EPFL

Thèse n° 10 542

Quantifying the Unknown: Data-Driven Approaches and Applications in Energy Systems

Présentée le 22 mars 2024

Faculté des sciences et techniques de l'ingénieur
Laboratoire d'automatique 3
Programme doctoral en génie électrique

pour l'obtention du grade de Docteur ès Sciences

par

Paul SCHARNHORST

Acceptée sur proposition du jury

Prof. M. Kamgarpour, présidente du jury
Prof. C. N. Jones, Dr B. Schubnel, directeurs de thèse
Prof. G. Hug, rapporteuse
Prof. J. Stoustrup, rapporteur
Prof. D. Kuhn, rapporteur

École polytechnique fédérale de Lausanne

2024

Acknowledgements

There are a lot of people without whom my PhD would not have been the same, and I am very grateful to every single one of them, although I will probably fail to mention everyone who deserves to be mentioned. First, I would like to thank my two supervisors, Professor Colin Jones and Dr Baptiste Schubnel, for their abundant support. Colin always provided me with valuable feedback and ideas, and on top of that fostered and embodied a great atmosphere of curiosity, respect, and kindness in the lab. Baptiste supported me in all my scientific endeavors, always encouraging me to be as precise as possible while keeping an eye on what matters most.

I would also like to thank Professor Gabriela Hug, Professor Jakob Stoustrup, Professor Daniel Kuhn, and Professor Maryam Kamgarpour for being a part of my thesis committee and taking the time to provide me with feedback on my work.

I would like to acknowledge the funding of the CSEM Data Program during my doctoral studies. My group at CSEM played a central role in my PhD journey, and I would like to extend my gratitude to the whole team, consisting of wonderfully different but matching individuals. Thank you Pierre-Jean, Baptiste, Rafael, Tomasz, Diya, Jelena, Max, and Renaud.

I mentioned the atmosphere in the Automatic Control Laboratory before, and of course, it also would not be possible without all the great people involved. Special thanks to my collaborators Emilio and Yuning, as well as Christophe, Loris, Roland, Johannes, Manuel, Peter, Yang, Wenjie, Jicheng, Yingzhao, Philippe, Clara, Andrea, Sohail, Mustafa, and Navid, to name just a few.

Thanks to all my friends who supported me on this journey. The ones I made in Switzerland, comprising Emilio, Vivi, Shubham, Pietro, Enrico, the LIP, and Rosa, as well as my old friends, who helped me through everything with unwavering love: Sebastian, Maurin, Christoph, Felix, Elli, Jonas, and Maren.

Last but not least, my family. Being able to conclude this part of my life is the result of a life filled with trust, love, and occasional teasing. My heartfelt gratitude to my parents Christina and Peter, my brother Enno, and his wife Ryeoyeong.

Neuchâtel, January 16, 2024 Paul Scharnhorst

Abstract

In light of the challenges posed by climate change and the goals of the Paris Agreement, electricity generation is shifting to a more renewable and decentralized pattern, while the operation of systems like buildings is increasingly electrified. This calls for new control methods to increase the efficiency and coordination of systems, to both lower energy needs, and offer consumption flexibility as a service to the grid. One key feature for the acceptance of those new control methods, which often rely on the availability of data and some form of data-driven modeling, is the guarantee of constraint satisfaction, either operational or related to, e.g., user comfort. This thesis considers the problem of guarantees in data-driven control, focusing on the robust setting. It covers a spectrum from a more general learning perspective to a more application-driven perspective, targeting energy systems.

The first part of the thesis discusses kernel methods for function approximation, such as Kernel Ridge Regression or Support Vector Regression. Being non-parametric, these methods offer a way to approximate functions, such as the response of a dynamical system, with varying complexity, based on the choice of hyperparameters and the available data samples. Depending on assumptions on the complexity of the ground truth function, deterministic error bounds of the approximations are developed, bounding the difference of the approximation to the true function, under noisy sampling. These bounds are further improved by formulating the bounding problem as an infinite-dimensional variational problem and reformulating it into a finite-dimensional version, using representer-theorem arguments. The tightness of those bounds is demonstrated through different simulation examples.

In the second part of the thesis, an open-source tool for controller benchmarking is introduced, to bridge the gap between the general control setting, and the specific application of building control. This Python library, called Energym, collects different building models from the simulation tools EnergyPlus and Modelica and interfaces them for direct usage in Python. Through an API that resembles the one of the reinforcement learning benchmarking library Gym, control signals can be sent to the individual models, and performance can be evaluated based on predefined metrics.

In the third part, a method to estimate the consumption flexibility potential of individual buildings is presented. By learning the parameters of a virtual battery model and express-

ing uncertainties as parameter uncertainties, this method combines robust estimation and its application to buildings. The flexibility potential of individual buildings is represented by flexibility envelopes, which are used in the formulation of a coordination problem of a pool of buildings. By solving a mixed-integer problem, a schedule of activation is fixed, while the actual flexibility requests are dispatched by a heuristic algorithm. This coordination is demonstrated in large-scale simulations, using building models from Energym, for the scenarios of self-consumption and peak reduction.

Key words: Kernel methods, error estimation, robustness, building control, benchmarking, consumption flexibility, building coordination

Zusammenfassung

Angesichts der Herausforderungen des Klimawandels und der Ziele des Pariser Abkommens verlagert sich die Stromerzeugung hin zu erneuerbaren und dezentralisierten Strukturen, während der Betrieb von Systemen wie Gebäuden zunehmend elektrifiziert wird. Dies erfordert neue Regelungsmethoden, um die Effizienz und Koordination der Systeme zu erhöhen, den Energiebedarf zu senken und dem Netz eine flexible Nutzung zu ermöglichen. Ein Schlüsselelement für die Akzeptanz dieser neuen Regelungsmethoden, die oft auf der Verfügbarkeit von Daten und einer Form der datengesteuerten Modellierung beruhen, ist die Garantie der Erfüllung von Randbedingungen, die entweder betrieblich oder z. B. in Bezug auf den Benutzerkomfort sind. Diese Arbeit befasst sich mit dem Problem der Garantien in der datengesteuerten Regelung und konzentriert sich dabei auf den robusten Bereich. Sie deckt ein Spektrum ab, das von einer allgemeineren Lernperspektive bis hin zu einer eher anwendungsorientierten Perspektive reicht, die auf Energiesysteme abzielt.
Der erste Teil der Arbeit befasst sich mit Kernel-Methoden zur Funktionsapproximation, wie z.B. Kernel-Ridge-Regression oder Support-Vector-Regression. Da diese Methoden nichtparametrisch sind, bieten sie eine Möglichkeit, Funktionen, wie z. B. die Reaktion eines dynamischen Systems, mit unterschiedlicher Komplexität zu approximieren, die auf der Wahl von Hyperparametern und den verfügbaren Daten basiert. Abhängig von den Annahmen über die Komplexität der wahren Funktion werden deterministische Fehlergrenzen für die Approximationen entwickelt, die die Differenz der Approximation zur wahren Funktion bei verrauschten Daten bestimmen. Diese Schranken werden weiter verbessert, indem das Problem der Fehlerbegrenzung als unendlich-dimensionales Variationsproblem formuliert und in eine endlich-dimensionale Version umformuliert wird, wobei Argumente des Repräsentationstheorems verwendet werden. Die Genauigkeit dieser Schranken wird anhand verschiedener Simulationsbeispiele demonstriert.
Im zweiten Teil der Arbeit wird ein Open-Source-Tool für den Vergleich von Reglern vorgestellt, um die Lücke zwischen dem allgemeinen Regelungsumfeld und der spezifischen Anwendung der Gebäuderegelung zu schließen. Diese Python-Bibliothek mit dem Namen Energym vereint verschiedene Gebäudemodelle aus den Simulationswerkzeugen EnergyPlus und Modelica und stellt sie für die direkte Verwendung in Python bereit. Über eine API, die derjenigen der Reinforcement-Learning-Benchmarking-Bibliothek Gym ähnelt, können Regelungssignale an die einzelnen Modelle gesendet und die Leistung

anhand von vordefinierten Metriken bewertet werden.
Im dritten Teil wird eine Methode zur Abschätzung des Verbrauchsflexibilitätspotenzials einzelner Gebäude vorgestellt. Indem die Parameter eines virtuellen Batteriemodells erlernt und Unsicherheiten als Parameterunsicherheiten ausgedrückt werden, kombiniert diese Methode eine robuste Schätzung und ihre Anwendung auf Gebäude. Das Flexibilitätspotenzial einzelner Gebäude wird durch Flexibilitätsbereiche dargestellt, die in der Formulierung eines Koordinationsproblems für einen Gebäudebestand verwendet werden. Durch die Lösung eines gemischt-ganzzahligen Problems wird ein Zeitplan für die Aktivierung festgelegt, während die tatsächlichen Flexibilitätsanforderungen durch einen heuristischen Algorithmus abgewickelt werden. Diese Koordination wird in groß angelegten Simulationen mit Gebäudemodellen von Energym für die Szenarien Eigenverbrauch und Spitzenlastreduzierung demonstriert.

Stichwörter: Kernel-Methoden, Fehlerabschätzung, Robustheit, Gebäuderegelung, Benchmarking, Verbrauchsflexibilität, Gebäudekoordination

Résumé

À la lumière des défis posés par le changement climatique et des objectifs de l'Accord de Paris, la production d'électricité évolue vers un modèle plus renouvelable et décentralisé, tandis que l'exploitation de systèmes tels que les bâtiments est de plus en plus électrifiée. Cela nécessite de nouvelles méthodes de contrôle pour accroître l'efficacité et la coordination des systèmes, afin de réduire les besoins en énergie et d'offrir une flexibilité de consommation en tant que service au réseau. Une caractéristique clé pour l'acceptation de ces nouvelles méthodes de contrôle, qui dépendent souvent de la disponibilité des données et d'une certaine forme de modélisation basée sur les données, est la garantie de la satisfaction des contraintes, qu'elles soient opérationnelles ou liées, par exemple, au confort de l'utilisateur. Cette thèse examine le problème des garanties dans le contrôle piloté par les données, en se concentrant sur le cadre robuste. Elle couvre un spectre allant d'une perspective d'apprentissage plus générale à une perspective plus orientée vers les applications, ciblant les systèmes énergétiques.

La première partie de la thèse traite des méthodes à noyau pour l'approximation de fonctions, telles que la régression de crête à noyau ou la régression à vecteur de support. Étant non paramétriques, ces méthodes offrent un moyen d'approximer des fonctions, telles que la réponse d'un système dynamique, avec une complexité variable, basée sur le choix des hyperparamètres et des échantillons de données disponibles. En fonction des hypothèses sur la complexité de la fonction de vérité de base, des limites d'erreur déterministes des approximations sont développées, limitant la différence de l'approximation par rapport à la vraie fonction, dans le cas d'un échantillonnage bruyant. Ces bornes sont encore améliorées en formulant le problème de borne comme un problème variationnel à dimension infinie et en le reformulant en une version à dimension finie, à l'aide d'arguments du théorème du représentant. L'étanchéité de ces bornes est démontrée à l'aide de différents exemples de simulation.

Dans la deuxième partie de la thèse, un outil open-source pour l'évaluation comparative des contrôleurs est introduit, afin de combler le fossé entre le réglage général du contrôle et l'application spécifique du contrôle des bâtiments. Cette bibliothèque Python, appelée Energym, recueille différents modèles de bâtiments à partir des outils de simulation EnergyPlus et Modelica et les interface pour une utilisation directe en Python. Grâce à une API qui ressemble à celle de la bibliothèque d'analyse comparative de l'apprentissage

par renforcement Gym, des signaux de contrôle peuvent être envoyés aux différents modèles et les performances peuvent être évaluées sur la base de paramètres prédéfinis. La troisième partie présente une méthode permettant d'estimer le potentiel de flexibilité de la consommation de chaque bâtiment. En apprenant les paramètres d'un modèle de batterie virtuelle et en exprimant les incertitudes comme des incertitudes de paramètres, cette méthode combine une estimation robuste et son application aux bâtiments. Le potentiel de flexibilité des bâtiments individuels est représenté par des enveloppes de flexibilité, qui sont utilisées dans la formulation d'un problème de coordination d'un ensemble de bâtiments. La résolution d'un problème mixte permet de fixer un calendrier d'activation, tandis que les demandes de flexibilité réelles sont réparties par un algorithme heuristique. Cette coordination est démontrée dans des simulations à grande échelle, en utilisant des modèles de bâtiments d'Energym, pour les scénarios d'autoconsommation et de réduction des pics.

Mots clefs : Méthodes à noyau, estimation d'erreur, robustesse, contrôle des bâtiments, analyse comparative, flexibilité de la consommation, coordination des bâtiments

Contents

List of Figures

List of Tables

Acronyms

HP Heat Pump

EV Electric Vehicle

DR Demand Response

ML Machine Learning

MPC Model Predictive Control

TCL Thermostatically Controlled Load

GP Gaussian Process

SoC State of Charge

KRR Kernel Ridge Regression

CVaR Conditional Value at Risk

MILP Mixed Integer Linear Program

RL Reinforecement Learning

SP Setpoint

KPI Key Performance Indicator

PV Photovoltaic

PID Proportional Integral Derivative

HVAC Heating Ventilation and Air Conditioning

RC Resistance Capacitance

FMI Functional Mockup Interface

FMU Functional Mockup Unit

DHW Domestic Hot Water

Acronyms

AHU Air Handling Unit

RBC Rule-based Controller

ARX AutoRegressive model with eXogenous input

IREC Institut de Recerca en Energia de Catalunya

NN Neural Network

SVR Support Vector Regression

RKHS Reproducing Kernel Hilbert Space

PD Positive Definite

RBF Radial Basis Function

UNAV Universidad de Navarra

ADMM Alternating Direction Method of Multipliers

Chapter 1

Introduction

1.1 Energy systems and robustness

Climate change poses one of the largest challenges for humanity today. To reach the goals of the Paris Agreement and to limit the impact of global warming, drastic changes are required across many aspects of daily life. Among the most important levers to address the issue is the operation of energy systems, including topics like energy efficiency, and renewable and distributed electricity generation. To adopt changes in the control of these systems, different requirements need to be fulfilled: new control algorithms need to be scalable to be easily transferred between systems, they need to be efficient, compared to more classical methods, and they need to come with guarantees to be accepted by practitioners. This dissertation aims to contribute to the rich body of literature for addressing these three challenges.

Buildings will be the main type of energy system considered in this thesis for different reasons: first, buildings are responsible for a large share of the final energy consumption and CO_2 emissions worldwide (about 34% and 37 % respectively in 2021 (Global Alliance for Buildings and Construction, 2022)) and exhibit a great potential for energy savings through the use of advanced control strategies in contrast with the prevalent classical control approaches (Drgoňa et al., 2020; Mařík et al., 2011). Second, they are promising candidates for helping grid operators balance the power grid through frameworks like Demand Response (DR) since their energy consumption patterns show a certain flexibility in which they can be changed without negatively impacting the occupants (Darwazeh et al., 2022; H. Li et al., 2021). Third, due to increased electrification of, e.g., heating systems and adoption of local generation through Photovoltaic (PV) systems, the potential impact of buildings with respect to the two previously mentioned measures is expected to increase even further (IEA, 2022).

Data-driven methods for building control have proven to be promising in addressing

the mentioned challenges (Maddalena et al., 2020). Relying on data to build prediction models, controllers, or estimate available consumption flexibility, makes the approaches in principle scalable to many different assets since the need for in-depth modeling is circumvented. However, the quality of the results strongly depends on the amount and quality of the available data. This does not only hold true for data-driven methods for buildings, but data-driven methods and Machine Learning (ML) in general (Budach et al., 2022). Sources of uncertainty for the learning, but also the real-time operation, are to be found in erroneous measurements (Dong et al., 2019), unmeasured disturbances, like occupancy (Oldewurtel, Sturzenegger, & Morari, 2013), and inaccurate forecasts, e.g., of the weather conditions (Oldewurtel et al., 2012).

The field of uncertainty quantification (Smith, 2013) aims at assessing the magnitude and influence of the different sources of uncertainty, which can then be dealt with in the form of stochastic (Mesbah, 2016) or robust (Bhattacharyya, 2017; Petersen & Tempo, 2014) methods. The former assumes information about the underlying distribution of the uncertainty-generating process and the latter usually requires the uncertainty to lie in a predefined set. These methods are also vital in the case of building operation, to guarantee occupant comfort despite various uncertainties and disturbances, and thus to achieve a broad acceptance of advanced control among practitioners and occupants (O'Grady et al., 2021).

This thesis contributes by proposing a method for robust uncertainty quantification in the framework of kernel methods, which can, e.g., be used to verify that certain control actions respect constraints, a building model library for comparing control algorithms with predefined metrics, and a framework to estimate and coordinate consumption flexibility of buildings, while considering uncertainty with a user-defined risk level. The contributions are explained in more detail next.

1.2 Contributions

This dissertation is organized into three chapters with individual conclusions and next steps. The main contributions are summarized as follows:

Chapter 2: Robust uncertainty quantification in a non-parametric setting

In this chapter, we address uncertainty quantification in data-driven function approximation with kernel methods. Utilizing their non-parametric nature, error bounds are developed that depend only on the sample locations. They rely on prior knowledge about the complexity of the data-generating function and the magnitude of possible noise corruption of the samples, which is assumed to be bounded without further knowledge of an underlying distribution. Different types of bounds are developed, starting with closed-form ones that are centered around a nominal model, to optimal model-free bounds that rely on

solving a quadratically-constrained linear program, as well as its dual formulation, which can provide an overapproximation of the actual bound for each suboptimal but feasible solution. The distinguishing factor of our bounds is that they provide deterministic error estimation instead of a probabilistic one. The efficacy of the bounds is demonstrated in different simulation examples, from a standard function approximation to a nonlinear system identification, to the verification of the feasibility of control actions. The results have been developed in collaboration with Emilio T. Maddalena, Yuning Jiang, and Colin N. Jones, and this chapter is based on the following publications:

- P. Scharnhorst, E. T. Maddalena, Y. Jiang, & C. N. Jones. (2023). Robust Uncertainty Bounds in Reproducing Kernel Hilbert Spaces: A Convex Optimization Approach. *IEEE Transactions on Automatic Control*, 68 (5), 2848-2861, https://doi.org/10.1109/TAC.2022.3227907.
- E. T. Maddalena, P. Scharnhorst, & C. N. Jones. (2021). Deterministic error bounds for kernel-based learning techniques under bounded noise. *Automatica*, 134, 109896, https://doi.org/10.1016/j.automatica.2021.109896.

An MPC scheme based on the closed-form bounds, which is not covered in this thesis, was presented in

- E. T. Maddalena, P. Scharnhorst, Y. Jiang, C.N. & Jones. (2021). KPC: Learning-Based Model Predictive Control with Deterministic Guarantees. *Proceedings of the 3rd Conference on Learning for Dynamics and Control, in Proceedings of Machine Learning Research*, 144, 1015-1026.

as well as in the dissertation of Emilio T. Maddalena.

Chapter 3: A controller benchmarking tool for buildings

The Python library Energym for controller benchmarking on buildings is presented in this part. The library provides an intuitive interface to different building models from the simulation tools EnergyPlus and Modelica, inspired by the Reinforecement Learning (RL) benchmarking library Gym. The covered models range from a single zone, lightweight residential building with controllable Heat Pump (HP) power, to a seminar center model with temperature Setpoint (SP) control of 22 zones. Predefined Key Performance Indicators (KPIs) are tracked and evaluated during the simulations. To the best of our knowledge, this library constituted the most extensive building control benchmarking library at the time of publication. Since its release, new libraries have been released and existing ones have been upgraded. We furthermore provide an example of the usage of Energym in an MPC scheme. This collaborative work is based on the publication

- P. Scharnhorst, B. Schubnel, C. Fernández Bandera, J. Salom, P. Taddeo, M. Boegli, T. Gorecki, Y. Stauffer, A. Peppas, & C. Politi. (2021). Energym: A Building Model Library for Controller Benchmarking. *Applied Sciences*, 11(8), 3518, https://doi.org/10.3390/app11083518.

The calibration of the different building models was done by the collaborators in the scope of the European project SABINA under grant agreement n°731211 and is therefore not covered in this dissertation.

Chapter 4: Uncertainty-aware flexibility estimation, scheduling, and dispatch in buildings

This chapter discusses the development of a consumption flexibility estimation and coordination framework for energy systems, specifically buildings and other thermal systems. Risk measures are used to quantify uncertainty in a data-driven virtual battery modeling approach, which is in turn used to quantify feasible request trajectories. Request trajectories with different characteristics are considered in the estimation, which leads to a flexibility representation in the form of flexibility envelopes. Those envelopes are used in the formulation of a Mixed Integer Linear Program (MILP) to schedule the activation of a pool of assets for flexibility provision, based on an external request signal. The actual dispatch of incoming requests is done by a heuristic algorithm, using the active assets according to the determined schedule. This approach combines the strengths of data-driven methods, which allow its application to many assets without the need for in-depth modeling, uncertainty quantification, which results in a freely choosable risk level in the estimation, and an efficient MILP formulation, which permits the use of many assets in a day ahead scheduling scheme. Both the flexibility estimation and the scheduling and dispatch are tested in simulation for buildings from Energym. The following works are the base for this part:

- P. Scharnhorst, B. Schubnel, R. E. Carrillo, P. -J. Alet, & C. N. Jones. (2023). Uncertainty-aware Flexibility Envelope Prediction in Buildings with Controller-agnostic Battery Models. *2023 American Control Conference (ACC), San Diego, CA, USA*, 583-590, https://doi.org/10.23919/ACC55779.2023.10156041.
- P. Scharnhorst, B. Schubnel, R. E. Carrillo, P. -J. Alet, & C. N. Jones. (2023). Risk-aware scheduling and dispatch of flexibility events in buildings. *ArXiv preprint*, https://arxiv.org/abs/2311.05402.

