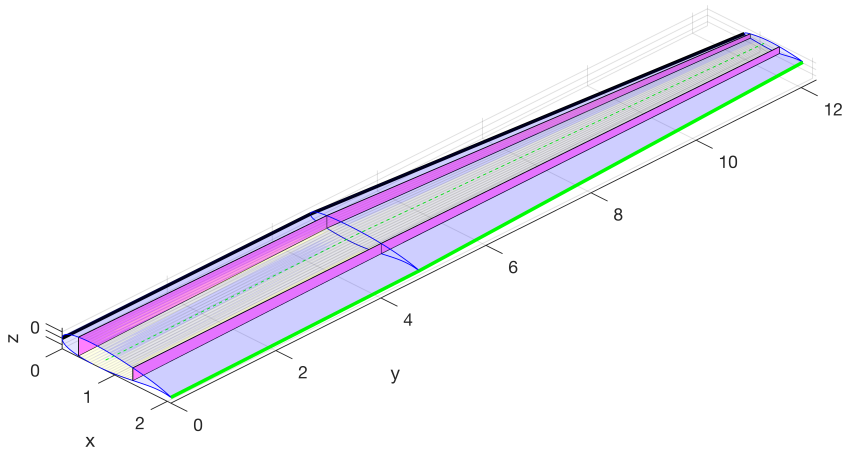


Figure 4.6: Wing structure consisting of wingbox and airfoil shape.

by a set of parameters $\mathbf{x}_i^{lam}$.

$$\begin{aligned} \mathbf{x} = &\left\{ \mathbf{x}_1^{lam}, \mathbf{x}_2^{lam}, ..., \mathbf{x}_{n_p}^{lam} \right\} \in \mathbb{R}^{108} \\ \text{with} \quad &\mathbf{x}_i^{lam} = \left\{ \xi_1^A, \xi_2^A, \xi_3^A, \xi_4^A, \xi_1^D, \xi_2^D, \xi_3^D, \xi_4^D, t \right\} \in \mathbb{R}^9. \end{aligned} \tag{4.28}$$

Based on the classical laminate theory, the following constitutive equations are used to relate the distributed forces N and moments M, with the in-plane ϵ^0 and curvature κ strains

$$\begin{bmatrix} N \\ M \end{bmatrix} = \begin{bmatrix} \mathbf{A}(\mathbf{x}) & \mathbf{0} \\ \mathbf{0} & \mathbf{D}(\mathbf{x}) \end{bmatrix} \begin{bmatrix} \epsilon^0 \\ \kappa \end{bmatrix} \tag{4.29}$$

The so-called ABD-matrix can be calculated by means of lamination parameters according to Tsai and Pagano (1968) as follows:

$$\begin{aligned} \mathbf{A}(\mathbf{x}) &= t(\mathbf{\Gamma_0} + \mathbf{\Gamma_1}\xi_1^A + \mathbf{\Gamma_2}\xi_2^A + \mathbf{\Gamma_3}\xi_3^A + \mathbf{\Gamma_4}\xi_4^A) \\ \mathbf{D}(\mathbf{x}) &= \frac{t^3}{12}(\mathbf{\Gamma_0} + \mathbf{\Gamma_1}\xi_1^D + \mathbf{\Gamma_2}\xi_2^D + \mathbf{\Gamma_3}\xi_3^D + \mathbf{\Gamma_4}\xi_4^D) \end{aligned} \tag{4.30}$$

where $\mathbf{\Gamma_i}$ are material invariants, defined in Tsai and Pagano (1968). Equation (4.30) encodes the dependency of the design variables $\mathbf{x}$ with the stiffness of the system (Daniel and Ishai, 2006). The constraints result from the incorporation of two loadcases. These multiple loadcases are often one of the reasons why the number of constraints can become very high. The aforementioned constraints arise from the multidisciplinary analyses, summarised in Table 4.2 and leading to a total number of $G = 1786$, similarly depending on the input variables $\mathbf{x}$. More information of the aeroelastic tailoring optimisation problem can be found in Maathuis et al. (2024).

Table 4.2: Aeroelastic tailoring constrained optimisation problem.

Type	Parameter	Symbol	#	
Objective	Minimise Wing Mass [kg]	f		
Design Variables (D)	Lamination Parameter Laminate Thickness	$\mathbf{x}$		
Constraints (G)	Laminate Feasibility	$\boldsymbol{c}_{lf}$	72	
	Static Strength	$\boldsymbol{c}_{tw}$	96	/loadcase
	Buckling	$\boldsymbol{c}_{b}$	768	/loadcase
	Aeroelastic Stability	$\boldsymbol{c}_{ds}$	10	/loadcase
	Aileron Effectiveness	c_{ae}	1	/loadcase
	Local Angle of Attack	$\boldsymbol{c}_{AoA}$	18	/loadcase

Apart from the mathematical reasoning to find a latent space of the output data, the premise of the introduced methodology lies in the consistency of the physics governing the constraints across loadcases, where eventually only the load changes. This stresses the potential for compressing this information due to the unchanged underlying physics for varying loadcases.

The lamination parameter feasibility constraints are, however, closed-form equations. These analytical equations do not need to be modelled via surrogates since their behaviour is known in the design space. Thus, these constraints are taken into account inherently within the sampling process via rejection sampling. Every candidate point in N_c is only added if not violating one of these feasibility constraints.

The aforementioned aeroelastic tailoring model is used to compute the DoE

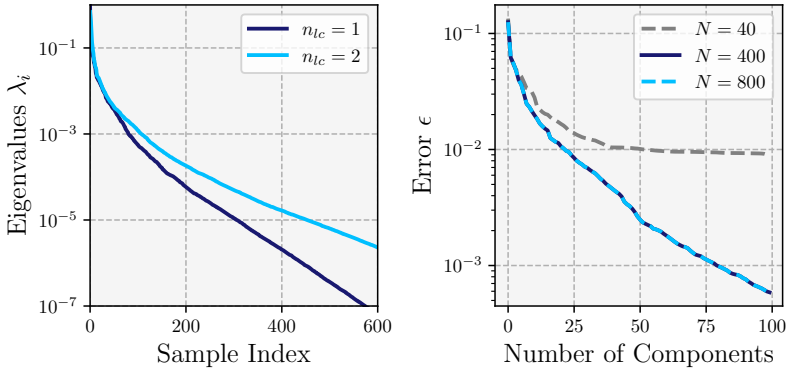


Figure 4.7: Investigating the constraints in $\mathcal{D}$.

$\mathcal{D}_0$ with $N = 416$ samples. Sampling was performed via LHS. One evaluation of this low-fidelity model takes ≈ 10s due to parallelisation. Anyway, subsequently PCA is applied on the matrix $\mathbf{C}$ to investigate its eigenvalues. Figure 4.7 depicts the decay of these computed eigenvalues. If the same error metric as in Subsection 4.4.1, eigenvalues up to approx $\lambda_i \approx 10^{-2}$, thus $g = 29$ principal components might be enough to construct a lower dimensional subspace of sufficient accuracy.

As previously noted, the high number of constraints stems from the incorporation of multiple loadcases. Consequently, it becomes intriguing to explore how the eigenvalues vary when the number of loadcases is altered. Recall that the eigenvalues denote the importance of their corresponding eigenvector, which serves as a measure of where to truncate the projection matrix. Beyond that, in Figure 4.7 we compared the eigenvalues of $n_{lc} = 1$ and $n_{lc} = 2$ loadcases. It can be observed that, even though the number of constraints in the original space has doubled, from $G = 893$ to $G = 1786$, if the eigenvalues $\lambda_i > 10^{-2}$ are used, no more principal components have to be taken into account. For $\lambda_i > 10^{-3}$, however, only 27 more components are needed to maintain the same error. Beyond that, the threshold of the eigenvalues is commonly set based on experience, thus can be seen as a hyper-parameter of the method.

To compute the projection error, some unseen data $\mathbf{C}_*$, is mapped onto the lower dimensional subspace $\tilde{\mathbf{C}}_* = \mathbf{\Psi}_g^\top \mathbf{C}_*$. Since PCA is a linear mapping, the inverse mapping can be simply computed by $\hat{\mathbf{C}}_* = \tilde{\mathbf{C}}_* \mathbf{\Psi}$. The approximation error can then be computed by

$$\epsilon = \frac{\| \mathbf{C}_* - \hat{\mathbf{C}}_* \|_F^2}{\| \mathbf{C}_* \|_F^2}. \tag{4.31}$$

In Figure 4.7 (right), the trend reveals that including more components leads to a reduced error, even for unseen data. Furthermore, to investigate how the construction of the lower-dimensional subspace behaves with sample size variation, the error ϵ is shown for $N = 40$, $N = 416$ and $2N$ samples. It can be seen that the error is approximately the same for the latter two cases. As anticipated, an insufficient initial sample size N results in limited information availability during the subspace construction, consequently leading to a larger error. Moreover, the conclusion drawn is that even with $N = 416$ samples, sufficient data is available to attain a reasonable subspace. Furthermore, increasing the number of samples in the DoE does not contribute to higher accuracy. To mitigate this issue, the projection matrix is recalculated in every iteration of the optimisation process to incorporate as much data as possible.

Figure 4.8 (left) shows the results for the $108D$ aeroelastic tailoring problem, comparing the results of SCBO, (k)PCA-GP SCBO, Random Search and Covariance Matrix Adaptation Evolution Strategy (CMA-ES) (Hansen, 2006). Again, kPCA-GP SCBO uses the Gaussian kernel defined in Equation (4.27). A total of 5 experiments

are performed per method on a conventional computer with: Intel Xeon w3-2423, 6 cores, 32gb RAM. The original SCBO method crashes due to insufficient memory after the first iteration, while trying to construct 1786 high-dimensional GP surrogates. However, a good convergence can be observed for the PCA-GP SCBO and kPCA-GP SCBO, with $g = 35$, where again PCA performs better than kPCA. Additionally, it is important to note that given the size of the DoE $\mathcal{D}_0$ being $N = D$, a feasible design point can be efficiently identified, even if all points in the DoE at iteration $k = 0$ were initially infeasible. This is also highlighted by the results of the random search and CMA-ES which both fail in finding a feasible point.

Due to the high-dimensional design space and the high number of constraints, the probability of finding a feasible point where no constraints are violated is extremely low with random search, which was unable to find a single feasible design point. Therefore, the proposed method renders the observed advantage of finding efficiently feasible points even when $\mathcal{D}_0$ only contains infeasible ones. Figure 4.8 (right) illustrates the size of the TR over the number of model evaluations for three randomly chosen runs. It can be observed that the size generally decreases. However, as seen for instance in the dark blue curve, the optimiser occasionally gets stuck, increases the TR size to escape the locality while the evaluation budget is not exhausted, and then restarts to decrease it. Furthermore, for this specific example, we can alternatively perform a gradient-based optimisation for comparison. Using this approach, an objective value of $f^* = 402.06$ kg is obtained.

For the sake of completeness, we compare the proposed method to the so-called

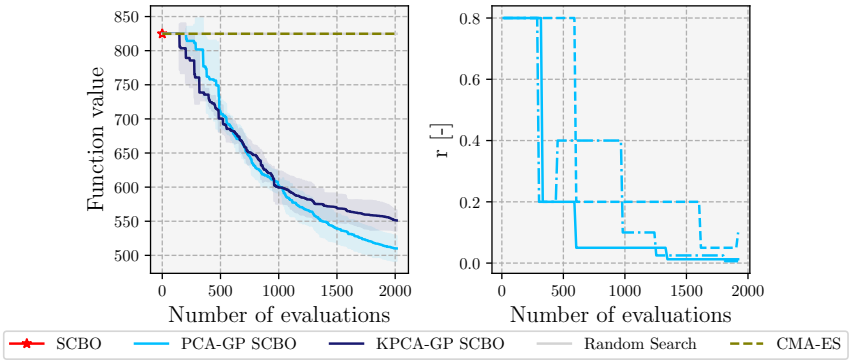


Figure 4.8: Optimisation results of aeroelastic tailoring case (**left**) and history of TR hyper-rectangle size L (**right**).

constraint aggregation approach, using the Kreisselmeier–Steinhauser (KS) function,

written as

$$KS(\mathbf{x}) = c_{max} + \frac{1}{\rho} \log \left[\sum_{j=1}^{m} e^{\rho c_j(\mathbf{x})} \right]. \tag{4.32}$$

This function aggregates multiple constraints, arising for example from a buckling or strength analysis into one constraint function. We implement this to lower the number of needed surrogates and compare the results against the best candidate so far. We aggregate the strain and buckling constraints for each loadcase individually for which we construct the GP, while the other constraints are modelled independently, leading to a reduced reduced number of constraints $g = 66$. It should be noted that, compared to PCA-GP SCBO/ kPCA-GP SCBO where $g = 35$ principal components were used, in the aggregation approach 66 surrogate models need to be constructed, needing approximately twice as long for surrogate construction. Thus, downsides are the increased number of needed surrogate models in high-dimensional space as well as the additional hyperparameters needed to define which constraints to aggregate as well as the hyperparameter ρ which we set in this case to $\rho = 100$. For more information the reader is referred to Martins and Poon (2005).

It should be pointed out that in the constraint aggregation case not only requires more GPs to be constructed, increasing the need for computational resources but also the structure of the constraints need to be known such that only constraints arising from one discipline are aggregated. This is additionally needed knowledge which might be not available, drastically lowering the generality of this approach. The corresponding results can be found in Figure 4.9 where we compare SCBO with the aggregation technique with PCA-GP SCBO.

Hypothesising why the aggregation method performed worse than the herein intro-

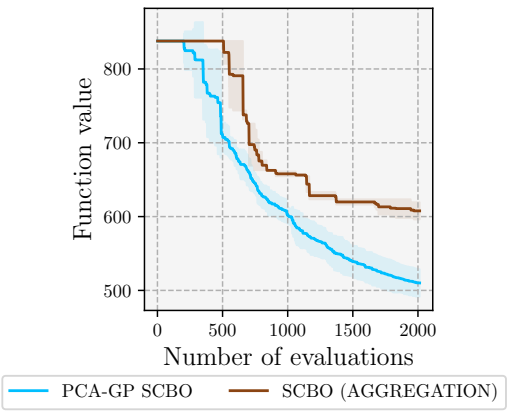


Figure 4.9: Comparison of best result with constraint aggregation.

duced approaches is first of all its conservativeness and second, the high-order of the output function due to approximating all the constraints, possibly leading to a quasi-non-smooth function which is cumbersome to approximate. However, further research has to be performed to confirm these statements.

4.5. Conclusion and Future Research

The aeroelastic tailoring problem exemplifies a high-dimensional multidisciplinary design optimisation challenge characterised by large-scale constraints. Conducting a global design space search is inherently complex, particularly when dealing with black-box optimisation problems where computing gradients is problematic. CBO faces scalability issues due to the extensive number of constraints involved. To mitigate the scalability shortcomings of the aforementioned methods, GPs are constructed on the latent space of the high-dimensional outputs in combination with TR-based approach. By significantly reducing the number of required GPs, substantial computational savings can be realised, making certain problems feasible and aligning with the objectives to reduce computational expenses. These savings are even more pronounced in high-dimensional settings where the training of each GP is critical.

Within aeroelastic tailoring, feasible designs can be found relatively easily by increasing the thickness of each panel. However, this simplicity does not extend to other problems. The presented approach demonstrates the capability to drastically reduce computational time, thus making SCBO feasible for such problems. Numerical investigations confirm the applicability of this method to aeroelastic tailoring, showcasing its effectiveness for multiple load cases with minimal additional principal components required.

An analytical example further illustrates that the proposed method converges to approximately the same objective function value. While our work primarily addresses aeroelastic tailoring, the method's generality allows for application to various problems involving large-scale constraints. This flexibility is supported by numerical evidence showing the ease of application to diverse high-dimensional constraint problems.

Additionally, any dimensionality reduction method, such as autoencoders, can be seamlessly integrated into the methodology. When compared to other methods for handling large-scale constraints, such as penalty and constraint aggregation methods, our proposed method demonstrates superior results without relying on specific knowledge about constraint categories. While the herein presented method works with a fixed user-defined or eigenvalue-based number of principal components g, a promising path could be an extension of this method, using an adapting number g such that the approximation error of the latent space is minimised. This might further improve the method. Moreover, future research will focus on simultaneously reducing input

and output spaces. Our current methodology requires training latent GPs on the full-dimensional input space, limiting the scalability. Approaches like REMBO (Wang et al., 2016), ALEBO (Letham et al., 2020) and (k)PCA-BO (Raponi et al., 2020, Antonov et al., 2022) offer promising avenues for further reducing computational costs during hyperparameter tuning. Simultaneously reducing input and output space would highly increase the scalability of this approach. Moreover, the efficient utilisation of gradients, if available, will be explored to combine gradient-based and surrogate approaches. This could facilitate the use of active subspaces, potentially enhancing performance.

Besides its application in BO, this research also holds promise for design under uncertainty. GPs offer a distinct advantage in providing a measure of variance. When addressing systems with high-dimensional outputs, this method becomes particularly advantageous. By leveraging PCA, we facilitate an efficient mapping back to the original high-dimensional space. This approach is particularly pertinent for engineering challenges where multiple model outputs are commonplace, offering a scalable solution for variability assessment. Furthermore, our method's potential application in a multi-fidelity optimisation strategy will be explored to bolster computational efficiency and practical feasibility.

Bibliography

M. A. Alvarez, L. Rosasco, and N. D. Lawrence. Kernels for Vector-Valued Functions: a Review. arXiv, Apr. 2012. URL http://arxiv.org/abs/1106.6251. arXiv:1106.6251 [cs, math, stat].

K. Antonov, E. Raponi, H. Wang, and C. Doerr. High dimensional bayesian optimization with kernel principal component analysis. 2022. doi: arXiv:2204.13753v2.

M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*, 2020. doi: https://doi.org/10.48550/arXiv.1910.06403.

M. Binois and N. Wycoff. A Survey on High-dimensional Gaussian Process Modeling with Application to Bayesian Optimization. *ACM Transactions on Evolutionary Learning and Optimization*, 2(2):1–26, June 2022. ISSN 2688-299X, 2688-3007. doi: 10.1145/3545611.

E. V. Bonilla, K. Chai, and C. Williams. Multi-task gaussian process prediction. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/66368270ffd51418ec58bd793f2d9b1b-Paper.pdf.

W. P. Bruinsma, E. Perim, W. Tebbutt, J. S. Hosking, A. Solin, and R. E. Turner. Scalable Exact Inference in Multi-Output Gaussian Processes, 2019. Version Number: 3.

I. M. Daniel and O. Ishai. *Engineering mechanics of composite materials*. Oxford University Press, New York, 2nd ed edition, 2006. ISBN 978-0-19-515097-1. OCLC: ocm57285865.

J. K. Dillinger, T. Klimmek, M. M. Abdalla, and Z. Gürdal. Stiffness Optimization of Composite Wings with Aeroelastic Constraints. *Journal of Aircraft*, 50(4):1159–1168, July 2013. ISSN 0021-8669, 1533-3868. doi: 10.2514/1.C032084.

D. Eriksson and M. Jankowiak. High-Dimensional Bayesian Optimization with Sparse Axis-Aligned Subspaces. June 2021. doi: https://doi.org/10.48550/arXiv.2103.00349. arXiv:2103.00349 [cs, stat].

D. Eriksson and M. Poloczek. Scalable Constrained Bayesian Optimization. Feb. 2021. doi: https://doi.org/10.48550/arXiv.2002.08526. arXiv:2002.08526 [cs, stat].

D. Eriksson, M. Pearce, J. R. Gardner, R. Turner, and M. Poloczek. Scalable Global Optimization via Local Bayesian Optimization. Feb. 2020. doi: https://doi.org/10.48550/arXiv.1910.01739. arXiv:1910.01739 [cs, stat].

P. I. Frazier. A Tutorial on Bayesian Optimization. July 2018. doi: https://doi.org/10.48550/arXiv.1807.02811. arXiv:1807.02811 [cs, math, stat].

J. R. Gardner, M. J. Kusner, and G. Jake. Bayesian Optimization with Inequality Constraints. 2014. URL `https://proceedings.mlr.press/v32/gardner14.html`.

J. R. Gardner, G. Pleiss, D. Bindel, K. Q. Weinberger, and A. G. Wilson. Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Advances in Neural Information Processing Systems*, 2018. doi: https://doi.org/10.48550/arXiv.1809.11165.

M. A. Gelbart, J. Snoek, and R. P. Adams. Bayesian Optimization with Unknown Constraints. 2014. doi: https://doi.org/10.48550/arXiv.1403.5607.

N. Hansen. The cma evolution strategy: A comparing review. 2006. URL `https://github.com/CMA-ES/pycma`.

J. M. Hernández-Lobato, M. A. Gelbart, R. P. Adams, M. W. Hoffman, and Z. Ghahramani. A General Framework for Constrained Bayesian Optimization using Information-based Search. Sept. 2016. doi: https://doi.org/10.48550/arXiv.1511.09422. arXiv:1511.09422 [stat].

J. M. Hernández-Lobato, J. Requeima, E. O. Pyzer-Knapp, and A. Aspuru-Guzik. Parallel and Distributed Thompson Sampling for Large-scale Accelerated Exploration of Chemical Space. 2017. doi: 10.48550/ARXIV.1706.01825. Publisher: arXiv Version Number: 1.

D. Higdon, J. Gattiker, B. Williams, and M. Rightley. Computer Model Calibration Using High-Dimensional Output. *Journal of the American Statistical Association*, 103(482):570–583, June 2008. ISSN 0162-1459, 1537-274X. doi: 10.1198/016214507000000888.

I. T. Jolliffe and J. Cadima. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202, Apr. 2016. ISSN 1364-503X, 1471-2962. doi: https://doi.org/10.1007/b98835.

K. Kandasamy, J. Schneider, and B. Poczos. High Dimensional Bayesian Optimisation and Bandits via Additive Models. May 2016. doi: https://doi.org/10.48550/arXiv.1503.01673. arXiv:1503.01673 [cs, stat].

A. Lemonge, H. Barbosa, C. Borges, and F. Silve. Constrained optimization problems in mechanical engineering design using a real-coded steady-state genetic algorithm. *Mecánica Computacional Vol XXIX*, pages 9287–9303, 2010.

B. Letham, R. Calandra, A. Rai, and E. Bakshy. Re-Examining Linear Embeddings for High-Dimensional Bayesian Optimization. Oct. 2020. doi: https://doi.org/10.48550/arXiv.2001.11659.

H. F. Maathuis, R. De Breuker, and S. G. Castro. High-Dimensional Bayesian Optimisation with Large-Scale Constraints - An Application to Aeroelastic Tailoring. In *AIAA SCITECH 2024 Forum*, Orlando, FL, Jan. 2024. American Institute of Aeronautics and Astronautics. ISBN 978-1-62410-711-5. doi: 10.2514/6.2024-2012.

W. J. Maddox, M. Balandat, A. G. Wilson, and E. Bakshy. Bayesian Optimization with High-Dimensional Outputs. Oct. 2021. doi: https://doi.org/10.48550/arXiv.2106.12997. arXiv:2106.12997 [cs, stat].

J. R. R. A. Martins and N. M. K. Poon. On structural optimization using constraint aggregation. In *Proceedings of the 6th World Congress on Structural and Multidisciplinary Optimization*, Rio de Janeiro, Brazil, May 2005.

J. Mockus. *Bayesian Approach to Global Optimization: Theory and Applications*. Springer Netherlands, Dordrecht, 1989. ISBN 978-94-009-0909-0. doi: https://doi.org/10.1007/978-94-009-0909-0. OCLC: 851374758.

Mockus, J., Tiesis, V., and Zilinskas, A. The Application of Bayesian Methods for Seeking the Extremum. pages 117–129, 1978.

R. Priem. *Optimisation bayésienne sous contraintes et en grande dimension appliquée à la conception avion avant projet*. PhD thesis, 2020.

E. Raponi, H. Wang, M. Bujny, S. Boria, and C. Doerr. High dimensional bayesian optimization assisted by principal component analysis. 2020. doi: arXiv:2007.00925v1.

C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass, 2006. ISBN 978-0-262-18253-9. OCLC: ocm61285753.

M. Santoni, E. Raponi, R. De Leone, and C. Doerr. Comparison of high-dimensional bayesian optimization algorithms on bbob. 2023. doi: arXiv:2303.00890v2.

P. Saves, N. Bartoli, Y. Diouane, T. Lefebvre, J. Morlier, C. David, E. Nguyen Van, and S. Defoort. Multidisciplinary design optimization with mixed categorical variables for aircraft design. In *AIAA SCITECH 2022 Forum*, San Diego, CA & Virtual, Jan. 2022. American Institute of Aeronautics and Astronautics. ISBN 978-1-62410-631-6. doi: 10.2514/6.2022-0082.

I. J. Schoenberg. Metric spaces and positive definite functions. *Transactions of the American Mathematical Society*, 44(3):522–536, 1938. ISSN 0002-9947, 1088-6850. doi: 10.1090/S0002-9947-1938-1501980-0.

B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, 10(5):1299–1319, July 1998. ISSN 0899-7667, 1530-888X. doi: 10.1162/089976698300017467.

M. H. Shirk, T. J. Hertz, and T. A. Weisshaar. Aeroelastic tailoring - Theory, practice, and promise. *Journal of Aircraft*, 23(1):6–18, Jan. 1986. ISSN 0021-8669, 1533-3868. doi: 10.2514/3.45260.

W. R. Thompson. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika*, 25(3/4):285, Dec. 1933. ISSN 00063444. doi: 10.2307/2332286.

S. W. Tsai and N. J. Pagano. Invariant properties of composite materials, ad668761. Technical report, Air Force Materials Laboratory, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio, 1968.

Z. Wang, F. Hutter, M. Zoghi, D. Matheson, and N. de Freitas. Bayesian Optimization in a Billion Dimensions via Random Embeddings. Jan. 2016. doi: https://doi.org/10.48550/arXiv.1301.1942. arXiv:1301.1942 [cs, stat].

N. P. Werter. *Aeroelastic Modelling and Design of Aeroelastically Tailored and Morphing Wings*. PhD thesis, Delft University of Technology, 2017.

W. Xing, A. Shah, and P. Nair. Reduced dimensional Gaussian process emulators of parametrized partial differential equations based on Isomap. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2174): 20140697, Feb. 2015. ISSN 1364-5021, 1471-2946. doi: 10.1098/rspa.2014.0697.

W. Xing, V. Triantafyllidis, A. Shah, P. Nair, and N. Zabaras. Manifold learning for the emulation of spatial fields from computational models. *Journal of Computational Physics*, 326:666–690, Dec. 2016. ISSN 00219991. doi: 10.1016/j.jcp.2016.07.040.

S. Zhe, W. Xing, and R. M. Kirby. Scalable high-order gaussian process regression. 89:2611–2620, 16–18 Apr 2019. URL `https://proceedings.mlr.press/v89/zhe19a.html`.

J. Ziomek and H. Bou-Ammar. Are Random Decompositions all we need in High Dimensional Bayesian Optimisation? Jan. 2023. doi: https://doi.org/10.48550/arXiv.2301.12844. arXiv:2301.12844 [cs, stat].

5

Autoencoder-enhanced Joint Input-Output Dimensionality Reduction for Constrained Bayesian Optimisation

This chapter is based on the following publication and has been reproduced with minor adjustments to notation and formatting for consistency within the thesis. Maathuis, H., & De Breuker, R. & Castro, S.G.P. (2025); Autoencoder-enhanced Joint Input-Output Dimensionality Reduction for Constrained Bayesian Optimisation. IOP Machine Learning Science and Technology, DOI 10.1088/2632-2153/ae0efe.

Abstract BO is a sample-efficient method for optimising expensive black-box functions, making it particularly suitable for engineering problems where gradients are unavailable and evaluating the objective or constraints is computationally costly. However, such problems often involve high-dimensional inputs and a large number of constraints, posing significant challenges for standard BO frameworks. While prior research has addressed scalability with respect to high-dimensional inputs in constrained settings, efficiently handling large numbers of constraints, i.e. high-dimensional outputs, remains an open problem. This work introduces Autoencoder-Enhanced Joint Dimensionality Reduction for Constrained Bayesian Optimisation (AERO-BO), a framework that performs dimensionality reduction in both the input

(design variable) and output (objective and constraint) spaces via autoencoders. These autoencoders are trained online, requiring no pre-training, and their respective latent representations are connected through GPs, which serve as surrogate models during optimisation. By operating in a joint latent space, AERO-BO enables scalable and efficient optimisation in settings with hundreds of design variables and thousands of black-box constraints.

5.1. Introduction

Engineering problems often aim to optimise performance, cost, and other objectives. In many cases, it is not straightforward to explore the global design space to identify the best combination of parameters. These challenges arise, for example, in the design of aerospace structures, where numerous constraints must be satisfied to ensure stability (Maathuis et al., 2024b), or in drug design, where factors such as synthesisability and the compounds toxicity may impose additional restrictions (Heifetz, 2024). However, many of those problems involve complex models where obtaining analytical or numerical gradients is impractical or impossible. The objective function and constraints in such problems are not only computationally expensive to evaluate but may also be noisy, further complicating the optimisation process. These challenges, often encountered in engineering disciplines, give rise to so-called black-box problems, where only the input-output relationship of the model can be observed. BO has emerged as a powerful and efficient method for addressing this challenge. It leverages a probabilistic model, frequently a GP, to approximate the objective function and constraints and intelligently guide the search for optimal solutions. By balancing exploration of the search space and exploitation of promising regions, BO is particularly well-suited for problems where function evaluations are costly. However, the aforementioned engineering problems are frequently characterised by a high number of decision variables (inputs), as well as often involving thousands of constraints (outputs), incorporating diverse disciplines to analyse and ensure feasibility. This high dimensionality of the input and output space renders traditional optimisation methods computationally prohibitive. The problem at hand can be formulated as

$$\begin{aligned} \min_{\mathbf{x}\in\mathcal{X}\subseteq\mathbb{R}^D} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & c_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, G \end{aligned} \tag{5.1}$$

with $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^D$ denoting a design point in the input space, $f : \mathcal{X} \to \mathbb{R}$ being the objective function mapping from the input space to a scalar value and $\mathbf{c} : \mathcal{X} \to \mathcal{Y} \subseteq \mathbb{R}^G$ being a collection of constraints $\mathbf{c}(\mathbf{x}) \in \mathbb{R}^G$. The scalability of the BO methodology for these types of problems remains limited.

Despite the fact that high-dimensional BO in unconstrained settings is already challenging to optimise, Eriksson and Poloczek (2021) propose SCBO, a promising method for efficiently handling high-dimensional and constrained optimisation problems. However, problems with possibly hundreds of design variables and thousands

of constraints remain prohibitive due to computational resources. This limitation arises because SCBO builds the probabilistic surrogate models in the full input space and requires a separate surrogate model for every constraint.

To address these computational challenges, this work aims to jointly reduce the dimensionality of both. The goal is to construct a joint latent space, or in other words a space of reduced dimensionality, in which the probabilistic models for the objective and the constraints are built and learned online directly from the acquired data. In this context, we propose a novel framework called AERO-BO, which integrates autoencoders for dimensionality reduction in both input and output spaces. Autoencoders are neural networks designed to learn low-dimensional representations of high-dimensional data in an unsupervised manner. By combining autoencoders with GPs, we aim to develop a scalable and efficient optimisation framework that is well-suited for high-dimensional input-output problems.

Our main contributions are:

1. Introducing AERO-BO for high-dimensional input-output problems,
2. Demonstrating performance and scalability on benchmark cases,
3. Applying AERO-BO on a multi-disciplinary real-world design problem from aerospace engineering.

Structure of this chapter. The remainder of this chapter is structured as follows. First, we introduce the theoretical fundamentals and review the relevant literature. Next, we define autoencoders for dimensionality reduction. This is followed by the presentation of AERO-BO and an evaluation of its performance against a selection of existing methods. Finally, we conclude with a summary and discussion of the findings.

5.2. Constrained Bayesian Optimisation via Gaussian Processes

Consider the constrained optimisation problem in Equation 5.1. The optimal solution $\mathbf{x}^* \in \mathcal{X}_f \subseteq \mathcal{X}$ lies within the feasible space $\mathcal{X}_f$ defined by the constraints:

$$\mathcal{X}_f = \{\mathbf{x} \in \mathcal{X} \mid c_i(\mathbf{x}) \leq 0,\, i = 1, \ldots, G\}. \tag{5.2}$$

In many practical applications, evaluating $f(\mathbf{x})$ and especially $c_i(\mathbf{x})\ \forall\ i = 1, ..., G$ can be computationally expensive or analytically intractable. BO, firstly introduced in Kushner (1962, 1964), addresses this by using probabilistic surrogate models to approximate the objective and constraints, allowing for efficient exploration and exploitation of the design space. GPs are frequently employed due to their flexibility and ability to provide uncertainty estimates (Frazier, 2018).

Commonly, in constrained Bayesian optimisation, separate GPs are used to model the objective function and each constraint function. Let $\mathcal{D}_t = \{(\mathbf{x}_j, f_j, \mathbf{c}_j)\}_{j=1}^{N_t}$ be the dataset of N_t observations. Based on this data set we further define the observation vector $\mathbf{f} = [f_1, \ldots, f_{N_t}]^\top$ or for the i-th constraint $\mathbf{c}_i = [c_{i,1}, \ldots, c_{i,N_t}]^\top$ and the corresponding input matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_{N_t}]^\top$. The GP model is defined by a mean function $\mu : \mathcal{X} \to \mathbb{R}$ and a covariance (kernel) function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, written as

$$f(\mathbf{x})|\mathcal{D} \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \tag{5.3}$$

Similarly, each constraint $c_i(\mathbf{x})$ is modelled using a separate GP:

$$c_i(\mathbf{x})|\mathcal{D} \sim \mathcal{GP}(\mu_i(\mathbf{x}), k_i(\mathbf{x}, \mathbf{x}')) \ \forall i = 1, ..., G. \tag{5.4}$$

The models are trained by optimising the marginal likelihood (Rasmussen and Williams, 2006), written as

$$\log p(\mathbf{f} \mid \mathbf{X}, \boldsymbol{\theta}) = -\frac{1}{2}\mathbf{f}^\top \mathbf{K}^{-1}\mathbf{f} - \frac{1}{2}\log|\mathbf{K}| - \frac{N}{2}\log 2\pi, \tag{5.5}$$

with $\boldsymbol{\theta}$ being some trainable hyperparameters. This expression penalises both poor data fit and high model complexity, preventing overfitting. After training the model, the posterior distributions for both the objective and each constraint at a new query point $\mathbf{x}_+ \in \mathcal{X}$ remain Gaussian:

$$\begin{aligned} f(\mathbf{x}_+)|\mathcal{D}, \mathbf{x}_+ &\sim \mathcal{N}(\mu(\mathbf{x}_+), \sigma^2(\mathbf{x}_+)) \\ c_i(\mathbf{x}_+)|\mathcal{D}, \mathbf{x}_+ &\sim \mathcal{N}(\mu_i(\mathbf{x}_+), \sigma_i^2(\mathbf{x}_+)) \ \forall i = 1, \ldots, G. \end{aligned} \tag{5.6}$$

The posterior predictive distribution of a GP is analytically tractable due to the conjugacy of the Gaussian prior and likelihood. Common choices for the kernel function include the RBF, Matérn, and rational quadratic kernels, each parametrised by hyperparameters $\boldsymbol{\theta}$. Then, for any new query point $\mathbf{x}_+$, the GP predictive posterior mean $\mu(\mathbf{x}_+)$ and variance $\sigma^2(\mathbf{x}_+)$ are given by:

$$\begin{aligned} \mu(\mathbf{x}_+) &= k(\mathbf{x}_+, \mathbf{X})\mathbf{K}^{-1}\mathbf{y}, \\ \sigma^2(\mathbf{x}_+) &= k(\mathbf{x}_+, \mathbf{x}_+) - k(\mathbf{x}_+, \mathbf{X})\mathbf{K}^{-1}k(\mathbf{X}, \mathbf{x}_+), \end{aligned} \tag{5.7}$$

where $\mathbf{K} \in \mathbb{R}^{N_t \times N_t}$ is the kernel matrix with entries $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ and $k(\mathbf{x}_+, \mathbf{X}) \in \mathbb{R}^N$ is the covariance between the new point and the training data.

Next, an acquisition function $\alpha(\mathbf{x}; \mathcal{D}) : \mathcal{X} \to \mathbb{R}$ makes use of the probabilistic surrogate model by encoding a utility policy to guide the selection of the next query point $\mathbf{x}_+$, balancing exploration and exploitation. For CBO, the acquisition function needs to account for both the objective and the constraints, thus trying to find a next feasible query point:

$$\mathbf{x}_+ = \underset{\mathbf{x} \in \mathcal{X}_f}{\operatorname{argmax}}\, \alpha(\mathbf{x}; \mathcal{D}). \tag{5.8}$$

Examples of constrained acquisition functions, such as CEI (Gardner et al., 2014, Gelbart et al., 2014) and their logarithmic extension LogCEI (Ament et al., 2023), PESC (Hernández-Lobato et al., 2016) or CTS, as used in SCBO (Eriksson and Poloczek, 2021). These methods incorporate probabilistic estimates from GPs to guide the search towards regions that improve the objective while satisfying the constraints. By iteratively updating the GP models, constrained Bayesian optimisation efficiently explores the design space.

However, high-dimensional input and output spaces introduce significant challenges, primarily due to the curse of dimensionality and storage limitations. As noted in Equation 5.6, the objective and all constraints need to be modelled via a separate or correlated GP. When dealing with potentially thousands of outputs, this becomes computationally infeasible, as GPs scale cubically with the number of sample points $\mathcal{O}(N^3)$ and storage requirements of $\mathcal{O}(N^2)$. Since high-dimensional problems normally require hundreds up to thousands of samples, the development of alternative approaches becomes necessary, which are discussed in the following section.

5.3. Bayesian Optimisation in High Dimensions

This section reviews recent developments in BO for tackling problems with high-dimensional inputs and outputs, for unconstrained and constrained settings.

5.3.1. Bayesian Optimisation with High-Dimensional Inputs

Scaling BO to high-dimensional input problems poses three main challenges: increased predictive uncertainty, more model hyperparameters, and computational difficulty in optimising the acquisition function (Binois and Wycoff, 2022).

A prominent line of research to mitigate these problems focuses on reducing the dimensionality of the input space. Linear projection methods such as REMBO (Wang et al., 2016), ALEBO (Letham et al., 2020), and HeSBO (Nayebi et al., 2019) assume the objective varies in a lower-dimensional subspace. These methods rely on random or adaptive linear projections to restrict the search domain. Given a random matrix $\mathbf{A} \in \mathbb{R}^{d \times D}$, optimisation is performed in the subspace $\tilde{\mathcal{X}} \subseteq \mathbb{R}^d$, where $d \ll D$, with the function approximated as $g(\tilde{\mathbf{x}}) \approx f(\mathbf{A}\mathbf{x})$.

More recent work has explored nonlinear dimensionality reduction techniques in the input space. Notably, autoencoder-based methods (Gómez-Bombarelli et al., 2018, Tripp et al., 2020, Grosnit et al., 2021, Maus et al., 2023) have been proposed, leveraging their ability to learn more flexible, data-adaptive representations of the optimisation space. For example, Gómez-Bombarelli et al. (2018) pioneered latent space optimisation for molecular design, using a variational autoencoder to encode discrete molecular graphs into a continuous latent manifold. Tripp et al. (2020) and Grosnit et al. (2021) extended this idea to more general structured and high-dimensional inputs, demonstrating improved sample efficiency and optimisation in

learned latent spaces. Maus et al. (2023) further propose the use of encoders to reduce the dimensionality of high-dimensional intermediate components in composite functions, improving the efficiency in grey-box BO. However, these methods have primarily addressed unconstrained optimisation tasks and focus exclusively on input space compression. They do not account for high-dimensional constraint outputs or incorporate feasibility modelling into the latent optimisation loop.

SAASBO (Eriksson and Jankowiak, 2021) takes a different approach, using a sparsity-inducing prior over GP length scales to identify relevant dimensions, gradually expanding the subspace as more data becomes available. Although effective, this approach incurs significant overhead. Trust region-based methods such as TuRBO (Eriksson et al., 2019) and BAxUS (Papenmeier et al., 2023) limit the search to a local region, balancing exploration and exploitation. BAxUS combines this with gradually expanding subspaces. More recently, Hvarfner et al. (2024) demonstrated that standard GPs with scaled length-scale priors can perform well in high-dimensional settings without embedding.

In constrained settings, SCBO (Eriksson and Poloczek, 2021) extends TuRBO by using TRs and CTS, training one GP per constraint. Similarly, VBO with logLogCEI (Ament et al., 2023) provides a scalable baseline but also requires a separate GP for each constraint, limiting applicability in problems with many constraints. Section 5.4 details how AERO-BO addresses these issues through joint input-output dimensionality reduction.

5.3.2. Bayesian Optimisation with High-Dimensional Outputs

When multiple outputs (e.g., objectives and constraints) are involved, a straightforward approach is to model each independently using batched GPs, as in SCBO. However, modelling cross-output correlations can yield performance improvements.

MTGP (Bonilla et al., 2007) model output dependencies via structured covariance matrices using the Intrinsic or Linear Co-regionalisation Model (ICM/LMC). When all outputs are observed at all points, the full covariance adopts a Kronecker structure $\mathbf{K}_{XX} \otimes \mathbf{K}_f \in \mathbb{R}^{NG \times NG}$, enabling some computational savings. Nevertheless, their inference and memory complexity $\mathcal{O}(N^3G^3)$ and $\mathcal{O}(N^2G^2)$, respectively remains prohibitive for large-scale problems. HOGPs (Zhe et al., 2019) extend GPs to matrix and tensor outputs. Maddox et al. (2021) improved sampling efficiency using Matheron's rule, reducing complexity to $\mathcal{O}(N^3 + G^3)$ in the MTGP case, but scalability remains limited.

Another direction is output space dimensionality reduction. Higdon et al. (2008) proposed using PCA to project outputs onto a lower-dimensional space, where GPs are trained. Variants include kPCA-GP and IsoMap-GP (Xing et al., 2015, 2016).

These reduce inference costs, but rely on linear (or fixed nonlinear) embeddings. Maathuis et al. (2024b) combine PCA-GP with SCBO to handle many constraints efficiently. However, these methods assume fixed embeddings and are not learned adaptively during BO.

AERO-BO builds upon this line of work by jointly learning input and output embeddings via autoencoders during optimisation. This enables tractable modelling of high-dimensional constraints and scalable acquisition in joint latent spaces.

5.4. AERO-BO: Constrained Bayesian Optimisation in a Joint Input-Output Latent Space

As outlined earlier, only a few methods have been proposed for high-dimensional CBO. These include SCBO and standard BO with scaled length-scale priors and logCEI (Hvarfner et al., 2024), here referred to as VBO. Both approaches construct one surrogate model per constraint in addition to the objective. However, most high-dimensional BO methods reviewed in Section 5.3 target unconstrained problems. When constraints are considered, they are often handled using soft formulations such as penalty terms, as in BAxUS (Papenmeier et al., 2023) and SAASBO (Eriksson and Jankowiak, 2021). BAxUS, for instance, shows that combining subspace modelling with TR strategies improves scalability. Nevertheless, extending random subspace methods to constrained settings remains challenging: as the number of constraints and the dimensionality grow, the likelihood of a randomly projected subspace containing feasible points declines significantly, limiting their applicability. In such cases, additional supervision during subspace construction may be required.

In parallel, dimensionality reduction using autoencoders has gained traction for addressing the curse of dimensionality in BO. Yet, prior work has focused exclusively on unconstrained problems and compressed only the input space. Reducing the dimensionality of the output space, particularly in the presence of numerous constraints, remains underexplored, despite its potential to yield substantial computational savings. We address this gap by employing autoencoders to reduce both the input and output dimensions, enabling BO for problems with hundreds of design variables and thousands of constraints.

This is achieved using two autoencoders, see Section 5.4.1: one for the inputs and one for the outputs. Each maps the high-dimensional data into a lower-dimensional latent representation that retains essential structural information. GPs are trained to model the relationships between the latent input and output spaces, as illustrated in Figure 5.1. Following Maus et al. (2024), the autoencoders and GPs are trained jointly to ensure that the latent representations remain coherent and aligned with the optimisation task. This joint training allows the latent spaces to adapt dynamically based on surrogate model feedback. The decoders are then used to project the

optimised latent variables back into the original high-dimensional space for training purposes.

Additionally, AERO-BO trains both autoencoders in an online manner during the optimisation process, relying exclusively on data gathered within the BO loop. This eliminates the need for a costly offline pre-training phase, which may be impractical when function evaluations are expensive or data is limited. The online training scheme is a further novelty of our approach, enabling the latent representations to adapt in response to the evolving optimisation landscape. While autoencoders have been used previously in unconstrained BO, AERO-BO is, to our knowledge, the first to extend this concept to constrained high-dimensional problems by jointly learning latent representations for both design variables and constraint responses. This allows BO to scale to problem sizes that are otherwise intractable with existing methods. In the following, we briefly introduce autoencoders before presenting our proposed method in detail.

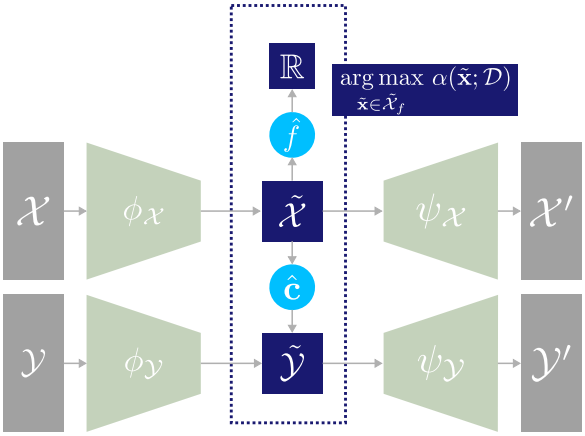


Figure 5.1: AERO-BO Architecture: Two autoencoders map the high-dimensional input and constraint output spaces to corresponding latent spaces which are connected via GPs. These latent space surrogates are then used within the acquisition of the next points.

5.4.1. Manifold-learning via Autoencoders

This section briefly introduces autoencoders (Rumelhart et al., 1986) as a manifold learning technique for nonlinear dimensionality reduction. Unlike PCA, which is limited to linear transformations via eigendecomposition, autoencoders are capable of capturing nonlinear data structures by learning mappings from the high-dimensional input space to a lower-dimensional latent space. This enables the representation of complex data manifolds that PCA may fail to model.

An autoencoder $\mathcal{A}(\mathbf{x}) = \psi \circ \phi(\mathbf{x})$ consists of two components: an encoder $\phi : \mathbb{R}^K \to \mathbb{R}^k$

that maps the input $\mathbf{x} \in \mathbb{R}^K$ to a lower-dimensional latent representation $\tilde{\mathbf{x}} \in \mathbb{R}^k$, and a decoder $\psi : \mathbb{R}^k \to \mathbb{R}^K$ that reconstructs an approximation of the original input. The input dimensionality K is determined by the data, while the latent dimension $k \ll K$ is a user-defined parameter. Throughout this work, both encoder and decoder are implemented as single-layer feedforward neural networks. The encoder applies a linear transformation followed by a Rectified Linear Unit (ReLU) activation:

$$\phi(\mathbf{x}; \boldsymbol{\theta}_\phi) = \text{ReLU}(\mathbf{W}_\phi \mathbf{x} + \mathbf{b}_\phi), \tag{5.9}$$

where $\boldsymbol{\theta}_\phi = \{\mathbf{W}_\phi, \mathbf{b}_\phi\}$ with $\mathbf{W}_\phi \in \mathbb{R}^{k \times K}$, $\mathbf{b}_\phi \in \mathbb{R}^k$ and $\text{ReLU}(\cdot) : \mathbb{R} \to \mathbb{R}$ denotes the element-wise rectified linear unit defined by $\text{ReLU}(z) = \max(0, z)$. The decoder performs a similar transformation using a sigmoid activation to constrain outputs to $[0, 1]^K$:

$$\psi(\tilde{\mathbf{x}}; \boldsymbol{\theta}_\psi) = \sigma(\mathbf{W}_\psi \tilde{\mathbf{x}} + \mathbf{b}_\psi), \tag{5.10}$$

with $\boldsymbol{\theta}_\psi = \{\mathbf{W}_\psi, \mathbf{b}_\psi\}$, where $\mathbf{W}_\psi \in \mathbb{R}^{K \times k}$, $\mathbf{b}_\psi \in \mathbb{R}^K$ and $\sigma(\cdot) : \mathbb{R} \to \mathbb{R}$ is the sigmoid activation function applied element-wise, defined as $\sigma(z) = [1+\exp(-z)]^{-1}$. Together, these network components define a parametrised nonlinear mapping $\mathcal{A}(\mathbf{x}) = \psi \circ \phi(\mathbf{x})$ from the input space to an approximation of itself. The autoencoder is trained to minimise the reconstruction loss:

$$\mathcal{L}(\mathbf{x}; \boldsymbol{\theta}_\phi, \boldsymbol{\theta}_\psi) = ||\mathbf{x} - \psi(\phi(\mathbf{x}; \boldsymbol{\theta}_\phi); \boldsymbol{\theta}_\psi)||^2, \tag{5.11}$$

using stochastic gradient-based optimisation. This results in a compact representation that preserves the essential structure of the input data. Once trained, the encoder ϕ serves as a dimensionality reduction tool, efficiently mapping new samples $\mathbf{x}_*$ to their corresponding latent representations $\tilde{\mathbf{x}}_* = \phi(\mathbf{x}_*; \boldsymbol{\theta}_\phi)$.