Chapter 2

Robust uncertainty quantification in a non-parametric setting

2.1 Introduction

An integral part of employing advanced control strategies like MPC is the availability of an accurate model of the controlled system. The approaches to get such a model are usually grouped into three categories: white-box, grey-box, or black-box modeling. An overview of these three paradigms in the context of building systems can be found in (Boodi et al., 2018; Drgoňa et al., 2020). While white-box models solely rely on in-depth equation-based modeling, see e.g., (Salakij et al., 2016), black-box models are fully data-driven, using e.g. Neural Networks (NNs) (Y. Li & Tong, 2021), and grey-box models use data to fit the parameters of models with incorporated physical knowledge (Di Natale et al., 2022). The former, while potentially being very accurate, suffer from the effort and knowledge required to build the models, which especially makes it difficult to scale up to a large set of systems (H. Gao et al., 2019). Data-driven approaches on the other hand become more viable due to the increasing number of sensors and increased data collection for control systems (Soudbakhsh et al., 2023). Depending on the amount of available data, different methods can be applied to achieve the best results: linear models often provide a good approximation while having low data requirements, non-parametric methods like GPs or KRRs can represent nonlinear relationships with low to medium data requirements, and NNs require large amounts of data to learn complex structures and dependencies (Drgoňa et al., 2020). The framework of data-driven modeling promises an easy application and scalability, however, its advantages come with a drawback: poor data quality, e.g., through noisy measurements or incomplete data, can have a large effect on the resulting prediction and control quality, potentially affecting the ability of the controller to satisfy constraints (Maddalena et al., 2020).

Uncertainty quantification deals with the assessment of, e.g., experimental uncertainties, modeling errors, or numerical errors (Smith, 2013). In this chapter, we focus on the uncertainty quantification problem in modeling or more generally in learned functional relationships. This is done in the setting of kernel methods, which use evaluations of so-called kernel functions at sample locations as basis functions, to create a surrogate of the true underlying function. Using basis functions based on available data instead of fixed ones is what gives them the attribute "non-parametric". Popular methods in this domain encompass GP regression (Rasmussen & Williams, 2005), KRR, or SVR (Murphy, 2012; Schölkopf & Smola, 2018).

In the noise-free case, deterministic generalization error bounds have been developed in the field of scattered data approximation (Wendland, 2004). The popularity of GPs is partly due to their ability to provide probabilistic error intervals for the predictions, based on available data affected by Gaussian noise and prior assumptions like the choice of the kernel function. Numerous works make use of these Bayesian confidence levels in predictive control schemes to mitigate the impact of uncertainty, see e.g., (Hewing et al., 2020; Nghiem & Jones, 2017).

A frequentist way of quantifying uncertainty, based on assumptions on the complexity of an underlying ground-truth function in a function space, in the form of a norm upper bound in an RKHS, is given in (Koller et al., 2018) for the application of safe exploration in learning-based MPC. Similar types of probabilistic error bounds have been developed, focusing on the out-of-sample prediction error quantification while assuming the noise affecting the data samples to be (sub-)Gaussian. Especially in the multi-armed bandit setting, where the goal is to find the maximum of an unknown function from noisy samples while minimizing the cumulative regret, these bounds are used. (Srinivas et al., 2012) provides one of the earliest examples of such bounds in the context of the mentioned setting, followed by (Chowdhury & Gopalan, 2017), both relying on the information-gain for the error quantification. Utilizing the results from (Chowdhury & Gopalan, 2017), (Fiedler et al., 2021) forgoes the need for the information-gain, thus deriving uncertainty bounds that are practically usable in control settings. (Lederer et al., 2019) uses a different approach to derive uniform error bounds for GP regression with Gaussian noise, relying on Lipschitz constants for the underlying kernel, instead of a direct assumption on the ground-truth complexity. The bounds are targeted for the control setting as well. Extending the approach to online learning through using a model composed of local GPs, the bounds are made available for efficiently incorporating new data samples in (Lederer et al., 2021). Another extension is presented in (Capone et al., 2022), to provide bounds that hold in the case of hyperparameter misspecification in GPs.

The previously mentioned approaches all present probabilistic methods for error quantification, whereas in robust control settings often deterministic guarantees are desired, comprising situations where noise is bounded rather than Gaussian. The importance of bounded noise in the control context has been acknowledged multiple decades ago

(Fogel & Huang, 1982), and also plays an important role in other branches of science (d'Onofrio, 2013). Despite this, only a limited number of works are available tackling this problem in RKHSs. Exploiting the bounded noise assumption and building on results from (Srinivas et al., 2012), (Hashimoto et al., 2022) presents deterministic error bounds around a nominal GP model. A potentially model-free way of bounding the values of a ground-truth function, without the complexity assumption, is given in (Fiedler et al., 2022). Through the use of hard shape constrained kernel machines (Aubin-Frankowski & Szabo, 2020), the authors are able to incorporate geometric constraints into the construction of deterministic uncertainty sets, to ensure e.g. differentiability of the resulting bounding functions.

In this chapter, we derive deterministic error bounds, in closed form and through a convex programming approach, for the function approximation problem in an RKHS with samples affected by bounded noise. Building on the results for noise-free interpolation bounds, we propose bounds that are applicable to any approximation expressed as a sum of partial kernel evaluations at the sample locations and demonstrate their application to the examples of the KRR and SVR regressors. The tightest possible bounds, considering the given information, are formulated as the pointwise solution to an infinite-dimensional problem and reformulated as a convex program through representer theorem arguments. These bounds are not centered around a fixed nominal model, and it is shown how the dual formulation can be used to get feasible bounds in an iterative scheme. All the bounds are tested and compared on different learning and control-related examples.

2.2 Kernels and RKHSs

We start by reviewing the theory behind kernels and their associated function spaces, after which we formalize the problem setting considered in this chapter. For further details, see e.g. (Schaback & Wendland, 2006).

Definition 1 (Kernel Function)**.** *A function $k : \Omega \times \Omega \to \mathbb{R}$ that is symmetric, defined on a non-empty input set Ω, is called a kernel function.*

Definition 2 (Kernel Matrix)**.** *For a kernel function k and $X = \{x_1, \ldots, x_N\} \subset \Omega$, the matrix K_{XX} defined by*

$$[K_{XX}]_{i,j} = k(x_i, x_j) \tag{2.1}$$

is called the kernel matrix.

In the following analysis, we restrict ourselves to a specific class of kernel functions, namely Positive Definite (PD) kernels. This class still contains many interesting kernel choices, e.g., the popular squared-exponential kernel, which is known to have the universal approximation property over compact domains (Micchelli et al., 2006).

Definition 3 (Positive Definite Kernels)**.** *A kernel function that additionally fulfills*

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j k(x_i,x_j) > 0, \quad \forall n \in \mathbb{N}, \forall \alpha_1,\dots,\alpha_n \in \mathbb{R}\backslash\{0\}, \forall x_1,\dots,x_n \in \Omega \tag{2.2}$$

is called a PD kernel.

Remark 1. *A kernel function can be interpreted as a measure of similarity between elements of Ω. Considering $\Omega \subset \mathbb{R}^m$ for simplicity, an example of this is given by the squared exponential kernel, also known as the Radial Basis Function (RBF) kernel since its value only depends on the distance of its arguments:*

$$k(x_1,x_2) = \exp\left(-\frac{\|x_1-x_2\|^2}{2\ell^2}\right) \tag{2.3}$$

with a lengthscale parameter ℓ. Like this, it can be seen that $0 < k(x_1,x_2) \leq 1$ for the squared exponential kernel, with $k(x_1,x_2) = 1$ iff $x_1 = x_2$.

Each kernel spans a unique space of functions, called Reproducing Kernel Hilbert Space (RKHS), as defined in the following:

Definition 4 (Reproducing Kernel Hilbert Space)**.** *Let Ω be a non-empty set and $\mathcal{H} \subset \mathbb{R}^{\Omega}$ a Hilbert space with inner product $\langle\cdot,\cdot\rangle_{\mathcal{H}} : \mathcal{H}\times\mathcal{H} \to \mathbb{R}$. If there exists a kernel $k : \Omega\times\Omega \to \mathbb{R}$ with*

$$g(x) = \langle g, k(x,\cdot)\rangle_{\mathcal{H}}, \forall g \in \mathcal{H}, x \in \Omega \tag{2.4}$$

and

$$\mathcal{H} = clos(span\{k(x,\cdot), x \in \Omega\}) \tag{2.5}$$

then $\mathcal{H}$ is called an RKHS with reproducing kernel k.

The property (2.4) is the so-called reproducing property. For each RKHS there exists a unique reproducing kernel and vice versa (Aronszajn, 1950). Property (2.5) shows that we can express elements of $\mathcal{H}$ as weighted sums of partial kernel evaluations. To express the norm of an element of an RKHS, we focus on elements with a finite expansion, while for elements with infinite expansion, we can consider the limit of a corresponding sequence: let $g(\cdot) = \sum_{i=1}^{N}\alpha_i k(x_i,\cdot), N \in \mathbb{N}, \alpha_i \in \mathbb{R}, x_i \in \Omega, i = 1,\dots,N$. Then we can express its norm as

$$\|g\|_{\mathcal{H}}^2 = \left\langle \sum_{i=1}^{N}\alpha_i k(x_i,\cdot), \sum_{i=1}^{N}\alpha_i k(x_i,\cdot)\right\rangle_{\mathcal{H}} = \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j k(x_i,x_j) = \alpha^\top K_{XX}\alpha \tag{2.6}$$

where $\alpha = [\alpha_1,\dots,\alpha_N]^\top$, $X = [x_1,\dots,x_N]^\top$.

This norm can be seen as a measure of complexity of the element in the RKHS, where a low norm corresponds to a low complexity, i.e. a slow-changing function, and a high norm corresponds to a high complexity and a fast-changing function. This can be observed by bounding the possible change of a function $g \in \mathcal{H}$ between two sample locations $\hat{x}_1, \hat{x}_2 \in \Omega$, assuming a finite expansion $g(\cdot) = \sum_{i=1}^{N} \alpha_i k(x_i, \cdot)$, as well as an RBF kernel:

$$|g(\hat{x}_1) - g(\hat{x}_2)| = |\langle g, k(\hat{x}_1, \cdot)\rangle_{\mathcal{H}} - \langle g, k(\hat{x}_2, \cdot)\rangle_{\mathcal{H}}| = |\langle g, k(\hat{x}_1, \cdot) - k(\hat{x}_2, \cdot)\rangle_{\mathcal{H}}| \tag{2.7}$$

$$\leq \|g\|_{\mathcal{H}} \sqrt{\langle k(\hat{x}_1, \cdot) - k(\hat{x}_2, \cdot), k(\hat{x}_1, \cdot) - k(\hat{x}_2, \cdot)\rangle_{\mathcal{H}}} \tag{2.8}$$

$$= \|g\|_{\mathcal{H}} \sqrt{2(1 - k(\hat{x}_1, \hat{x}_2))} \leq \sqrt{2}\|g\|_{\mathcal{H}} \tag{2.9}$$

where (2.7) follows from the reproducing property and the bilinearity of the inner product, (2.8) from the Cauchy-Schwarz inequality, and (2.9) again by the bilinearity and the positivity and boundedness of RBF kernels.

As a last introductory step, we consider a finite subset $X \subset \Omega$ and define the power function $P_X : \Omega \to \mathbb{R}_{\geq 0}$ of a kernel k as

$$P_X(x) = \sqrt{k(x, x) - K_{xX} K_{XX}^{-1} K_{Xx}} \tag{2.10}$$

where $K_{Xx} = \begin{bmatrix} k(x, x_1) & \dots & k(x, x_N) \end{bmatrix}^\top \in \mathbb{R}^N$ and $K_{xX} = K_{Xx}^\top$. Whenever clear from the context, the reference to X will be omitted. $P_X(x)$ can be interpreted as a form of statistical covariance, it is positive $\forall x \in \Omega, x \notin X$, and evaluates to zero $\forall x \in X$.

2.3 The learning problem

With the tools of the previous section at hand, consider the problem of learning an unknown map $f : \Omega \to \mathbb{R}$, referred to as the ground-truth or target function.[I] Herein $\Omega \subset \mathbb{R}^m$ is assumed to be compact.

Assumption 1. *Given a kernel k, we assume that the ground-truth f belongs to its RKHS, $\mathcal{H}$. Additionally, an upper bound for its norm $\|f\|_{\mathcal{H}} \leq \Gamma$ is available.*

This assumption will be discussed in more detail in the next section, here we can use it to make observations about the complexity of different approximations, with the same kernel k.

In order to reconstruct f, we collect a finite dataset

$$D = \{(x_i, y_i) \,|\, i = 1, \dots, N\} \tag{2.11}$$

[I] The more general case $f : \Omega \to \mathbb{R}^k$ can be handled by learning each output dimension separately.

composed of sites x_i and noisy evaluations of the ground truth

$$y_i = f(x_i) + \delta_i,\ i = 1, \dots, N \tag{2.12}$$

Assumption 2. *The data sites $X = \{x_1, \dots, x_N\}$ in D are assumed to be pairwise distinct.*

This assumption is needed for the invertibility of the corresponding kernel matrix. However, it is not very restrictive, since we can always choose a subset of the dataset that fulfills the assumption. In Section 2.5 we will lift this assumption and show a model-free way of computing error bounds that deals with multiple data samples at the same input location.

Assumption 3. *The measurement noise $\delta = \begin{bmatrix}\delta_1 & \dots & \delta_N\end{bmatrix}^\top$ is bounded $|\delta_i| \leq \bar{\delta}$, with $\bar{\delta} \in \mathbb{R}_{\geq 0}$ known.*

For the learning problem, we restrict our attention to models $s : \Omega \to \mathbb{R}$ built as a weighted sum of kernels that are centered at the data locations, for a given kernel k

$$s(x) = \sum_{i=1}^{N} \alpha_i k(x_i, x) = \alpha^\top K_{Xx}\,. \tag{2.13}$$

Solutions to a number of optimal fitting problems have this form as discussed next. Since the considered basis functions are defined by the kernel and the sample locations, constructing a model is equivalent to deciding the α coefficients.

2.3.1 The noise-free case

Before discussing the approximation case, we take a look at the interpolation case for $\bar{\delta} = 0$, i.e., for noise-free samples. In the absence of noise, the labels in D perfectly represent f. We can then solve the approximation problem by finding an $s \in \mathcal{H}$ such that the evaluations $s(x_i)$ match $f(x_i) =: f_{x_i}$ for all points in D. This can be posed as the variational problem

$$\bar{s} = \underset{s \in \mathcal{H}}{\arg\min} \quad \|s\|_{\mathcal{H}}^2 \tag{2.14a}$$

$$\text{s.t.} \quad s(x_i) = f_{x_i} \quad \forall i = 1, \dots, N \tag{2.14b}$$

in which the objective favors low-complexity solutions, measured by the norm $\|\cdot\|_{\mathcal{H}}$.

The optimal recovery property, see e.g., (Wendland, 2004), states that the minimizer to (2.14) exists and assumes the form (2.13). The solution $\bar{s}$ can be therefore found by simply solving the linear system of equations $K_{XX}\alpha = f_X$ for α, where $f_X = \begin{bmatrix}f(x_1) & \dots & f(x_N)\end{bmatrix}^\top$. Given that the kernel k is PD and Assumption 2, K_{XX} is

positive-definite and hence invertible. Therefore, $\alpha = K_{XX}^{-1} f_X$ and the unique optimizer of (2.14) is

$$\bar{s}(x) = f_X^\top K_{XX}^{-1} K_{Xx} \tag{2.15}$$

Because of (2.6), we see that its norm can be expressed in terms of the data values as $\|\bar{s}\|_{\mathcal{H}}^2 = f_X^\top K_{XX}^{-1} f_X$.

Remark 2. *It holds that $\|\bar{s}\|_{\mathcal{H}} \leq \|f\|_{\mathcal{H}}$ independently of the number of samples in D. This stems from f being the solution to* (2.14) *when the equality constraints are imposed for all $x \in \Omega$.*

2.3.2 The noisy case

To handle noise in the data, i.e. $\bar{\delta} > 0$ in Assumption 3, we present two approximation techniques, namely KRR and SVR. The goal of these approximations, in contrast to interpolations, is to closely fit the data, while smoothing the response to not follow noise-related fluctuations.

The KRR problem looks as follows:

$$s^* = \arg\min_{s \in \mathcal{H}} \ \frac{1}{N} \sum_{i=1}^{N} (y_i - s(x_i))^2 + \lambda \|s\|_{\mathcal{H}}^2 \tag{2.16}$$

This unconstrained problem tackles exactly this balance, with a data term to achieve a close fit, and a norm term to favor a lower complexity function. The balance is dictated by the regularization weight $\lambda \in \mathbb{R}_{\geq 0}$, where $\lambda = 0$ recovers the solution to the interpolation problem. The KRR problem only makes indirect use of the noise assumption 3, by informing the choice of λ. A more direct use of this assumption can be made in the case of SVR, given in the following form:

$$\begin{aligned} s^\star = \arg\min_{s \in \mathcal{H}} \quad & \|s\|_{\mathcal{H}}^2 && \text{(2.17a)} \\ \text{s.t.} \quad & |s(x_i) - y_i| \leq \epsilon \quad \forall i = 1, \ldots, N \end{aligned}$$

which can be seen as ϵ-SVR with hard margins (Schölkopf & Smola, 2018) and $\epsilon \geq 0$. When choosing $\epsilon = 0$, the interpolation problem is again recovered, whereas a choice of $\epsilon = \bar{\delta}$ gives the minimum norm solution that is consistent with our data and assumptions. In particular, we have $\|s^\star\|_{\mathcal{H}} \leq \|\bar{s}\|_{\mathcal{H}}$ and $\|s^\star\|_{\mathcal{H}} \leq \|f\|_{\mathcal{H}}$ if ϵ is chosen in that way.

In the formulation of both the KRR and the SVR problem, we are dealing with an infinite-dimensional problem over the RKHS $\mathcal{H}$. However, both solutions can be expressed

in the form (2.13), as stated by the representer theorem, a general form of which will be given next, as presented in (Schölkopf et al., 2001):

Theorem 1 (Representer theorem)**.** *Given a nonempty set Ω, a PD kernel k, its corresponding RKHS $\mathcal{H}$, a dataset D as in* (2.11)*, a strictly monotonically increasing real-valued function h on $[0, \infty)$, and an arbitrary cost function $l : (\Omega \times \mathbb{R}^2)^N \to \mathbb{R} \cup \{\infty\}$. Then any $s \in \mathcal{H}$ minimizing*

$$l((x_1, y_1, s(x_1)), \ldots, (x_N, y_N, s(x_N))) + h(\|s\|_{\mathcal{H}}) \tag{2.18}$$

can be represented in the form (2.13)*.*

Proof. See (Schölkopf et al., 2001). □

The insights of the representer theorem allow us to reformulate problems (2.16) and (2.17) into finite-dimensional problems of determining the coefficients α in (2.13). For KRR, those coefficients can be expressed in closed form, using the dataset D and the regularization parameter λ. We have that

$$s^*(x) = \alpha^{*\top} K_{Xx} \tag{2.19}$$

with $\alpha^* = (K_{XX} + N\lambda I)^{-1} y$, where I denotes the identity matrix of appropriate size. The RKHS norm of s^* can be expressed as $\|s^*\|_{\mathcal{H}} = \alpha^{*\top} K_{XX} \alpha^* = c^\top K_{XX}^{-1} c$, where $c = K_{XX}(K_{XX} + N\lambda I)^{-1} y \in \mathbb{R}^N$ denotes the vector of function values of s^* at the sample locations, i.e. $s^*(x_i) = c_i, i = 1, \ldots, N$.

2.4 Closed form error bounds in RKHSs

Having multiple ways to approximate a function from data or even interpolate the data points, a natural question arises: What is the accuracy of the learned functions in approximating the ground truth? More specifically, we look at establishing out-of-sample error bounds for all kinds of models of the form (2.13). As a first step, we review some classic results from the field of scattered data approximation in the noise-free setting, informing the development of error bound in the case that the data samples are affected by bounded noise.

2.4.1 The noise-free case

In this section, we consider again the case that $\bar{\delta} = 0$ in Assumption 3. As an approximation of the ground truth function f given the dataset D, we consider the interpolant $\bar{s}$, given by (2.15).