5.4.2. AERO-BO: Architecture

As previously mentioned, we employ two autoencoders to efficiently reduce the dimensionality of the input $\mathcal{X}$ and output $\mathcal{Y}$ spaces, respectively, defined as $\mathcal{A}_\mathcal{X}(\mathbf{x}) = \psi_\mathcal{X} \circ \phi_\mathcal{X}(\mathbf{x})$ and $\mathcal{A}_\mathcal{Y}(\mathbf{y}) = \psi_\mathcal{Y} \circ \phi_\mathcal{Y}(\mathbf{y})$. The encoders map data to lower-dimensional latent spaces, $\phi_\mathcal{X} : \mathcal{X} \subseteq \mathbb{R}^D \to \tilde{\mathcal{X}} \subseteq \mathbb{R}^d$ and $\phi_\mathcal{Y} : \mathcal{Y} \subseteq \mathbb{R}^G \to \tilde{\mathcal{Y}} \subseteq \mathbb{R}^g$, while the decoders reconstruct approximations of the original data:

$$\begin{aligned} \tilde{\mathbf{x}} &= \phi_\mathcal{X}(\mathbf{x}; \boldsymbol{\theta}_{\phi,\mathcal{X}}), \quad \tilde{\mathbf{y}} = \phi_\mathcal{Y}(\mathbf{y}; \boldsymbol{\theta}_{\phi,\mathcal{Y}}) \\ \mathbf{x}' &= \psi_\mathcal{X}(\tilde{\mathbf{x}}; \boldsymbol{\theta}_{\psi,\mathcal{X}}), \quad \mathbf{y}' = \psi_\mathcal{Y}(\tilde{\mathbf{y}}; \boldsymbol{\theta}_{\psi,\mathcal{Y}}). \end{aligned} \tag{5.12}$$

We denote the reconstructed data as $\mathbf{x}'$ and $\mathbf{y}'$, respectively, acknowledging that reconstruction may introduce an error, which the training process aims to minimise. The trainable parameters, $\boldsymbol{\theta}_{\phi,\mathcal{X}}, \boldsymbol{\theta}_{\phi,\mathcal{Y}}, \boldsymbol{\theta}_{\psi,\mathcal{X}}, \boldsymbol{\theta}_{\psi,\mathcal{Y}}$, govern the encoding and decoding transformations, ensuring optimal representation learning. When trained independently, these models remain decoupled. To address this, we adopt the approach proposed by Maus et al. (2024) to couple them through variational GPs, see Section 2.2.2, modelling the constraints in this joint latent space and train all

models together. Specifically, we construct an approximate GP (Hensman et al., 2014) for the objective function f, mapping from the latent input space $\tilde{\mathcal{X}}$ to a scalar value $\hat{f} : \tilde{\mathcal{X}} \to \mathbb{R}$. Additionally, we construct independent approximate GPs for the constraints $\mathbf{c}$, mapping from the latent input space to the latent output space $\hat{\mathbf{c}} : \tilde{\mathcal{X}} \to \tilde{\mathcal{Y}}$. In total, this results in $n_m = 2 + 1 + g$ interconnected models: two autoencoders, one objective GP and g GPs for the latent space constraints.

These relationships can be expressed as:

$$\begin{aligned} \tilde{f} \mid \tilde{\mathbf{X}} &\sim \mathcal{GP}(\mu(\tilde{\mathbf{X}}), k(\tilde{\mathbf{X}}, \tilde{\mathbf{X}})) \\ \tilde{c}_i \mid \tilde{\mathbf{X}} &\sim \mathcal{GP}(\mu_i(\tilde{\mathbf{X}}), k_i(\tilde{\mathbf{X}}, \tilde{\mathbf{X}})) \ \forall i = 1, \ldots, g. \end{aligned} \tag{5.13}$$

The n_m models are trained by minimising the joint loss:

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}) = &\log p(f \mid \phi_{\mathcal{X}}(\mathbf{x}; \boldsymbol{\theta}_{\phi,\mathcal{X}}), \boldsymbol{\theta}_f) + \log p(\phi_{\mathcal{Y}}(\mathbf{c}; \boldsymbol{\theta}_{\phi,\mathcal{Y}}) \mid \phi_{\mathcal{X}}(\mathbf{x}; \boldsymbol{\theta}_{\phi,\mathcal{X}}), \boldsymbol{\theta}_{c_i}) \\ &+ ||\mathbf{x} - \psi_{\mathcal{X}}(\phi_{\mathcal{X}}(\mathbf{x}; \boldsymbol{\theta}_{\phi,\mathcal{X}}); \boldsymbol{\theta}_{\psi,\mathcal{X}})||^2 + ||\mathbf{c} - \psi_{\mathcal{Y}}(\phi_{\mathcal{Y}}(\mathbf{c}; \boldsymbol{\theta}_{\phi,\mathcal{Y}}); \boldsymbol{\theta}_{\psi,\mathcal{Y}})||^2 \end{aligned} \tag{5.14}$$

where the terms ensure the accurate reconstruction of inputs and outputs while maintaining the coherence of GP predictions. Here, $\boldsymbol{\theta}$ represents not only the aforementioned hyperparameters of the autoencoders but also those of the GPs, such that $\boldsymbol{\theta} = \{\boldsymbol{\theta}_f, \boldsymbol{\theta}_{c_i}, \boldsymbol{\theta}_{\phi,\mathcal{X}}, \boldsymbol{\theta}_{\psi,\mathcal{X}}, \boldsymbol{\theta}_{\phi,\mathcal{Y}}, \boldsymbol{\theta}_{\psi,\mathcal{Y}}\}$. By leveraging automatic differentiation alongside the Adam optimiser (Kingma and Ba, 2017), we efficiently train these interconnected models. We embed this modelling strategy into a TR heuristic, akin to SCBO (Eriksson and Poloczek, 2021). The proposed method is summarised in Algorithm 6.

During the acquisition strategy we first sample from each latent space model's

Algorithm 6 AERO-BO

Require: Input space $\mathcal{X}$, Number of initial samples N, Number of candidates N_c, batch size q_c, SCBO hyperparameters

1: Compute initial DoE $\mathcal{D}_0 = \{\mathbf{x}_i, f(\mathbf{x}_i), \mathbf{c}(\mathbf{x}_i)\}_{i=1:N}$
2: Initialise SCBO state
3: Initialise models $\tilde{f}, \tilde{\boldsymbol{c}}, \mathcal{A}_{\mathcal{X}}$ and $\mathcal{A}_{\mathcal{Y}}$
4: $t = 0$
5: **while** Computational budget is not exhausted **do**
6: $\quad \mathbf{x}_+ \leftarrow$ AcquisitionStrategy (see Algorithm 7)
7: $\quad$ Evaluate $\mathbf{x}_+$ and observe $f(\mathbf{x}_+)$, $\mathbf{c}(\mathbf{x}_+)$
8: $\quad \mathcal{D}_{k+1} = \mathcal{D}_k \cup \{\mathbf{x}_+, f(\mathbf{x}_+), \mathbf{c}(\mathbf{x}_+)\}$
9: $\quad$ Update SCBO state
10: $\quad$ Update models $\hat{f}, \hat{\boldsymbol{c}}, \mathcal{A}_{\mathcal{X}}, \mathcal{A}_{\mathcal{Y}}$ jointly (see Eq. 5.14)
11: $\quad t \leftarrow t + 1$
12: **end while**

posterior to obtain $\hat{\tilde{f}}, \hat{\tilde{c}}_1, ..., \hat{\tilde{c}}_g$. Subsequently, a set of randomly sampled candidate points $\mathbf{X}_c$ is mapped to the latent space $\tilde{\mathbf{X}}_c \in \tilde{\mathcal{X}}$ using the encoder $\phi_{\mathcal{X}}$ which allows us to conduct constrained Thompson Sampling within the joint latent space, employing the aforementioned latent space GPs (Equation 5.13) for the objective and constraints. Therein, we use an utility function $\alpha(\tilde{\mathbf{x}})$, written as

$$\alpha(\tilde{\mathbf{x}}; \mathcal{D}) = \begin{cases} \text{argmin}_{\tilde{\mathbf{x}} \in \tilde{\mathcal{X}}} \hat{\tilde{f}}(\tilde{\mathbf{x}}) & \text{if } \mathcal{F} \neq \emptyset \\ \text{argmin}_{\tilde{\mathbf{x}} \in \tilde{\mathcal{X}}} \sum_{j=1}^{g} \max\{\hat{\tilde{c}}_j(\tilde{\mathbf{x}}), 0\} & \text{else} \end{cases} \tag{5.15}$$

with the set of feasible points, defined as $\mathcal{F} = \{\tilde{\mathbf{x}}_i \mid \hat{\tilde{c}}_j(\tilde{\mathbf{x}}_i) \leq 0,\ j = 1, ..., g\}$. By solving

$$\tilde{\mathbf{x}}_+ = \underset{\tilde{\mathbf{x}} \in \tilde{\mathcal{X}}_f}{\text{argmin}}\ \alpha(\tilde{\mathbf{x}}, \mathcal{D}) \tag{5.16}$$

a batch of q points in the latent space $\tilde{\mathcal{X}}$ can be obtained. Instead of using the decoder to map the points back, the indices are used to select the corresponding points in the original space to ensure they lie within the bounds (Maus et al., 2024).

Summarising, from Equations 5.16 and 6.29 it can be seen that the constrained Thompson sampling (Thompson, 1933, Eriksson and Poloczek, 2021) is performed in the latent space, making use of the encoded inputs $\tilde{\mathbf{x}}_i$ and outputs $\tilde{\mathbf{c}}_i$. This heuristic is summarised in Algorithm 7. During the first iteration of the BO algorithm, when

Algorithm 7 AcquisitionStrategy in AERO-BO

Require: Input space $\mathcal{X}$, Number of candidates N_c, batch size q_c, acquisition function $\alpha(\bullet)$, Samples form the GP posteriors in latent space $(\hat{\tilde{f}}, \hat{\tilde{c}}_1, \cdots, \hat{\tilde{c}}_g)$, autoencoder models $\mathcal{A}_{\mathcal{X}}, \mathcal{A}_{\mathcal{Y}}$

1: Generate N_c candidate $\mathbf{X}_c \in \mathbb{R}^{D \times N_c}$ with $\mathbf{x}_c^i \in \mathcal{X}_{tr}$
2: Map candidates into latent space $\tilde{\mathbf{X}}_c = \phi_{\mathcal{X}}(\mathbf{X}_c)$
3: Construct acquisition function $\tilde{\mathbf{A}} = \alpha(\tilde{\mathbf{X}}_c; \mathcal{D}_t)$ (see Equation 6.29)
4: Choose the q next points $\mathbf{X}_+ \leftarrow \mathbf{X}_c[\text{argmin}\, \tilde{\mathbf{A}}]$

$t = 0$, all four models are initialised and trained, then continuously updated after each batch of q new points is acquired. The TR in this process acts to restrain the N_c candidate points locally by using a hyperrectangle. Thus, if we denote $\mathcal{X}_{tr}$ as the TR-confined subspace of the original design space $\mathcal{X}$, we can write $\mathbf{X}_c \in \mathcal{X}_{tr} \subseteq \mathcal{X}$. This TR is centred around the current best point in the original space $\mathbf{x}^* \in \mathcal{X}$. Initialised with an initial length $r = r_0$, the length is increased or decreased according to the progress of the optimisation. Therefore, the algorithm counts the number of successes and failures which record whether a better point has been found or not. In doing so, the centre of the TR moves with success. Once a defined number of failures τ_f or successes τ_s is exceeded, the TR length is either decreased in size $r \leftarrow \frac{r}{2}$ or increased $r \leftarrow \min\{2r, r_{max}\}$. If $r < r_{min}$, a new TR is initialised.

5.5. Experiments

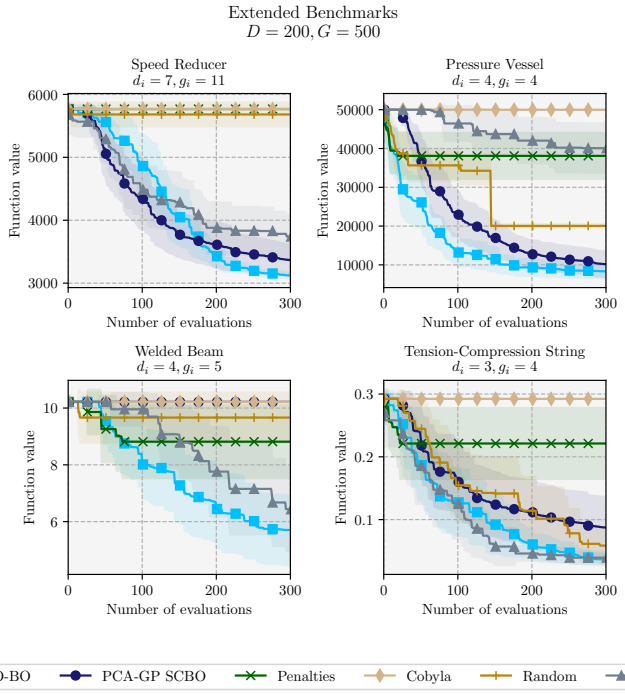


Figure 5.2: Performance comparison over four physics-based and synthetic benchmarks all embedded in a higher dimensional space and augmented with synthetic constraints.

In the following, we evaluate AERO-BO on a range of high-dimensional input–output benchmark problems, as well as a real-world aircraft design task. As described in Section 5.4.1, both encoder and decoder are implemented as single-layer feedforward neural networks. To assess performance, AERO-BO is compared against several baseline methods, including PCA-GP SCBO (Maathuis et al., 2024b), soft constraint handling via quadratic penalty terms (Eriksson and Jankowiak, 2021), COBYLA (Powell, 1994), the CMA-ES (Hansen, 2006), and a random search heuristic. Since SCBO and VBO require training a separate surrogate model for each constraint, quickly exceeding memory limitations in large-scale problems, comparisons to these methods are discussed separately in Section 5.5.3. AERO-BO is implemented using `GPyTorch` (Gardner et al., 2018) and `BoTorch` (Balandat et al., 2020). The code for reproducibility is available at: `https://github.com/haukemaa/aerobo`.

In line with Hernández-Lobato et al. (2016), we adopt the principle that a feasible solution is always preferable over an infeasible one when comparing optimisation outcomes. Therefore, infeasible solutions are assigned the value of the worst feasible

objective value found. Additionally, since PCA-GP SCBO also operates with a user-defined output latent dimension, we select the same latent dimension for PCA-GP SCBO as we do for AERO-BO, ensuring a fair comparison between the two methods.

5.5.1. Benchmarks

We adopt four physics-based test problems: the *Speed Reducer* (Lemonge et al., 2010), *Pressure Vessel* (Coello Coello and Mezura Montes, 2002), *Welded Beam Design* (Hedar and Fukushima, 2006) and *Tension Compression String* (Hedar and Fukushima, 2006). Additionally, we consider the 128D *MOPTA08* problem with 68 black-box constraints (Anjos, M., 2008). For more information, please refer to Appendix 5.7.1. To emulate the characteristics of many engineering problems having high-dimensional inputs and outputs, all benchmark problems are embedded in a $D = 200$ space and the number of constraints is artificially increase to $G = 500$ without altering the feasible optimal value of the optimisation problems (see Appendix 5.7.2). Thus, the original dimensionality (reduced space) and number of constraints of these problems are denoted by d_i and g_i, respectively, while D and G represent the new, extended dimensionality (original space) and number of constraints.

Physics-based and synthetic benchmarks. The results for the four physics-based test problems and the synthetic *Ackley* function are summarised in Figure 5.2. We use a total budged of 300 evaluations, a batch size of $q = 5$, 10 initial samples and perform 20 experiments per method and benchmark. For the sake of generality, we apply the same hyperparameters for all benchmarks, namely learning rate $\alpha = 0.01$, number of epochs $N_e = 10$, dimension of the input latent space $\dim(\tilde{\mathcal{X}}) = 50$, dimension of the output latent space $\dim(\tilde{\mathcal{Y}}) = 10$. For the *Speed Reducer* problem, we note that while PCA-GP SCBO and CMA-ES initially converge slightly faster, AERO-BO ultimately identifies a superior solution compared to the other two methods. All other competing methods either fail to find a feasible point or struggle to make meaningful progress. In the *Pressure Vessel* problem, AERO-BO and PCA-GP SCBO deliver the best performance, closely followed by the Random Search heuristic. By contrast, constraints handled via Penalties, COBYLA, and CMA-ES either fail to identify a feasible point or remain stuck, unable to improve further. The *Welded Beam* benchmark, on the other hand, appears challenging for all algorithms, exhibiting large variances across methods. While PCA-GP SCBO and COBYLA fail to find a feasible point and Penalties stagnates, CMA-ES is ultimately outperformed by AERO-BO. In the *Tension-Compression String* benchmark, AERO-BO and CMA-ES demonstrate strong performances, dominating the results. PCA-GP SCBO and Random Search perform slightly worse, while COBYLA and Penalties show limited progress. Lastly, the synthetic *Ackley* function is dominated by PCA-GP SCBO, followed by AERO-BO and Penalties, while all other methods fail to locate a feasible solution. Additionally, we observe that the COBYLA algorithm struggles with optimising the extended benchmark problems (we verified this by testing on the original benchmarks, where it could identify feasible points). However,

it is important to emphasise that the same hyperparameters and architectures, such as latent input and output dimensionalities, were applied across all methods. As demonstrated in the ablation study in Section 5.7.3, further improvements can be achieved by varying these parameters.

MOPTA08 benchmark. The *MOPTA08* benchmark, proposed by Anjos, M. (2008), originates from the automotive industry and is widely used for testing optimisation algorithms. The problem entails 124 dimensions and 68 black-box constraints. Again, we embed this benchmark into a 200D space and increase the number of constraints to 500. We utilise a total evaluation budget of 2000 samples, a batch size of $q = 15$, 100 initial samples and run 10 experiments. Furthermore, we set the latent dimension of the input autoencoder to $\dim(\tilde{\mathcal{X}}) = 60$ and for the latent output dimension to $\dim(\tilde{\mathcal{Y}}) = 20$. For training of AERO-BO we use $\alpha = 0.01$ and $N_e = 10$. The results of the *MOPTA08* benchmark are shown in Figure 5.3 (left). Most methods fail to locate a feasible point under these challenging conditions, with the exception of AERO-BO and PCA-GP SCBO. Notably, AERO-BO outperforms PCA-GP SCBO, further reinforcing the findings from the previously discussed benchmarks.

5.5.2. Aeroelastic Tailoring: A Multi-Disciplinary Design Optimisation Problem

Finally, we test AERO-BO and the aforementioned methods on a real-world aircraft wingbox design problem. The objective is to optimise the weight of a wingbox while satisfying thousands of constraints. The wingbox is divided into multiple design regions, with its stiffness and thickness adjustable through the use of composite materials. The thousands of constraints that need to be satisfied, arise from multi-disciplinary analyses, namely constraining the minimum strength of the structure, ensuring structural stability via a buckling analysis, as well as ensuring dynamic stability by analysing the dynamic aeroelastic capabilities of the structure, leading in total to $G = 1786$ constraints while $D = 108$ design variables describe the aforementioned structural properties. More on this problem can be found in Maathuis et al. (2024a).

In Figure 5.3 (right) the corresponding results are presented. AERO-BO and PCA-GP SCBO successfully identify a feasible solution after approximately 200 evaluations, even in this highly constrained design space. In contrast, all other methods fail to locate a feasible point, highlighting the robustness and effectiveness of AERO-BO. We use a total evaluation budget of 1500 samples, a batch size of $q = 15$, 100 initial samples and run 10 experiments. Furthermore, we set the latent dimension of the input autoencoder to $\dim(\tilde{\mathcal{X}}) = 60$ and the latent output dimension to $\dim(\tilde{\mathcal{Y}}) = 30$.

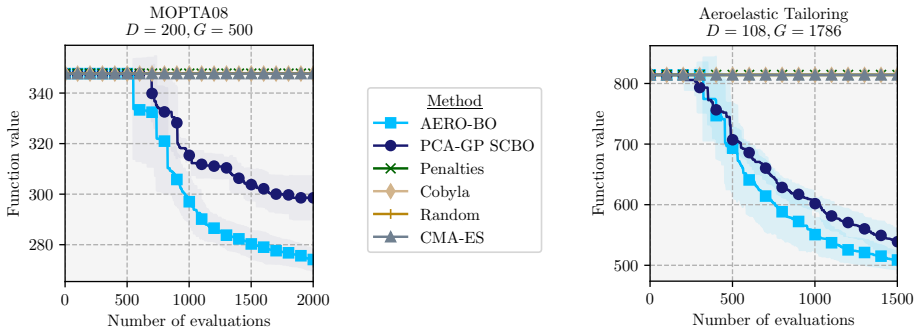


Figure 5.3: **(left)** 124D MOPTA08 benchmark with 68 black-box constraints. **(right)** 108D Aeroelastic Tailoring problem with 1786 black-box constraints arising from two different cases. The constraints ensure static strength, buckling, aeroelastic stability, aileron effectiveness, and local angle of attack of the system.

5.5.3. Comparison with SCBO and Vanilla Bayesian Optimisation

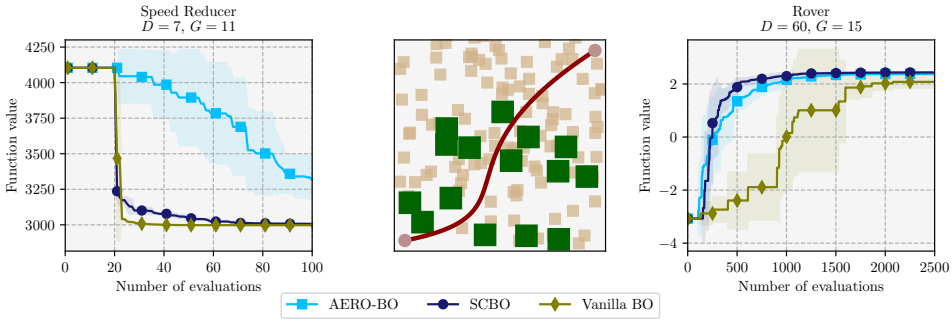


Figure 5.4: Comparison of AERO-BO with SCBO and VBO. **(left)** 7D *Speed Reducer* benchmark with 11 black-box constraints. (Lemonge et al., 2010) **(centre)** Example of an optimised trajectory. The green boxes are the impassable objects whereas the tanned objects are passable while adding a penalty to the objective function. **(right)** 60D *Rover trajectory* optimisation with 15 black-box constraints. (Wang et al., 2018)

To evaluate the capabilities and limitations of AERO-BO, we compare its performance with two baseline methods: (i) SCBO (Eriksson and Poloczek, 2021), which constructs a GP for each constraint and the objective in the full input space using a TR heuristic, and (ii) a VBO setup using the LogCEI acquisition function, together with a scaled length-scale prior, as proposed in Hvarfner et al. (2024). While AERO-BO is specifically designed for high-dimensional and large-constraint settings where SCBO and VBO become computationally intractable, this comparison serves to illustrate where each method excels, and how approximation errors

introduced by latent modelling affect performance. We assess performance on two benchmark problems: the 7*D Speed Reducer* problem with 11 black-box constraints (evaluated without up-projecting the input dimension and constraints), and the 60*D Rover* benchmark from Wang et al. (2018). In the latter, the task is to optimise a robot trajectory encoded by 30 spline control points, each consisting of an (x, y) coordinate. Following the setup in Eriksson and Poloczek (2021), we introduce impassable rectangular obstacles as constraints (visualised in Figure 5.4, centre).

For the *Speed Reducer* benchmark, we initialise with 20 samples and use a batch size of $q = 1$. AERO-BO uses arbitrary latent spaces of dimension $\dim(\tilde{\mathcal{X}}) = 4$ and $\dim(\tilde{\mathcal{Y}}) = 7$. For the Rover benchmark, we start with 100 initial samples and set $q = 100$, using latent dimensions $\dim(\tilde{\mathcal{X}}) = 20$ and $\dim(\tilde{\mathcal{Y}}) = 8$. The results, shown in Figure 5.4, reveal that AERO-BO underperforms relative to both SCBO and vanilla BO on the Speed Reducer benchmark. This is expected, as the additional approximation introduced by output-space compression may lead to reduced accuracy in low-data regimes. VBO with logCEI, performs best in this case, likely due to the modest problem dimensionality and constraint count, which allow full-dimensional GP modelling without significant computational burden. SCBO also performs well, likewise benefiting from constructing the constraint surrogates directly. In contrast, on the high-dimensional *Rover* benchmark, VBO fails to scale effectively, while AERO-BO performs almost as good as SCBO despite training significantly fewer surrogate models, demonstrating improved scalability. This comparison underscores several important insights. First, SCBO remains a strong choice for problems of moderate dimensionality and constraint count, particularly in early optimisation stages where surrogate models benefit from low uncertainty. However, its scalability is fundamentally limited, as it constructs and updates one full-dimensional GP per constraint. Second, AERO-BO introduces a trade-off: by reducing the dimensionality of the constraint space, it enables optimisation in previously intractable regimes, but at the cost of introducing an approximation. The quality of this approximation depends on the existence of a meaningful low-dimensional manifold in the output space, as well as the availability of sufficient training data. If these conditions are not met, as appears to be the case in the Speed Reducer problem, AERO-BO may fail to model high-frequency or strongly coupled constraint behaviour accurately, resulting in diminished performance.