A first bound on the approximation error at a location $x \in \Omega$ is then given by the following classic result, see e.g. (Wendland, 2004):

Lemma 1 (Simple noise-free error bound)**.** *Let $\bar{s}$ be the minimum norm interpolant of the samples f_X at the sample locations X and Γ a norm upper-bound for the ground truth function f as in Assumption 1. Then for $x \in \Omega$, we have*

$$|\bar{s}(x) - f(x)| \leq P_X(x)\|f\|_{\mathcal{H}}^2 \leq P_X(x)\Gamma^2 \tag{2.20}$$

Proof. See e.g. (Fasshauer, 2011). □

Remark 3 (On the necessity of Γ)**.** *The bound proposed in Lemma 1 and also the following results depend on the availability of a ground truth norm upper bound Γ. Data alone are not sufficient to compute any out-of-sample bounds when considering functions $f \in \mathcal{H}$, regardless of the number of samples $N < \infty$. Given any tentative bound ϵ at $x \notin X$, there exists $f_\rho \in \mathcal{H}$ consistent with the dataset that will violate the bound, that is, $f_\rho(x) > \epsilon + \rho$, for any pre-specified violation level $\rho > 0$. This is simply due to the existence of maps that can interpolate any finite set of samples. Restricting the search to the Γ-ball in $\mathcal{H}$ limits the flexibility of the considered functions, thus allowing for guarantees to be established. An analogous argument can be made in the space of Lipschitz functions. If no bound is posed on the Lipschitz constant of the ground truth, assuming Lipschitz continuity per se becomes vacuous. An example of how to empirically estimate the RKHS norm from data is given in Appendix 2.8.5.*

This error bound is on the conservative side, since it uses only information about the sample locations in the form of the power function and about the ground truth complexity, but neglects information about the values y and therefore the resulting interpolant complexity. This is different for the following, tighter bound (Fasshauer, 2011), for which we will also take a look at the proof to draw some insights for the proofs in the noisy case:

Lemma 2. *Given a dataset D, the interpolating model $\bar{s}$ admits the error bound*

$$|\bar{s}(x) - f(x)| \leq P_X(x)\sqrt{\Gamma^2 - \|\bar{s}\|_{\mathcal{H}}^2} \tag{2.21}$$

for any $x \in \Omega$, where f is the unknown ground-truth and $\|\bar{s}\|_{\mathcal{H}}^2 = f_X^\top K_{XX}^{-1} f_X$.

Proof. Let $x \in \Omega$ be a fixed query point, which is not in D. Denote by $\bar{s}_+$ the function of the form (2.13) interpolating all known points f_X in D and the unknown value $f_x := f(x)$.

We then have

$$\begin{aligned}\|\bar{s}_+\|^2_{\mathcal{H}} &= \begin{bmatrix} f_X \\ f_x \end{bmatrix}^\top \begin{bmatrix} K_{XX} & K_{Xx} \\ K_{xX} & K_{xx} \end{bmatrix}^{-1} \begin{bmatrix} f_X \\ f_x \end{bmatrix} \\ &= \begin{bmatrix} f_X \\ f_x \end{bmatrix}^\top \begin{bmatrix} K_{XX}^{-1} & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} \begin{bmatrix} f_X \\ f_x \end{bmatrix} + P_X^{-2}(x) \begin{bmatrix} f_X \\ f_x \end{bmatrix}^\top \begin{bmatrix} K_{XX}^{-1} K_{Xx} \\ -1 \end{bmatrix} \begin{bmatrix} K_{XX}^{-1} K_{Xx} \\ -1 \end{bmatrix}^\top \begin{bmatrix} f_X \\ f_x \end{bmatrix} \\ &= \|\bar{s}\|^2_{\mathcal{H}} + P_X^{-2}(x)(\bar{s}(x) - f_x)^2 \\ &\leq \Gamma^2\end{aligned}$$

where the second equality follows from the matrix inversion lemma (see Appendix 2.8.3), and the inequality follows from Remark 2. Finally, the last two lines imply $|\bar{s}(x) - f(x)| \leq P_X(x)\sqrt{\Gamma^2 - \|\bar{s}\|^2_{\mathcal{H}}}$.

If, on the other hand, the query point x belongs to the dataset D, the bound evaluates to zero, and thus it holds tightly. □

Both the bound itself and the proof provide us with some additional knowledge. First, we observe that if the bound on the ground truth norm is tight, i.e. $\|f\|_{\mathcal{H}} = \Gamma$, and the interpolant norm reaches that bound, the interpolant and the ground truth function take the same value in all $x \in \Omega$. Second, we observe from the proof how the interpolant norm changes when adding a new datapoint to D. The two deciding factors are the value of the power function at the new sample location, and the difference of the true value and the value of the previous interpolant at that location. This gives an intuition on where to sample new datapoints for decreasing the error bound overall: Points that lead to a high value in the power function, i.e. points with a large distance to the available samples in D, and points with a large difference in previous prediction and actual value, i.e. points that give new information, are preferred.

Through Lemma 2, evaluations of f for every $x \in \Omega$ can be bounded according to $f^{\min}(x) \leq f(x) \leq f^{\max}(x)$ with $f^{\min}(x) = \bar{s}(x) - P_X(x)\sqrt{\Gamma^2 - \|\bar{s}\|^2_{\mathcal{H}}}$ and $f^{\max}(x) = \bar{s}(x) + P_X(x)\sqrt{\Gamma^2 - \|\bar{s}\|^2_{\mathcal{H}}}$. Lemma 3 below establishes that the interval containing the ground truth is non-growing after the addition of a new datapoint.

Lemma 3. *Let x be any fixed query point in the domain Ω. Let Z be an augmented set of distinct data-sites, i.e., $Z = X \cup \{z\}$, $z \in \Omega, z \notin X$. Then we have*

$$[f_Z^{\min}(x), f_Z^{\max}(x)] \subseteq [f_X^{\min}(x), f_X^{\max}(x)] \tag{2.23}$$

Proof. See Appendix 2.8.1. □

Intuitively, this favorable property states that augmenting the dataset D with any new pair $(x, y) \in \Omega \times \mathbb{R}$ (while still satisfying Assumption 2) either preserves or sharpens the inequality (2.21). Moreover, this holds everywhere in the domain.

2.4.2 The noisy case

We turn our attention to the case $\bar{\delta} > 0$. Through the added noise, interpolating the noisy samples directly might lead to a function with a higher norm than the actual ground truth norm. We will quantify this potential norm change in comparison with the noise-free interpolant as a first step:

Lemma 4. *Let Assumptions 2 and 3 hold. Let moreover $\bar{s}(x) = f_X^\top K_{XX}^{-1} K_{Xx}$ be the model interpolating the noise-free values f_X, and $\tilde{s}(x) = y^\top K_{XX}^{-1} K_{Xx}$ the model interpolating the noisy values y. Then*

$$\nabla \leq \|\tilde{s}\|_{\mathcal{H}}^2 - \|\bar{s}\|_{\mathcal{H}}^2 \leq \Delta \tag{2.24}$$

where Δ denotes the maximum and ∇ the minimum of $(-\delta^\top K_{XX}^{-1}\, \delta + 2\, y^\top K^{-1}\, \delta)$ subject to $\|\delta\|_\infty \leq \bar{\delta}$.

Proof. It follows from expanding $\|\tilde{s}\|_{\mathcal{H}}^2$ as $\|\bar{s}\|_{\mathcal{H}}^2$ plus a perturbation term, and recalling the definitions of Δ and ∇. □

Using this lemma, we can separate the pointwise error by an approximation into a noise-free error term of the form (2.21) and an additional term taking the noise effect into account, all of which will be derived for a generic kernel function with finite expansion, as in (2.13).

Theorem 2. *Let $s(x) = \alpha^\top K_{Xx}$, for a given $\alpha \in \mathbb{R}^N$. Then, for any $x \in \Omega$, the ground-truth is bounded by $s(x) - S(x) \leq f(x) \leq s(x) + S(x)$ with*

$$S(x) = P_X(x)\, \sqrt{I^2 + \tilde{\Delta}} + \bar{\delta}\, \left\| K_{XX}^{-1} K_{Xx} \right\|_1 + |\tilde{s}(x) - s(x)| \tag{2.25}$$

and $\tilde{s}(x) = y^\top K_{XX}^{-1} K_{Xx}$. Here, the constant $\tilde{\Delta}$ is the minimum of the unconstrained convex problem

$$\min_{\nu \in \mathbb{R}^d} \left\{ \frac{1}{4} \nu^\top K_{XX} \nu + \nu^\top y + \bar{\delta}\, \|\nu\|_1 \right\}.$$

Proof. For any given $s(x) = \alpha^\top K_{Xx}$, we have

$$
\begin{aligned}
&|f(x) - s(x)| \\
&= |f(x) - \tilde{s}(x) + \tilde{s}(x) - s(x)| && (2.26)\\
&\leq |f(x) - (f_X + \delta_X)K_{XX}^{-1}K_{Xx}| + |\tilde{s}(x) - s(x)| && (2.27)\\
&\leq |f(x) - \bar{s}(x)| + \bar{\delta}\left\|K_{XX}^{-1}K_{Xx}\right\|_1 + |\tilde{s}(x) - s(x)| && (2.28)\\
&\leq P_X(x)\sqrt{I^2 - \|\bar{s}\|_{\mathcal{H}}^2} + \bar{\delta}\left\|K_{XX}^{-1}K_{Xx}\right\|_1 + |\tilde{s}(x) - s(x)| && (2.29)\\
&\leq P_X(x)\sqrt{I^2 + \Delta - \|\tilde{s}\|_{\mathcal{H}}^2} + \bar{\delta}\left\|K_{XX}^{-1}K_{Xx}\right\|_1 + |\tilde{s}(x) - s(x)| && (2.30)
\end{aligned}
$$

with f_X being the vector of true function values at the sample locations in X and δ_X the vector of additive measurement noise for the samples y. (2.27) follows from the triangle inequality and the additive noise property of y. Using the triangle inequality again, we arrive at (2.28), where $\bar{s}$ denotes the noise-free interpolant of f_X. The noise-free interpolation error bound gives the estimation in the first term of (2.29), while (2.30) follows from (2.24), with $\Delta = \max_{\|\delta\|_\infty \leq \bar{\delta}}(-\delta^\top K_{XX}^{-1}\delta + 2y^\top K_{XX}^{-1}\delta)$. A standard dualization procedure as the one presented in Appendix 2.8.4 leads to the dual problem

$$
\min_{\nu \in \mathbb{R}^d} \frac{1}{4}\nu^\top K_{XX}\nu + \nu^\top y + \bar{\delta}\|\nu\|_1 + y^\top K_{XX}^{-1}y \tag{2.31}
$$

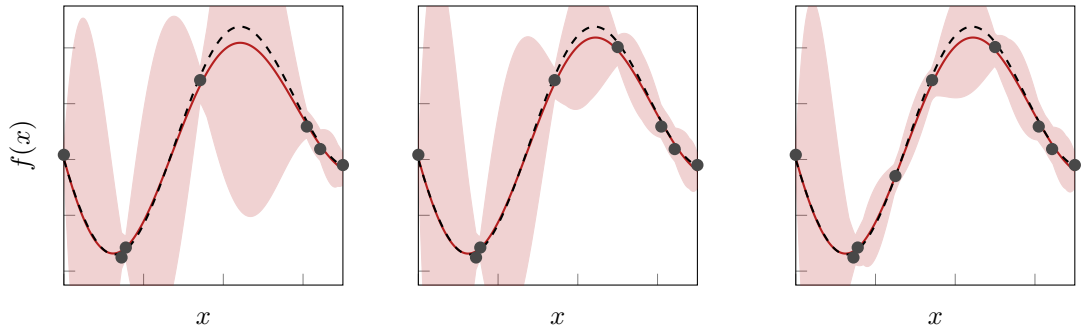
for Δ. Notice that the last term in (2.31) is constant and the same as the squared interpolant norm $\|\tilde{s}\|_{\mathcal{H}}^2$. Therefore, these terms cancel in (2.30) and we are left with

$$
|f(x) - s(x)| \leq P(x)\sqrt{I^2 + \tilde{\Delta}} + \bar{\delta}\left\|K_{XX}^{-1}K_{Xx}\right\|_1 + |\tilde{s}(x) - s(x)| \tag{2.32}
$$

where $\tilde{\Delta}$ represents (2.31) without the constant term. □

In the proof, we use the dual formulation of the problem to determine Δ from (2.24). Therefore, we have three distinct parts, one corresponding to the noise-free interpolation, one for the effect of the noise, and one for the difference of the noisy interpolating model and the one considered here. Since the first two parts are independent of the considered model, we see that the tightest bound is achieved for $\tilde{s}$. This indicates the suboptimality of those bounds, while still being very useful in practice, as will be shown in the numerical experiments. As a result of this, an equivalent result to Lemma 3 can not be derived for the closed form bounds in (2.25): The addition of a datapoint in one region of the domain might degrade the tightness of the bounds in another region (see Figure 2.1). However, since we are dealing with deterministic bounds, taking the intersection of intervals defined by successive bounds leads to a feasible and shrinking bound.

A fundamental reason for this suboptimality is the fact that the bounds are centered around a fixed model, while it is not clear what the optimal model is, or whether the bounds need to be symmetric around that model. Therefore, the model-free bounds



(a) Ground truth function, KRR model, and error bounds for 7 noisy samples

(b) Ground truth function, KRR model, and error bounds for 8 noisy samples

(c) Ground truth function, KRR model, and error bounds for 9 noisy samples

Figure 2.1 – Behavior of the closed form bounds (red shaded area) when adding new noisy samples (gray dots) of the ground truth function (black dotted line) to the KRR model (red solid line).

presented in the next section provide an alternative that circumvents the mentioned issue.

2.5 Optimal error bounds through convex programming

We consider a more general setting in this section, compared to the one in the previous parts. In particular, we consider having a dataset $D = \{(x_i, \mathsf{y}_i)\}_{i=1}^{N}$ is given to us, being composed of inputs $x_i \in \Omega$ and outputs $\mathsf{y}_i \in \mathbb{R}^{n_i}$, $\mathsf{y}_i = [y_{i,1}, \ldots, y_{i,n_i}]^\top$ that contain n_i scalar samples collected at the same input location x_i. The dataset carries information about the underlying ground-truth map $f \in \mathcal{H}$ according to

$$y_{i,j} = f(x_i) + \delta_{i,j}, \tag{2.33}$$

where $\delta_{i,j}$ represents a bounded additive measurement noise in line with Assumption 3, i.e. $|\delta_{i,j}| \leq \bar{\delta}, \forall i, j$.

The question we are posing in this section is: What are the minimum and maximum values that any function that fulfills our assumptions can take at a given sample location? We can formulate the answer to that question in the form of two infinite-dimensional variational problems, with a query point $x \in \Omega$ as a parameter:

$$\mathrm{G}(x) = \sup_{g \in \mathcal{H}} \{g(x) : \|g\|_{\mathcal{H}} \leq \Gamma, \|\Lambda g_X - \mathsf{y}\|_\infty \leq \bar{\delta}\}, \tag{2.34}$$

$$\mathrm{F}(x) = \inf_{g \in \mathcal{H}} \{g(x) : \|g\|_{\mathcal{H}} \leq \Gamma, \|\Lambda g_X - \mathsf{y}\|_\infty \leq \bar{\delta}\}, \tag{2.35}$$

where Λ is a matrix of zeros and ones to repeat a function value whenever multiple

samples with the same input location exist. It is defined in Appendix 2.8.2.

Due to Assumption 1 on the ground truth norm, the function values of this ground truth are bounded, and therefore, the supremum and infimum in (2.34) and (2.35) are attained. The boundedness of the function values can be seen by

$$|f(x)| = |\langle f, k(x,\cdot)\rangle_{\mathcal{H}}| \leq \|f\|_{\mathcal{H}} \|k(x,\cdot)\|_{\mathcal{H}} = \|f\|_{\mathcal{H}} \sqrt{k(x,x)} < \infty \tag{2.36}$$

through applying the Cauchy-Schwarz inequality.

Given a query location x, $\mathrm{G}(x)$ yields the tightest upper bound for $f(x)$ over all members $g \in \mathcal{H}$ of our hypothesis space that are consistent with our dataset, as well as our knowledge on the ground-truth complexity $\|f\|_{\mathcal{H}} \leq \Gamma$. The equivalent applies to $\mathrm{F}(x)$ as the tightest lower bound, thus we directly get that

$$\mathrm{F}(x) \leq f(x) \leq \mathrm{G}(x) \quad \forall x \in \Omega. \tag{2.37}$$

Due to the property of being the tightest bounds consistent with our assumptions, we refer to them as the optimal bounds.

Since solving an infinite-dimensional problem to bound function values is not realistic in practice, we consider two finite-dimensional ones and show their equivalence to (2.34). We focus on the upper bound for now, but the following results hold equivalently for the lower bounding problem.

Consider the convex parametric quadratically-constrained linear program

$$\mathrm{C}(x) = \max_{c \in \mathbb{R}^N, c_x \in \mathbb{R}} \quad c_x \tag{2.38a}$$

$$\text{s.t.} \begin{bmatrix} c \\ c_x \end{bmatrix}^\top \begin{bmatrix} K_{XX} & K_{Xx} \\ K_{xX} & k(x,x) \end{bmatrix}^{-1} \begin{bmatrix} c \\ c_x \end{bmatrix} \leq \Gamma^2 \tag{2.38b}$$

$$\|\Lambda c - \mathsf{y}\|_\infty \leq \bar{\delta} \tag{2.38c}$$

for any $x \in \Omega \backslash X$, and extend its value function to points $x = x_i \in X$ with the solution of

$$\mathrm{C}(x_i) = \max_{c \in \mathbb{R}^N} c_i \tag{2.39a}$$

$$\text{s.t. } c^\top K_{XX}^{-1} c \leq \Gamma^2 \tag{2.39b}$$

$$\|\Lambda\, c - \mathsf{y}\|_\infty \leq \bar{\delta} \tag{2.39c}$$

where c_i is the i-th component of c.

This can be thought of as finding a map that interpolates the points $\{(x_i, c_i)\}_{i=1}^N$ and maximizes its value c_x at the input location x. The two cases are distinguished due to the

matrix in (2.38b) becoming singular for any $x \in X$, and since it allows for one decision variable to be eliminated. Finally, the connection between (2.34) and (2.38) is stated next.

Theorem 3 (Finite-dimensional equivalence)**.** *The objective in* (2.34) *attains its supremum in* $\mathcal{H}$ *and* $\mathrm{G}(x) = \mathrm{C}(x)$ *for any* $x \in \Omega$.

Proof. The proof uses arguments along the lines of the classical representer theorem proofs.

Let $\mathbb{X} := X \cup \{x\}$ and define the finite-dimensional subspace $\mathcal{H}^{\|} = \{g \in \mathcal{H} : g \in \operatorname{span}(k(x_i, \cdot), x_i \in \mathbb{X})\}$. Furthermore, let $\mathcal{H}^{\perp} = \{g \in \mathcal{H} : \langle g, h^{\|} \rangle_{\mathcal{H}} = 0, \forall h^{\|} \in \mathcal{H}^{\|}\}$ be the orthogonal complement of $\mathcal{H}^{\|}$. Then, we have $\mathcal{H} = \mathcal{H}^{\|} \oplus \mathcal{H}^{\perp}$ and for all $g \in \mathcal{H}$, $\exists g^{\|} \in \mathcal{H}^{\|}, g^{\perp} \in \mathcal{H}^{\perp} : g = g^{\|} + g^{\perp}$. By employing the latter decomposition and using the reproducing property, we can reformulate (2.34) in terms of $\mathcal{H}^{\|}$ and $\mathcal{H}^{\perp}$ as

$$
\begin{aligned}
&\sup_{\substack{g^{\|} \in \mathcal{H}^{\|} \\ g^{\perp} \in \mathcal{H}^{\perp}}} \left\{ \begin{array}{l} \langle g^{\|} + g^{\perp}, k(x, \cdot) \rangle_{\mathcal{H}} : \\ \|g^{\|} + g^{\perp}\|_{\mathcal{H}}^2 \leq \Gamma^2, \left\| \Lambda (g^{\|} + g^{\perp})_X - \mathsf{y} \right\|_{\infty} \leq \bar{\delta} \end{array} \right\} \\
&\overset{(i)}{=} \sup_{\substack{g^{\|} \in \mathcal{H}^{\|} \\ g^{\perp} \in \mathcal{H}^{\perp}}} \left\{ g^{\|}(x) : \|g^{\|}\|_{\mathcal{H}}^2 + \|g^{\perp}\|_{\mathcal{H}}^2 \leq \Gamma^2, \left\| \Lambda g_X^{\|} - \mathsf{y} \right\|_{\infty} \leq \bar{\delta} \right\} \\
&\overset{(ii)}{=} \sup_{g^{\|} \in \mathcal{H}^{\|}} \left\{ g^{\|}(x) : \|g^{\|}\|_{\mathcal{H}}^2 \leq \Gamma^2, \left\| \Lambda g_X^{\|} - \mathsf{y} \right\|_{\infty} \leq \bar{\delta} \right\}
\end{aligned} \tag{2.40}
$$

In (i), the $g^{\perp}$ component vanished from the cost and from the last constraint due to orthogonality w.r.t. $k(x_i, \cdot) \in \mathcal{H}^{\|}$ for any $x_i \in \mathbb{X}$; moreover, the Pythagorean relation $\|g\|_{\mathcal{H}}^2 = \|g^{\|}\|_{\mathcal{H}}^2 + \|g^{\perp}\|_{\mathcal{H}}^2$ was also used. To arrive at the second equality (ii), one only has to note that the objective is insensitive to $g^{\perp}$ and that any $g^{\perp} \neq 0_{\mathcal{H}}$ would tighten the first constraint.

The attainment of the supremum is addressed next. Consider (2.40) and denote the members of $\mathcal{H}^{\|}$ simply as g. $\|g\|_{\mathcal{H}}^2 \leq \Gamma^2$ is a closed and bounded constraint as it is the sublevel set of a norm. We transform $\|\Lambda g_X - \mathsf{y}\|_{\infty} \leq \bar{\delta}$ into $|g(x_i) - y_{i,j}| \leq \bar{\delta}$, $i = 1, \ldots, N$, $j = 1, \ldots, n_i$. Sets of the form $\{a \in \mathbb{R} : |a| \leq b\}$ are clearly closed in $\mathbb{R}$, hence

$$\{g(x_i) \in \mathbb{R} : |g(x_i) - y_{i,j}| \leq \bar{\delta}, \forall i, j\}$$

is also closed. For any x_i, the evaluation functional $L_{x_i}(g) = g(x_i)$ is a linear operator and thus pre-images of closed sets are also closed. Consequently,

$$\{g \in \mathcal{H}^{\|} : |g(x_i) - y_{i,j}| \leq \bar{\delta}, \forall i, j\}$$

is closed in $\mathcal{H}^{\|}$. The intersection of a finite number of closed sets is necessarily closed, thus all constraints present in (2.40) define a closed feasible set. Since $\mathcal{H}^{\|}$ is finite-dimensional, any closed and bounded subset of it is compact (Heine–Borel); therefore, the continuous objective $L_x(g) = g(x)$ in (2.40) attains a maximum by the Weierstrass extreme value theorem.

Finally, we establish the connection between (2.34) and (2.38). From the above arguments, an optimizer for (2.34) must lie in $\mathcal{H}^{\|}$. The members $g \in \mathcal{H}^{\|}$ have the form $g(z) = \alpha^\top K_{\mathbb{X}z}$, being defined by the α weights. Due to the positive-definiteness of k, there exists a bijective map between outputs at the $\mathbb{X}$ locations $g_{\mathbb{X}} = \begin{bmatrix} g(x_1) & \dots & g(x_N) & g(x) \end{bmatrix}^\top$ and the weights α, namely $\alpha = K_{\mathbb{X}\mathbb{X}}^{-1} g_{\mathbb{X}}$. $K_{\mathbb{X}\mathbb{X}}$ denotes the kernel matrix associated with the set $\mathbb{X} = X \cup \{x\}$. Consequently, optimizing over $g \in \mathcal{H}^{\|}$ is equivalent to optimizing over $\begin{bmatrix} g(x_1) & \dots & g(x_N) & g(x) \end{bmatrix}^\top =: \begin{bmatrix} c^\top & c_x \end{bmatrix}^\top$. The bounded norm condition can be recast as

$$\|g\|_{\mathcal{H}}^2 = \langle g, g \rangle_{\mathcal{H}} = \alpha^\top K_{\mathbb{X}\mathbb{X}} \alpha = \begin{bmatrix} c^\top & c_x \end{bmatrix} K_{\mathbb{X}\mathbb{X}}^{-1} \begin{bmatrix} c^\top & c_x \end{bmatrix}^\top .$$

The remaining constraint and the objective are straightforward. Noting that this reformulation is valid for any $x \in \Omega$ concludes the proof. □

In an analogous way, a finite-dimensional equivalence can be shown for (2.35) and

$$\begin{aligned} \mathrm{B}(x) \;=\; & \min_{c \in \mathbb{R}^N, c_x \in \mathbb{R}} \quad c_x && \text{(2.41a)} \\ & \text{subj. to} \quad \begin{bmatrix} c \\ c_x \end{bmatrix}^\top \begin{bmatrix} K_{XX} & K_{Xx} \\ K_{xX} & k(x,x) \end{bmatrix}^{-1} \begin{bmatrix} c \\ c_x \end{bmatrix} \leq \Gamma^2 && \text{(2.41b)} \\ & \qquad \|\Lambda c - \mathsf{y}\|_\infty \leq \bar{\delta} && \text{(2.41c)} \end{aligned}$$

for any $x \in \Omega \backslash X$, and extended to

$$\mathrm{B}(x_i) = \min_{c \in \mathbb{R}^N} \{ c_i \,|\, c^\top K_{XX}^{-1} c \leq \Gamma^2,\ \|\Lambda c - \mathsf{y}\|_\infty \leq \bar{\delta} \}$$

for $x = x_i \in X$.

Therefore, we have

$$\mathrm{B}(x) \leq f(x) \leq \mathrm{C}(x) \quad \forall x \in \Omega. \tag{2.42}$$

2.5.1 Connections to the closed-form bounds

Naturally, the question arises as to how the optimal bounds relate to the closed-form ones presented in Theorem 2. As we will see, a relaxation of the constraints to the

problem (2.38) (and equivalently (2.41)) reveals the closed-form bounds. To make this connection, we assume that only one sample is present at each input location, i.e., $\mathsf{y}_i = y_i$ for $i = 1, \ldots, N$, so that $\mathsf{y} = y$, and focus on problem (2.38) only to avoid redundancy.

As a first step, we consider a change of variables in (2.38), by considering $\delta = c - y$. This gives

$$\mathrm{C}(x) = \max_{\delta \in \mathbb{R}^N, c_x \in \mathbb{R}} \quad c_x \tag{2.43a}$$

$$\text{s.t.} \begin{bmatrix} \delta + y \\ c_x \end{bmatrix}^\top \begin{bmatrix} K_{XX} & K_{Xx} \\ K_{xX} & k(x,x) \end{bmatrix}^{-1} \begin{bmatrix} \delta + y \\ c_x \end{bmatrix} \leq I^2 \tag{2.43b}$$

$$\|\delta\|_\infty \leq \bar{\delta} \tag{2.43c}$$

Using the matrix inversion identity from Appendix 2.8.2, reformulation (2.67), and solving for c_x in constraint (2.43b), we get

$$c_x \leq P(x)\sqrt{I^2 - y^\top K_{XX}^{-1} y - \delta^\top K_{XX}^{-1}\delta + 2y^\top K_{XX}^{-1}\delta} + y^\top K_{XX}^{-1} K_{Xx} + \delta^\top K_{XX}^{-1} K_{Xx} \tag{2.44}$$

Note that the terms involving y and not δ are related to the noisy interpolant $\tilde{s}$ introduced in Lemma 4, namely $\tilde{s}(x) = y^\top K_{XX}^{-1} K_{Xx}$ and $\|\tilde{s}\|_{\mathcal{H}}^2 = y^\top K_{XX}^{-1} y$. Since the goal is to maximize c_x under constraint (2.44), we can directly consider maximizing the right-hand side of (2.44). Therefore, we get

$$\max_{\|\delta\|_\infty \leq \bar{\delta}} \tilde{s}(x) + P(x)\sqrt{I^2 - \|\tilde{s}\|_{\mathcal{H}} - \delta^\top K_{XX}^{-1}\delta - 2y^\top K_{XX}^{-1}\delta} + \delta^\top K_{XX}^{-1} K_{Xx} \tag{2.45}$$

Relaxing the problem by allowing δ to take different values in the square root term and outside the square root, we observe the separability of the new objective

$$\tilde{s}(x) + P(x)\sqrt{I^2 - \|\tilde{s}\|_{\mathcal{H}} - \delta_1^\top K_{XX}^{-1}\delta_1 - 2y^\top K_{XX}^{-1}\delta_1} + \delta_2^\top K_{XX}^{-1} K_{Xx} \tag{2.46}$$

with $\|\delta_1\|_\infty \leq \bar{\delta}, \|\delta_2\|_\infty \leq \bar{\delta}$. To arrive at the closed-form bound (2.25), we make the following observations: First, $\tilde{\Delta}$, as in Theorem 2, is the solution of the dual problem to

$$\max_{\|\delta_1\|_\infty \leq \bar{\delta}} \left\{ -\delta_1^\top K_{XX}^{-1}\delta_1 + 2y^\top K_{XX}^{-1}\delta_1 - \|\tilde{s}\|_{\mathcal{H}}^2 \right\}.$$

Second, we have

$$\max_{\|\delta_2\|_\infty \leq \bar{\delta}} \delta_2^\top K_{XX}^{-1} K_{Xx} = \bar{\delta} \left\| K_{XX}^{-1} K_{Xx} \right\|_1.$$

To fully recover the closed-form bound (2.25) for a general model $s(x)$, we make use of the triangle inequality $|f(x) - s(x)| \leq |f(x) - \tilde{s}(x)| + |\tilde{s}(x) - s(x)|$.

From (2.45), the noise variable δ is seen to increase the maximum in two distinct ways: through the inner product $\delta^\top K_{XX}^{-1} K_{Xx}$, and via a norm augmentation corresponding to $\tilde{\Delta}$. One source of conservativeness in Theorem 2 is taking into account the worst-possible inner-product and norm increase jointly. Despite this fact, they yield competitive results for moderate noise levels, as shown numerically in Section 2.6. We moreover note that in the noise-free scenario, (2.45) and (2.46) are the same, and Theorem 2 simplifies to the classical bounds in the interpolation case in Lemma 2.

Remark 4. *The closed-form bounds presented in Theorem 2 feature a nominal model at their center, which can be desirable in practical situations that require a nominal prediction and not only the bounds. In the optimal bounds scenario, the $\bar{\delta}$-SVR model $s^\star(x)$ can be used as a nominal model. This choice is guaranteed to lie completely within $C(x)$ and $B(x)$, although not necessarily in the middle, since the map $s^\star$ belongs to $\mathcal{H}$ and is a feasible solution for* (2.34).

2.5.2 Properties of the optimal bounds

In the following, we explore the properties and behavior of the optimal bounds (2.42), e.g. their minimum width and the influence of added data.

First, given our knowledge of the noise bound $\bar{\delta}$, it is natural to ask what the limits of the uncertainty quantification technique considered herein are. For example, is the width of the envelope $\mathrm{C}(x) - \mathrm{B}(x)$ restricted to a certain minimum value that cannot be reduced even with the addition of new data? From (2.38c), it is clear that at any input location $x_i \in X$, $\mathrm{C}(x_i)$ and $\mathrm{B}(x_i)$ cannot be more than $2\bar{\delta}$ apart. In addition to that, the presence of the complexity constraint (2.38b) can decrease the width of this envelope, potentially leading to samples lying outside bounds. An illustration of this is given in Figure 2.2 (left).

Proposition 1 (Width smaller than the noise bound)**.** *If $\exists \mathsf{y}_i$ such that $y_{i,j} \geq C(x_i)$ or $y_{i,j} \leq B(x_i)$ for some j, then $C(x_i) - B(x_i) \leq \bar{\delta}$.*

Proof. Let $y_{i,j} \geq C(x_i)$, the result for $y_{i,j} \leq B(x_i)$ follows equivalently. We have

$$\begin{aligned} & y_{i,j} \geq C(x_i) \geq B(x_i) \geq y_{i,j} - \bar{\delta} \\ \Rightarrow & C(x_i) - B(x_i) \leq y_{i,j} - (y_{i,j} - \bar{\delta}) = \bar{\delta} \end{aligned}$$

□

As a special case, we consider sampling (x_i, y_i) with $\mathsf{y}_i = \begin{bmatrix} y_{i,1} & y_{i,2} \end{bmatrix}^\top$, $y_{i,1} = f(x_i) + \bar{\delta}$

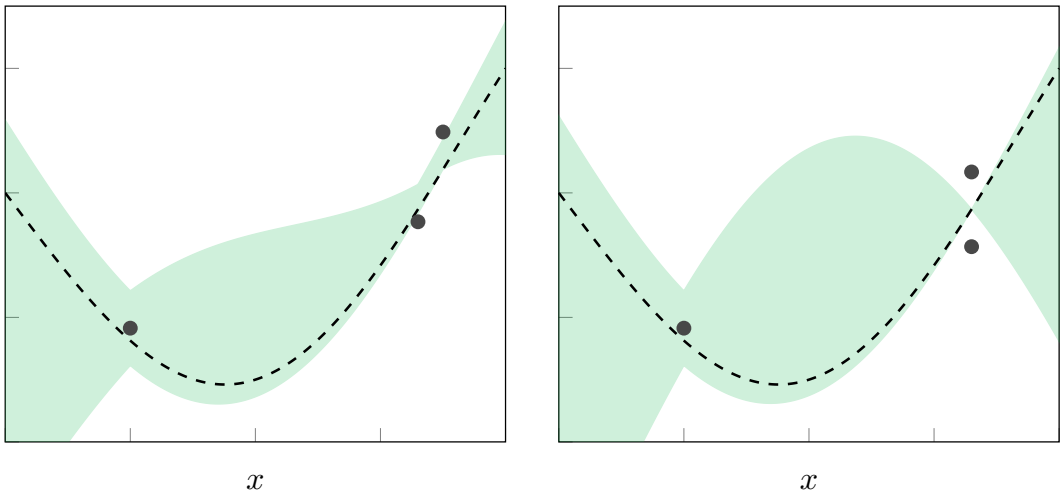
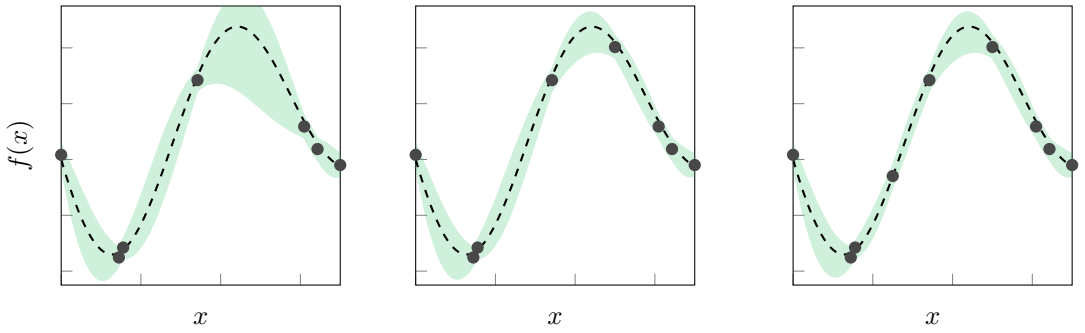


Figure 2.2 – (Left) Samples lying outside the uncertainty envelope, implying that the width at these locations is smaller than the noise bound. (Right) Redundant information is used to shrink the uncertainty envelope. In this scenario, we recover the ground-truth value at the sample location.

and $y_{i,2} = f(x_i) - \bar{\delta}$. Like this, $f(x_i) = (y_{i,1} + y_{i,2})/2$ is the only possible value attainable by the ground truth and the uncertainty envelope for this sample location reduces to a singleton as shown in Figure 2.2 (right). Regardless, adding data to an existing dataset, be it in the form of a new output at an already sampled location or a completely new input-output pair, can only reduce the uncertainty (see Figure 2.3).

Proposition 2 (Decreasing uncertainty)**.** *Let $C_1(x)$ denote the solution of* (2.38) *with a dataset $D_1 = \{(x_i, \mathsf{y}_i)\}_{i=1}^N$, and $C_2(x)$ the solution with $D_2 = D_1 \cup \{(x_{N+1}, \mathsf{y}_{N+1})\}$. Then $C_2(x) \leq C_1(x)$ for any $x \in \Omega$.*

Proof. Denote by $\mathbb{P}1$ the problem solved with D_1 and decision variables $\begin{bmatrix} c & c_x \end{bmatrix}$. Similarly, $\mathbb{P}2$ is associated with the dataset D_2 and the decision variables $\begin{bmatrix} c & c_x & c_z \end{bmatrix}$, where c_z are due to the additional input in D_2. Since D_2 contains all members of D_1, the ∞-norm constraint of $\mathbb{P}2$ can be recast as that of $\mathbb{P}1$ and an additional constraint for c_z and the new outputs. Let $\mathbb{X} := X \cup \{x\}$, $\bar{c} := \begin{bmatrix} c^\top c_x \end{bmatrix}^\top$ and $z := x_{N+1}$ be shorthand variables to



(a) Ground truth function and optimal error bounds for 7 noisy samples

(b) Ground truth function and optimal error bounds for 8 noisy samples

(c) Ground truth function and optimal error bounds for 9 noisy samples

Figure 2.3 – Behavior of the optimal bounds (green shaded area) when adding new noisy samples (gray dots) of the ground truth function (black dotted line).

ease notation. The complexity constraint of $\mathbb{P}2$ is then

$$\begin{bmatrix} \bar{c} \\ c_z \end{bmatrix}^\top \begin{bmatrix} K_{\mathbb{X}\mathbb{X}} & K_{\mathbb{X}z} \\ K_{z\mathbb{X}} & k(z,z) \end{bmatrix}^{-1} \begin{bmatrix} \bar{c} \\ c_z \end{bmatrix} \leq \Gamma^2 \tag{2.47a}$$

$$\overset{(i)}{\Leftrightarrow} \ \bar{c}^\top K_{\mathbb{X}\mathbb{X}}^{-1} \bar{c} + P_{\mathbb{X}}^{-2}(z) \left\| \begin{bmatrix} K_{\mathbb{X}\mathbb{X}}^{-1} K_{\mathbb{X}z} \\ -1 \end{bmatrix} \begin{bmatrix} \bar{c} \\ c_z \end{bmatrix} \right\|_2^2 \leq \Gamma^2 \tag{2.47b}$$

$$\overset{(ii)}{\Leftrightarrow} \begin{bmatrix} c \\ c_x \end{bmatrix}^\top \begin{bmatrix} K_{XX} & K_{Xx} \\ K_{xX} & k(x,x) \end{bmatrix}^{-1} \begin{bmatrix} c \\ c_x \end{bmatrix} \tag{2.47c}$$
$$+ P_{\mathbb{X}}^{-2}(z) \left(\bar{c}^\top K_{\mathbb{X}\mathbb{X}}^{-1} K_{\mathbb{X}z} - c_z \right)^2 \leq \Gamma^2,$$

using the matrix identity from Appendix 2.8.3 in (i) and $P_{\mathbb{X}}^2(z) = k(z,z) - K_{z\mathbb{X}} K_{\mathbb{X}\mathbb{X}}^{-1} K_{\mathbb{X}z}$. In (ii), the definitions of $\bar{c}$ and $\mathbb{X}$ were used. Thanks to $P_{\mathbb{X}}(z) \geq 0, \forall z$ and the quadratic term multiplying it, we conclude that for any choice of the decision variable c_z, (2.47c) is a tightened version of the complexity constraint of $\mathbb{P}1$, which is (2.38b). As a result, the maximum of $\mathbb{P}2$ is lower or equal than that of $\mathbb{P}1$. □

Let us take a closer look at the tightened constraint (2.47c). The term $\bar{c}^\top K_{\mathbb{X}\mathbb{X}}^{-1} K_{\mathbb{X}z} =: s(z)$ represents an interpolating model passing through the output values $\bar{c}$, that is, c and c_x. If the difference $s(z) - c_z$ can be made small, then the tightening will also be reduced, whereas it will be significant if the difference is large. This is in line with the observation on the norm increase of an interpolating model when adding a new sample, as seen in the discussion of Lemma 2. The result is of course dictated by the ∞-norm constraint, since c_z cannot be more than $\bar{\delta}$ away from all the outputs y available at z. Therefore, a new datum will cause significant shrinkage of the envelope at a point $z \in \Omega$ when the new output causes $s(z) - c_z$ to be large, which intuitively can be seen as a measure of gained

information through the new sample. Finally, this process is weighted by the inverse of the power function $P_{\mathbb{X}}^{-2}(z)$, which does not depend on any output, but only on the input locations.

Remark 5. *Recovering the ground truth as shown in Figure 2.2 (right) requires the noise realizations to match $\bar{\delta}$ and $-\bar{\delta}$; it is thus necessary to have tight noise bounds for it to happen. On the other hand, Proposition 2 guarantees the decreasing uncertainty property regardless of how accurate $\bar{\delta}$ is. Although not explicitly stated, a completely analogous result holds for the lower part of the envelope $B(x)$.*

2.5.3 A dual formulation

One of the fundamental sources of computational complexity in kernel learning lies in the inverse term K_{XX}^{-1}. Scaling these techniques to large datasets in a principled manner is still a topic of active research (Burt et al., 2020; Y. Zhang et al., 2013). Notice that K_{XX}^{-1} is explicitly present in (2.38), thus limiting its applicability to small and medium-sized problems due to the cubic time complexity associated with the inverse operation. In this section we discuss alternative formulations that can be solved more efficiently.