These results highlight a fundamental limitation of latent-space BO approaches: if the effective dimensionality of the output space is high or the autoencoder is not trained with adequate data, compression may lead to loss of information and poor surrogate performance. This effect is explicitly visible in the reconstruction loss as in Equation 5.14 and is further compounded when the latent embedding is poorly aligned with the constraint geometry. Conversely, when a low-dimensional structure exists and is well captured, as in the *Rover* benchmark, AERO-BO offers a scalable alternative to conventional methods, maintaining competitive optimisation

performance while drastically reducing the number of GPs.

In summary, AERO-BO fills a practical niche in the constrained BO landscape: it enables efficient optimisation in settings characterised by high input dimensionality and a large number of black-box constraints. However, its success is contingent upon the representational capacity of the autoencoders and the quality of the learned latent spaces. In settings where the number of constraints is low, methods such as SCBO or standard BO may still offer superior performance.

5.5.4. Ablation Study

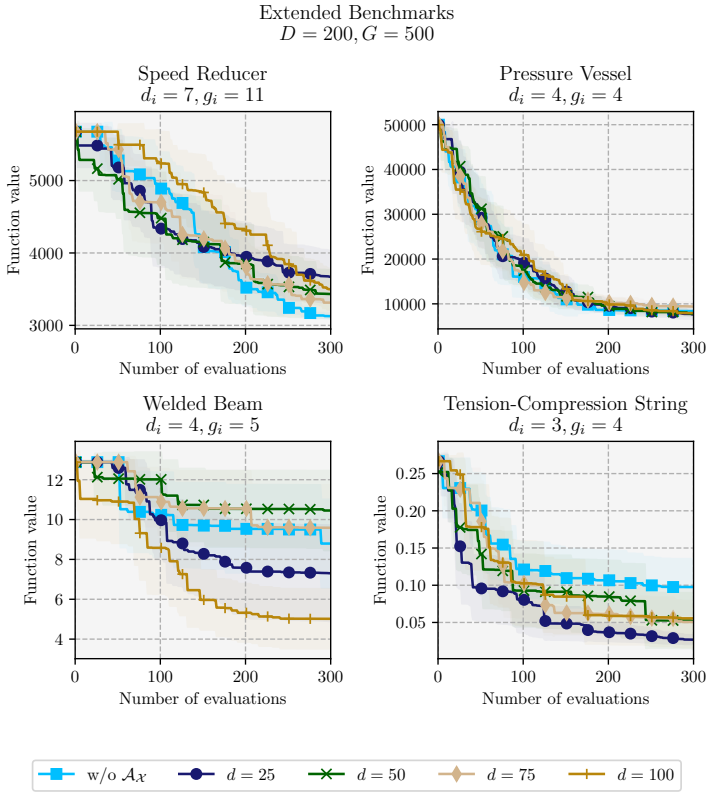


Figure 5.5: Input Dimension variation, investigating the influence of the input reduction on the performance of the method.

To better understand the behaviour and design trade-offs of AERO-BO, we conduct a series of ablation studies focusing on its key components and hyperparameters. In this subsection, we investigate the influence of the latent input space dimensionality, i.e. the output size of the encoder $\phi_\mathcal{X}$, on optimisation performance. Figure 5.5

presents results for the previously introduced, up-projected benchmark problems. We also include a baseline that omits input dimensionality reduction altogether, replacing the encoder $\phi_{\mathcal{X}}$ with the identity function. This baseline corresponds to a variant of SCBO that compresses only the output (constraint) space using an autoencoder rather than PCA.

The results show that input compression via $\mathcal{A}_{\mathcal{X}}$ is beneficial in most cases. However, the optimal choice of latent dimension is problem-specific. In the first two benchmarks, namely *Speed Reducer* and *Pressure Vessel*, performance differences between latent dimensions are relatively small. This suggests that the problems either admit a wide range of effective latent dimensionalities or that the search space is not strongly sensitive to compression in those cases. Consequently, no single latent size clearly dominates, and several configurations yield near-equivalent performance. In contrast, the third and fourth benchmarks (Welded Beam and Tension-Compression String) exhibit more distinct trends, depicting that performance can benefit from input dimensionality reduction. In the third case, a larger latent space ($\dim(\tilde{\mathcal{X}}) = 100$) performs best, followed $\dim(\tilde{\mathcal{X}}) = 25$, while $\dim(\tilde{\mathcal{X}}) = 50$ underperforms. This non-monotonic behaviour likely reflects trade-offs between expressiveness and regularisation: $\dim(\tilde{\mathcal{X}}) = 100$ may better capture complex interactions, whereas dimension $\dim(\tilde{\mathcal{X}}) = 25$ may promote smoother, more regular models. In the fourth benchmark, a smaller latent space ($\dim(\tilde{\mathcal{X}}) = 25$) yields the best performance, followed by dimension $\dim(\tilde{\mathcal{X}}) = 100$, then $\dim(\tilde{\mathcal{X}}) = 50$. This inversion indicates that for some problems, aggressive compression may help by suppressing noise and preventing overfitting.

As is common in latent variable modelling, the optimal latent dimensionality is both problem- and data-dependent. When the input–output mapping lies on a well-defined low-dimensional manifold and enough training data are available, small latent spaces can suffice. However, for intrinsically high-dimensional problems or in low-data regimes, excessive compression may discard informative variation, impairing surrogate fidelity and optimisation performance. These findings underscore the importance of selecting the latent dimension carefully. While AERO-BO is generally robust to this parameter, practitioners may benefit from heuristic guidance such as examining PCA eigenvalue spectra or monitoring surrogate reconstruction loss.

Further ablation studies on learning rate, training epochs, training window size, and the use of the TR heuristic (Eriksson and Poloczek, 2021) are provided in Appendix 5.7.3. These experiments confirm that AERO-BO performs robustly across a wide range of configurations, reinforcing its suitability for practical use.

5.6. Conclusion

In this work, we introduce AERO-BO, a novel approach for high-dimensional BO that effectively handles problems with hundreds of design variables as well as thousands

of constraints. Unlike many traditional high-dimensional BO methods, which rely on random input embeddings and face challenges in constrained settings, AERO-BO employs a unique strategy: two jointly trained autoencoders that map both inputs and outputs to lower-dimensional latent spaces. This approach eliminates the need to construct separate surrogate models for each constraint, making it computationally efficient and scalable. We demonstrate the effectiveness of AERO-BO across a variety of benchmarks, including existing and extended high-dimensional test problems and a complex real-world multidisciplinary design optimisation problem from aerospace engineering. The results showed that AERO-BO consistently outperforms competing methods in finding feasible solutions for highly constrained problems, which are often encountered in engineering optimisation. This capability is particularly crucial when other methods struggle to find feasible points, highlighting the reliability and robustness of AERO-BO.

5.7. Appendix

5.7.1. Benchmark Problems

This section briefly describes the benchmark cases used in this work.

7D Speed Reducer with 11 black-box constraints This benchmark poses an optimisation problem to minimise the weight of the so-called speed reducer. The design variables include geometrical measures like the length of two shafts and the width of the face, also the number of teeth on a pinion and the module of teeth (Lemonge et al., 2010)

4D Pressure Vessel with 4 black-box constraints The goal of this benchmark is to minimise the cost of the design. The design variables include the shell and head thicknesses, as well as the inner radius and length of the cylindrical section including some bounds (Coello Coello and Mezura Montes, 2002).

4D Welded Beam with 5 black-box constraints The Welded Beam benchmark aims to minimise the cost by considering mechanical limits on the structure, such as shear stress, bending stress, buckling load and the deflection of the beam (Hedar and Fukushima, 2006).

3D Tension-Compression Spring with 4 black-box constraints This problem also takes into account mechanical constraints such as a limit on the deflection, shear stress and surge frequency. In addition a geometrical parameter is added, describing the outside diameter. While considering these constraints, the aim is to minimise the weight of the tension compression string (Hedar and Fukushima, 2006).

10D Ackley with 2 black-box constraints. This benchmark is a synthetic function, considering two black-box constraints and is known to be difficult to optimise taken from Eriksson and Poloczek (2021).

128D MOPTA08 with 68 black-box constraints. Lastly, the MOPTA08 benchmark was proposed by Anjos, M. (2008) and stems form the automotive industry. The design variables herein describe the material and shape of the structure as well as gages while taking into account some performance constraints.

60D Rover with 15 black-box constraints. This problem was originally considered by Wang et al. (2018), optimising the trajectory of a rover by fitting a B-spline to 30 design points, described by x and a y coordinate. The objective function is $f(x) = c(x) + 5$ where $c(x)$ penalises collisions with objects by -20. Additionally, Eriksson and Poloczek (2021) propose to add 15 impassable objects. More on the constraint formulation can be found in their paper.

5.7.2. Extending Benchmark Problems

To evaluate the performance of optimisation algorithms in high-dimensional and highly constrained scenarios, we extend existing benchmark problems in terms of dimensionality and the number of constraints. Considering a d_i-dimensional optimisation problem including g_i black-box constraints as follows

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \quad \text{s.t.} \quad \mathbf{c}(\mathbf{x}) \leq 0, \tag{5.17}$$

with $\mathbf{c} \in \mathbb{R}^{g_i}$. The goal is to extend the existing problem to D dimensions and G constraints, where $d_i \ll D$ and $g_i \ll G$. To achieve this, two projection matrices are defined: $\mathbf{P}_d : \mathbb{R}^D \to \mathbb{R}^{d_i}$, mapping the high-dimensional input space back to the original space, and $\mathbf{P}_{g_i} : \mathbb{R}^G \to \mathbb{R}^{g_i}$, mapping the extended constraint space to the original constraint space. Using these projections, the extended optimisation problem can be written as

$$\min_{\mathbf{x}'\subseteq\mathcal{X}'\in\mathbb{R}^D} f(\mathbf{P}_{d_i}\mathbf{x}') \quad \text{s.t.} \quad \mathbf{c}(\mathbf{P}_{d_i}\mathbf{x}') \leq 0, \tag{5.18}$$

where $\mathbf{P}_{d_i}$ retains only the original d_i dimensions, ensuring that the objective function $f(\mathbf{P}_{d_i}\mathbf{x}') = f(\mathbf{x})$ is unchanged. Similarly, the constraints satisfy $\mathbf{c}(\mathbf{P}_{d_i}\mathbf{x}') = \mathbf{c}(\mathbf{x})$. Here, the extended variable $\mathbf{x}'$ is decomposed into $\mathbf{x}' = [\mathbf{x}^\top, \tilde{\mathbf{x}}^\top]^\top$ where $\tilde{\mathbf{x}} \in \mathbb{R}^{(D-d_i)}$ represents some artificial dimensions. To introduce additional constraints, we define the artificial constraints as follows:

$$\tilde{\mathbf{c}}(\tilde{\mathbf{x}}) = \mathbf{A}\tilde{\mathbf{x}} - \mathbf{b}, \tag{5.19}$$

where $\mathbf{A} \in \mathbb{R}^{(G-g_i)\times(D-d_i)}$ and $\mathbf{b} \in \mathbb{R}^{(D-d_i)}$ are a random matrix and vector, respectively, where $\mathbf{b}$ ensures that the artificial constraints are non-violated by construction. The extended constraints are then formulated by combining the original and artificial constraints:

$$\mathbf{c}'(\mathbf{x}') = [\mathbf{c}^\top(\mathbf{P}_{d_i}\mathbf{x}'), \tilde{\mathbf{c}}^\top(\tilde{\mathbf{x}})]^\top \in \mathbb{R}^G. \tag{5.20}$$

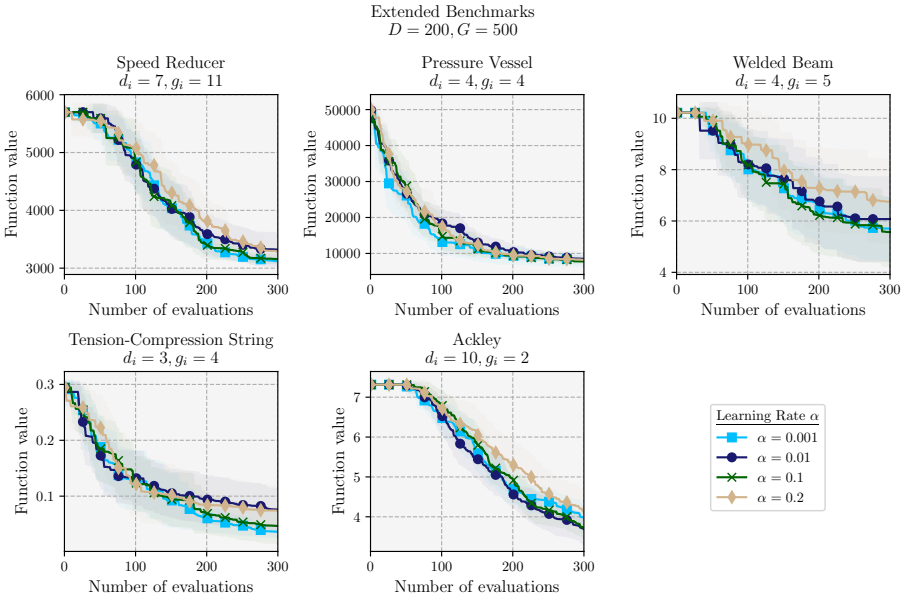


Figure 5.6: Sensitivity analysis of the learning rate α.

Thus, the final extended optimisation problem becomes:

$$\min_{\mathbf{x}' \subseteq \mathcal{X}' \in \mathbb{R}^D} f(\mathbf{x}') \quad \text{s.t.} \quad \mathbf{c}'(\mathbf{x}') \leq 0. \tag{5.21}$$

This approach ensures that the characteristics of the original problem are preserved, while allowing for scalability in terms of dimensionality and constraint complexity.

5.7.3. Additional Ablation Studies and Sensitivity Analyses

In this section we present some additional ablation studies to investigate the performance of AERO-BO with respect to some of its hyperparameters as well as the influence of the TR heuristic.

5.7.4. Training Hyperparameters

We analyse the performance of AERO-BO concerning the learning rate α, used to jointly train the models. Figure 5.6 shows that the algorithm exhibits robustness across different values of α, with the best performance achieved at $\alpha = 0.001$. The impact of the number of training epochs N_e is investigated while keeping the learning rate fixed. Figure 5.7 demonstrates robust performance across different values of N_e. Notably, when the models are not updated ($N_e = 0$), performance significantly deteriorates in some cases, underscoring the importance of periodic

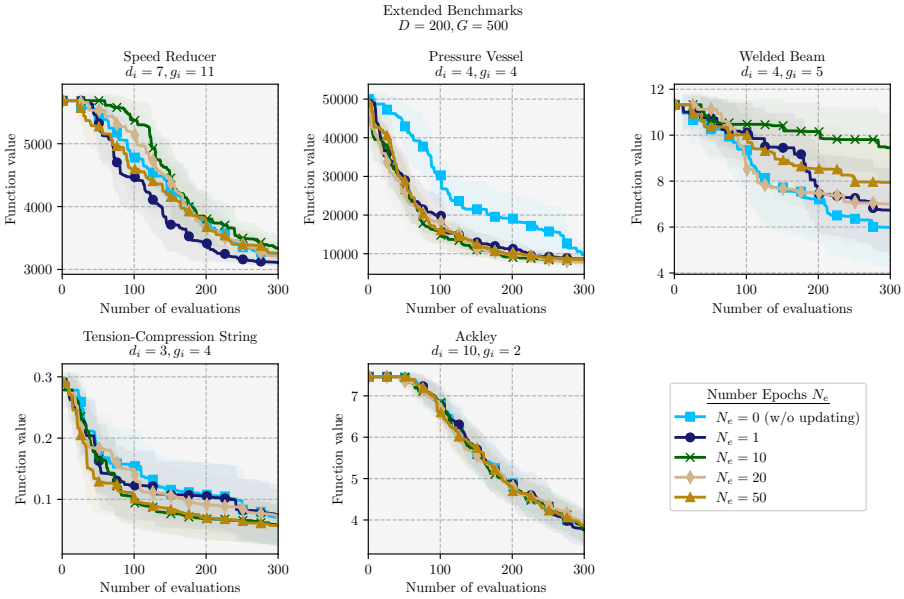


Figure 5.7: Sensitivity analysis of the number of training epochs N_e.

updates. Lastly, the influence of the training window size or also called lookback factor N_b is investigated, inspired by Maus et al. (2024). The lookback factor determines the number of recent samples within the current dataset $\mathcal{D}$ used during training. Figure 5.8 illustrates that utilising all samples ($N_b = N$) generally yields the best results, except for the Welded Beam benchmark. However, considering all samples may increase training costs.

5.7.5. Influence of the Trust Region Heuristic

The role of the trust region heuristic is evaluated, as introduced by Eriksson and Poloczek (2021). Figure 5.9 compares two scenarios: Enforcing that N_c candidate points lie within a trust region centred around the best solution so far and allowing candidate sampling across the entire design space. The use of perturbation probability, as noted by Rashidi et al. (2024), remains integral to the SCBO framework and is used in both cases. The results in Figure 5.9 suggest that while the trust region heuristic has limited impact on the speed of finding a feasible point, it plays a crucial role in identifying higher-quality solutions later in the optimisation process.

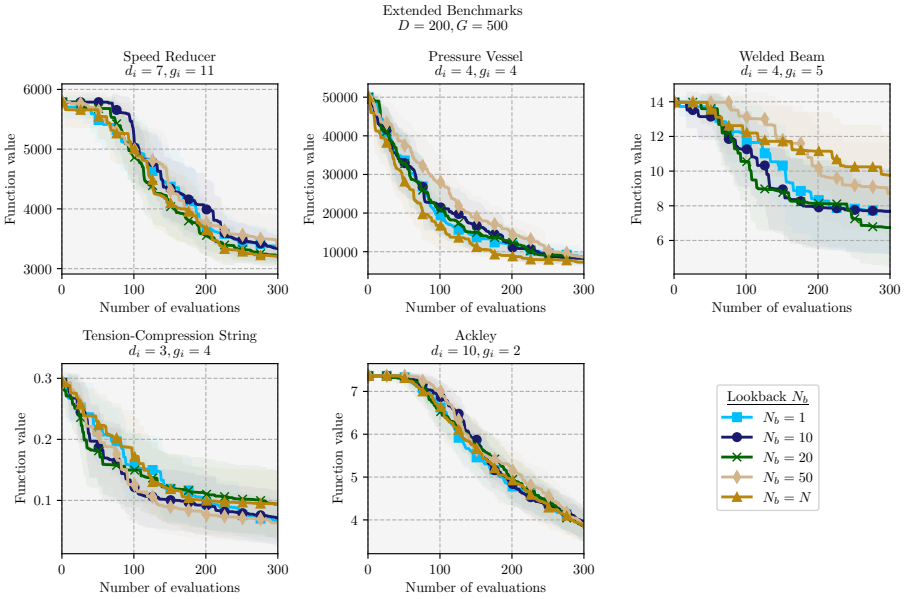


Figure 5.8: Ablation study of the lookback factor N_b with $N_b = N$ denoting the case where all samples within the current $\mathcal{D}$ are taken into account for training.

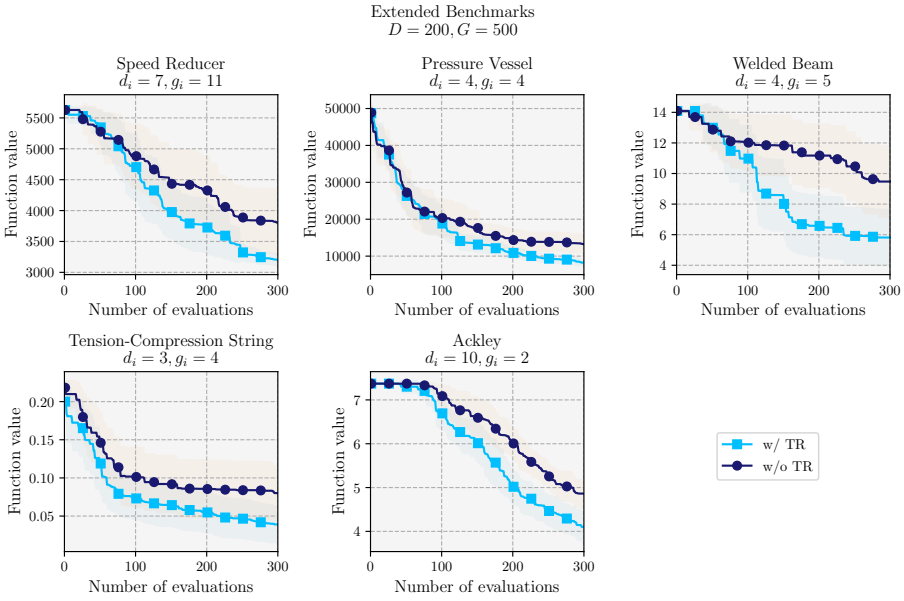


Figure 5.9: Ablation study investigating the influence of the TR heuristic on the performance of AERO-BO.

BIBLIOGRAPHY

S. Ament, S. Daulton, D. Eriksson, M. Balandat, and E. Bakshy. Unexpected Improvements to Expected Improvement for Bayesian Optimization, Jan. 2023. URL http://arxiv.org/abs/2310.20708. arXiv:2310.20708 [cs].

Anjos, M. The MOPTA 2008 Benchmark, 2008. URL http://www.miguelanjos.com/jones-benchmark.

M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*, 2020. URL http://arxiv.org/abs/1910.06403.

M. Binois and N. Wycoff. A Survey on High-dimensional Gaussian Process Modeling with Application to Bayesian Optimization. *ACM Transactions on Evolutionary Learning and Optimization*, 2(2):1–26, June 2022. ISSN 2688-299X, 2688-3007. doi: 10.1145/3545611. URL https://dl.acm.org/doi/10.1145/3545611.

E. V. Bonilla, K. Chai, and C. Williams. Multi-task gaussian process prediction. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/66368270ffd51418ec58bd793f2d9b1b-Paper.pdf.

C. A. Coello Coello and E. Mezura Montes. Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. *Advanced Engineering Informatics*, 16(3):193–203, July 2002. ISSN 14740346. doi: 10.1016/S1474-0346(02)00011-3. URL https://linkinghub.elsevier.com/retrieve/pii/S1474034602000113.

D. Eriksson and M. Jankowiak. High-Dimensional Bayesian Optimization with Sparse Axis-Aligned Subspaces, June 2021. URL http://arxiv.org/abs/2103.00349. arXiv:2103.00349 [cs, stat].

D. Eriksson and M. Poloczek. Scalable Constrained Bayesian Optimization, Feb. 2021. URL http://arxiv.org/abs/2002.08526. arXiv:2002.08526 [cs, stat].

D. Eriksson, M. Pearce, J. Gardner, R. D. Turner, and M. Poloczek. Scalable Global Optimization via Local Bayesian Optimization. 2019.

P. I. Frazier. A Tutorial on Bayesian Optimization, July 2018. URL http://arxiv.org/abs/1807.02811. arXiv:1807.02811 [cs, math, stat].

J. R. Gardner, M. J. Kusner, and G. Jake. Bayesian Optimization with Inequality Constraints. 2014.

J. R. Gardner, G. Pleiss, D. Bindel, K. Q. Weinberger, and A. G. Wilson. Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Advances in Neural Information Processing Systems*, 2018.

M. A. Gelbart, J. Snoek, and R. P. Adams. Bayesian Optimization with Unknown Constraints. 2014.

A. Grosnit, R. Tutunov, A. M. Maraval, R.-R. Griffiths, A. I. Cowen-Rivers, L. Yang, L. Zhu, W. Lyu, Z. Chen, J. Wang, J. Peters, and H. Bou-Ammar. High-Dimensional Bayesian Optimisation with Variational Autoencoders and Deep Metric Learning, Nov. 2021. URL http://arxiv.org/abs/2106.03609. arXiv:2106.03609 [cs].

R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Central Science*, 4(2):268–276, Feb. 2018. ISSN 2374-7943, 2374-7951. doi: 10.1021/acscentsci.7b00572. URL https://pubs.acs.org/doi/10.1021/acscentsci.7b00572.

N. Hansen. The cma evolution strategy: A comparing review. In *Towards a New Evolutionary Computation*, 2006. URL https://api.semanticscholar.org/CorpusID:13968591.

A.-R. Hedar and M. Fukushima. Derivative-Free Filter Simulated Annealing Method for Constrained Continuous Global Optimization. *Journal of Global Optimization*, 35(4):521–549, Aug. 2006. ISSN 0925-5001, 1573-2916. doi: 10.1007/s10898-005-3693-z. URL http://link.springer.com/10.1007/s10898-005-3693-z.

A. Heifetz, editor. *High Performance Computing for Drug Discovery and Biomedicine*, volume 2716 of *Methods in Molecular Biology*. Springer US, New York, NY, 2024. ISBN 978-1-07-163448-6 978-1-07-163449-3. doi: 10.1007/978-1-0716-3449-3. URL https://link.springer.com/10.1007/978-1-0716-3449-3.

J. Hensman, A. Matthews, and Z. Ghahramani. Scalable Variational Gaussian Process Classification, Nov. 2014. URL http://arxiv.org/abs/1411.2005. arXiv:1411.2005 [stat].