Following a standard dualization procedure, which can be found in Appendix 2.8.4, a Lagrangian dual for (2.38) can be the convex problem

$$\min_{\nu\in\mathbb{R}^{\tilde{N}},\,\lambda>0} \quad \frac{1}{4\lambda}\nu^\top \Lambda K_{XX}\Lambda^\top\nu + \left(\mathsf{y} - \frac{1}{2\lambda}\Lambda K_{Xx}\right)^\top \nu + \bar{\delta}\|\nu\|_1 + \frac{1}{4\lambda}k(x,x) + \lambda\Gamma^2 \quad (2.48)$$

for any query point $x \in \Omega\backslash X$. In our notation the dimension $\tilde{N} = \sum_{i=1}^{N} n_i$ is the total number of outputs, that is, the size of y. As detailed in Appendix 2.8.4, under the assumption of the complexity constraint not being active, the dual of (2.39) is also given by (2.48), meaning that it could be used $\forall x \in \Omega$.

Contrary to the primal formulation, (2.48) only involves the kernel matrix itself and not its inverse, avoiding thus the aforementioned adversity. Furthermore, the query point x enters (2.48) through the terms K_{Xx} and $k(x,x)$. The former measures the similarity between the query point x and each of the inputs in X; the latter is simply a constant term for translation-invariant kernels and always evaluates to 1 in the specific case of the squared-exponential kernel.

The optimization problem above is convex since it is a quadratic-over-linear function with $\Lambda K_{XX}\Lambda^\top \succeq 0$ and λ restricted to the positive reals. The objective can moreover be decomposed into a differentiable part and a single non-differentiable term $\|\nu\|_1$, with ν unconstrained. This class of problems has long been studied and mature numerical algorithms exist to solve them, notably different flavors of splitting methods such as the Alternating Direction Method of Multipliers (ADMM) (Boyd et al., 2011, Section 6).

Alternatively, a standard linear reformulation could be employed to substitute $\|\nu\|_1$ by $\sum_i \eta_i$, with additional constraints $-\nu \leq \eta$, $\nu \leq \eta$. The result is a completely differentiable objective but with extra decision variables and linear constraints. Next, a mild condition is given, ensuring a zero duality gap between the primal and dual problems.

Proposition 3 (Strong duality)**.** *If $\bar{\delta} > \delta_{i,j}, \forall i, j$ and $\Gamma > \|f\|_{\mathcal{H}}$, then no duality gap exists, i.e., the solutions to* (2.38) *and* (2.48) *are the same.*

Proof. Consider the primal problem (2.38) and select $c = f_X$ and $c_x = f(x)$. Let $\mathbb{X} := X \cup \{x\}$ and $K_{\mathbb{X}\mathbb{X}}$ denote the kernel matrix associated with $\mathbb{X}$. Thanks to the optimal recovery property (Wendland, 2004, Theorem 13.2), $\begin{bmatrix} c^\top & c_x \end{bmatrix} K_{\mathbb{X}\mathbb{X}} \begin{bmatrix} c^\top & c_x \end{bmatrix}^\top \leq \|f\|_{\mathcal{H}}^2$, which in turn is strictly smaller than Γ^2 by assumption. Also,

$$\begin{aligned} \|\Lambda c - \mathsf{y}\|_\infty &= \|\Lambda f_X - \mathsf{y}\|_\infty \\ &= \left\| \begin{bmatrix} \delta_{1,1} & \dots & \delta_{2,1} & \dots \end{bmatrix}^\top \right\|_\infty < \bar{\delta}. \end{aligned}$$

Therefore, the ground-truth values constitute a feasible solution that lies in the interior of the primal problem feasible set. As a result, Slater's condition is met and, since the primal is convex, there is no duality gap. □

An alternating optimization procedure

Solving the dual problem to any accuracy leads to an overbound on $\mathrm{C}(x)$ thanks to duality. In other words, any feasible suboptimal solution of (2.48) establishes a conservative uncertainty estimate. This motivates the study of light methods that could trade off computational time and accuracy. In what follows, we describe a block coordinate minimization scheme to tackle the problem, which is later shown to yield reasonable results after only a small number of iterations.

Whenever λ is fixed to a particular positive value $\lambda^* > 0$, the problem (2.48) simplifies to an unconstrained quadratic program (QP) in ν of the form $\min_{\nu \in \mathbb{R}^{\tilde{N}}} \tilde{C}_x(\lambda^*, \nu)$. On the other hand, if ν is fixed to $\nu^* \in \mathbb{R}^{\tilde{N}}$, the dual objective takes the form

$$\min_{\lambda \in \mathbb{R}_{>0}} \tilde{C}_x(\lambda, \nu^*) = \min_{\lambda \in \mathbb{R}_{>0}} \frac{c_1}{\lambda} + c_2 \lambda + c_3 \tag{2.49}$$

with the constants

$$\begin{aligned} c_1 &= \frac{1}{4} \begin{bmatrix} \Lambda^\top \nu^* \\ -1 \end{bmatrix}^\top K_{\mathbb{X}\mathbb{X}} \begin{bmatrix} \Lambda^\top \nu^* \\ -1 \end{bmatrix}, \\ c_2 &= \Gamma^2, \quad c_3 = \mathsf{y}^\top \nu^* + \bar{\delta} \|\nu^*\|_1 \end{aligned} \tag{2.50}$$

and $K_{\mathbb{X}\mathbb{X}} \succeq 0$. We have

$$\frac{\partial \tilde{C}_x(\lambda, \nu^*)}{\partial \lambda} = \frac{-c_1}{\lambda^2} + c_2,$$

which gives the candidate solution $\lambda^* = \sqrt{c_1/c_2}$ for (2.49). We have $c_2 > 0$ and $c_1 \geq 0$ for any $x \in \Omega$, and $c_1 > 0$ for any $x \in \Omega \setminus X$. Therefore, λ^* is a feasible solution to (2.49) and thus (λ^*, ν^*) a feasible solution to the dual problem. Furthermore, λ^* is indeed a minimizer of (2.49) for $x \in \Omega \setminus X$ since its curvature is positive, i.e., $\frac{\partial^2 \tilde{C}_x(\lambda^*, \nu^*)}{\partial \lambda^2} > 0$. In closed-form, λ^* takes the following form.

$$\lambda^* = \frac{\sqrt{\nu^{*\top} \Lambda K_{XX} \Lambda^\top \nu^* - 2(\Lambda K_{Xx})^\top \nu^* + k(x,x)}}{2\Gamma} \tag{2.51}$$

Note that $\lambda^* = 0$ is only possible if $\begin{bmatrix} \Lambda^\top \nu^* \\ -1 \end{bmatrix}$ is in the nullspace of matrix $K_{\mathbb{X}\mathbb{X}}$, which is only possible if $x \in X$. In this case, after fixing $\lambda^* = 0$, the problem to solve for ν reduces to

$$\min_{\nu} \ \mathrm{y}^\top \nu + \bar{\delta} \left\| \nu \right\|_1 \ \text{ s.t. } \ \begin{bmatrix} \Lambda^\top \nu \\ -1 \end{bmatrix} \in \mathrm{Null}(K_{\mathbb{X}\mathbb{X}}) \, .$$

We formulate the alternating optimization algorithm for a maximum number of iterations L and a termination threshold ϵ in the following way.

Algorithm 1: Alternating minimization

Result: Upper bound $\tilde{C}(x)$ of the ground-truth at point x
Input: x, λ_0, L, ϵ
$\lambda_0^* = \lambda_0$
$k = 0$
do
  $\nu^* = \arg\min_{\nu \in \mathbb{R}^{\tilde{N}}} \tilde{C}_x(\lambda_k^*, \nu)$
  $\lambda_{k+1}^* = \frac{\sqrt{\nu^{*\top} \Lambda K_{XX} \Lambda^\top \nu^* - 2(\Lambda K_{Xx})^\top \nu^* + k(x,x)}}{2\Gamma}$
  $k = k + 1$
while $k < L$ *and* $|\lambda_k^* - \lambda_{k-1}^*| > \epsilon$;
$\tilde{C}(x) = \tilde{C}_x(\lambda_k^*, \nu^*)$

Remark 6 (Numerical properties)**.** *Recall the convex dual objective function* (2.48)*. Since the non-differentiable term* $\|\nu\|_1$ *is separable and the remainder of the objective is differentiable, a tuple* (ν^*, λ^*) *that simultaneously minimizes both sub-problems also necessarily minimizes the whole objective* (2.48)*. For non-asymptotic sublinear convergence rates of alternating minimization algorithms applied to convex programs, the reader is referred to the work (Beck, 2015).*

2.6 Numerical results

In this section, we demonstrate the performance of the different bounds in varying examples and conditions, starting out with the closed-form ones, before also reviewing the optimal ones.

2.6.1 1D comparison of closed-form bounds

In the first example, we compare the closed-form bounds from Theorem 2, centered around a KRR and SVR model, to the GP bound with bounded noise, proposed in (Hashimoto et al., 2022). These bounds take the form

$$|s^\diamond(x) - f(x)| \leq \sigma(x)\sqrt{\Gamma^2 - y^\top (K_{XX} + \bar{\delta}^2 I)^{-1} y + N} \tag{2.52}$$

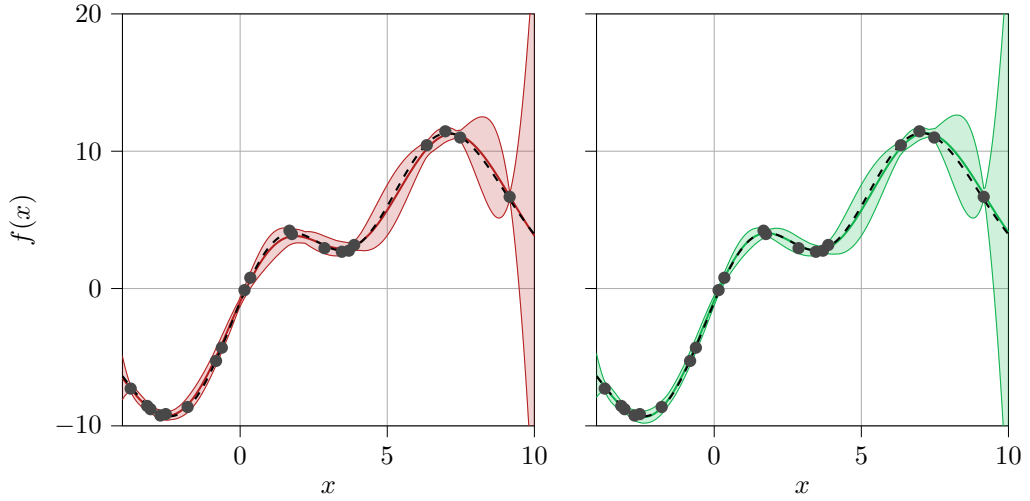
where $\sigma^2(x) = k(x,x) - K_{xX}(K_{XX} + \bar{\delta}^2 I)^{-1} K_{Xx}$ and $s^\diamond(x) = K_{xX}(K_{XX} + \bar{\delta}^2 I)^{-1} y$ is their approximation model.

We consider a function $f : \mathbb{R} \to \mathbb{R}$ composed of partial kernel evaluations. Specifically, we consider the squared exponential kernel (2.3)

$$k(x_1, x_2) = \exp\left(-\frac{\|x_1 - x_2\|^2}{2\ell^2}\right) \tag{2.53}$$

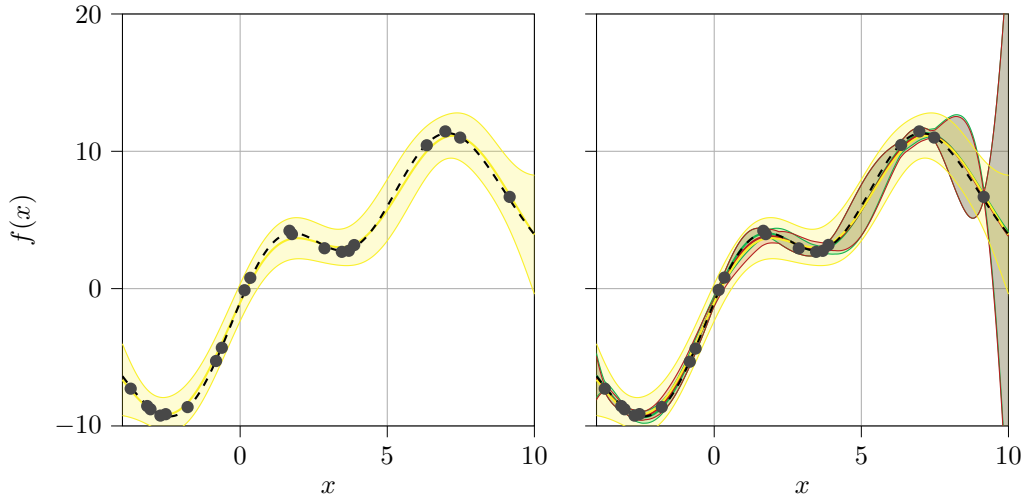
with a lengthscale $\ell = 2.5$ and the resulting $f(x) = -k(x,0) + 4\,k(x,2) + 3\,k(x,3) + 6\,k(x,5)$. The resulting RKHS norm of f is $\|f\|_{\mathcal{H}} \approx 21.181$ and we employ an overapproximation of that to get $\Gamma = 23.299 \approx 1.1\|f\|_{\mathcal{H}}$. Samples are drawn from f with a uniformly bounded noise and $\bar{\delta} = 0.2$, considering the domain $\Omega = \{x \in \mathbb{R} \mid -4 \leq x \leq 10\}$. Two cases are examined: having a dataset with samples at random locations and having a dataset with evenly spaced samples. For both cases, we have $N = 20$ samples, and the results are displayed in Figure 2.4c for the random sample locations and in Figure 2.5 for the evenly spaced sample locations.

The nominal prediction models all yield very similar results, but a clear difference in bound behavior is observable: For random sample locations, both KRR and SVR bounds present a tight fit in areas with many samples, but a degradation in lower sampled regions, especially at the extremes of the considered domain, is observable (see Figure 2.4a and Figure 2.4b). This is mostly due to the noise-bound term in (2.25). The GP bounds exhibit a more uniform behavior, but are less tight almost everywhere (see Figure 2.4c). In this particular example, this was due to two reasons. Firstly, recall (2.10), and notice that the power function $P_X(x)$ will always evaluate to a number smaller than $\sigma(x)$ in (2.52) due to $(K_{XX} + \bar{\delta}^2 I) \succ K_{XX}$. Secondly, (2.52) has a direct dependence on the number of samples N, which is not present in the other bounds.



(a) KRR model (solid line) as an approximation of the ground truth function (dotted line) with error approximation (shaded area)

(b) SVR model (solid line) as an approximation of the ground truth function (dotted line) with error approximation (shaded area)



(c) GP model (solid line) as an approximation of the ground truth function (dotted line) with error approximation (shaded area)

(d) The three different approximation models (solid lines) of the ground truth function (dotted line) with error approximations (shaded areas)

Figure 2.4 – Comparison of KRR (red), SVR (green), and GP (yellow) with corresponding error approximations for the ground truth function (black dotted line) and noisy samples (grey dots) at random input locations.

Considering the case with equally spaced sample locations, the difference between the proposed bounds and the GP bounds is even more pronounced. Due to the included samples at the extremes of the domain, no degradation is observable in Figures 2.5a and 2.5b. Therefore, the GP bounds encompass the other bounds across the whole domain, emphasizing the benefit of regular-spaced samples for our proposed bounds.

2.6.2 Closed-form bounds in two dimensions

Next, consider the dynamics of the Tinkerbell chaotic system first coordinate

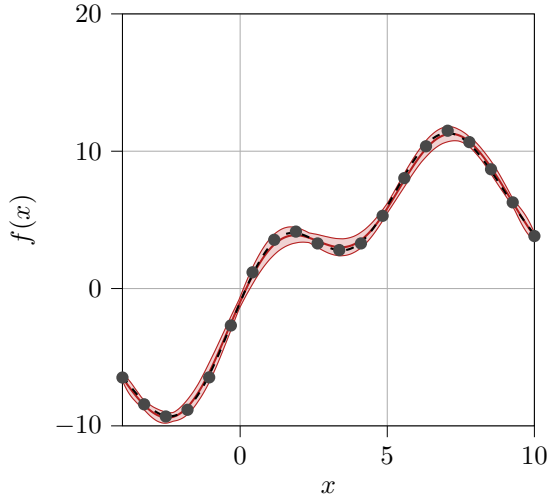
$$f(x) = x_1^2 - x_2^2 + 0.8\,x_1 - 0.6\,x_2 \tag{2.54}$$

on the domain $\Omega = \{x \in \mathbb{R}^2 \,|\, [-5 \;\; -5]^\top \leq x \leq [5 \; 5]^\top\}$. With a slight abuse of notation, x_1 and x_2 denote the first and second components of x. Two training datasets of $N = 625$ points were collected: one forming a perfect grid across the domain, and one drawn randomly from a uniform distribution. Bounded measurement noise with $\bar{\delta} = 0.5$ was considered in both cases. The squared-exponential kernel was chosen, and the lengthscale $\ell = 1.62$ was determined by maximizing the log-likelihood objective for a sensible variance value. Γ was estimated by collecting noiseless evaluations f_X of the ground truth and determining the norm of the associated interpolant; a final value of $\Gamma = 175$ was adopted after the use of a safety factor, as explained in Appendix 2.8.5. A KRR model with $\lambda = 1 \times 10^{-3}$and SVR model were used to reconstruct f. The ground truth values and the error bounds of the approximation models are shown in Figures 2.6 and 2.7.

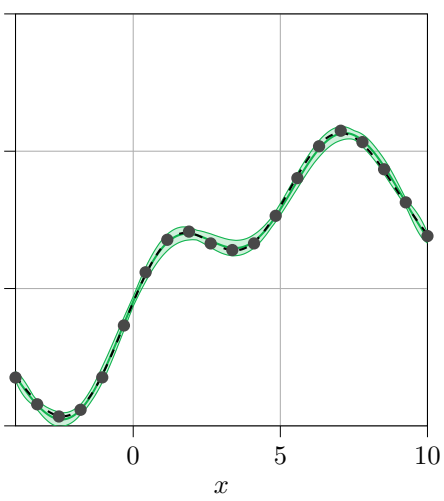
The bounds in both Figures 2.6a and 2.7a show a tight and uniform behavior with evenly spaced samples, whereas they behaved badly under the scattered ones, showing high peaks, especially at the border of the domain in Figures 2.6b and 2.7b. Notice nevertheless that the center part of the error bounds remain relatively tight. Finally, the random dataset was augmented with 32 points collected from the domain boundary, and the results are presented in Figures 2.6c and 2.7c. Incorporating these extra points was enough to significantly dampen the bounds increase, not only at the borders but also in internal regions. The average bound size, i.e. the distance from the nominal predictions to the upper and lower bounds, is shown in Table 2.1. Note that the SVR bounds are tighter on average for the displayed cases, which is due to the choice of the regularization parameter λ in the KRR. Decreasing the regularization will lead to shrinking bounds, as mentioned in the discussion of Theorem 2.

Model	Grid	Random	Random + boundary
KRR	2.37	5.29	3.40
SVR	2.11	4.88	3.07

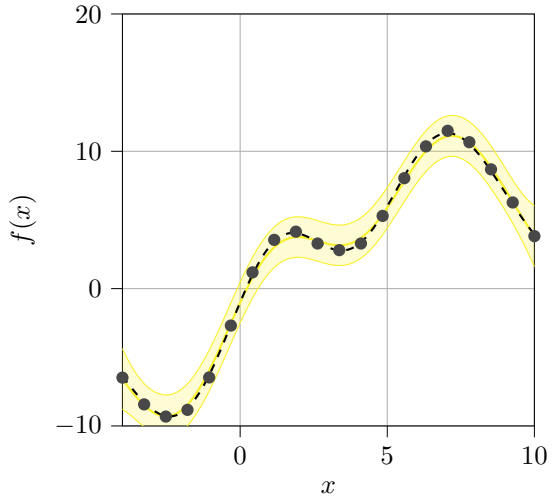
Table 2.1 – Average distance of the nominal predictions to the error bounds under different sampling strategies.



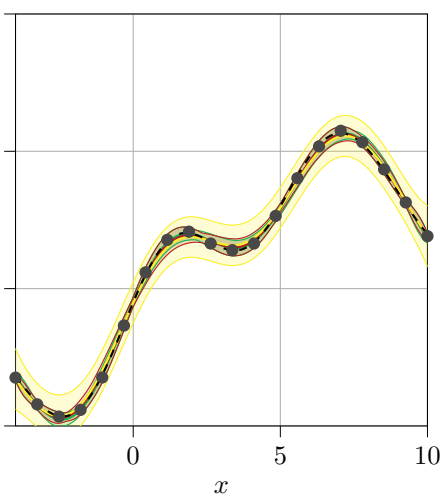
(a) KRR model (solid line) as an approximation of the ground truth function (dotted line) with error approximation (shaded area)



(b) SVR model (solid line) as an approximation of the ground truth function (dotted line) with error approximation (shaded area)

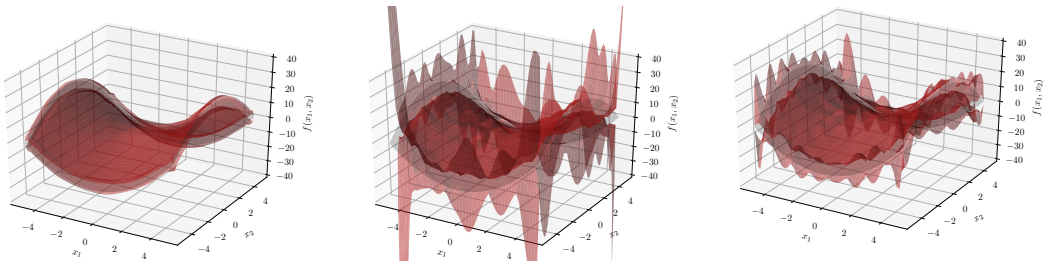


(c) GP model (solid line) as an approximation of the ground truth function (dotted line) with error approximation (shaded area)



(d) The three different approximation models (solid lines) of the ground truth function (dotted line) with error approximations (shaded areas)

Figure 2.5 – Comparison of KRR (red), SVR (green), and GP (yellow) with corresponding error approximations for the ground truth function (black dotted line) and noisy samples (grey dots) at equidistant input locations.