J. M. Hernández-Lobato, M. A. Gelbart, R. P. Adams, M. W. Hoffman, and Z. Ghahramani. A General Framework for Constrained Bayesian Optimization using Information-based Search, Sept. 2016. URL http://arxiv.org/abs/1511.09422. arXiv:1511.09422 [stat].

D. Higdon, J. Gattiker, B. Williams, and M. Rightley. Computer Model Calibration Using High-Dimensional Output. *Journal of the American Statistical Association*, 103(482):570–583, June 2008. ISSN 0162-1459, 1537-274X. doi:

10.1198/016214507000000888. URL https://www.tandfonline.com/doi/full/10.1198/016214507000000888.

C. Hvarfner, E. O. Hellsten, and L. Nardi. Vanilla Bayesian optimization performs great in high dimensions. In R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 20793–20817. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/hvarfner24a.html.

D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization, Jan. 2017. URL http://arxiv.org/abs/1412.6980. arXiv:1412.6980 [cs].

H. J. Kushner. A versatile stochastic model of a function of unknown and time varying form. *Journal of Mathematical Analysis and Applications*, 5(1):150–167, Aug. 1962. ISSN 0022247X. doi: 10.1016/0022-247X(62)90011-2. URL https://linkinghub.elsevier.com/retrieve/pii/0022247X62900112.

H. J. Kushner. A New Method of Locating the Maximum Point of an Arbitrary Multipeak Curve in the Presence of Noise. *Journal of Basic Engineering*, 86(1):97–106, Mar. 1964. ISSN 0021-9223. doi: 10.1115/1.3653121. URL https://asmedigitalcollection.asme.org/fluidsengineering/article/86/1/97/392213/A-New-Method-of-Locating-the-Maximum-Point-of-an.

A. C. C. Lemonge, H. J. C. Barbosa, C. C. H. Borges, and F. B. dos Santos Silva. Constrained optimization problems in mechanical engineering design using a real-coded steady-state genetic algorithm. 2010. URL https://api.semanticscholar.org/CorpusID:54994542.

B. Letham, R. Calandra, A. Rai, and E. Bakshy. Re-Examining Linear Embeddings for High-Dimensional Bayesian Optimization, Oct. 2020. URL http://arxiv.org/abs/2001.11659. arXiv:2001.11659 [cs, stat].

H. Maathuis, R. D. Breuker, and S. G. P. Castro. High-Dimensional Bayesian Optimisation with Large-Scale Constraints – An Application to Aeroelastic Tailoring. In *AIAA SCITECH 2024 Forum*, Jan. 2024a. doi: 10.2514/6.2024-2012. URL http://arxiv.org/abs/2312.08891. arXiv:2312.08891 [cs].

H. F. Maathuis, R. D. Breuker, and S. G. P. Castro. High-Dimensional Bayesian Optimisation with Large-Scale Constraints via Latent Space Gaussian Processes, Dec. 2024b. URL http://arxiv.org/abs/2412.15679. arXiv:2412.15679 [cs].

W. J. Maddox, M. Balandat, A. G. Wilson, and E. Bakshy. Bayesian Optimization with High-Dimensional Outputs, Oct. 2021. URL http://arxiv.org/abs/2106.12997. arXiv:2106.12997 [cs].

N. Maus, H. T. Jones, J. S. Moore, M. J. Kusner, J. Bradshaw, and J. R. Gardner. Local Latent Space Bayesian Optimization over Structured Inputs, Feb. 2023. URL http://arxiv.org/abs/2201.11872. arXiv:2201.11872 [cs].

N. Maus, Z. J. Lin, M. Balandat, and E. Bakshy. Joint Composite Latent Space Bayesian Optimization, July 2024. URL http://arxiv.org/abs/2311.02213. arXiv:2311.02213 [cs].

A. Nayebi, A. Munteanu, and M. Poloczek. A framework for Bayesian optimization in embedded subspaces. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 4752–4761, 2019. URL https://proceedings.mlr.press/v97/nayebi19a.html.

L. Papenmeier, L. Nardi, and M. Poloczek. Increasing the Scope as You Learn: Adaptive Bayesian Optimization in Nested Subspaces, Apr. 2023. URL http://arxiv.org/abs/2304.11468. arXiv:2304.11468 [cs].

M. J. D. Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. 1994. URL https://api.semanticscholar.org/CorpusID:118045691.

B. Rashidi, K. Johnstonbaugh, and C. Gao. Cylindrical Thompson sampling for high-dimensional Bayesian optimization. In S. Dasgupta, S. Mandt, and Y. Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 3502–3510. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/rashidi24a.html.

C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass, 2006. ISBN 978-0-262-18253-9. OCLC: ocm61285753.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. *Learning internal representations by error propagation*, page 318–362. MIT Press, Cambridge, MA, USA, 1986. ISBN 026268053X.

W. R. Thompson. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika*, 25(3/4):285, Dec. 1933. ISSN 00063444. doi: 10.2307/2332286. URL https://www.jstor.org/stable/2332286?origin=crossref.

A. Tripp, E. Daxberger, and J. M. Hernández-Lobato. Sample-Efficient Optimization in the Latent Space of Deep Generative Models via Weighted Retraining, Oct. 2020. URL http://arxiv.org/abs/2006.09191. arXiv:2006.09191 [cs].

Z. Wang, F. Hutter, M. Zoghi, D. Matheson, and N. de Freitas. Bayesian Optimization in a Billion Dimensions via Random Embeddings, Jan. 2016. URL http://arxiv.org/abs/1301.1942. arXiv:1301.1942 [cs, stat].

Z. Wang, C. Gehring, P. Kohli, and S. Jegelka. Batched large-scale bayesian optimization in high-dimensional spaces, 2018. URL https://arxiv.org/pdf/1706.01445.

W. Xing, A. A. Shah, and P. B. Nair. Reduced dimensional Gaussian process emulators of parametrized partial differential equations based on Isomap. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2174): 20140697, Feb. 2015. ISSN 1364-5021, 1471-2946. doi: 10.1098/rspa.2014.0697. URL https://royalsocietypublishing.org/doi/10.1098/rspa.2014.0697.

W. Xing, V. Triantafyllidis, A. Shah, P. Nair, and N. Zabaras. Manifold learning for the emulation of spatial fields from computational models. *Journal of Computational Physics*, 326:666–690, Dec. 2016. ISSN 00219991. doi: 10.1016/j.jcp.2016.07.040. URL https://linkinghub.elsevier.com/retrieve/pii/S0021999116303722.

S. Zhe, W. Xing, and R. M. Kirby. Scalable high-order gaussian process regression. In K. Chaudhuri and M. Sugiyama, editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 2611–2620. PMLR, 16–18 Apr 2019. URL https://proceedings.mlr.press/v89/zhe19a.html.

6

Constrained Bayesian Optimisation with Multiple Information Sources

This chapter is based on the following preprint and has been reproduced with minor adjustments to notation and formatting for consistency within the thesis.
Maathuis, H., & De Breuker, R. & Castro, S.G.P. & Osborne, M. (2025); Constrained Bayesian Optimisation with Multiple Information Sources. Preprint

Abstract BO under unknown constraints is particularly challenging when feasible regions are small. In such settings, existing methods that typically rely solely on evaluations of the true objective and constraints struggle to efficiently explore the design space. However, many real-world applications offer auxiliary data sources (e.g. surrogate models or simplified simulations) that can support early exploration. Despite this potential, their integration into CBO remains largely unexplored. We propose a general multi-source framework that extends constrained Max-value Entropy Search, capturing inter-source correlation while balancing evaluation cost and information gain. Experiments on both synthetic and physics-based benchmarks show that our method efficiently identifies feasible and optimal solutions, even when auxiliary data are only weakly correlated. The proposed approach consistently outperforms existing methods, particularly in early-stage exploration.

6.1. Introduction

BO is a principled framework for optimising expensive black-box functions. Such problems are prevalent in science and engineering applications, including materials

discovery, drug design and simulation-based engineering. In these domains, where each evaluation may involve expensive physical experiments or high-resolution simulations, sample efficiency becomes critical. Constraints, often black-box functions themselves, are often equally expensive to evaluate, yet essential to ensure feasible designs. In many applications the feasible region is small, discontinuous, or highly non-linear, making even the discovery of a single feasible point challenging, especially in high-dimensional spaces where data is sparse, as commonly seen in crashworthiness design (Raponi et al., 2019).

In many engineering applications, however, practitioners may have access to multiple information sources. Examples include coarse simulations (Wu et al., 2019), simplified physics-based models (Maathuis et al., 2024, Aretz et al., 2025), or analytical approximations (Anand et al., 2024), that are cheaper to evaluate but potentially biased or noisy. In drug design, this may include inexpensive simulations alongside costly laboratory experiments. While these information sources are often correlated with the target information source (ground truth), the strength and structure of this correlation can vary significantly. Yet, even when information sources are only weakly correlated with the target, they can still provide valuable information for optimisation and help populate the design space, particularly in data-sparse scenarios.

Unlike traditional BO methods that rely on a single information source, multi-source BO leverages multiple models in a cost-aware manner, using cheap sources to guide exploration while reserving expensive evaluations to maintain accuracy. These approaches, often referred to as multi-fidelity in literature, were initially modelled through autoregressive processes (Kennedy and O'Hagan, 2000, Forrester et al., 2007), with later work extending acquisition strategies to this setting (Kandasamy et al., 2017, Poloczek et al., 2016, Wu et al., 2019, Takeno et al., 2020). While much of the literature uses the term multi-fidelity, our work adopts the more general *multi-source* perspective, which does not assume a strict fidelity hierarchy.

Despite recent progress, the integration of multi-source modelling with CBO remains under-explored, as most methods address either constraints or multiple sources in isolation. In particular, leveraging low-cost sources for early-stage exploration, especially when feasible points are unknown, has received little attention. To close this gap, we propose a scalable framework for CBO with multiple information sources, targeting problems with expensive target evaluations, hard-to-locate feasible regions, and sparse data. We formalise the constrained optimisation problem with multiple sources as

$$\begin{aligned} \min_{\mathbf{x}\in\mathcal{X}\subset\mathbb{R}^d} \quad & f^{(L)}(\mathbf{x}) \\ \text{s.t.} \quad & c_i^{(L)}(\mathbf{x}) \leq 0, \quad i = 1, \ldots, g \end{aligned} \tag{6.1}$$

where $f^{(L)} : \mathcal{X} \to \mathbb{R}$ and $c_i^{(L)} : \mathcal{X} \to \mathbb{R}\ \forall i = 1, ..., G$ respectively denote the objective and constraints at the target information source L (most accurate but costly). We

assume that the objective and all constraints are always evaluated jointly at a given source. While this assumption may not hold in all domains, it is natural in computational engineering. The design space $\mathcal{X} \subset \mathbb{R}^D$ may be high-dimensional. To reduce queries to source L, we additionally leverage cheaper information sources $f^{(\ell)}$, $c_i^{(\ell)}$ with $\ell \in \mathcal{S} = \{0, 1, \ldots, L\}$ which provide biased or noisy estimates at lower cost. These auxiliary sources help explore the design space and identify promising regions, while target source evaluations refine solutions near the feasibility boundary or optima. The goal is to efficiently optimise $f^{(L)}$ subject to the constraints $c_i^{(L)} \leq 0$ under a limited budget by balancing information gain and query cost across sources. Our work makes the following key contributions:

- A unified framework for CBO with multiple sources, combining cheap auxiliary samples and a TR heuristic for scalability in to high-dimensional input spaces.
- A systematic comparison of scaling multi-source GP models.
- Extensive experiments on real-world and synthetic benchmarks, showing clear improvements over existing CBO methods.

6.2. Related Work

Early approaches to multi-source or multi-fidelity modelling used hierarchical GP with nested fidelities (Forrester et al., 2007, Kennedy and O'Hagan, 2000), later extended to non-nested settings via autoregressive and discrepancy-based models (Gratiet, 2013, Perdikaris et al., 2017). In BO, Kandasamy et al. (2017) proposed a bandit-based fidelity selection method, while Poloczek et al. (2016) introduced the MISO framework, treating each source as a biased observation of a latent function. Further developments include trace-aware knowledge gradients (Wu et al., 2019), multi-fidelity MES (Takeno et al., 2020), and safeguards against uninformative sources (Mikkola et al., 2022). Additionally, recent pre-prints (Foumani and Bostanabad, 2025, Cordelier et al., 2025) explore constrained multi-source BO. The former introduces a simple cost-aware heuristic based on expected improvement with feasibility filtering, while the latter performs sequential optimisation over design and fidelity. However, both approaches remain limited in empirical validation.

A parallel line of research has explored information-theoretic acquisition functions which aim to maximise expected information gain about the global optimum. PES (Hernández-Lobato et al., 2014) and MES (Wang and Jegelka, 2018) estimate information gain about the global optimum. Extensions include Fast Information-Theoretic Bayesian Optimisation (FITBO) (Ru et al., 2018) and General-purpose Information-Based Bayesian Optimisation (GIBBON) (Moss et al., 2021). Constrained variants incorporate feasibility modelling via PES with constraints PESC (Hernández-Lobato et al., 2016) and CMES (Perrone et al., 2019).

For scalability to high dimensions with constraints, proposed strategies include

TR heuristics (Eriksson and Poloczek, 2021), its extension of feasibility-aware refinements (Ascia et al., 2025), and scaled length-scale priors (Hvarfner et al., 2024, Papenmeier et al., 2025), often in combination with LogCEI (Ament et al., 2023). High-dimensional problems with thousands of black-box constraints have been addressed in Maathuis et al. (2025), while Om et al. (2025) employ flow-based ensembles to improve GP scalability.

Despite this progress, the joint treatment of constraints and multiple sources to scale BO algorithms remains under-explored. Early discovery of feasible regions is especially critical, yet existing methods rarely exploit auxiliary sources for this purpose. We address this gap by proposing a scalable, unified, information-theoretic framework for CBO with multiple sources.

6.3. Bayesian Optimisation with Black-Box Constraints and Multiple Information Sources

In this section, we first review BO with unknown constraints and its use of GPs. We then show how this framework can be extended to incorporate multiple information sources.

6.3.1. Constrained Bayesian Optimisation

We consider the constrained optimisation problem in Equation 6.1, where the feasible set under the target information source L is defined as:

$$\mathcal{X}_f^{(L)} = \{\mathbf{x} \in \mathcal{X} \mid c_i^{(L)}(\mathbf{x}) \leq 0,\ i = 1, \ldots, G\}. \tag{6.2}$$

Since constraint predictions may vary across sources, feasibility must be defined with respect to the target source L. BO (Kushner, 1962, 1964) addresses expensive, black-box problems by learning probabilistic surrogate models, typically GPs, to approximate the objective and constraints, allowing for efficient exploration and exploitation of the design space (Frazier, 2018). An acquisition function $\alpha(\mathbf{x}; \mathcal{D}_t) : \mathcal{X} \to \mathbb{R}$ balances this exploration-exploitation trade-off by selecting the next evaluation point $\mathbf{x}_+ \in \mathcal{X}$. In the constrained setting, feasibility such that $\mathbf{x}_+ \in \mathcal{X}_f$, must be accounted for by the acquisition function:

$$\mathbf{x}_+ = \underset{\mathbf{x} \in \mathcal{X}_f}{\operatorname{argmax}}\, \alpha(\mathbf{x}; \mathcal{D}_t). \tag{6.3}$$

Popular constrained acquisition functions have been proposed in Gardner et al. (2014), Gelbart et al. (2014), Ament et al. (2023), Hernández-Lobato et al. (2016), Eriksson and Poloczek (2021). These methods leverage GP-based uncertainty estimates to guide the search towards promising, feasible regions, enabling efficient optimisation despite limited budgets and unknown constraint landscapes.

6.3.2. Gaussian Process Regression

GPs provide a flexible, non-parametric prior over functions. A function $u : \mathcal{X} \to \mathbb{R}$, where $\mathcal{X} \subset \mathbb{R}^D$, is said to follow a GP prior if for any finite collection of input points $\{\mathbf{x}_i\}_{i=1}^N \subset \mathcal{X}$. The corresponding vector of function values follows a multivariate normal distribution $u \sim \mathcal{GP}(\mu, \Sigma)$ such that

$$[u(\mathbf{x}_1), \ldots, u(\mathbf{x}_N)]^\top \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}), \tag{6.4}$$

where $\mu : \mathcal{X} \to \mathbb{R}$ is the mean function with $\mu_i = \mu(\mathbf{x}_i)$, and $\Sigma : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the covariance function with $K_{ij} = \Sigma(\mathbf{x}_i, \mathbf{x}_j)$. In practice, the mean is often taken to be zero, while typical covariance functions include the squared exponential or Matérn kernels, parametrised by hyperparameters that are obtained by optimising the marginal likelihood (Rasmussen and Williams, 2006).

6.3.3. Extending Gaussian Processes to Multiple Data Sources

To extend GPs to the multi-source setting, we model evaluations indexed by $\ell \in \mathcal{S} = \{0, 1, \ldots, L\}$, where $\ell = L$ denotes the target source and $\ell = 0$ the cheapest auxiliary source. The joint domain is then the source–augmented input space $\mathcal{S} \times \mathcal{X}$, with $\mathcal{X} \subset \mathbb{R}^D$. Following the MISO framework (Poloczek et al., 2016), we consider each source specific function $u^{(\ell)} : \mathcal{S} \times \mathcal{X} \to \mathbb{R}$:

$$u^{(\ell)}(\mathbf{x}) = u^{(L)}(\mathbf{x}) + \Delta^{(\ell)}(\mathbf{x}), \quad \text{with } \Delta^{(L)} \equiv 0. \tag{6.5}$$

where $u^{(L)} : \mathcal{X} \to \mathbb{R}$ is a latent target function and $\Delta^{(\ell)} : \mathcal{X} \to \mathbb{R}$ a source-dependent discrepancy. We place independent GP priors over both:

$$\begin{aligned} u^{(L)} &\sim \mathcal{GP}(\mu_L, \Sigma_L), \\ \Delta^{(\ell)} &\sim \mathcal{GP}(\mu_\ell, \Sigma_\ell), \quad \forall\, \ell < L. \end{aligned} \tag{6.6}$$

We typically assume $\mu_\ell \equiv 0$, whereas Σ_L and Σ_ℓ are chosen from standard kernel families and learned jointly via marginal likelihood maximisation. Since the sum of independent GPs is again a GP, the combined model defines a joint GP over the source-augmented input space: $u : \mathcal{S} \times \mathcal{X} \to \mathbb{R}$ with $u(\ell, \mathbf{x}) := u^{(\ell)}(\mathbf{x}) \sim \mathcal{GP}(\mu, \Sigma)$ with mean and covariance functions given by:

$$\begin{aligned} \mu(\ell, \mathbf{x}) &= \mathbb{E}[u^{(\ell)}(\mathbf{x})] = \mathbb{E}[u^{(L)}(\mathbf{x})] + \mathbb{E}[\Delta^{(\ell)}(\mathbf{x})], \\ \Sigma\big((\ell, \mathbf{x}), (\ell', \mathbf{x}')\big) &= \Sigma_L(\mathbf{x}, \mathbf{x}') + \delta(\ell, \ell')\Sigma_\ell(\mathbf{x}, \mathbf{x}'), \end{aligned} \tag{6.7}$$

where $\delta(\ell, \ell')$ is the Kronecker delta. This structure captures inter-source correlations through a shared latent process, while allowing source-specific discrepancies. Unlike naïve multi-output GPs (Bonilla et al., 2007) or multi-fidelity models with a sequential hierarchy (Kennedy and O'Hagan, 2000), the additive characteristic explicitly encodes a source structure, allowing for joint inference over the latent function $u^{(L)}$,

discrepancies $\Delta^{(\ell)}$ and hyperparameters from observations $\mathcal{D}_t = \{(\ell_i, \mathbf{x}_i, y_i)\}_{i=1}^{N_t}$, while maintaining flexibility.

In practice, we fit a separate MISO model for each black-box function, i.e. one for the objective f and one for each constraint c_i $\forall i = \{1, ..., G\}$, all defined over the same source-augmented domain $\mathcal{S} \times \mathcal{X}$. Figure 6.1 illustrates how the model can approximate the target source $f^{(L)}$ from only two target source observations $\mathcal{D}^{(L)}$ by leveraging six auxiliary observations $\mathcal{D}^{(\ell)}$.

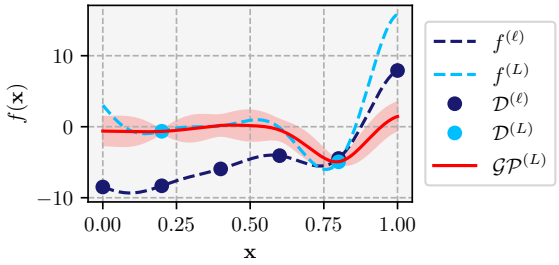


Figure 6.1: Multi-source GP approximation of the target function.

6

6.4. Constrained Max-Value Entropy Search with Multiple Information Sources

We propose Multi-Source Constrained Max-value Entropy Search (MS-CMES), extending the MES (Wang and Jegelka, 2018) principle to settings where both objective and constraints can be queried at multiple information sources. In each iteration we aim to choose the next pair $(\mathbf{x}_+, \ell_+)$ by balancing three factors: (i) expected information gain about the best feasible solution at the target source, (ii) the correlation between auxiliary and target sources, and (iii) the cost of each source, allowing the method to exploit cheap, approximate sources early on. This is achieved by solving

$$(\mathbf{x}_+, \ell_+) = \underset{\mathbf{x}\in\mathcal{X}, \ell\in\mathcal{S}}{\operatorname{argmax}} \frac{\alpha_n(\mathbf{x}, \ell)}{\lambda(\mathbf{x}, \ell)}, \tag{6.8}$$

where $\lambda(\mathbf{x}, \ell)$ denotes the cost of evaluating source ℓ. In the following we assume that each function $f^{(\ell)}$ and constraint $c_i^{(\ell)}$ $\forall i \in \{1, ..., G\}$ can be observed through any source $\ell \in \mathcal{S}$.

Mutual Information Gain. Following the MES principle, the function α_t maximises the expected information gain about the constrained optimum

$$f^* = \max_{\mathbf{x}\in\mathcal{X}_f^{(L)}} f^{(L)}(\mathbf{x}) \tag{6.9}$$

at the target source L:

$$\alpha_t(\mathbf{x}, \ell) = \mathbb{I}\left(\mathbf{u}^{(\ell)}(\mathbf{x}); f^*\right). \tag{6.10}$$

with $\mathbf{u}^{(\ell)}(\mathbf{x}) = [f^{(\ell)}(\mathbf{x}), c_1^{(\ell)}(\mathbf{x}), \ldots, c_G^{(\ell)}(\mathbf{x})] \in \mathbb{R}^{G+1}$. In the following we will omit that $\mathbf{u}^{(\ell)}$ depends on $\mathbf{x}$. This expression measures how much querying $(\mathbf{x}, \ell)$ reduces uncertainty over f^*, with all quantities modelled via the joint GP framework described in Section 6.3.3. As f^* is unknown, we approximate it from the GP posteriors via:

$$f^* := \begin{cases} \max\limits_{\mathbf{x} \in \mathcal{X}_f^{(L)}} f^{(L)}(\mathbf{x}) & \text{if } \mathcal{X}_f^{(L)} \neq \emptyset \\ f^{(L)}\left(\operatorname{argmin}\limits_{\mathbf{x} \in \mathcal{X}} \sum_j \bar{c}_j^{(L)}\right) & \text{else} \end{cases} \tag{6.11}$$

with $\bar{c}_j^{(L)} = \max\left(0, c_j^{(L)}(\mathbf{x})\right)$. This fallback ensures that, in the absence of any feasible point at the target source (i.e. when $\mathcal{X}_f^{(L)} = \emptyset$), the algorithm focuses on reducing uncertainty around the most promising infeasible region. Based on this definition of f^*, mutual information can be written as

$$\begin{aligned} \mathbb{I}(\mathbf{u}^{(\ell)}; f^*) &= \mathbb{E}_{f^*}\left[\mathrm{D}_{\mathrm{KL}}\left(p(\mathbf{u}^{(\ell)} \mid f^*) \,\|\, p(\mathbf{u}^{(\ell)})\right)\right] \\ &= \mathbb{E}_{f^*}\left[\mathbb{E}_{\mathbf{u}^{(\ell)} \mid f^*}\left[\log \frac{p(\mathbf{u}^{(\ell)} \mid f^*)}{p(\mathbf{u}^{(\ell)})}\right]\right] \end{aligned} \tag{6.12}$$

However, $p(\mathbf{u}^{(\ell)} \mid f^*)$ is intractable to compute directly. Therefore, we adopt the variational lower bound on mutual information as derived by (Takeno et al., 2022), adapted to our multi-source context. By introducing a variational distribution $q(\mathbf{u}^{(\ell)} \mid f^*)$, we can write the information gain as:

$$\mathbb{I}(\mathbf{u}^{(\ell)}; f^*) \geq \underbrace{\mathbb{E}_{f^*}\left[\mathbb{E}_{\mathbf{u}^{(\ell)} \mid f^*} \log \frac{q(\mathbf{u}^{(\ell)} \mid f^*)}{p(\mathbf{u}^{(\ell)})}\right]}_{:= \alpha_n(\mathbf{x}, \ell)}. \tag{6.13}$$

This lower bound quantifies the expected information gain from evaluating $(\mathbf{x}, \ell)$, i.e. whether the outcome falls within the feasible region defined by the current value of the constrained optimum f^*. The variational distribution $q(\mathbf{u} \mid f^*)$ is defined as the normalised posterior over the joint outputs, restricted to the feasible set $\mathcal{F} := (-\infty, f^*] \times (-\infty, 0]^g \subset \mathbb{R}^{G+1}$:

$$q(\mathbf{u}^{(\ell)} \mid f^*) = \begin{cases} \dfrac{p(\mathbf{u}^{(\ell)})}{\Pr(\mathbf{u}^{(\ell)} \in \mathcal{F})} & \text{if } \mathbf{u}^{(\ell)} \in \mathcal{F} \\ 0 & \text{otherwise} \end{cases} \tag{6.14}$$

Substituting Equation (6.14) into the variational lower bound in Equation (6.13) yields the acquisition function

$$\begin{aligned}\alpha_n(\mathbf{x}, \ell) &:= \mathbb{E}_{f^*}\left[\mathbb{E}_{\mathbf{u}^{(\ell)}|f^*} \log \frac{q(\mathbf{u}^{(\ell)} \mid f^*)}{p(\mathbf{u}^{(\ell)})}\right] \\ &\approx -\frac{1}{K}\sum_{k=1}^{K} \log \Pr(\mathbf{u}^{(\ell)} \in \mathcal{F})\end{aligned} \tag{6.15}$$

where

$$\begin{aligned}&\Pr(\mathbf{u}^{(\ell)} \in \mathcal{F}) \\ &\quad = \Pr\left(f^{(\ell)}(\mathbf{x}) \leq f^*\right) \prod_{i=1}^{G} \Pr\left(c_i^{(\ell)}(\mathbf{x}) \leq 0\right)\end{aligned} \tag{6.16}$$

The expectation is estimated by drawing K samples $\{f_k^*\}_{k=1}^K$ of the constrained optimum using discrete TS from the GP posterior at the target source L. A full derivation is provided in Appendix 6.7.1.

Variance correction. As shown in Equations (6.15) and (6.16), we are interested in $\Pr(\mathbf{u}^{(\ell)} \in \mathcal{F})$. However, this introduces a mismatch: while $\mathcal{F}$ is defined at the target source L, $\mathbf{u}^{(\ell)}$ is potentially computed on a different source with $\ell \neq L$. Thus, a candidate may appear feasible under source ℓ, yet violating constraints or optimality at source L, since $\mathcal{X}_f^{(L)} \neq \mathcal{X}_f^{(\ell)}$. The authors in Moss et al. (2020) show that $\mathbf{u}^{(\ell)} \mid f^{(L)}$ is not a truncated Gaussian but rather an extended skew Gaussian distribution, which they approximate in (Moss et al., 2020, 2021) via a variance-correction by quantifying the correlation $\rho(\mathbf{x}, \ell) \in (0, 1]$ between source ℓ and target L. This mechanism ensures that weakly informative sources are automatically penalised, preventing spurious feasibility or infeasibility from dominating early-stage acquisition decisions.

Since the MISO model defines a jointly correlated Gaussian process across all fidelities, observations at auxiliary sources do influence the target posterior through the cross-covariance structure. However, the optimisation objective concerns only the target fidelity $f^{(L)}$. Therefore, the truncation event defining the optimum $f^\star$ is applied to the marginal posterior of the target process. Auxiliary fidelities are not truncated directly, instead, their contribution to learning about the truncated target is mediated through their correlation $\rho(\mathbf{x}, \ell)$ with the target process. Under the Gaussian approximation used in the acquisition function, the joint posterior remains Gaussian, and we retain the truncated Gaussian approximation in Equation (6.15). Following Moss et al. (2021), we introduce a variance correction term to account for the reduced information content contributed by auxiliary sources relative to the target fidelity, yielding:

$$\tilde{\sigma}^{(\ell)}(\mathbf{x}) \approx \sigma^{(L)}(\mathbf{x}) \left(1 - \rho^2(\mathbf{x}, \ell)\Psi\left(\gamma^{(L)}\right)\right), \tag{6.17}$$

where $\gamma^{(L)} = \frac{t-\mu^{(L)}(\mathbf{x})}{\sigma^{(L)}(\mathbf{x})}$ and $\Psi(\gamma) = \frac{\phi(\gamma)}{\Phi(\gamma)}\left(\gamma + \frac{\phi(\gamma)}{\Phi(\gamma)}\right)$. We set $t = f^*$ for the objective and $t = 0$ for the constraints, respectively. Hence, the adjusted feasibility probabilities from Equation (6.16) can be computed via:

$$\begin{aligned} \Pr\left(f^{(\ell)}(\mathbf{x}) \leq f^*\right) &\approx \Phi\left(\frac{f^* - \mu_f^{(\ell)}(\mathbf{x})}{\tilde{\sigma}_f^{(\ell)}(\mathbf{x})}\right) \\ \Pr\left(c_i^{(\ell)}(\mathbf{x}) \leq 0\right) &\approx \Phi\left(-\frac{\mu_{c_i}^{(\ell)}(\mathbf{x})}{\tilde{\sigma}_{c_i}^{(\ell)}(\mathbf{x})}\right). \end{aligned} \tag{6.18}$$

with $\Phi(\bullet)$ being the cumulative distribution function of a standard Gaussian. We emphasise that the corrected variance $\tilde{\sigma}^{(\ell)}$ is always computed with respect to the predictive mean $\mu^{(L)}$ and variance $\sigma^{(L)}$ of data source L, since the goal is to reduce uncertainty about the target-source optimum f^*. This follows from the fact that the mutual information gained by evaluating at source ℓ can be interpreted as upper bounded:

$$\mathbb{I}(\mathbf{u}^{(\ell)}(\mathbf{x}), f^*) \leq \mathbb{I}(\mathbf{u}^{(L)}(\mathbf{x}), f^*), \tag{6.19}$$

with reduction scaled by $\rho^2(\mathbf{x}, \ell)$, as illustrated in Figure 6.2. Importantly, the scaling also prevents overconfidence of auxiliary models which may appear more certain due to being trained on a larger data set.

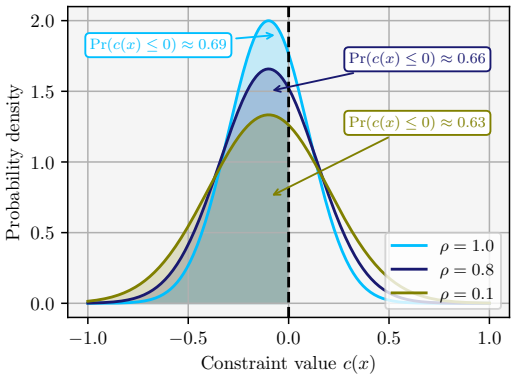


Figure 6.2: Illustration of the variance correction (see Equation (6.18)), depending on the correlation coefficient $\rho \in [1.0, 0.8, 0.1]$ whereas $\rho = 1.0$ denotes perfect correlation.

This correction is crucial in early-stage optimisation, where feasible regions are unknown and auxiliary sources may be misleading. By inflating variances at untrusted sources (when $\rho^2 \approx 0$), the model can still assign non-negligible feasibility probability as uncertainty is high. Rather than discarding such points, the acquisition function keeps encouraging exploration near the feasibility boundary. Details on how the correlation is computed in case of the MISO model can be found in Appendix 6.7.1

Algorithm 8 MS-CMES: Multi-Source Constrained Max-value Entropy Search

Require: Initial multi-source $\mathcal{GP}$ models for the objective and constraints, budget B, cost model $\lambda(x, \ell)$, Number of MC samples K

1: Initialise data $\mathcal{D}_t \leftarrow \{(\mathbf{x}_i, \ell_i, f^{(\ell_i)}(\mathbf{x}_i), c_j^{(\ell_i)}(\mathbf{x}_i))\}_{i=1}^{N_t}$
2: **while** Computational budget is not exhausted **do**
3: Sample $\{f_k^*\}_{k=1}^K$ using $\mathcal{GP}_{(\cdot)}^{(L)}$ posteriors in $\mathcal{T}$
4: Solve $(\mathbf{x}_+, \ell_+) \leftarrow \operatorname{argmax}_{\mathbf{x}\in\mathcal{T},\, \ell\in\mathcal{S}} \frac{\alpha_n(\mathbf{x},\ell)}{\lambda(\mathbf{x},\ell)}$
5: Query all outputs at $(\mathbf{x}_+, \ell_+)$
6: $\mathcal{D}_{t+1} \leftarrow \mathcal{D}_t \cup \{(\mathbf{x}_+, \ell_+, f^{(\ell_+)}(\mathbf{x}_+), c_j^{(\ell_+)}(\mathbf{x}_+))\}$
7: Update GP models using $\mathcal{D}_{t+1}$
8: **end while**

Trust region acquisition optimisation. To further increase the efficiency and scalability, we restrict acquisition optimisation to a dynamically updated TR, centred at the best observed point $\mathbf{x}^\dagger \in \mathcal{D}_t$, restricted to all points where $\ell = L$, following the idea of (Eriksson and Poloczek, 2021):

$$\mathbf{x}^\dagger = \underset{(\mathbf{x}_i, L)\in\mathcal{D}_t}{\operatorname{argmax}} \; f^{(L)}(\mathbf{x}_i) \quad \text{s.t.} \quad c_j^{(L)}(\mathbf{x}_i) \leq 0, \; \forall j, \tag{6.20}$$

or, if no feasible point is known, as the one with minimal total constraint violation. The TR is defined as a hypercube around $\mathbf{x}^\dagger$ with side length r, as

$$\mathcal{T}(\mathbf{x}^\dagger, r) = \left\{\mathbf{x} \in \mathcal{X} \,\middle|\, \mathbf{x}^\dagger - \tfrac{r}{2} \leq \mathbf{x} \leq \mathbf{x}^\dagger + \tfrac{r}{2}\right\}, \tag{6.21}$$

clipped to the domain $\mathcal{X} = [0, 1]^D$. Hence, the K samples $\{f_k^*\}_{k=1}^K$ are generated within $\mathcal{T}$, and candidate points are then selected by

$$(\mathbf{x}_+, \ell_+) = \underset{\mathbf{x}\in\mathcal{T}(\mathbf{x}^\dagger, r),\, \ell\in\mathcal{S}}{\operatorname{argmax}} \; \frac{\alpha_t(\mathbf{x}, \ell)}{\lambda(\mathbf{x}, \ell)}. \tag{6.22}$$

Depending on the progress of the optimisation, r shrinks or expands. More details can be found in Appendix 6.7.1. A summary of the acquisition strategy is provided in Algorithm 8, where the optimisation problem in Line 4 is solved using gradient ascent. The complexity is discussed in Appendix 6.7.2.

Summarising, the key novelty lies in combining a constrained, entropy-based information gain criterion with explicit multi-source modelling and variance correction. Unlike previous single-source constrained BO methods (Takeno et al., 2022) or multi-source methods without constraints (Poloczek et al., 2016), MS-CMES provides the first principled acquisition rule for constrained optimisation that automatically down-weights uninformative sources.

6.5. Numerical Experiments

We present numerical experiments that first compare multi-source models and their scalability with increasing dimensionality, then benchmark the proposed MS-CMES acquisition strategy against state-of-the-art methods, followed by an ablation study and parameter sensitivity analysis.

6.5.1. Scalability of Gaussian Processes with Multiple Data Sources

We compare the performance of the MISO model, Subsection 6.3.3, against three representative alternatives: the KOH model (Kennedy and O'Hagan, 2000, Forrester et al., 2007), MTGP (Bonilla et al., 2007), and a standard GP trained only on target-source data. As test function we use the Rosenbrock function (Rosenbrock, 1960) in dimensions $d \in \{10, 50, 100\}$. The target source contains $n_L = D$ data points, while the auxiliary source contains $n_\ell = 4D$. Auxiliary signals with no, weak, and strong correlation are constructed following the procedure in Appendix 6.7.4. Figure 6.3 reports the normalised RMSE averaged over 10 random seeds.

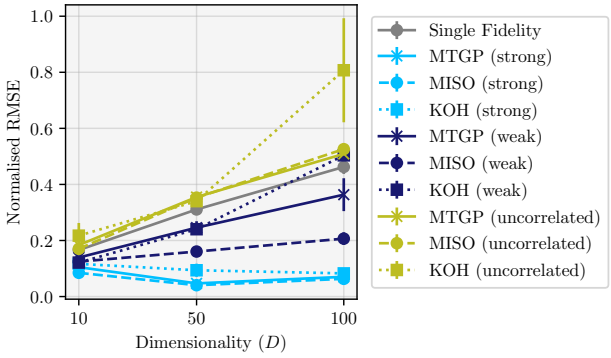


Figure 6.3: Model comparison across problem dimensionalities $D \in \{10, 50, 100\}$ using the Rosenbrock benchmark. Results show the normalised RMSE averaged over ten random seeds, error bars indicate 1σ standard deviation.

When auxiliary and target sources are strongly correlated, all multi-source models achieve comparable accuracy and consistently outperform the single-source GP. This behaviour is expected in settings such as mesh-refined aerodynamic solvers, where lower-cost simulations preserve much of the structural information of their high-cost counterparts.

By contrast, strong correlations cannot always be assumed. In scenarios where simulators represent the same physical system but differ substantially, or where simulation data complements experimental data, the outputs are often weakly correlated or structurally divergent. In such cases, models such as KOH and MTGP,

which impose a strict hierarchy or assume shared latent structure, show degraded performance. MISO, by explicitly modelling source-specific discrepancies, remains robust across a wide range of inter-source correlations. Notably, even when auxiliary sources are entirely uninformative or uncorrelated, the performances of MISO and MTGP do not degrade much below that of a single-source GP trained only on high-fidelity data. In contrast, KOH, which assumes a strict hierarchy between sources, performs significantly worse in such settings. This robustness makes MISO particularly well-suited for general multi-source optimisation, especially when the relationships between sources are unknown, weak, or heterogeneous.

6.5.2. Benchmark Tests

To evaluate performance and scalability, we select five constrained benchmark problems with dimensionalities ranging from $D = 4-100$. These include the physics-based *Pressure Vessel* benchmark (Coello Coello and Mezura Montes, 2002), the *Rosenbrock* function with two constraints (Rosenbrock, 1960), the *(Rotated) Rastrigin* and the *Different Powers* function (Dufossé et al., 2022). More information can be found in Appendix 6.7.7. We use the approach presented in Appendix 6.7.4 to derive a corrupted, weakly correlated signal for each objective and constraint of the respective function. Moreover, we focus on the practically relevant regime of low to moderate evaluation budgets, specifically up to 200 evaluations of the target information source, as common in scenarios where evaluation costs of the target source are immense (Pretsch et al., 2025). The corresponding code can be found at `github.com/haukemaa/ms-cmes`.

MS-CMES Setup. In our MS-CMES implementation, we adopt a constant cost model, depending only on the data source, with cost weights $c_L = 1000$ for the target source and $c_\ell = 1$ for the auxiliary source (more information on the cost function can be found in Appendix 6.7.6). This translates to an evaluation budget of $c \approx 2 \cdot 10^5$. We use $K = 32$ samples to approximate f^* and initialise the optimisation with a ratio of $n_\ell / n_L = 5$ points during the design of experiments.

Baseline methods. We compare our method against several state-of-the-art single- and multi-source approaches for constrained BO. These include SCBO (Eriksson and Poloczek, 2021), FuRBO (Ascia et al., 2025), a VBO baseline with dimensionality-scaled length-scale priors (Hvarfner et al., 2024) using LogCEI (Ament et al., 2023) as the acquisition function, and a random search baseline. We also compare against Constrained MES via Information Lower Bound (CMES-IBO), the information-theoretic constrained BO method proposed by Takeno et al. (2022), which our work extends. In Appendix 6.7.5, we introduce a minor modification to this method and refer to the modified version as CMES-IBO+. Additionally, we include Constrained Multi-Fidelity Bayesian Optimisation (CMFBO) from Foumani and Bostanabad (2025) as a representative fidelity-aware baseline. We do not include PESC (Hernández-Lobato et al., 2017) in our comparison, as it has previously been shown to be outperformed

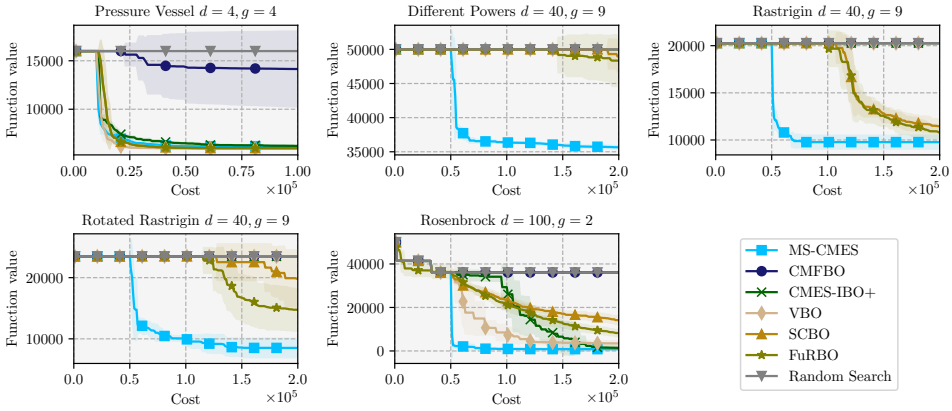


Figure 6.4: Comparison of optimisation performance across five constrained benchmark problems with dimensionalities up to $d = 100$. Results are averaged over ten random seeds, shaded regions indicate 1σ standard deviation.

by SCBO (Eriksson and Poloczek, 2021).

Results. All GP models use Matérn kernels and are trained jointly across sources. For benchmarks with $D \geq 40$, we use a batch size of $q = 5$ and $N_0 = 50$ initial samples, while for the Pressure Vessel benchmark we use $q = 1$ and $N_0 = 10$ initial samples. For multi-source methods we use $5N_0$ auxiliary source samples. The results are depicted in Figure 6.4, averaged over ten random seeds with mean and variance reported. We emphasise that in multi-source methods only outputs on the target source $f^{(L)}, c_1^{(L)}, ..., c_G^{(L)}$ determine the optimality and feasibility of a sample. Moreover, when no feasible point has been found, we assign its value to the highest feasible objective discovered so far.

On the *Pressure Vessel* benchmark, all methods perform similarly, though MS-CMES achieves slightly better results than CMES-IBO+. CMFBO struggles to converge and random search fails to discover a feasible point. On the *Rastrigin* family (standard, rotated) and the *Different Powers* benchmark, most methods find it difficult to locate the first feasible point. Here, MS-CMES stands out, significantly outperforming alternatives, whereas CMFBO, CMES-IBO+, VBO, and random never find a single feasible sample. SCBO performs somewhat better but is eventually surpassed by FuRBO. On the *Rosenbrock* benchmark, CMFBO and random again fail to converge, while VBO and CMES-IBO+ achieve promising final values but are clearly outperformed by MS-CMES.

Overall, MS-CMES delivers the strongest performance, particularly on the more challenging high-dimensional problems. On the *Pressure Vessel* benchmark, per-

formance is broadly comparable across methods, though MS-CMES still surpasses CMES-IBO+, highlighting the benefits of incorporating auxiliary information. The advantage of our approach becomes especially evident in higher dimensions: whereas most baselines fail to identify any feasible point even after many iterations, MS-CMES discovers feasible solutions immediately after the initial design. Although the auxiliary signals are only weakly correlated (see Appendix 6.7.4), exploiting them proves highly beneficial, enabling MS-CMES to locate feasible regions early and substantially enhance optimisation performance.

6.5.3. Ablation and Parameter Study

We further examine the effect of the TR heuristic by comparing MS-CMES with and without TR-constrained acquisition optimisation, see Figure 6.9 in Appendix 6.7.8. In the unconstrained variant, f^* is sampled and Equation 6.22 is optimised over the full domain $\mathcal{X}$ rather than within the adaptive TR space $\mathcal{T}$. Both variants perform similarly on *Pressure Vessel* and *Rosenbrock*, but the TR heuristic provides clear efficiency gains on the more challenging high-dimensional problems. This indicates that dynamically restricting the search space helps concentrate exploration and prevents drifting into unpromising regions.

We also investigate the impact of the number of Monte Carlo samples K used to estimate the entropy bound. Figure 6.8 in Appendix 6.7.8 shows that while performance improves slightly for larger K, moderate values as $K = 10$ are already sufficient. This underlines the robustness of the method, consistent with the findings of Takeno et al. (2022).

6.6. Conclusion

We introduced MS-CMES, the first information-theoretic framework for constrained Bayesian optimisation with multiple information sources. By explicitly modelling inter-source correlations and evaluation costs, MS-CMES leverages inexpensive, approximate sources for early exploration while reserving costly target evaluations for refinement. Across a diverse set of benchmarks, we demonstrated that MS-CMES consistently outperforms established baselines in both identifying feasible regions and converging towards optimal solutions. These results highlight the promise of multi-source modelling for constrained optimisation in domains ranging from engineering design to drug discovery.

Limitations. Our current formulation assumes that the objective and all constraints are evaluated jointly at a given source. While this assumption holds naturally in many simulation-based settings, it is restrictive in cases where partial observations are available (e.g. objective-only or constraint-only measurements). Extending MS-CMES to accommodate selective output queries would require acquisition strategies capable of balancing heterogeneous information gains. Moreover, the method currently presumes that auxiliary sources exist for every objective and constraint.

In problems where this is not the case, the framework could easily be adapted by defaulting to single-source models for missing signals. Finally, performance depends on the quality of the user-specified cost function $\lambda(\mathbf{x}, \ell)$. If costs are poorly estimated, the optimisation may misallocate evaluations, making careful cost modelling crucial in practice.

Future Work. Several promising directions remain. Extending the framework to support partial-output queries would increase flexibility in settings where objectives and constraints can be decoupled. In addition, scalable generative models trained on archival data offer a particularly promising avenue. We believe that by treating such models as auxiliary information sources, organisations could recycle past experiments and simulations, effectively turning historical data into a reusable asset that accelerates and guides future optimisation.

6.7. Appendix

6.7.1. Details on MS-CMES

This appendix complements Section 6.4 by providing detailed derivations for the variational bound used in MS-CMES, the associated variance correction mechanism that accounts for discrepancies between information sources and the TR heuristic.