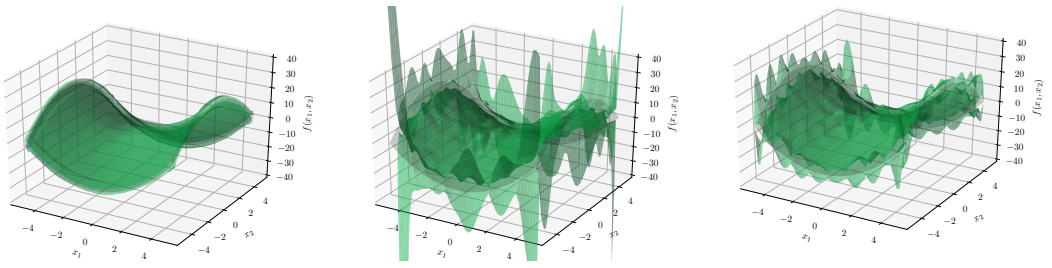


(a) 625 samples from a regular grid

(b) 625 samples drawn uniformly at random

(c) 625 random samples, plus 32 samples from the domain boundary

Figure 2.6 – Ground truth function (gray shaded area) with closed-form error bounds (red shaded area) around the KRR model under different sampling strategies



(a) 625 samples from a regular grid

(b) 625 samples drawn uniformly at random

(c) 625 random samples, plus 32 samples from the domain boundary

Figure 2.7 – Ground truth function (gray shaded area) with closed-form error bounds (green shaded area) around the SVR model under different sampling strategies

2.6.3 A comparison with the optimal bounds

As a third example, consider the function below, which represents the first component update map of a Hénon chaotic attractor with an additional sinusoidal forcing term

$$f(z_1, z_2) = 1 - az_1^2 + z_2 + b\sin(c\, z_2). \tag{2.55}$$

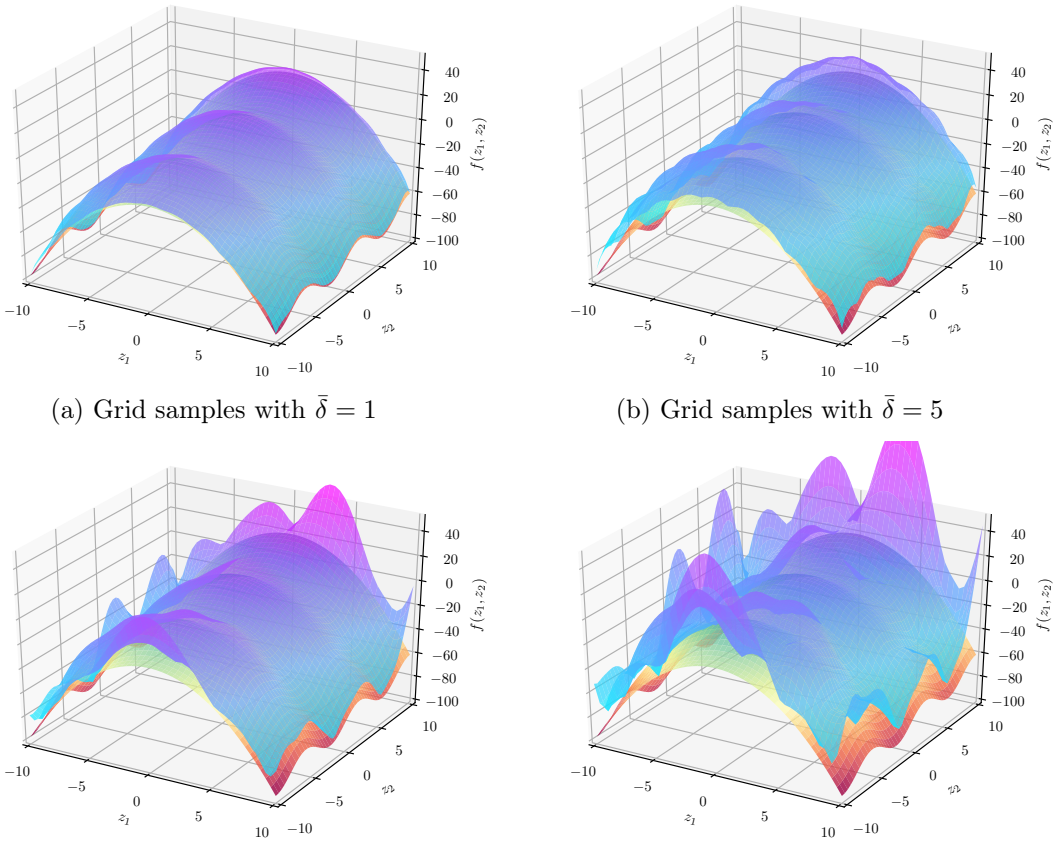
Here, the parameters are $a = 0.8$, $b = 8$ and $c = 0.8$, and its domain is the box $\Omega = \begin{bmatrix} -10 & 10 \end{bmatrix} \times \begin{bmatrix} -10 & 10 \end{bmatrix}$. A squared-exponential kernel with lengthscale $l = 5$ was chosen for our experiments, which was empirically estimated by gridding the search space and performing posterior validation. Γ was obtained through the procedure described in Appendix 2.8.5 with a final value of $\Gamma = 1200$. $N = 100$ samples were collected using two strategies: inputs lying in an equidistant grid, and inputs being drawn randomly from a uniform distribution. Noise was sampled uniformly throughout the tests with $\bar{\delta} = 1$ and $\bar{\delta} = 5$ to examine the sensitivity of the bounds to noise.

Optimal bounds in 2D

The obtained optimal upper bound $\mathrm{C}(z)$ is displayed in Figure 2.8 along with the ground truth function f. Consider the scenarios where $\bar{\delta} = 1$ (Figures 2.8a and 2.8c). Whereas the $\mathrm{C}(z)$ surface is overall tight for the grid-based dataset, with an average distance of 3.01 to the latent function, randomized data yielded a less regular bound with an average distance of 8.02. These numbers were increased respectively to 9.57 and 18.97 when the noise levels were risen to $\bar{\delta} = 5$ (Figures 2.8b and 2.8d). The plots illustrate the same behavior as observed in the previous examples for the closed-form bounds, where relying on completely randomized input locations degrades the tightness of the bounds, especially the borders of $\mathrm{C}(z)$. An equidistant grid of points is highly favorable since it not only fills the domain well but also ensures a minimum separation distance so that no two inputs are too close to cause numerical problems when handling the kernel matrix K_{XX}.

Comparing a cross-section

To compare the closed form bounds for a KRR model to the optimal ones and the ones obtained by Algorithm 1, $f(z_1, z_2)$ was sliced at $z_1 = -5.9$ and the entire envelope $\mathrm{B}(z) \leq z \leq \mathrm{C}(z)$ was computed. The two previous datasets with $\bar{\delta} = 1$ were used and the obtained results are displayed in Figure 2.9, always showing the optimal bounds in comparison with another bound type. The bounds obtained by Algorithm 1 were computed for 6 and 10 alternating steps, showing the difference of the obtained results. As can be seen from the plots, the optimal approach yielded tighter uncertainty intervals than the alternatives, although the bounds computed with Algorithm 1 and 10 alternating steps are almost indistinguishable. The average distance of upper and lower bounds is shown in Table 2.2. Additional results for $\bar{\delta} = 5$ are shown in Appendix 2.8.6.



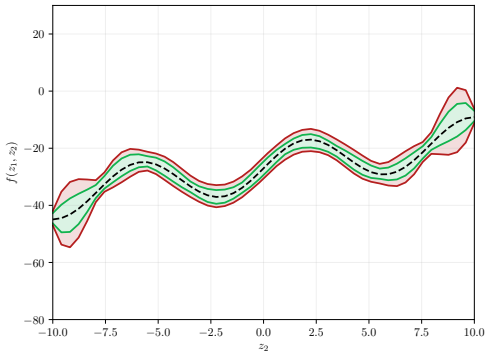
(a) Grid samples with $\bar{\delta} = 1$

(b) Grid samples with $\bar{\delta} = 5$

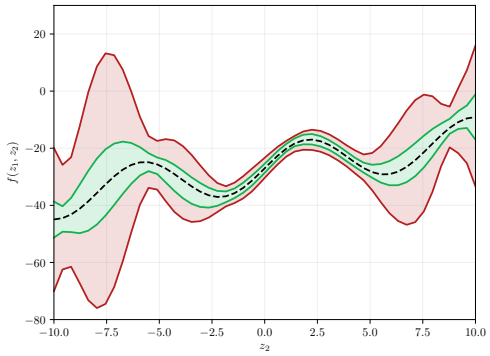
(c) Random samples with $\bar{\delta} = 1$

(d) Random samples with $\bar{\delta} = 5$

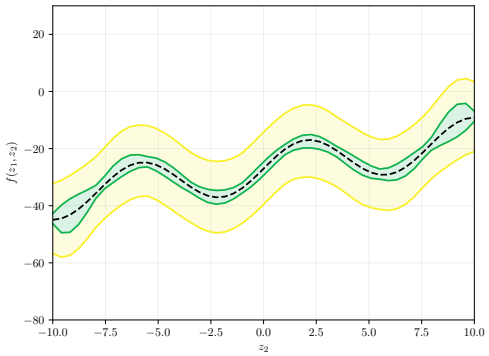
Figure 2.8 – Ground truth function and optimal upper bound under grid or random sampling with changing noise bound.



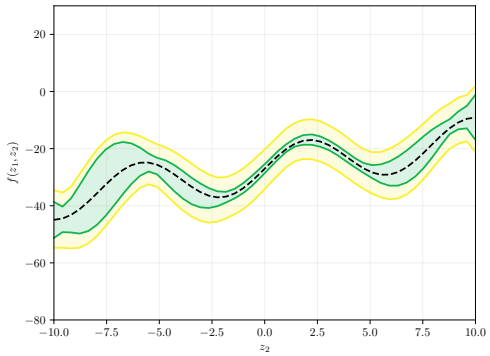
(a) Optimal bounds (green) vs KRR bounds (red) under grid sampling



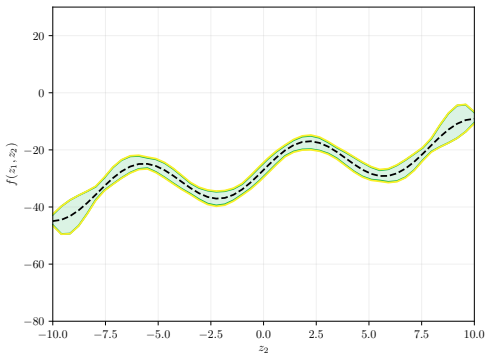
(b) Optimal bounds (green) vs KRR bounds (red) under random sampling



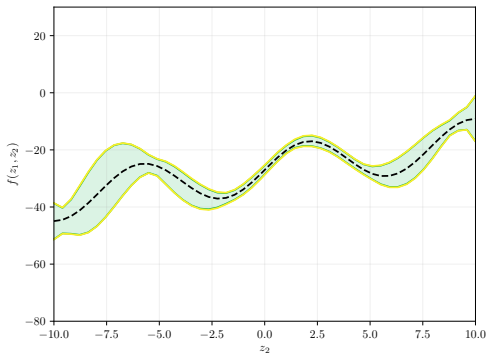
(c) Optimal bounds (green) vs approximate optimal bounds with 6 alternating steps (yellow) under grid sampling



(d) Optimal bounds (green) vs approximate optimal bounds with 6 alternating steps (yellow) under random sampling



(e) Optimal bounds (green) vs approximate optimal bounds with 10 alternating steps (yellow) under grid sampling



(f) Optimal bounds (green) vs approximate optimal bounds with 10 alternating steps (yellow) under random sampling

Figure 2.9 – Comparison of the optimal bounds to the KRR bounds, and approximate optimal bounds with 6 or 10 alternating steps on a cross-section of f, for grid sampling or random sampling, with $\bar{\delta} = 1$.

Sampling strategy	Optimal bounds	KRR bounds	6 step approx.	10 step approx.
Grid sampling	5.36	9.55	25.34	5.78
Random sampling	8.45	26.84	17.85	8.70

Table 2.2 – Average distance of the upper and lower bounds under different sampling strategies for $\bar{\delta} = 1$.

Sensitivity to overapproximations

Next, we consider the Gaussian process bounds proposed in (Berkenkamp et al., 2017, Lemma 3) (see also the closely related works (Chowdhury & Gopalan, 2017; Koller et al., 2018)) and analyze how they compare to the optimal and KRR based ones. Overloading notation for the sake of clarity, these bounds have the form

$$|\mu(z) - f(z)| \leq \beta\, \sigma(z) \tag{2.56a}$$

$$\text{with} \quad \beta = \Gamma + 4\lambda\sqrt{\gamma + 1 + \ln(1/\delta)}, \tag{2.56b}$$

where $\mu(z)$ is the GP mean, $\sigma(z)$ is its standard deviation, λ is the strength of the sub-Gaussian noise, γ is the maximum information capacity for a fixed number of samples, and $1 - \delta$ is the confidence of the inequality.

In order to compare the GP uncertainty bounds (2.56) to their deterministic counterparts, the following approach was adopted. First, a lower bound for the maximum information gain γ was used since the problem of exactly computing such a quantity is in general NP-hard (Srinivas et al., 2012). Note how this decision favors the GP bounds by shrinking them. The chosen lower bound was the information gain of our inputs X, which in our zero-mean Gaussian noise setting with variance λ^2 is $\frac{1}{2}\ln(\det(I + \lambda^{-2}K_{XX}))$ (Srinivas et al., 2012). As for the noise realizations, we proceeded as follows. Starting from our hard noise limit $\bar{\delta}$, we considered a zero-mean Gaussian distribution with variance such that its samples would lie in the $[-\bar{\delta}, \bar{\delta}]$ band with confidence 0.99, i.e., a standard deviation of $\lambda = \frac{\bar{\delta}}{2.58}$. The noise was then drawn from the normal distribution and clipped to the interval $[-\bar{\delta}, \bar{\delta}]$ to fulfill Assumption 3. Finally, the probabilistic inequality (2.56) was evaluated for a final confidence of 99%.

The data, $N = 100$ samples, corrupted by the same noise realizations were used throughout the tests for all methods. Two parameters were then varied to understand how sensitive each method is to them: the RKHS norm estimate Γ and the noise bound $\bar{\delta}$, which were multiplied by a factor of 1, 1.5, and 2. The results for $\bar{\delta} = 1$ are summarized in Table 2.3 and the results for $\bar{\delta} = 5$ are shown in Appendix 2.8.7. The outcomes in all 18 different scenarios were unanimous in ranking the optimal bounds as the tightest method, followed by the KRR ones, and finally the GP approach. Indeed, the GP bounds always yielded average widths at least one order of magnitude greater than the optimal deterministic

ones. We attribute this difference, especially to the direct product of Γ and $\sigma(z)$ in (2.56), which causes them to be particularly sensitive to norm over-approximations. This effect is dampened in (2.25) due to the interaction with $\tilde{\Delta}$ (see the derivation in the proof of Theorem 2).

	Γ	1200			1800			2400		
	$\bar{\delta}$	1	1.5	2	1	1.5	2	1	1.5	2
Grid	OPT	6.21	8.35	10.34	7.45	9.75	11.90	8.50	10.94	13.20
	KRR	11.07	15.60	20.13	11.70	16.23	20.76	12.36	16.89	21.42
	GP	604.51	706.13	786.51	904.61	1055.89	1175.32	1204.71	1405.65	1564.12
Rand	OPT	14.62	19.02	22.89	18.05	23.08	27.51	20.85	26.39	31.26
	KRR	64.78	93.99	123.20	65.91	95.12	124.33	67.07	96.28	125.49
	GP	643.20	743.44	822.24	962.51	1111.67	1228.70	1281.82	1479.90	1635.17

Table 2.3 – Average distance between the upper and lower bounds for the optimal (OPT) and KRR deterministic cases, and the GP. Moderate noise level (true $\bar{\delta} = 1$), using factors of 1, 1.5, and 2 to augment $\bar{\delta}$ and Γ.

2.6.4 Robust optimization

Next, we use the ground truth function of the previous example (2.55) as an unknown constraint for a static problem (data-driven optimization with unknown constraints is typical in the field of real-time optimization (Chachuat et al., 2009)). Consider the following formulation

$$\min_{z \in \mathbb{R}^2} \quad (z_1 - 1)^2 + (z_2 - 5)^2 \tag{2.57a}$$

$$\text{subj. to} \quad f(z) \leq -10, \tag{2.57b}$$

where the function $f(z)$ that maps the decision variables to the constraint is not explicitly known but can be measured. Samples were used to establish an upper bound $\mathrm{C}(z)$ for $f(z)$, hence providing an inner approximation for the real feasible set. We considered the cases of having 64, 81, and 100 evaluations of $f(z)$ affected by noise with $\bar{\delta} = 1$ and, once more, the data were collected by means of a uniform random distribution and an equidistant grid. In the approximate optimization problems, the original constraint (2.57b) was replaced by $C(z) \leq -10$. Optimizers $z^\star$ were computed by gridding the domain, and the results along with the estimated feasible sets (shaded areas) are shown in Figure 2.10. Notice how in some instances the set of feasible decisions is not connected. Thanks to Proposition 2, the addition of new data points can only relax the approximate formulation, hence reducing the found minimum. Indeed, the average obtained costs over 10 runs for the approximate problems were 13.21, 11.36, and 10.96, respectively with 64, 81, and 100 samples taken randomly. Depending on the sampled dataset, the costs ranged from 7.53 to 29.64, 7.45 to 22.65, and 6.55 to 30.80 for the datasets of different sizes, and only one example for each dataset size is shown in Figure 2.10. When employing a grid,

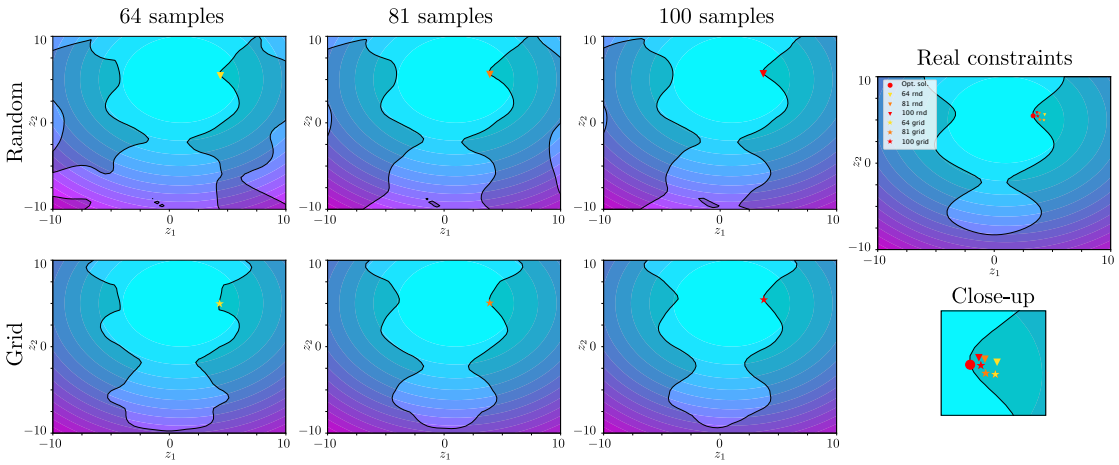


Figure 2.10 – Solutions and feasible sets (shaded areas) for problem (2.57) with 64, 81 and 100 samples of $f^\star(z)$. Top row: samples drawn uniformly. Bottom row: samples on an equidistant grid. The true feasible set and optimal solution are shown on the right.

the figures were 10.67, 8.48, and 7.67. The solution of the real problem, i.e., the one with the ground truth constraint, is 5.69.

2.7 Conclusion

In this chapter, we presented different types of error bounds for the function approximation problem in RKHSs. Starting from the general formalism of kernels, their properties and corresponding function spaces, we reviewed different methods to approximate a ground truth function from a set of noisy samples. The type of noise considered here was bounded noise under the assumption of having an upper bound on the maximum noise realization. Informed by the process of interpolating noise-free samples, the methods of KRR and SVR were reviewed. The commonality in obtaining all the different approximation models was in the minimization of a complexity measure in the form of the RKHS norm.