Variational Bound and Variance Correction in MS-CMES

We aim to compute the mutual information between a new observation $\mathbf{u}^{(\ell)}$ and the unknown constrained optimum f^*. This quantity is defined as:

$$\begin{aligned}\mathbb{I}(\mathbf{u}^{(\ell)}; f^*) &= \mathbb{E}_{f^*}\left[\mathrm{D}_{\mathrm{KL}}\left(p(\mathbf{u}^{(\ell)} \mid f^*) \,\|\, p(\mathbf{u}^{(\ell)})\right)\right] \\ &= \mathbb{E}_{f^*}\underbrace{\left[\mathbb{E}_{\mathbf{u}^{(\ell)}|f^*}\left[\log \frac{p(\mathbf{u}^{(\ell)} \mid f^*)}{p(\mathbf{u}^{(\ell)})}\right]\right]}_{(*)}\end{aligned} \tag{6.23}$$

This follows from the standard mutual information identity:

$$\mathbb{I}(X; Y) = \mathbb{E}_Y\left[\mathrm{D}_{\mathrm{KL}}\left(p(X \mid Y) \,\|\, p(X)\right)\right] \tag{6.24}$$

with D_{KL} being the Kullback-Leibler divergence. However, $p(\mathbf{u}^{(\ell)} \mid f^*)$ is intractable to compute directly. We continue to follow the approach of Takeno et al. (2022) who introduce a variational distribution $q(\mathbf{u}^{(\ell)} \mid f^*)$:

$$\begin{aligned}\mathbb{E}_{\mathbf{u}^{(\ell)}|f^*}\left[\log \frac{p(\mathbf{u}^{(\ell)} \mid f^*)}{p(\mathbf{u}^{(\ell)})}\right] &= \mathbb{E}_{\mathbf{u}^{(\ell)}|f^*}\left[\log \frac{q(\mathbf{u}^{(\ell)} \mid f^*)}{p(\mathbf{u}^{(\ell)})}\right] \\ &\quad + \mathrm{D}_{\mathrm{KL}}\left(q(\mathbf{u}^{(\ell)} \mid f^*) \,\|\, p(\mathbf{u}^{(\ell)} \mid f^*)\right) \\ &\geq \mathbb{E}_{\mathbf{u}^{(\ell)}|f^*}\left[\log \frac{q(\mathbf{u}^{(\ell)} \mid f^*)}{p(\mathbf{u}^{(\ell)})}\right]\end{aligned} \tag{6.25}$$

Since $\mathrm{D}_{\mathrm{KL}}(\cdot \,\|\, \cdot) \geq 0$ we end up at the information source-dependent lower bound. Inserting Equation (6.25) into Equation (6.23), we can write:

$$\mathbb{I}(\mathbf{u}^{(\ell)}; f^*) \geq \mathbb{E}_{f^*}\left[\mathbb{E}_{\mathbf{u}^{(\ell)}|f^*} \log \frac{q(\mathbf{u}^{(\ell)} \mid f^*)}{p(\mathbf{u}^{(\ell)}(\mathbf{x}))}\right] \tag{6.26}$$

This lower bound quantifies the expected information gain obtained by observing whether the evaluation $(\mathbf{x}, \ell)$ yields a realisation within the feasible region associated with the current value of the constrained optimum f^*. In the unconstrained case, the variational distribution $q(\mathbf{u} \mid f^*)$ corresponds to a truncated Gaussian. In the constrained setting, we define $q(\mathbf{u} \mid f^*)$ as the normalised posterior over the joint outputs, restricted to the feasible set $\mathcal{F} \subset \mathbb{R}^{g+1}$, i.e. $\mathcal{F} := (-\infty, f^*] \times (-\infty, 0]^g$. We define the feasibility probability $\Pr(\mathbf{u}^{(\ell)} \in \mathcal{F})$ at data source ℓ as:

$$\Pr(\mathbf{u}^{(\ell)} \in \mathcal{F}) = \Pr\left(f^{(\ell)}(\mathbf{x}) \leq f^*\right) \prod_{i=1}^{G} \Pr\left(c_i^{(\ell)}(\mathbf{x}) \leq 0\right) \tag{6.27}$$

where $\Phi(\bullet)$ is the commulative distribution function of a standard Gaussian. Using this, the variational distribution $q(\mathbf{u}^{(\ell)} \mid f^*)$ is defined as

$$q(\mathbf{u}^{(\ell)} \mid f^*) = \begin{cases} \dfrac{p(\mathbf{u}^{(\ell)})}{\Pr(\mathbf{u}^{(\ell)} \in \mathcal{F})} & \text{if } \mathbf{u}^{(\ell)} \in \mathcal{F} \\ 0 & \text{otherwise} \end{cases} \tag{6.28}$$

Substituting Equation (6.28) into the lower bound in Equation (6.26) gives:

$$\begin{aligned} \alpha_n(\mathbf{x}, \ell) &:= \mathbb{E}_{f^*}\left[\mathbb{E}_{\mathbf{u}^{(\ell)}|f^*} \log \frac{q(\mathbf{u}^{(\ell)} \mid f^*)}{p(\mathbf{u}^{(\ell)})}\right] \\ &= \mathbb{E}_{f^*}\left[\int p(\mathbf{u}^{(\ell)}|f^*) \log \frac{q(\mathbf{u}^{(\ell)}|f^*)}{p(\mathbf{u}^{(\ell)})}\, d\mathbf{u}^{(\ell)}\right] \\ &= \mathbb{E}_{f^*}\left[\int p(\mathbf{u}^{(\ell)}|f^*) \log \frac{p(\mathbf{u}^{(\ell)})}{\Pr(\mathbf{u}^{(\ell)} \in \mathcal{F}) p(\mathbf{u}^{(\ell)})}\, d\mathbf{u}^{(\ell)}\right] \\ &= \mathbb{E}_{f^*}\left[\int p(\mathbf{u}^{(\ell)}|f^*) \log \frac{1}{\Pr(\mathbf{u}^{(\ell)} \in \mathcal{F})}\, d\mathbf{u}^{(\ell)}\right] \\ &= \mathbb{E}_{f^*}\left[-\log \Pr(\mathbf{u}^{(\ell)} \in \mathcal{F}) \int p(\mathbf{u}^{(\ell)}|f^*)\, d\mathbf{u}^{(\ell)}\right] \\ &= \mathbb{E}_{f^*}\left[-\log \Pr(\mathbf{u}^{(\ell)} \in \mathcal{F})\right] \end{aligned} \tag{6.29}$$

Finally, the expected value is approximated with K samples as

$$\mathbb{E}_{f^*}\left[-\log \Pr(\mathbf{u}^{(\ell)} \in \mathcal{F})\right] \approx -\frac{1}{K} \sum_{k=1}^{K} \log \Pr(\mathbf{u}^{(\ell)} \in \mathcal{F}) \tag{6.30}$$

Variance Correction

Recalling the setup of the multi-source model in Equations (6.5) and (6.7), we can quantify the correlation (Rasmussen and Williams, 2006) with:

$$\begin{aligned}\rho(\mathbf{x}, \ell) &= \frac{\mathrm{Cov}\left(u^{(L)}(\mathbf{x}), u^{(\ell)}(\mathbf{x})\right)}{\sigma_L(\mathbf{x})\sigma_\ell(\mathbf{x})} \\ &= \frac{\sigma_L(\mathbf{x})}{\sigma_L(\mathbf{x}) + \sigma_\Delta(\mathbf{x})} \in (0, 1].\end{aligned} \tag{6.31}$$

with $\mathrm{Cov}\big(u^{(\ell)}(\mathbf{x}), u^{(L)}(\mathbf{x})\big) = \mathrm{Var}\big(u^{(L)}(\mathbf{x})\big) = {\sigma_L}^2$. From that, it follows that $\rho(\mathbf{x}, L) \equiv 1$. The discrepancy $\Delta^{(\ell)}$ increases predictive variance at more inaccurate data sources which changes the shape of the conditional distribution.

Trust-Region Constrained Optimisation of the Acquisition Function

We constrain the optimisation of the acquisition function to a dynamically updated TR in order to stabilise the search. This idea, inspired by Eriksson and Poloczek (2021), focuses optimisation on promising neighbourhoods around the current best observed solution, which is particularly beneficial in high-dimensional settings.

At iteration t, let the dataset of all evaluations be

$$\mathcal{D}_t = \{(\mathbf{x}_i, \ell_i, f^{(\ell_i)}(\mathbf{x}_i), c_1^{(\ell_i)}(\mathbf{x}_i), \ldots, c_G^{(\ell_i)}(\mathbf{x}_i))\}_{i=1}^{N_t} \tag{6.32}$$

We select $\mathbf{x}^\dagger$ from the subset of points evaluated at the target source L, i.e. $\{(\mathbf{x}_i, \ell_i) \in \mathcal{D}_t \mid \ell_i = L\}$. If at least one feasible point has been observed, $\mathbf{x}^\dagger$ is defined as the best feasible point:

$$\mathbf{x}^\dagger = \underset{(\mathbf{x}_i, L) \in \mathcal{D}_n}{\operatorname{argmax}} \; f^{(L)}(\mathbf{x}_i) \quad \text{s.t.} \quad c_j^{(L)}(\mathbf{x}_i) \leq 0 \; \forall j. \tag{6.33}$$

If no feasible point exists, $\mathbf{x}^*$ is chosen as the infeasible point with the smallest total constraint violation:

$$\mathbf{x}^* = \underset{(\mathbf{x}_i, \ell_i) \in \mathcal{D}_t, \; \ell_i = L}{\operatorname{argmin}} \sum_{j=1}^{G} \max\big(0, c_j^{(L)}(\mathbf{x}_i)\big). \tag{6.34}$$

The TR is defined as a hypercube centred at $\mathbf{x}^*$ with side length r:

$$\mathcal{T}(\mathbf{x}^\dagger, r) = \left\{\mathbf{x} \in \mathcal{X} \,\middle|\, x_z^\dagger - \tfrac{r}{2} \leq x_z \leq x_z^\dagger + \tfrac{r}{2}, \; z = 1, \ldots, d\right\}, \tag{6.35}$$

clipped to the global domain $\mathcal{X} = [0, 1]^D$.

The side length r is adapted dynamically. Let s_n and f_n denote counters of consecutive successes and failures, respectively. A *success* occurs if the new evaluation

improves upon the best feasible objective, or reduces total constraint violation relative to $\mathbf{x}^*$. Otherwise, a *failure* is recorded. When s_n reaches a tolerance $s_{\max}$, the region is expanded as $r \leftarrow \min(2r, r_{\max})$ and s_n is reset. Conversely, if f_n reaches $f_{\max}$, the region is shrunk as $r \leftarrow \frac{1}{2}r$ and f_n reset. If r falls below $r_{\min}$, the TR is restarted around the current best point.

At each iteration, new candidates are chosen by optimising the cost-scaled acquisition function within the TR:

$$(\mathbf{x}_+, \ell_+) = \underset{\mathbf{x}\in\mathcal{T}(\mathbf{x}^\dagger, r),\, \ell\in\mathcal{S}}{\operatorname{argmax}} \frac{\alpha_n(\mathbf{x}, \ell)}{\lambda(\mathbf{x}, \ell)}. \tag{6.36}$$

This strategy prevents the optimiser from wasting evaluations in unpromising regions (particularly when auxiliary sources are misleading) and provides a principled balance between local refinement and global exploration through adaptive expansion and contraction of the TR.

6.7.2. Computational Complexity

We briefly analyse the computational complexity of the proposed method, separating the cost of model training from that of acquisition optimisation.

Multi-Source Gaussian Process Model Let $N = \sum_{\ell\in\mathcal{S}} N_\ell$ denote the total number of observations across sources. The training and inference requires a Cholesky factorisation of the $N \times N$ covariance matrix, leading to the classical complexities for time and memory, $\mathcal{O}(N^3)$ and $\mathcal{O}(N^2)$. Optimisation of the hyperparameters requires repeated gradient evaluations, hence scales identically. Due to the additive structure, additional kernel parameters are introduced due to the discrepancy GP $\Delta^{(\ell)}$, compared to a standard, single-source GP. For large data sets N, inducing point methods such as Structured Kernel Interpolation (SKI) or Sparse GPs could be applicable.

Acquisition function (MS-CMES) The cost of evaluating the acquisition function decomposes as follows: We draw K samples from the posterior of the constrained optimum on a candidate set of size N_c to obtain f^*. This costs $\mathcal{O}(KN_c)$ plus the cost of GP posterior evaluation on N_c candidates. Evaluating $\alpha(\mathbf{x}, \ell)$, for each candidate $\mathbf{x}$, computing feasibility probabilities involves Gaussian CDFs of the posterior mean/variance. With g constraints, a batch of b candidates and $|\mathcal{S}|$ number of sources, the complexity yields $\mathcal{O}(bK(G+1)|\mathcal{S}|)$. In total this leads to $\mathcal{O}(KN_c + bK(G+1)|\mathcal{S}|)$.

Computational resources in this work. All experiments were carried out on a compute cluster equipped with Intel Xeon Gold 5218 CPUs (2.3 GHz). All jobs were executed on single CPU nodes without GPU acceleration. Runtime per experiment varied with problem dimensionality and evaluation budget but remained within a few

hours for the largest benchmarks, showing that the method does not need extensive compute.

6.7.3. Details on Implementation of Models

MISO Model We implement a variant of the MISO model based on the structure proposed by Poloczek et al. (2016). This subsection accompanies the mathematical description in Section 6.3.3. Our model assumes a hierarchical design of experiments, wherein each target source evaluation is accompanied by evaluations from all auxiliary sources at the same input location. This nested design is motivated by the assumption that auxiliary sources are inexpensive to query, allowing evaluations to be collected "for free" whenever a target source point is selected by the acquisition function. The model decomposes the objective into a shared latent function and source-specific discrepancies. Both the base kernel and the discrepancy kernels are instantiated as Matérn-5/2 kernels. Discrepancy kernels are masked to activate only on data from their associated source level. These components are combined additively in the covariance module. To capture the magnitude of inter-fidelity variation, we place a log-normal prior on the outputscale of each discrepancy kernel. The prior's mean is set to the empirical mean squared discrepancy between target and auxiliary source observations across the training data, extracted from the nested design. This reflects an informed prior belief on the relative strength of the discrepancy signal. Finally, the model exposes a utility function that returns the local squared correlation $\rho^2(x)$ between each auxiliary source and the target source, based on the kernel structure. We implemented the model with the help of `GPyTorch` (Gardner et al., 2021) and use it within `BoTorch` (Balandat et al., 2020) for optimisation.

KOH Model The KOH model (Kennedy and O'Hagan, 2000) is a seminal framework for multi-fidelity/ multi-source GP modelling, originally proposed for calibrating computer models using experimental data. It has since become a cornerstone in multi-source BO and surrogate modelling, particularly when relating a cheap but approximate simulator to a more expensive high-fidelity function.

In its canonical form, the KOH model assumes a hierarchical structure that expresses the high-fidelity response $f_L(\mathbf{x})$ as a scaled version of the low-fidelity model $f_\ell(\mathbf{x})$, plus a discrepancy term:

$$f_L(\mathbf{x}) = \rho f_\ell(\mathbf{x}) + \Delta(\mathbf{x}), \tag{6.37}$$

where $\rho \in \mathbb{R}$ is a scalar calibration or scaling parameter that is learned alongside the other hyperparameters defining the model, $f_\ell(\mathbf{x}) \sim \mathcal{GP}(0, k_\ell(\mathbf{x}, \mathbf{x}'))$ is a GP representing the low-fidelity simulator, $\Delta(\mathbf{x}) \sim \mathcal{GP}(0, k_\Delta(\mathbf{x}, \mathbf{x}'))$ is a discrepancy GP capturing the systematic mismatch between fidelities. Under this model, the joint prior over $f_\ell(\cdot)$ and $f_L(\cdot)$ is analytically tractable and Gaussian, and the posterior predictions retain closed-form expressions under Gaussian noise. This hierarchical design enforces that high-fidelity predictions are informed by low-fidelity evaluations through the shared term $f_\ell(\cdot)$, while allowing the discrepancy GP to correct for

systematic errors. We implemented this model with the help of `GPyTorch` (Gardner et al., 2021), using the Matérn-5/2 kernel.

MTGP Model The MTGP model proposed by Bonilla et al. (2007) offers a principled approach for jointly modelling multiple related outputs (tasks) using a shared GP framework. In the context of multi-source modelling, tasks correspond to different sources, providing a flexible and non-hierarchical alternative to models like KOH.

Let $f_\ell(\mathbf{x})$ denote the output of task $\ell \in \{1, \ldots, L\}$ at input $\mathbf{x} \in \mathcal{X}$. The MTGP assumes that the joint function $f(\mathbf{x}, \ell)$ is drawn from a GP:

$$f(\mathbf{x}, \ell) \sim \mathcal{GP}(0, k((\mathbf{x}, \ell), (\mathbf{x}', \ell'))), \tag{6.38}$$

with a covariance function decomposed as:

$$k((\mathbf{x}, \ell), (\mathbf{x}', \ell')) = k_x(\mathbf{x}, \mathbf{x}') \cdot k_\ell(\ell, \ell'). \tag{6.39}$$

Here k_x is a standard kernel (e.g. RBF or Matérn) defined over the input space, k_ℓ is a positive semi-definite covariance matrix $\mathbf{K}_\ell \in \mathbb{R}^{L \times L}$ that encodes task relationships.

This structure allows the MTGP to capture correlations across tasks (sources), enabling knowledge transfer from low-fidelity data to improve predictions at high fidelity. Importantly, unlike the KOH model, MTGPs do not impose any explicit hierarchy or discrepancy structure. Instead, task dependencies are entirely captured through the learned task covariance matrix. In this work, we make use of `GPyTorch`'s (Gardner et al., 2021) MTGP implementation (Bonilla et al., 2007), again using the Matérn-5/2 kernel.

6.7.4. Extending Benchmarks to Multi-Fidelities

Let $u^{(L)} : \mathbb{R}^D \to \mathbb{R}$ denote a target source objective or constraint function. We define a auxiliary source $u^{(\ell)}$ by applying an input-dependent oscillatory distortion to the function $u^{(L)}$. The distortion is scaled relative to the estimated magnitude of each function. Let $S_f > 0$ denote an empirical estimate of the objective's scale:

$$S := \mathbb{E}_{x \sim \mathcal{U}([0,1]^D)} \left[|u^{(L)}(x)| \right] \tag{6.40}$$

The oscillatory signal is defined as:

$$s(x) := \sin\left(\frac{2\pi}{d} \sum_{j=1}^{d} x_j \right) \tag{6.41}$$

For the weak correlation level, set the relative distortion factor to $\rho := 1$, for a strong correlation $\rho := 0.1$. We define the auxiliary source $\tilde{u}^{(\ell)}$ as:

$$u^{(\ell)}(x) := u^{(L)}(x) \cdot \left(1 + \frac{\rho S}{\max\left(|u^{(L)}(x)|, \varepsilon \right)} \cdot s(x) \right) + \xi \tag{6.42}$$

with $\xi \sim \mathcal{N}(0, \sigma_\xi^2)$.

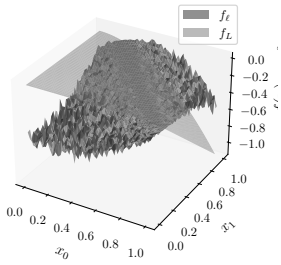


Figure 6.5: We use the 40-dimensional *Different Powers* objective function f and plot a slice along variable x_4 to visualise the difference between the target information source f_L and the weakly correlated and noisy auxiliary information source f_ℓ.

6.7.5. Baseline Methods: Experiment Setup

Multi-Source Constrained Max-value Entropy Search In MS-CMES we use the MISO model as discussed in Appendix 6.7.3 and employ discrete Thompson sampling to sample f^*. Moreover, we use OPTIMIZE_ACQF_MIXED in `BoTorch` Balandat et al. (2020), using 3 restarts and 200 raw samples to maximise the acquisition function.

Vanilla Bayesian Optimisation Here, we use the dimensionality-scaled length-scale prior from Hvarfner et al. (2024) and the logCEI, proposed in Ament et al. (2023). We use 64 MC samples and enable the SAMPLE_AROUND_BEST option, embedded in `BoTorch`'s acquisition function optimiser. More information can be found in Papenmeier et al. (2025).

Scalable Constrained Bayesian Optimisation We use the same parameters as in Eriksson and Poloczek (2021). The method is implemented in `BoTorch` and can be found here: `https://botorch.org/docs/tutorials/scalable_constrained_bo/`

Feasibility-Driven Trust Region Bayesian Optimisation FuRBO is directly built upon SCBO and differs only w.r.t. the TR heuristic. Here again, we employ the same user-defined parameters as presented in Ascia et al. (2025). The corresponding code was taken from `https://anonymous.4open.science/r/FuRBO`.

Constrained Max-value Entropy Search Additionally, we compare against the original CMES-IBO methods, proposed in Takeno et al. (2022). While the authors propose to define f^* as

$$f^* := \begin{cases} \max\limits_{\mathbf{x} \in \mathcal{X}_f} f(\mathbf{x}) & \text{if } \mathcal{X}_f \neq \emptyset \\ -\infty & \text{else,} \end{cases} \tag{6.43}$$

we found that our definition from Equation (6.11) yields better results as depicted in Figure 6.6 where we plot the mean and standard deviation over ten runs. For all benchmarks (this Section as well as Section 6.5.2) we use $K = 32$. Similar to before, we employ discrete Thompson sampling to sample f^* and use OPTIMIZE_ACQF in `BoTorch` Balandat et al. (2020), using 3 restarts and 200 raw samples.

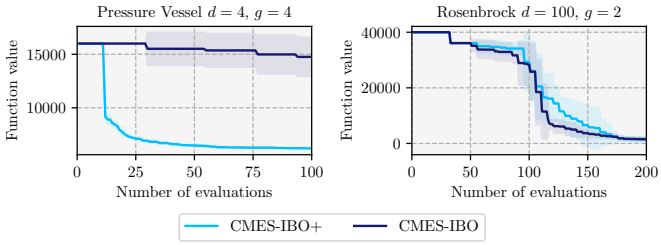


Figure 6.6: Comparison of the original definition of f^* (CMES-IBO) versus using the definition in Equation (6.11), here denoted with CMES-IBO+.

Constrained Multi-Fidelity Bayesian Optimisation We implemented our own version of the acquisition function in Foumani and Bostanabad (2025) in combination with the introduced MISO model. We use gradient-ascent to maximise the acquisition function. However, to obtain a similar ratio of auxiliary and target source evaluations we noted that we need to set the cost to $c_L = 0$. Like before, we employ OPTIMIZE_ACQF in `BoTorch` Balandat et al. (2020), using 3 restarts and 200 raw samples.

6.7.6. Details on Cost Function

As the cost function needs to be set up such that it reflects the magnitude of the acquisition function scale, we found that in MS-CMES $\lambda(\mathbf{x}, \ell) = 1 + \frac{\ell}{10^5} c_\ell$ balances the number of target and auxiliary source evaluations well. However, for real world problems, the choice of the cost function can be problem specific and needs to reflect the sources it is trying to balance. We emphasise that the choice of cost function is arbitrary as it solely scales the utility value of the acquisition function.

To showcase when a target source evaluation is queried, we depict in Figure 6.7 the accumulated costs over the accumulated evaluations. While auxiliary sources provide cost-effective guidance in early iterations, our acquisition function also continues to select the target source throughout the optimisation.

6.7.7. Details on Benchmark Problems

We evaluate our method on several well-known constrained test problems to stress different aspects of performance:

- The **Pressure Vessel** benchmark was proposed by Coello Coello and Mezura Montes

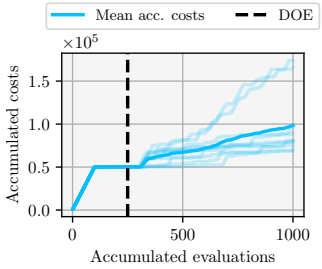


Figure 6.7: Accumulated evaluation cost as a function of the accumulated number of evaluations.

(2002) with dimensionality $d = 4$ and number of constraints $g = 4$. The aim is to minimise a total cost of designing the pressure vessel, including the shell and head thicknesses, as well as the inner radius and length of the cylindrical section including some bounds.

- The benchmarks **Different Powers**, **Rastrigin** and **Rotated Rastrigin** (all $d = 40$ and $g = 9$) were drawn from the BBOB-constrained suite (Dufossé et al., 2022), These functions are constructed by applying non-linear transformations (e.g. rotations, asymmetric scaling, oscillatory distortions) to the base functions, then overlaying constraint functions so that the feasible region becomes non-trivial. For instance, the Rastrigin and rotated Rastrigin variants combine the highly multimodal base with global rotations, while Different Powers applies coordinate-wise power scaling to create differing levels of smoothness along axes. This set covers both multimodal, non-separable and ill-conditioned landscapes with constraints, making it a rigorous benchmark for constrained multi-source BO.
- Additionally, we test our method on the Rosenbrock objective (Rosenbrock, 1960) augmented with two nonlinear constraints ($d = 100$ and $g = 2$) defined in Eriksson and Poloczek (2021), which challenges the model in high dimensions and with narrow feasible channels.

6.7.8. Ablation and Parameter Study

We perform an ablation study to assess key parameters. First, we examine the sensitivity to the number of Monte Carlo samples K for the acquisition function (Figure 6.8). Beyond moderate K, stability gains are marginal, indicating that relatively few samples suffice. Next, we evaluate the TR heuristic during acquisition optimisation (Figure 6.9). Constraining candidate generation to the local TR improves efficiency and stabilises convergence.

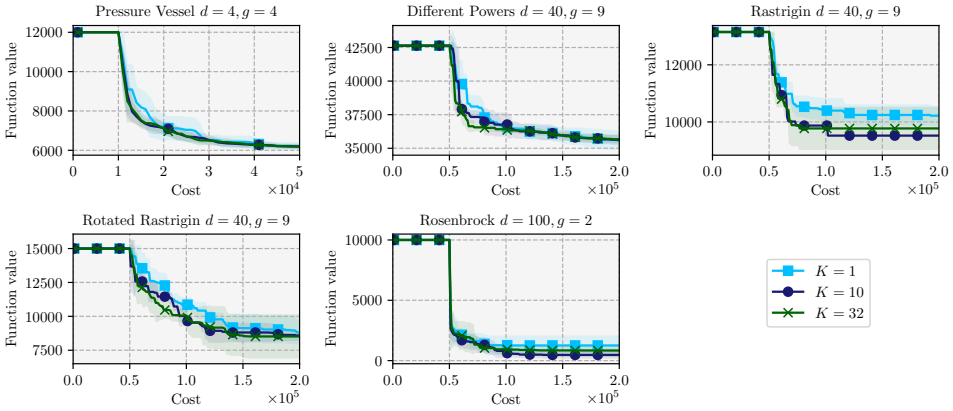


Figure 6.8: Parameter study on the number of MC samples K: optimisation performance for $K \in \{1, 10, 30\}$, showing that already moderate values yield stable results.

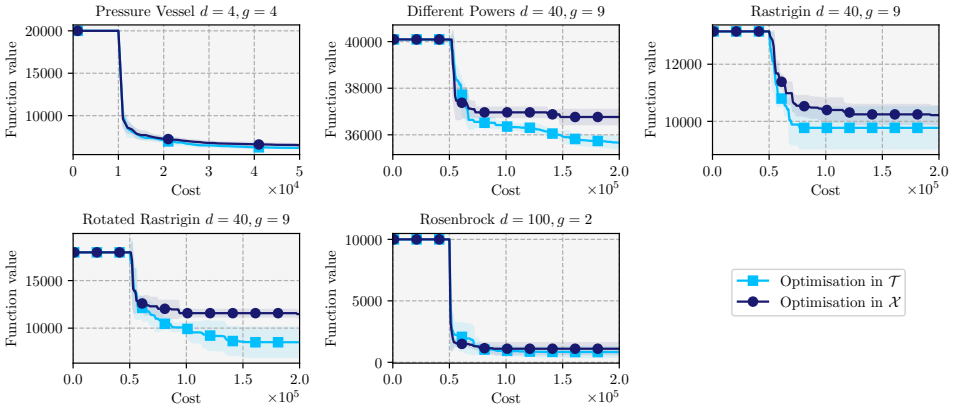


Figure 6.9: Ablation study on the TR heuristic: comparison of MS-CMES with and without TR–constrained acquisition optimisation.

Bibliography

S. Ament, S. Daulton, D. Eriksson, M. Balandat, and E. Bakshy. Unexpected Improvements to Expected Improvement for Bayesian Optimization, Jan. 2023. arXiv:2310.20708 [cs].

S. Anand, R. Alderliesten, and S. G. Castro. Crashworthiness in preliminary design: Mean crushing force prediction for closed-section thin-walled metallic structures. *International Journal of Impact Engineering*, 188:104946, June 2024. ISSN 0734743X. doi: 10.1016/j.ijimpeng.2024.104946.

N. Aretz, M. Gunzburger, M. Morlighem, and K. Willcox. Multifidelity uncertainty quantification for ice sheet simulations. *Computational Geosciences*, 29(1):5, Feb. 2025. ISSN 1420-0597, 1573-1499. doi: 10.1007/s10596-024-10329-3.

P. Ascia, E. Raponi, T. Bäck, and F. Duddeck. Feasibility-Driven Trust Region Bayesian Optimization, June 2025. arXiv:2506.14619 [cs].

M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*, 2020.

E. V. Bonilla, F. V. Agakov, and C. K. I. Williams. Kernel multi-task learning using task-specific features. In M. Meila and X. Shen, editors, *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics*, volume 2 of *Proceedings of Machine Learning Research*, pages 43–50, San Juan, Puerto Rico, 21–24 Mar 2007. PMLR.

C. A. Coello Coello and E. Mezura Montes. Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. *Advanced Engineering Informatics*, 16(3):193–203, July 2002. ISSN 14740346. doi: 10.1016/S1474-0346(02)00011-3.

O. Cordelier, Y. Diouane, N. Bartoli, and E. Laurendeau. Multi-Fidelity Constrained Bayesian Optimization with Application to Aircraft Wing Design. In *AIAA AVIATION FORUM AND ASCEND 2025*, Las Vegas, Nevada, July 2025. American Institute of Aeronautics and Astronautics. ISBN 978-1-62410-738-2. doi: 10.2514/6.2025-3474.

P. Dufossé, N. Hansen, D. Brockhoff, P. R. Sampaio, A. Atamna, and A. Auger. Building scalable test problems for benchmarking constrained optimizers. Technical report, Technical Report. http://numbbo. github. io/coco-doc/bbob-constrained/To be . . . , 2022.

D. Eriksson and M. Poloczek. Scalable Constrained Bayesian Optimization, Feb. 2021. URL `http://arxiv.org/abs/2002.08526`. arXiv:2002.08526 [cs, stat].

A. I. Forrester, A. Sóbester, and A. J. Keane. Multi-fidelity optimization via surrogate modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 463(2088):3251–3269, Dec. 2007. ISSN 1364-5021, 1471-2946. doi: 10.1098/rspa.2007.1900.

Z. Z. Foumani and R. Bostanabad. Constrained multi-fidelity Bayesian optimization with automatic stop condition, Mar. 2025. arXiv:2503.01126 [cs].

P. I. Frazier. A Tutorial on Bayesian Optimization, July 2018. arXiv:1807.02811 [cs, math, stat].

J. R. Gardner, M. J. Kusner, and G. Jake. Bayesian Optimization with Inequality Constraints. 2014.

J. R. Gardner, G. Pleiss, D. Bindel, K. Q. Weinberger, and A. G. Wilson. GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration, June 2021. arXiv:1809.11165 [cs].

M. A. Gelbart, J. Snoek, and R. P. Adams. Bayesian Optimization with Unknown Constraints. 2014.

L. L. Gratiet. Recursive co-kriging model for Design of Computer experiments with multiple levels of fidelity with an application to hydrodynamic, Jan. 2013. arXiv:1210.0686 [math].

J. M. Hernández-Lobato, M. W. Hoffman, and Z. Ghahramani. Predictive Entropy Search for Efficient Global Optimization of Black-box Functions, June 2014. arXiv:1406.2541 [stat].

J. M. Hernández-Lobato, M. A. Gelbart, R. P. Adams, M. W. Hoffman, and Z. Ghahramani. A General Framework for Constrained Bayesian Optimization using Information-based Search, Sept. 2016. URL `http://arxiv.org/abs/1511.09422`. arXiv:1511.09422 [stat].

J. M. Hernández-Lobato, J. Requeima, E. O. Pyzer-Knapp, and A. Aspuru-Guzik. Parallel and Distributed Thompson Sampling for Large-scale Accelerated Exploration of Chemical Space. 2017. doi: 10.48550/ARXIV.1706.01825. URL `https://arxiv.org/abs/1706.01825`. Publisher: arXiv Version Number: 1.

C. Hvarfner, E. O. Hellsten, and L. Nardi. Vanilla Bayesian Optimization Performs Great in High Dimensions, Dec. 2024. arXiv:2402.02229 [cs].

K. Kandasamy, G. Dasarathy, J. Schneider, and B. Poczos. Multi-fidelity Bayesian Optimisation with Continuous Approximations, 2017. Version Number: 1.

M. C. Kennedy and A. O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000. ISSN 00063444, 14643510.

H. J. Kushner. A versatile stochastic model of a function of unknown and time varying form. *Journal of Mathematical Analysis and Applications*, 5(1):150–167, Aug. 1962. ISSN 0022247X. doi: 10.1016/0022-247X(62)90011-2.

H. J. Kushner. A New Method of Locating the Maximum Point of an Arbitrary Multipeak Curve in the Presence of Noise. *Journal of Basic Engineering*, 86(1): 97–106, Mar. 1964. ISSN 0021-9223. doi: 10.1115/1.3653121.

H. Maathuis, S. G. P. Castro, and R. D. Breuker. Exploring Multi-Fidelity Aeroelastic Tailoring: Prospect and Model Assessment, Nov. 2024. arXiv:2411.03247 [cs].

H. F. Maathuis, R. De Breuker, and S. G. P. Castro. Scaling Bayesian Optimization for High-Dimensional and Large-Scale Constrained Spaces. *AIAA Journal*, pages 1–11, July 2025. ISSN 0001-1452, 1533-385X. doi: 10.2514/1.J065252.

P. Mikkola, J. Martinelli, L. Filstroff, and S. Kaski. Multi-Fidelity Bayesian Optimization with Unreliable Information Sources, 2022. Version Number: 2.

H. B. Moss, D. S. Leslie, and P. Rayson. MUMBO: MUlti-task Max-value Bayesian Optimization, June 2020. arXiv:2006.12093 [cs].

H. B. Moss, D. S. Leslie, J. Gonzalez, and P. Rayson. GIBBON: General-purpose Information-Based Bayesian OptimisatioN, Oct. 2021. arXiv:2102.03324 [cs].

K. Om, K. Sim, T. Yun, H. Kang, and J. Park. Posterior Inference in Latent Space for Scalable Constrained Black-box Optimization, July 2025. arXiv:2507.00480 [cs].

L. Papenmeier, M. Poloczek, and L. Nardi. Understanding High-Dimensional Bayesian Optimization, June 2025. arXiv:2502.09198 [cs].

P. Perdikaris, M. Raissi, A. Damianou, N. D. Lawrence, and G. E. Karniadakis. Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2198):20160751, Feb. 2017. ISSN 1364-5021, 1471-2946. doi: 10.1098/rspa.2016.0751.

V. Perrone, I. Shcherbatyi, R. Jenatton, C. Archambeau, and M. Seeger. Constrained Bayesian Optimization with Max-Value Entropy Search, Oct. 2019. arXiv:1910.07003 [stat].

M. Poloczek, J. Wang, and P. I. Frazier. Multi-Information Source Optimization, Nov. 2016. arXiv:1603.00389 [stat].

L. Pretsch, I. Arsenyev, N. Bartoli, and F. Duddeck. Bayesian optimization of cooperative components for multi-stage aero-structural compressor blade design. *Structural and Multidisciplinary Optimization*, 68(4):84, Apr. 2025. ISSN 1615-147X, 1615-1488. doi: 10.1007/s00158-025-03998-w.

E. Raponi, M. Bujny, M. Olhofer, N. Aulig, S. Boria, and F. Duddeck. Kriging-assisted topology optimization of crash structures. *Computer Methods in Applied Mechanics and Engineering*, 348:730–752, May 2019. ISSN 00457825. doi: 10.1016/j.cma.2019.02.002.

C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass, 2006. ISBN 978-0-262-18253-9. OCLC: ocm61285753.

H. H. Rosenbrock. An Automatic Method for Finding the Greatest or Least Value of a Function. *The Computer Journal*, 3(3):175–184, Mar. 1960. ISSN 0010-4620, 1460-2067. doi: 10.1093/comjnl/3.3.175.

B. Ru, M. McLeod, D. Granziol, and M. A. Osborne. Fast Information-theoretic Bayesian Optimisation, June 2018. arXiv:1711.00673 [stat].

S. Takeno, H. Fukuoka, Y. Tsukada, T. Koyama, M. Shiga, I. Takeuchi, and M. Karasuyama. Multi-fidelity Bayesian Optimization with Max-value Entropy Search and its parallelization, Feb. 2020. arXiv:1901.08275 [stat].

S. Takeno, T. Tamura, K. Shitara, and M. Karasuyama. Sequential and parallel constrained max-value entropy search via information lower bound. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 20960–20986. PMLR, 17–23 Jul 2022.

Z. Wang and S. Jegelka. Max-value Entropy Search for Efficient Bayesian Optimization, Jan. 2018. arXiv:1703.01968 [stat].

J. Wu, S. Toscano-Palmerin, P. I. Frazier, and A. G. Wilson. Practical Multi-fidelity Bayesian Optimization for Hyperparameter Tuning, 2019. Version Number: 1.

7

Conclusion and Future Research

Recap and Motivation Bayesian Optimisation (BO) provides an efficient and principled framework for global optimisation, particularly suitable for black-box problems where derivative information is unavailable or evaluations are expensive. Its flexibility makes it highly attractive for a wide range of real-world problems. However, this flexibility comes at a cost: the performance of BO degrades significantly in high-dimensional spaces due to the curse of dimensionality. Addressing this challenge remains one of the most significant open problems in optimisation. Moreover, while much of the existing literature has focused on unconstrained problems, developing scalable BO algorithms for constrained problems substantially broadens the applicability of BO, especially in engineering where constraints are the norm rather than the exception. Addressing both high-dimensionality and complex feasibility conditions typically requires strong structural assumptions or problem-specific heuristics. Recent advances in Trust Region (TR) methods have established state-of-the-art performance for Constrained Bayesian Optimisation (CBO), but standard engineering problems can be even more complex, involving hundreds or thousands of design variables and constraints. This thesis contributes to the field by developing methods that enable BO to scale to such settings, thereby advancing the frontier of constrained optimisation in practical applications.

Reflection on Research Questions. This section revisits the research questions posed in the introduction and discusses how each was addressed through the contributions of this thesis.

RQ 1 explored whether techniques from unconstrained BO can be effectively adapted to solve constrained high-dimensional problems. This question is rooted

in the observation that the transfer of many high-performing strategies from unconstrained BO to CBO, where feasibility is critical, is not always trivial. This work evaluated two such techniques in particular random subspace embeddings and dimensionality-scaled length-scale priors. Chapter 3 demonstrated that while random embeddings can reduce the effective input dimensionality and accelerate convergence in unconstrained settings, they often fail to capture the feasible region when constraints are active. This is primarily due to the misalignment between the embedded subspace and the often sparse or disconnected feasible region in the original space. As a response, the thesis proposed a novel, supervised embedding strategy, informed by objective and constraint values, to improve the alignment between the reduced subspace and the feasible set, see Section 3.3. This method led to consistently better feasibility discovery and optimisation performance in constrained problems, compared to random embeddings, as shown in Chapter 3.5. However, empirical results also revealed that often, full-dimensional modelling outperformed subspace methods. This underscores a key insight: embedding-based dimensionality reduction is not universally beneficial in constrained settings, and its success depends on structural assumptions about the problem, e.g. a shared low-dimensional subspaces across objectives and constraints. Regarding Dimensionality-Scaled Priors (DSPs), discussed in Section 3.4, the thesis showed that this simple model initialisation heuristic can improve the robustness of Gaussian Process (GP)-based BO methods in high-dimensional constrained problems. By adapting the initial smoothness assumptions to the dimensionality of the design space, these priors improved convergence rates and reduced sensitivity to hyperparameter choices. However, their effectiveness varied depending on the problem class and degree of anisotropy, suggesting that further work is needed to personalise or learn these priors adaptively. Taken together, these results provide a nuanced answer to RQ 1: while insights from unconstrained BO can be translated to constrained settings, they require adaptation and problem-aware modification to remain effective.

RQ 2 focused on the scalability of CBO in the presence of large-scale constraints, often encountered in engineering applications such as structural optimisation or aeroelastic tailoring. Chapter 4 addressed this by proposing a method to reduce the output dimensionality in which the constraints live, by using latent space models. Instead of independently modelling each constraint, which becomes computationally infeasible when thousands are present, the method projects the constraints onto a low-dimensional latent space using a PCA-based mapping. A GP surrogate is then fit in this compressed output space, yielding substantial memory and runtime savings. Importantly, this modelling strategy was embedded within a trust-region framework to ensure local feasibility and stable convergence, even in high-dimensional input spaces. This enabled the method to scale to problems with hundreds of design variables and thousands of constraints, demonstrating strong performance in a realistic aeroelastic tailoring optimisation setting. Notably, the dimensionality reduction introduced some approximation error, resulting in a modest trade-off in optimisation

performance. Nevertheless, the method remained effective, particularly in identifying feasible solutions efficiently. Chapter 5 further advanced this idea by jointly learning dimensionality-reduced representations of both the input and output spaces using two autoencoders. This joint input-output latent space enabled BO to focus its modelling capacity on the most informative subspaces of the problem, improving feasibility attainment and optimisation performance. A key observation is that such joint embeddings are particularly beneficial when the design variables and constraint outputs are strongly coupled, a common scenario in multi-physics simulations. These methods were benchmarked against classical penalty and aggregation strategies, showing consistent improvements in data efficiency and robustness. Additionally, it was shown that in high-data regimes, these methods can even compete with approaches that model each constraint using a separate GP. Thus, RQ 2 is addressed by proposing scalable approaches tailored for high-dimensional, highly-constrained settings, with clear advantages in practical engineering design problems.

RQ 3 explored how multiple information sources of varying accuracy and cost can be integrated into CBO frameworks. This is motivated by real-world engineering workflows, where both expensive but accurate simulations and approximate models often co-exist. Chapter 6 introduced a flexible multi-source GP model, capable of capturing both shared structure and source-specific discrepancies. Importantly, it allowed for learning inter-source correlations in a data-driven way, accommodating cases where sources are weakly informative or poorly aligned. To fully exploit this model, the thesis proposed a new acquisition function based on information-theoretic principles. This acquisition strategy estimates the expected information gain about the optimal solution, while accounting for constraint satisfaction and query cost. As a result, the optimiser is able to prioritise low-cost, informative sources early in the optimisation, while selectively querying the high-fidelity model when necessary. This trade-off is crucial in cost-constrained design settings. Empirical results showed that this method not only improves convergence speed but also reduces the number of high-cost evaluations needed to identify feasible and optimal solutions. RQ 3 is thus answered through the development of correlation-aware modelling and cost-sensitive acquisition strategies, enabling principled use of heterogeneous data sources.

Personal reflection. Looking back, one personal reflection concerns the rapid pace of development in this field. The area of BO is evolving so quickly that methods considered state-of-the-art today may already be superseded or reinterpreted by tomorrow. This dynamism makes it challenging to define research questions that remain relevant over several years. At the same time, it highlights the importance of principled insights, modular methods, and reusable ideas that can adapt to or inspire future developments. While specific algorithmic choices may become outdated, the broader goals, e.g. efficient learning under constraints, exploiting structure in simulation-based design, and making intelligent use of multiple information sources, are likely to remain central challenges. In this context, some form of meta-level

thinking becomes essential: rather than focusing solely on isolated techniques, it is equally important to reflect on how problems are framed, what kinds of assumptions are embedded into the models, and how these choices influence both short-term performance and long-term relevance. In this sense, the work aims not only to contribute to new methods, but also to provide conceptual building blocks and transferable insights that can support sustained progress in the field.

Future Work. Several promising directions emerge from this work:

Heterogeneous Length-Scale Priors. Chapter 3 used dimensionality-scaled length-scale priors assuming shared smoothness across all outputs. However, in practice, different constraints or objectives may exhibit different levels of smoothness. Future work could explore how to initialise or learn individualised priors in a data-efficient way.

Reusing Past Data. Engineering organisations generate vast amounts of simulation data, much of which remains unused. This data could be leveraged to pre-train latent spaces for constraint modelling, as in Chapters 4 and 5, or to train generative surrogate models serving as auxiliary information sources in multi-source BO, see Chapter 6.

Grey-Box Modelling and Gradient Information. Many engineering models expose partial derivatives or internal state variables that can be exploited to improve surrogate modelling. Incorporating such grey-box information, either through gradient-enhanced GP or grey-box surrogates, could yield more accurate models in data-scarce regimes. For instance, a hybrid strategy could use global surrogates to identify a promising feasible region, followed by local refinement, thereby reducing reliance on uncertain GP predictions in high-dimensional spaces. Moreover, while many simulators are treated as black boxes, intermediate state variables are often accessible and could be integrated in a grey-box fashion to enrich model expressiveness.

Prior Fitted Networks for Constrained Optimisation. Recent advances in Prior-Fitted Networkss (PFNs) offer a promising alternative to GP, particularly in settings with access to large prior datasets. PFNs can approximate Bayesian posteriors directly via offline meta-learning, bypassing the need for explicit kernel design or matrix inversion. Future work could explore PFNs tailored to constrained optimisation, potentially replacing GP-based models in high-dimensional or cost-constrained scenarios with weak inter-fidelity correlations. This could enable faster and more scalable multi-source optimisation strategies that retain uncertainty awareness while benefiting from learned structural priors.

Final Remarks. This thesis has proposed a set of algorithms and modelling strategies, namely BOOSTRE, (k)PCA-GP SCBO, AERO-BO and MS-CMES, that extend the applicability of BO to high-dimensional, constrained engineering problems. By combining dimensionality-aware priors, input- and output-space dimensionality reduction, and multi-source information integration, it makes progress towards solving large-scale optimisation problems that are otherwise intractable using existing methods. The resulting framework supports scalable, data-efficient, and constraint-aware optimisation, with promising applications across structural design, aerospace systems, and other simulation-based engineering domains. While theoretical and computational challenges remain, the methods developed here offer a foundation upon which more robust and general-purpose solutions can be built.

Despite these advances, a significant gap persists between the capabilities of state-of-the-art optimisation algorithms and their practical deployment in industry. For instance, many engineering organisations continue to rely on manual, sequential, and siloed design processes, where optimisation is applied in isolation, constraints are handled heuristically, and valuable simulation data is discarded after use. These limitations are not solely technical but reflect deeper issues in tooling, organisational structure, and cultural inertia. Bridging this gap will require more than algorithmic innovation. It will demand a shift in how engineering design is conceptualised and executed: from isolated workflows to integrated pipelines, from one-off simulations to reusable knowledge, and from local heuristics to global optimisation strategies. Importantly, optimisation models often operate under idealised assumptions and, at best, incorporate limited forms of model uncertainty. In practice, however, design decisions are shaped by real-world factors such as supply chain disruptions, manufacturing constraints, operational flexibility, and certification timelines. These considerations may lead to intentional deviations from theoretical optima in favour of solutions that are more robust, practical, or readily deployable. Ultimately, even these real-world constraints must be accounted for to arrive at the best possible design in practice. Companies and institutions that invest in aligning their simulation infrastructure, data strategy, and optimisation capabilities stand to benefit significantly, not only in terms of efficiency, but also in the quality and robustness of their early design decisions.

Ultimately, the vision underpinning this work is not just about developing better algorithms, but about embedding optimisation as a central, intelligent layer in the engineering design process. As the field continues to evolve rapidly, this kind of meta-level thinking about how problems are formulated, how models are structured, and how knowledge is reused will become increasingly important. Such considerations can play a key role in enabling the discovery of fundamentally new designs. In this sense, the contributions of this thesis aim to serve both as practical methods and as conceptual stepping stones toward more automated, adaptive, and intelligent design systems.

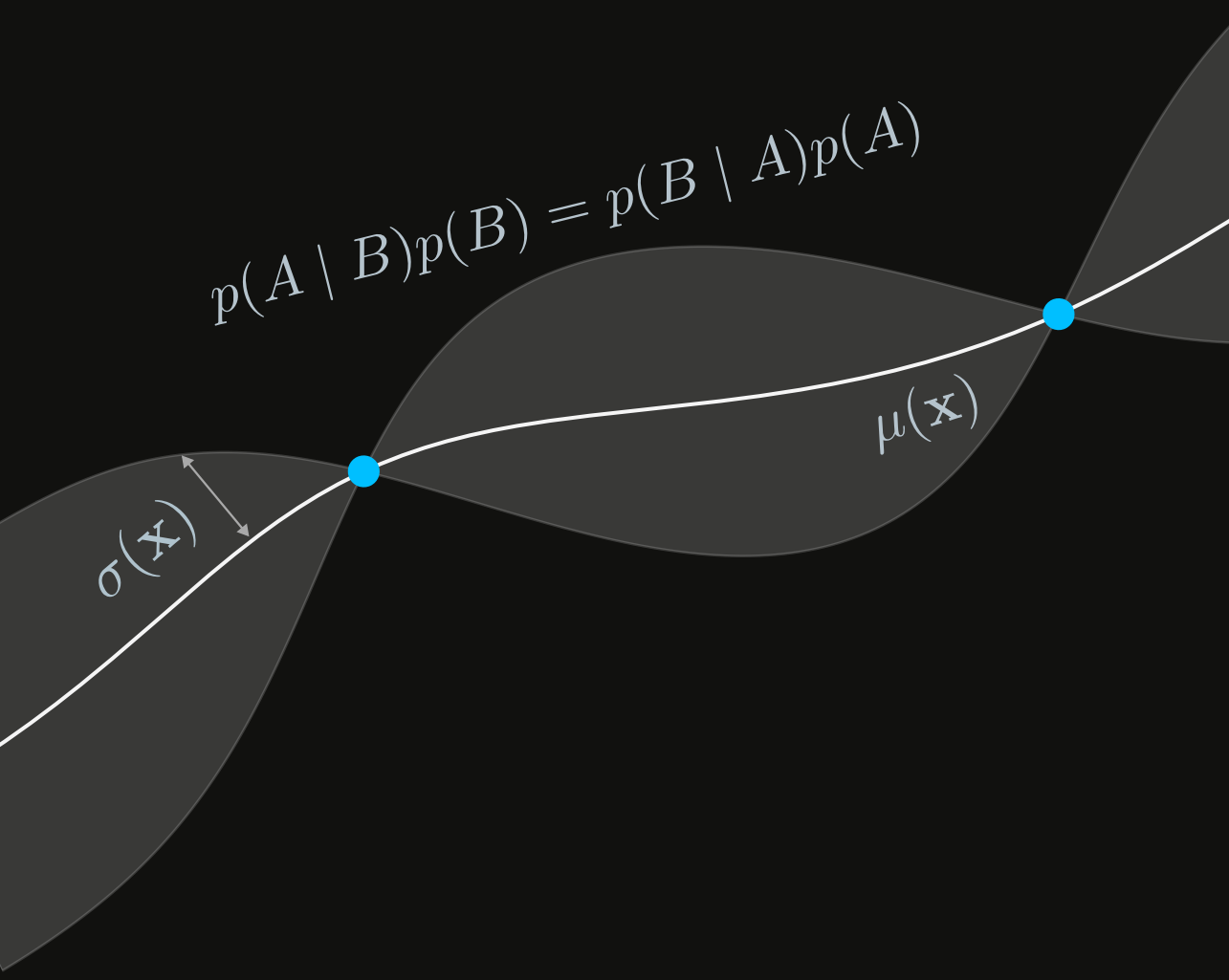
Curriculum Vitæ

Hauke Felix MAATHUIS

15-09-1996 Born in Nordhorn, Germany.

Education

2007–2015	Secondary Education Lise-Meitner-Gymnasium, Neuenhaus/Uelsen, Germany
2015–2018	BEng in Industrial Engineering (Dual Study Programme) University of Applied Sciences Osnabrück, Germany
2018–2021	MSc in Mechanical Engineering Gottfried Wilhelm Leibniz University Hanover, Germany
2019–2020	MSc Exchange Semester Université Paris-Saclay (ENS Paris-Saclay), France
2025	PhD Visiting Researcher University of Oxford, United Kingdom
2022–2026	PhD Candidate in Aerospace Engineering Delft University of Technology, The Netherlands



ISBN: 978-94-65-18-2