Assuming the availability of an upper bound on this complexity measure for the ground truth function allowed for the derivation of deterministic closed-form error bounds. These bounds assume a worst-case specification around a given approximation model, and while being useful and favorable in comparison with other alternatives, the conducted experiments showed a potentially high level of conservativeness, especially under random sampling schemes. To reduce the conservativeness, we introduced two infinite-dimensional variational problems for a model-free, pointwise uncertainty estimation, and showed the equivalence to finite-dimensional problems. Furthermore, we showed that these improved bounds enjoy favorable properties like strictly decreasing uncertainty on the whole domain after adding new data samples. Through a dualization procedure, we derived a bounding problem where the optimal solution corresponds to the optimal bounds and any feasible

solution constitutes an overbound compared to the optimal ones. This insight was used to inform an alternating optimization procedure to obtain approximations of the optimal bounds with increasing accuracy for an increasing number of alternating steps. The behavior, properties, and tightness of the different bounds were evaluated in multiple numerical examples, and compared to popular alternatives from the community, showing a better performance in all the comparisons. Lastly, the usage of the optimal bounds for approximating constraints in an optimization problem was demonstrated.

Various extensions to the proposed bounds are envisioned, and while being of theoretical interest, mostly address practical issues. The first set of extensions deals with the assumptions needed to apply the approach: The availability of upper bounds to the RKHS norm and noise level are needed to achieve deterministic guarantees as in Theorem 2 and (2.37). However a principled way of estimating the former is still missing in the literature, while the latter has been studied more extensively. Also a joint estimation of both quantities on the basis of available data is of interest for the practical application. Furthermore, the developed error bounds assume knowledge of the true underlying kernel of the ground truth function. How the misspecification of the kernel function influences the error bounds is a topic of future research.

The second set of extensions deals with the incorporation of the bounds in optimization or MPC schemes. One predictive control scheme relying on the closed-form bounds was presented in (Maddalena et al., 2021), however the approach relies on multi-step models, which becomes increasingly prohibitive with longer prediction horizons. To propagate uncertainty, incorporating input uncertainty into the kernel models is a necessary step. The dual bounds present a promising formulation to incorporate error estimation into constraints directly, and assume the dual variables as additional optimization variables directly. Solution approaches and properties of the resulting problem require additional investigation. Lastly, the computational needs that come with large datasets can become a bottleneck in the applicability of the approach. Achieving tight bounds with decreased computational complexity, e.g., through subsampling techniques, is a promising direction of future explorations.

2.8 Appendix

2.8.1 Proof of Lemma 3

In the following, we only prove $f_X^{\max}(x) \geq f_Z^{\max}(x)$, the inequality for $f^{\min}$ follows from the same arguments, which proves the interval containment. To get the interval at a point $x \in \Omega$, we consider the sets $X, \bar{X} = X \cup \{x\}, Z = X \cup \{z\}$ and $W = Z \cup \{x\}$. Additionally, we denote the interpolant of f_X by s_X and follow this convention for the other sets. We observe the following norm identities, derived as in the proof of Lemma 2

$$\begin{aligned}
\|s_W\|_{\mathcal{H}}^2 &= \|s_{\bar{X}}\|_{\mathcal{H}}^2 + P_{\bar{X}}^{-2}(z)(s_{\bar{X}}(z) - f_z)^2 && (2.58)\\
&= \|s_X\|_{\mathcal{H}}^2 + P_X^{-2}(x)(s_X(x) - f_x)^2 + P_{\bar{X}}^{-2}(z)(s_{\bar{X}}(z) - f_z)^2 && (2.59)\\
&= \|s_Z\|_{\mathcal{H}}^2 + P_Z^{-2}(x)(s_Z(x) - f_x)^2 && (2.60)\\
&\leq \Gamma^2 && (2.61)
\end{aligned}$$

This allows us to write $f_Z^{\max}(x)$ in two different ways

$$\begin{aligned}
f_Z^{\max}(x) &= s_Z(x) + P_Z(x)\sqrt{\Gamma^2 - \|s_Z\|_{\mathcal{H}}^2} && (2.62)\\
&= s_X(x) + P_X(x)\sqrt{\Gamma^2 - \|s_X\|_{\mathcal{H}}^2 - P_{\bar{X}}^2(s_{\bar{X}}(z) - f_z)^2} && (2.63)
\end{aligned}$$

From (2.63), we observe

$$f_Z^{\max}(x) \leq s_X(x) + P_X(x)\sqrt{\Gamma^2 - \|s_X\|_{\mathcal{H}}^2} = f_X^{\max}(x) \tag{2.64}$$

using the positivity of the power function. □

2.8.2 Data-selection matrix

Recall that $n_1, n_2, \ldots, n_N$ are the number of outputs available at the input locations $x_1, x_2, \ldots, x_N$. Λ has size $(\sum_i n_i) \times N$ and is defined as

$$\Lambda := \begin{bmatrix}
\mathbf{1}_{n_1} & \mathbf{0}_{n_1} & \mathbf{0}_{n_1} & \cdots & \mathbf{0}_{n_1} \\
\mathbf{0}_{n_2} & \mathbf{1}_{n_2} & \mathbf{0}_{n_2} & \cdots & \mathbf{0}_{n_2} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\mathbf{0}_{n_N} & \mathbf{0}_{n_N} & \mathbf{0}_{n_N} & \cdots & \mathbf{1}_{n_N}
\end{bmatrix} \tag{2.65}$$

where $\mathbf{1}_{n_i}$ and $\mathbf{0}_{n_i}$ are respectively column vectors of ones and zeros of size n_i. If only a single output is available at every input, Λ is an identity matrix.

2.8.3 A block matrix identity

Let $A \in \mathbb{R}^{d\times d}$ be invertible, $B \in \mathbb{R}^d$ and $c \in \mathbb{R}$. The following identity holds

$$\begin{bmatrix} A & B \\ B^\top & c \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + \frac{1}{d}A^{-1}BB^\top A^{-1} & -\frac{1}{d}A^{-1}B \\ -\frac{1}{d}B^\top A^{-1} & \frac{1}{d} \end{bmatrix} \tag{2.66}$$

where $d = c - B^\top A^{-1}B$.

This reformulation yields the following for $y \in \mathbb{R}^d, u \in \mathbb{R}$:

$$\begin{bmatrix} y \\ u \end{bmatrix}^\top \begin{bmatrix} A & B \\ B^\top & c \end{bmatrix}^{-1} \begin{bmatrix} y \\ u \end{bmatrix} = y^\top A^{-1}y + \frac{1}{d}(y^\top A^{-1}B - u)^2 \tag{2.67}$$

2.8.4 Dualization

Consider the case $x \notin X$. Let $z := \begin{bmatrix} c^\top & c_x \end{bmatrix}^\top$, $a := \begin{bmatrix} \mathbf{0}^\top & 1 \end{bmatrix}^\top$, $A := \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}$. The Lagrangian of (2.38) is

$$\begin{aligned} \mathcal{L}(z,\lambda,\beta,\gamma) =& a^\top z - \lambda(z^\top K_{\mathbb{X}\mathbb{X}}^{-1} z - I^2) \\ & - \beta^\top(\Lambda A z - \mathsf{y} - \bar{\delta}\mathbf{1}) - \gamma^\top(\mathsf{y} - \Lambda A z - \bar{\delta}\mathbf{1}) \end{aligned} \tag{2.68}$$

where $K_{\mathbb{X}\mathbb{X}}$ denotes the kernel matrix evaluated at $X \cup \{x\}$. Suppose $\lambda > 0$. Computing $\nabla_z \mathcal{L}(z^\star) = 0$ leads to

$$z^\star = -\frac{1}{2\lambda} K_{\mathbb{X}\mathbb{X}} \left(A^\top \Lambda^\top (\beta - \gamma) - a \right).$$

Defining the auxiliary variable $\nu = \beta - \gamma$, and substituting $z^\star$ into (2.68) gives the dual objective

$$\begin{aligned} g(\lambda,\nu) &= \frac{1}{4\lambda}\nu^\top \Lambda A K_{\mathbb{X}\mathbb{X}} A^\top \Lambda^\top \nu + \left(\mathsf{y} - \frac{1}{2\lambda}\Lambda A K_{\mathbb{X}\mathbb{X}} a\right)^\top \nu \\ &\quad + \bar{\delta}\|\nu\|_1 + \frac{1}{4\lambda} a^\top K_{\mathbb{X}\mathbb{X}} a + \lambda I^2 \qquad (2.69) \\ &= \frac{1}{4\lambda}\nu^\top \Lambda K_{XX} \Lambda^\top \nu + \left(\mathsf{y} - \frac{1}{2\lambda}\Lambda K_{Xx}\right)^\top \nu \\ &\quad + \bar{\delta}\|\nu\|_1 + \frac{1}{4\lambda} k(x,x) + \lambda I^2 \qquad (2.70) \end{aligned}$$

where in the second equality the matrix $K_{\mathbb{X}\mathbb{X}}$ was expanded and the resulting terms were reorganized. Since $\beta, \gamma \in \mathbb{R}^{\tilde{d}}_{\geq 0}$ and $\nu = \beta - \gamma$, ν is unconstrained.

Now if $\lambda = 0$, the Lagrangian (2.68) simplifies to $\mathcal{L}(z,\nu) = (a - A^\top \Lambda^\top \nu)^\top z + \nu^\top \mathsf{y} + \bar{\delta}\|\nu\|_1$,

which is linear in z. Its supremum w.r.t. z is only finite if $a = A^\top \Lambda^\top \nu$. Recalling the definitions of a, A and Λ, one can see that $\nexists \nu$ that could satisfy the latter condition. Therefore, $\lambda = 0 \implies \sup_z \mathcal{L}(z, \lambda, \nu) = +\infty$, meaning that the dual problem is infeasible. As a conclusion, the Lagrangian dual of (2.38) is precisely (2.48).

Next, consider the case $x \in X$, $x = x_i$. The objective of (2.39) can be written as $a^\top c$ with $a_i = 1$ and $a_n = 0, n \neq i$. When deriving its Lagrangian, one obtains again (2.68) with the simplifications: $z \leftarrow c$, $K_{\mathbb{X}\mathbb{X}} \leftarrow K_{XX}$ and $A \leftarrow \mathbf{I}$. We proceed by analyzing the two scenarios for λ as before. If $\lambda > 0$, the previous derivations apply, leading to the same quadratic-over-linear objective (2.70). However, if $\lambda = 0$, the Lagrangian becomes $\mathcal{L}(z, \nu) = (a - \Lambda^\top \nu)^\top z + \nu^\top \mathsf{y} + \bar{\delta}\|\nu\|_1$, whose supremum w.r.t. z is only finite if $a = \Lambda^\top \nu$. In contrast with the previous paragraph, this condition now can be satisfied. It is equivalent to $\nu_{i,1} + \cdots + \nu_{i,n_i} = 1$, where the variables are all the multipliers associated with the i-th input location x_i. The resulting expression can be minimized analytically, yielding the minimum $\min_j y_{i,j} + \bar{\delta}$, i.e., the smallest output available at x_i augmented by the noise bound. Finally, we conclude that the dual objective for (2.39) is

$$g(\lambda, \nu) = \begin{cases} (2.70), & \text{if } \lambda > 0 \\ \min_j y_{i,j} + \bar{\delta}, & \text{if } \lambda = 0 \end{cases} \tag{2.71}$$

As a last observation, a dual problem can also be derived for (2.41), calculating the lower part of the envelope. The formulation is analogous to (2.48), assuming the form

$$\begin{aligned} \max_{\nu \in \mathbb{R}^d, \lambda > 0} \quad & -\frac{1}{4\lambda} \nu^\top \Lambda K_{XX} \Lambda^\top \nu - \left(\mathsf{y} + \frac{1}{2\lambda} \Lambda K_{Xx}\right)^\top \nu \\ & - \bar{\delta}\|\nu\|_1 - \frac{1}{4\lambda} k(x, x) - \lambda \Gamma^2 \end{aligned} \tag{2.72}$$

Note that these are distinct objectives, not merely opposites. Therefore, two problems have to be solved to fully quantify the ground-truth uncertainty.

2.8.5 Norm estimation

To estimate the RKHS norm of an unknown map $f \in \mathcal{H}$, assume a set of samples $D = \{(x_i, f(x_i))\}_{i=1}^N$. Using the shorthand $f_X = \begin{bmatrix} f(x_1) \dots f(x_N) \end{bmatrix}^\top$, we have that

$$\hat{\Gamma} := \sqrt{f_X^\top K_{XX}^{-1} f_X} \leq \|f\|_{\mathcal{H}} \tag{2.73}$$

holds for any number of samples $N \in \mathbb{N}$ due to the optimal recovery property (Wendland, 2004). Moreover, the decomposition used in the proof of Lemma 2 shows that the quantity $\hat{\Gamma}$ can only increase with the addition of new data. Since $\|f\|_{\mathcal{H}}$ is the least upper bound for it, then this quantity can be used as an efficient lower estimate for the RKHS norm. In

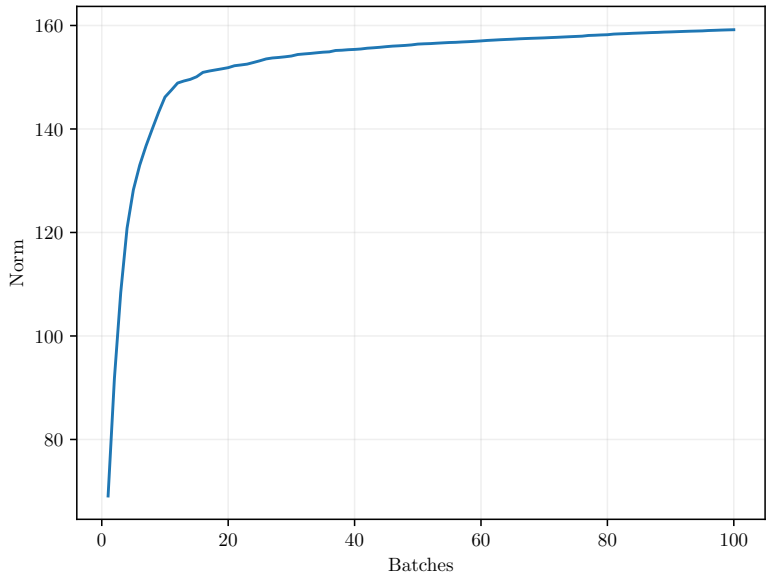
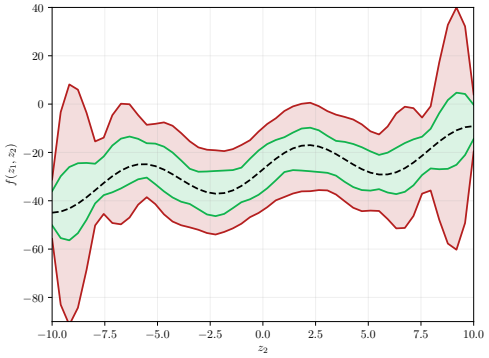


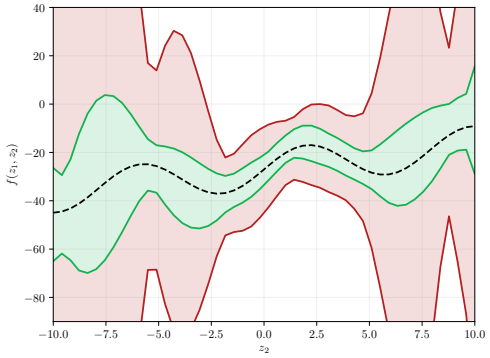
Figure 2.11 – Evolution of the RKHS norm of an interpolant fit with noise-free data and an increasing number of samples added in batches of 64.

a practical situation, expert knowledge should be elicited to augment $\hat{\Gamma}$ through a safety factor and hopefully transform it into an upper bound $\Gamma \geq \|f\|_{\mathcal{H}}$. Note however that no hard guarantees are offered, a situation similar to estimating Lipschitz constants purely from scattered observations. Finally, in case the outputs are corrupted by measurement noise, it is possible to quantify its worst-case effect on the estimation process as in Lemma 4. The approach is exemplified by estimating the norm of the function in the example in Section 2.6.2.

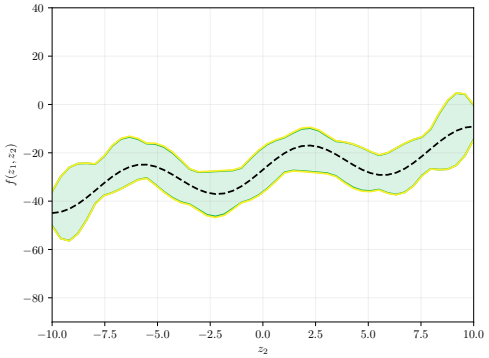
To estimate the RKHS norm of the ground truth function in (2.54), we employ the following approach: 6400 noise-free samples from a regular grid of the domain $\Omega = \{x \in \mathbb{R}^2 \,|\, [-5 \;\; -5]^\top \leq x \leq [5 \;\; 5]^\top\}$ are collected and an interpolation model with the specified kernel is used to fit the data. The RKHS norm of the resulting interpolant is then augmented by a factor of 1.1 to get the estimate of $\Gamma = 175$. To visualize how adding new data influences the norm of the interpolant, we fit an interpolant with 64 randomly subsampled points from the original dataset and determine its RKHS norm. Then, batches of 64 randomly subsampled points of the remaining dataset are added one by one to observe the norm increase. Figure 2.11 shows the procedure leading to a final interpolant norm of about 159.



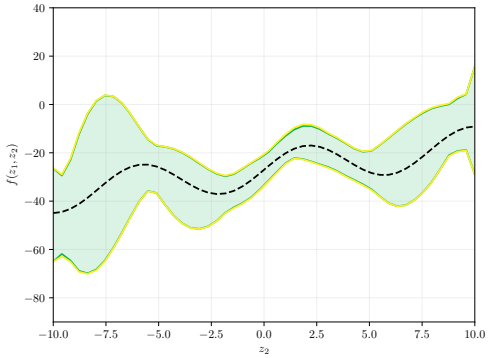
(a) Optimal bounds (green) vs KRR bounds (red) under grid sampling



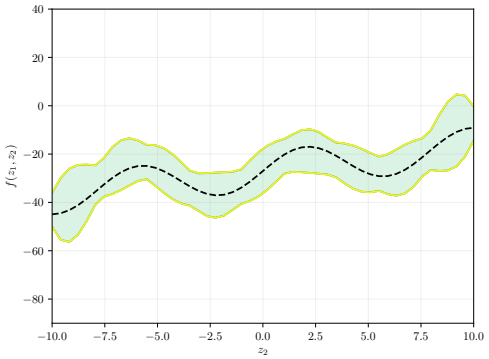
(b) Optimal bounds (green) vs KRR bounds (red) under random sampling



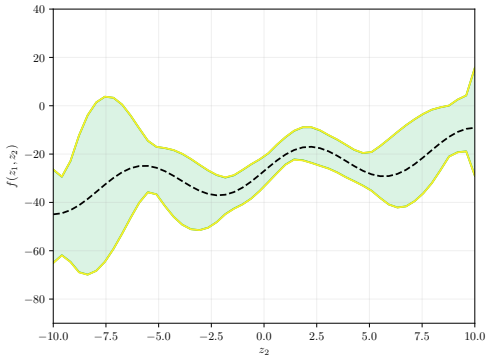
(c) Optimal bounds (green) vs approximate optimal bounds with 6 alternating steps (yellow) under grid sampling



(d) Optimal bounds (green) vs approximate optimal bounds with 6 alternating steps (yellow) under random sampling



(e) Optimal bounds (green) vs approximate optimal bounds with 10 alternating steps (yellow) under grid sampling



(f) Optimal bounds (green) vs approximate optimal bounds with 10 alternating steps (yellow) under random sampling

Figure 2.12 – Comparison of the optimal bounds to the KRR bounds, and approximate optimal bounds with 6 or 10 alternating steps on a cross-section of f, for grid sampling or random sampling, with $\bar{\delta} = 5$.

2.8.6 Example 3 plots

Sampling strategy	Optimal bounds	KRR bounds	6 step approx. optimal	10 step approx. optimal
Grid sampling	18.48	43.58	18.83	18.54
Random sampling	27.47	128.13	27.93	27.54

Table 2.4 – Average distance of the upper and lower bounds under different sampling strategies for $\bar{\delta} = 5$.

2.8.7 Example 3 GP comparison

	Γ	1200			1800			2400		
	$\bar{\delta}$	5	7.5	10	5	7.5	10	5	7.5	10
Grid	opt	20.29	28.57	36.39	22.54	31.31	39.58	24.41	33.56	42.17
	krr	49.15	71.79	94.44	49.81	72.46	95.11	50.48	73.13	95.78
	gp	1090.16	1247.34	1366.96	1624.19	1854.47	2028.24	2158.21	2461.60	2689.52
Rand	opt	39.95	53.43	65.41	47.00	62.15	75.57	52.76	69.32	83.89
	krr	312.44	458.51	604.57	313.61	459.68	605.74	314.79	460.85	606.91
	gp	1117.01	1268.40	1383.43	1664.18	1885.79	2052.67	2211.36	2503.18	2721.91

Table 2.5 – Average distance between the upper and lower bounds for the optimal (opt) and KRR (krr) deterministic cases, and the Gaussian process alternative (gp). High noise level (true $\bar{\delta} = 5$), using factors of 1, 1.5, and 2 to augment $\bar{\delta}$ and Γ.

Chapter 3

A controller benchmarking tool for buildings

3.1 Introduction

While nowadays automated building control still mostly relies on methods like Proportional Integral Derivative (PID) control or Rule-based Controllers (RBCs) (O'Neill et al., 2017), advanced control strategies promise a significant reduction in energy consumption (Mařík et al., 2011). Two recent reviews on the impact and saving potentials of building automation and control systems reported savings of up to 95% for the energy consumption of Heating Ventilation and Air Conditioning (HVAC) systems, with most of the studies reporting savings between 10% and 30% (Van Thillo et al., 2022; Vandenbogaerde et al., 2023). The large range of the results is due to different factors: the used control method, the considered building model, either in simulation or a real-world example, the climate and weather conditions, the baseline for comparison, etc.

Among the most popular approaches in the literature are MPC (Drgoňa et al., 2020; Maddalena et al., 2020) and RL (Nagy et al., 2023; Wang & Hong, 2020). The concept of MPC is to use a prediction model in an optimization problem, to determine the best control action considering future influences, and to repeat this in a receding horizon fashion to counteract disturbances or forecast errors. In RL, an agent explores the given state and action space and receives rewards based on the quality of the chosen action. Over time, it learns to select the best available action according to the current state of the system. Within these categories, many different flavors can be found. On the MPC side, examples are: Stochastic MPC (Oldewurtel, Sturzenegger, Esfahani, et al., 2013), robust MPC (Y. Gao et al., 2023), adaptive MPC (Sha et al., 2023), learning-based MPC (Terzi et al., 2020), and combinations of those (Yang et al., 2019). For RL, we have Q-learning (Kim & Lim, 2018), soft actor-critic (Coraci et al., 2021), proximal

policy optimization (Azuatalam et al., 2020), and many more (Nagy et al., 2023). Also, intermediate approaches that use RL on the basis of data or a model (Svetozarevic et al., 2022) or approximate MPC (Yang et al., 2021) that learn MPC controllers with NNs exist.

With this plethora of methods at hand, a fair comparison of their performance becomes extremely involved. Multiple software libraries have been developed to fill that gap and provide a set of simulation models for benchmarking controller performance. These works have different focuses, from direct building control to DR, from classic control methods to RL, and from reduced order modeling to high-fidelity simulations. Made for direct building control, BOPTEST (Blum et al., 2021) uses Modelica (Mattsson & Elmqvist, 1997) models and interfaces them with a RESTful API, currently providing 7 different test cases. To easily integrate RL agents, a Gym (Brockman et al., 2016) API for the BOPTEST models is provided through BOPTEST-Gym (Arroyo et al., 2021). Building on BOPTEST, ACTB (Marzullo et al., 2022) interfaces Spawn of EnergyPlus (Wetter et al., 2020) models either via a RESTful API or a Gym API. With this, it combines the modeling capabilities of EnergyPlus (Crawley et al., 2000) and Modelica. With the aim of emulating the interface to real buildings, building on top of VOLTTRON (Katipamula et al., 2016), V-BCT (Huang et al., 2023) provides building models of office buildings, based on EnergyPlus via the Functional Mockup Interface (FMI) standard (Blochwitz et al., 2012). BAC-Bench (Khayatian et al., 2023) also uses FMI for the simulation of an EnergyPlus-modeled apartment. The synchronization with other simulation models or physical systems is also made possible.

Many libraries focus on providing a benchmark for RL by providing a Gym or Gym-like API. COBS (T. Zhang & Ardakanian, 2020) works with EnergyPlus models, while not directly incorporating them in the library. One of the main features is the adaptability of occupancy schedules. Sinergym (Jiménez-Raboso et al., 2021) includes a multitude of simulation scenarios based on 6 Energyplus models via BCVTB (Wetter et al., 2008), which also allows for simulating Modelica models in the future, and offers Docker containers for the execution of the simulations. A similar route is taken by Gym-Eplus (Z. Zhang & Lam, 2018), using EnergyPlus with BCVTB. However, only one example model is included, and the library does not seem to be under current development. The RL-testbed for EnergyPlus (Moriyama et al., 2018) includes an EnergyPlus model of a data center, which can be simulated in a Docker container or via direct installation. How to use RL algorithms from the Ray RLlib on EnergyPlus models is demonstrated in RLlib-Energyplus (Galataud, 2023). BEAR (C. Zhang et al., 2023) uses physics-based or data-driven modeling of buildings, with the possibility to extract parameters from EnergyPlus models. 16 models are included for usage in simulation and both MPC and RL controllers have been demonstrated. Beobench (Findeis et al., 2022) uses a different approach, by providing a unified interface and Docker containers for different simulation libraries, namely BOPTEST, Sinergym, and Energym, presented in this chapter.

Another set of libraries targets DR applications instead of the single building control problem. CityLearn (Vázquez-Canteli et al., 2019) provides a Gym API for multi-agent RL and other control applications, by using pre-simulated EnergyPlus models. AlphaBuilding ResCommuity (Wang et al., 2021) also offers a Gym-like API for the control and coordination of Thermostatically Controlled Loads (TCLs). Resistance Capacitance (RC) models are used with parameters identified from the operation data of households. Incorporating models from both EnergyPlus and Modelica via Functional Mockup Units (FMUs), FlexDRL (Touzani et al., 2021) uses the OpenAI Gym API, too.

In this chapter, we present the building model library Energym, one of the earliest building control libraries using a Gym-like API and at the time of release one of the most extensive libraries for building controller benchmarking. It can be used for single building control or multi-building coordination, as demonstrated in Chapter 4. The 14 currently incorporated models are based on EnergyPlus and Modelica via FMUs, making it flexible for the addition of further models. During the simulations, pre-implemented KPIs are tracked, relating to the objectives of thermal comfort, energy consumption, and emissions. Different weather files allow controller testing under varying conditions, and the implemented evaluation mode permits comparability over different runs. The provided models were obtained in collaboration with Institut de Recerca en Energia de Catalunya (IREC), Universidad de Navarra (UNAV), Athens University, and CSEM S.A.

3.2 Building models

Energym includes 14 simulation models to date, six Modelica models, and eight EnergyPlus models of EnergyPlus version 9.5. An overview of the installed technical equipment, the controllability, the simulation software, and the location is given in Table 3.1. A description of each model's inputs and outputs is provided in Appendix 3.6.1. The models differ in size, number of rooms, usage profile, technical equipment, controllability, and climate zone. The six building envelopes that are the base for the 14 models are listed below. Four of them are available in multiple versions, differing in the control (e.g., controlling thermostat setpoints vs. controlling the equipment directly) or the installed equipment. Most of the buildings have been fully or partly calibrated from real measurement data, the work on that was mostly performed by collaborators in the scope of a European project and is therefore not included in this thesis. Details on this can be found in the accompanying publication. The buildings have the following characteristics.

Apartments

The Apartments building is a residential building with four stories, each being one apartment split into two thermal zones. It is located in Spain and has a central geothermal HP providing heat to all apartments. The building envelope is fictive, based on typical

Environment	Th	HP	Bat	AHU	EV	PV	Soft.	Loc.
ApartmentsThermal-v0	✔	✔	✔	✖	✔	━	E+	ESP
ApartmentsGrid-v0	✔	━	✔	✖	✔	━	E+	ESP
Apartments2Thermal-v0	✔	✔	✔	✖	✔	━	E+	ESP
Apartments2Grid-v0	✔	━	✔	✖	✔	━	E+	ESP
OfficesThermostat-v0	✔	✖	✖	✖	✖	━	E+	GRC
MixedUseFanFCU-v0	✔	✖	✖	✔	✖	✖	E+	GRC
SeminarcenterThermostat-v0	✔	━	✖	━	✖	━	E+	DNK
SeminarcenterFull-v0	✔	✔	✖	✔	✖	━	E+	DNK
SimpleHouseRad-v0	✖	✔	✖	✖	✖	━	Mod	CHE
SimpleHouseRSla-v0	✖	✔	✖	✖	✖	━	Mod	CHE
SwissHouseRSlaA2W-v0	✖	✔	✖	✖	✖	━	Mod	CHE
SwissHouseRSlaW2W-v0	✖	✔	✖	✖	✖	━	Mod	CHE
SwissHouseRSlaTank-v0	✖	✔	✖	✖	✖	━	Mod	CHE
SwissHouseRSlaTankDhw-v0	✖	✔	✖	✖	✖	━	Mod	CHE

Table 3.1 – Equipment of the different models in Energym. Th: Thermostat, HP: Heat Pump, Bat: Battery, AHU: Air Handling Unit, EV: Electric Vehicle, PV: Photovoltaic. ✔: present and controllable, ━: present but not controllable, ✖: absent.

Spanish construction materials used in the period from 1991 to 2007, but the HP was calibrated with a real HP located in the IREC laboratory. The envelope is the same for both Apartments and Apartments2 buildings; see Figure 3.2. The active surface area of the PV panels is 58m^2 with an inclination of 40° and south-oriented. The PV EnergyPlus component has a rated electric power output of 10.75 kW and the inverter efficiency is 0.95. In addition, occupancy, appliances and lighting consumptions follow stochastic profiles that differentiate each dwelling behavior, resulting in different energy demands. The DHW profiles are based on the European standard (EN16147, 2011).

The difference between Apartments and Apartments2 lies in their thermal systems. Apartments has a central geothermal HP, directly connected to hot water tanks (1 per Apartment) used only for Domestic Hot Water (DHW) consumption, and to a heating loop providing heat to the entire building. Apartments2 does not have this central heating system, but possesses four storage tanks (supplying heating and DHW to each apartment), each being alimented by a dedicated air-to-water HP.

Both buildings possess a stationary battery with a capacity of 10 kWh, maximum power for charging and discharging of 4 kW. In apartments, there is one Electric Vehicle (EV) with a capacity of 20 kWh and a maximum power for charging of 3.7 kW. For Apartments2, two EVs with the same characteristics are present. Usage schedules are stochastic and forecasts are provided via the forecast API.

The evaluation weather file used for Apartments and Apartments2 is given by the identifier `ESP_CT_Barcelona_ElPratAP1` and should not be used in the training process. Control

inputs and the most relevant outputs for the Apartments and Apartments2 models are listed in Table 3.4.

Offices

This building is located in Greece and includes 25 conditioned rooms with a total area of 643.73m^2 (see Figure 3.3). Of those 25 rooms, 14 are controllable with thermostats (2 storage rooms, 2 lobbies, 4 seminar rooms, 1 meeting room, and 5 offices). Water-to-air fan coil units are used to condition the spaces, where either water heating is provided by an oil boiler or water cooling by an electrical air-to-water chiller. Both the envelope and the technical systems were calibrated with the corresponding test site data.

The evaluation weather file for the Offices building is given by the identifier `GRC_TC_Lamia1`. The inputs and some outputs are described in Table 3.5.

MixedUse

The MixedUse building is a 566.38m^2 building located in Greece with 13 thermal zones, of which eight are controllable with thermostats (see Figure 3.4). The HVAC system installed consists of two Air Handling Units (AHUs), one dedicated exclusively to thermal zones 5, 6, and 7, and a second one serving the remaining thermal zones.

The first system, dedicated to TZ-5, 6, and 7, is composed of an air loop, an AHU that includes water coils, and two supply water loops: one with a HP Water Heater and the other with a chiller for cooling.

The second system, serving the entire facility, consists of an air loop with an AHU that has direct expansion coils. In addition, the zones that are affected by this system have variable refrigerant flow terminal units as part of the air-conditioning system. Both the envelope and the technical systems were calibrated with the corresponding test site data.

The evaluation weather file for the MixedUse building is given by the identifier `GRC_TC_Lamia1`. The control inputs and KPI related outputs are displayed in Table 3.6.

Seminarcenter

The Seminarcenter building is a one-story building located in Denmark and includes 22 conditioned rooms on 1278.94m^2 (see Figure 3.5). Five of the 22 rooms are divided into two thermal zones, and 18 rooms are controllable with thermostats.

Water convectors provide heating of the rooms with hot water from a buffer tank. For the buffer tank and the DHW, air-to-water HPs are used to supply the heating demand,

and an additional gas boiler is available in case the HPs can not provide enough heating.

The evaluation weather file for the Seminarcenter buildings is given by the identifier `DNK_MJ_Horsens2`. The control inputs to both simulation models and some outputs are described in Table 3.7. Both the envelope and the technical systems were calibrated with the corresponding test site measurements.

SimpleHouse

This building is a standard single-room house with HP and sun heating effects through glazing. Two versions exist, one with a standard radiator (SimpleHouseRad) and the other one with floor heating (SimpleHouseSlab). The first-order envelope model is designed based on thermal peak power and minimal outdoor temperature.

The evaluation weather file is given by the identifier `CH_ZH_Maur`. The control inputs to both simulation models and some outputs are described in Table 3.8.

SwissHouse

This building is a large-scale version of the SimpleHouse building with underfloor heating. It has been designed with parameters (thermal peak power, outdoor temperature) from a real house. Two versions of the model come with a HP, specifically an air-to-water HP (SwissHouseRSlaA2W-v0) and a water-to-water HP (SwissHouseRSlaW2W-v0). Another two versions have added tanks that supply the underfloor heating and are connected to a HP. Those are SwissHouseRSlaTank-v0 and SwissHouseRSlaTankDhw-v0, which has another added tank for DHW. The inputs of the models with and without tank differ, as given in Table 3.9. The evaluation weather file is given by the identifier `CH_ZH_Maur`, the same as for the SimpleHouse models.

3.3 Functionality and usage

Energym is designed to work with different controller types including RBCs, MPC controllers, and RL-based controllers. Hence, the building environments and their interface are provided, but the controller structure is not prescribed and is left free to the user. Moreover, model performance evaluation is not based on fixed rewards (like in Gym) but implemented via KPIs that can be computed by the user after an evaluation run. The main features of the library are outlined below [I].

[I] A full documentation of the library, describing usage and installation, is available at https://bsl546.github.io/energym-pages/.

Standardized evaluation

For each model outlined in Table 3.1, a physical objective to be reached is predefined. This objective might be, e.g., the minimization of the CO_2 emissions related to the building operation. The controllers also have to satisfy thermal constraints to guarantee occupant comfort. These two quantities - objective and constraints - are tracked with the implemented KPIs; see Section 3.3.2, Table 3.2. For each building, the evaluation phase with the predefined KPIs is run over a definite period of time and under predefined weather conditions.

Wrappers

Simulation wrappers are implemented to cope with different controller needs. In particular, wrappers are provided to scale inputs and/or outputs between values in a min-max fashion. The scaling can be beneficial for optimization-based controllers like MPC, due to the used model and solver structure. For RL controllers, an RL wrapper is provided to change the outputs of the `step` method and provide exactly the same outputs as in the Gym library, i.e. `outputs, reward, done, info`. One slight change with respect to Gym, however, is that the reward design is left free to the user and must be specified at wrapper initialization. This design choice was made for users to be free in the reward design phase, the main objective of any controller being to minimize the predefined KPIs. Similarly, for controller speed-up (in particular for MPC), a downsampling wrapper is provided to optimize computation time, making it possible to solve the problem less frequently than what the standard `step` method would impose.

Forecasting capabilities

For designing controllers such as MPC, it is important to have descriptions of external disturbances. For this, we provide weather forecasts (including irradiance and temperatures), optionally given by the exact values in the used weather files or by stochastic variations of those. Furthermore, we provide forecasts that are highly relevant for certain models: EV usage schedules for the Apartments and Apartments2 buildings, and electricity mix forecasts for the Seminarcenter. Random seeds to generate the forecasts are fixed in evaluation mode to ensure reproducibility of the results.

3.3.1 Usage

After importing Energym, a model can be created by calling the `make` method and specifying the name of the model and other optional parameters, i.e. the starting day of the simulation, the number of simulated days, the used weather file, and the used KPIs, all of which use default values if not specified upon initialization. The interaction

with the model, i.e. passing control inputs and receiving outputs, is done with the `step` method. Control inputs are Python dictionaries, with the setpoint name as key and input as value (possibly a list with multiple entries for multiple consecutive inputs), e.g., `{"Z01_T_Thermostat_sp":[21]}` (or `{"Z01_T_Thermostat_sp":[21, 22, 21]}`). Outputs are also defined as dictionaries using the predefined output names as keys. The main inputs and outputs for each model are given in Appendix 3.6.1. A full list is available in the online documentation. The `Wrapper` class is implemented to provide input-output wrapper functionalities. Weather and stochastic disturbances forecasts are available with the `get_forecast` method.

For the tracking of the KPIs, a `KPI` object is initialized for each model, and it automatically records the necessary data. Calling the method `get_kpi` returns the evaluation for a specified time interval (by default all the completed steps) as a dictionary. More details on handling the KPIs and the default ones are discussed in Section 3.3.2.

A simple usage example is given in Section 3.4.1.

3.3.2 Performance evaluation

A pre-compiled FMU is provided for each building model and can be used with different weather files. This is done for both Linux and Windows-based platforms, making Energym usable in diverse environments. Like this, users can train their controllers (i.e. RL agents or models for MPC) with different weather files, while the weather file for final evaluation is fixed. These fixed weather conditions for a predefined period of time ensure comparability of the control performances via the implemented KPIs. The characteristics of these fixed evaluation scenarios are displayed in Table 3.2. The defined KPIs fall into the categories of thermal comfort (related to temperature constraints) and objective KPIs (related to the objective to minimize).

KPI definition

For the thermal comfort, a range of acceptable temperatures is defined. The tracked KPIs are the average deviation from the target temperatures for each controlled thermal zone and the total number of range violations. Let the desired temperature range be defined by the interval $I = [a, b]$. Then the average deviation $d(T, I)$ for temperature measurements $T = \{t_i : i = 1, \ldots, H\}$ is defined as

$$d(T, I) := \frac{1}{H} \sum_{i=1}^{H} \|t_i\|_I \tag{3.1}$$

Model	Simulation period	Temperature constraints (°C)	Objective KPI
ApartmentsThermal-v0	Jan.-April	19-24	Grid exchange
ApartmentsGrid-v0	Entire year	19-24	Grid exchange
Apartments2Thermal-v0	Jan.-April	19-24	Grid exchange
Apartments2Grid-v0	Entire year	19-24	Grid exchange
OfficesThermostat-v0	Entire year	19-24	Power demand
MixedUseFanFCU-v0	Entire year	19-24	Power demand
SeminarcenterThermostat-v0	Jan.-May	21-24	CO_2 emissions
SeminarcenterFull-v0	Jan.-May	21-24	CO_2 emissions
SimpleHouseRad-v0	Jan.-April	19-24	Power demand
SimpleHouseRSla-v0	Jan.-April	19-24	Power demand
SwissHouseRSlaA2W-v0	Jan.-April	19-24	Power demand
SwissHouseRSlaW2W-v0	Jan.-April	19-24	Power demand
SwissHouseRSlaTank-v0	Jan.-April	19-24	Power demand
SwissHouseRSlaTankDhw-v0	Jan.-April	19-24	Power demand

Table 3.2 – Fixed evaluation scenarios for the simulation models.

$$\text{where } \|t\|_I = \begin{cases} a - t, & \text{if } t < a \\ 0, & \text{if } t \in I \\ t - b, & \text{if } t > b \end{cases}.$$

The number of total violations $v(T, I)$ is defined as

$$v(T, I) := \sum_{i=1}^{H} \delta(t_i, I) \tag{3.2}$$

$$\text{where } \delta(t, I) = \begin{cases} 0, & \text{if } t \in I \\ 1, & \text{if } t \notin I \end{cases}.$$

The average energy exchanged with the grid is tracked for the models based on the Apartments and Apartments2 buildings. Let $E_{prod} = \{e_{prod,i} : i = 1, \ldots, H\}$ be the set of H consecutive measurements of produced energy and $E_{con} = \{e_{con,i} : i = 1, \ldots, H\}$ of consumed energy. Then the average energy exchange $e(E_{prod}, E_{con})$ is defined as

$$e(E_{prod}, E_{con}) := \frac{1}{H} \sum_{i=1}^{H} |e_{prod,i} - e_{con,i}|. \tag{3.3}$$

In the evaluation scenario, the goal is to minimize this quantity and therefore maximize the self-consumption of produced energy.

The objective for the Offices, MixedUse, SimpleHouse and SwissHouse buildings is to minimize their power consumption. Let the mean power demand for H simulation steps be given by $D = \{d_i : i = 1, \ldots, H\}$. The minimization objective is again given by averaging over the measurements, so the average power demand $p(D)$ is defined as

$$p(D) := \frac{1}{H} \sum_{i=1}^{H} d_i. \tag{3.4}$$

The environments based on the Seminarcenter building track the CO_2 emissions for the installed gas boiler and the varying electricity mix. A minimization of this emission is the focus of their evaluation scenario. Let the emission values be given by $C = \{c_i : i = 1, \ldots, H\}$. The computed KPI for those measurements is the average emission $g(C)$ defined as

$$g(C) := \frac{1}{H} \sum_{i=1}^{H} c_i. \tag{3.5}$$

Instead of using predefined KPIs, it is also possible to define custom KPIs. An example of this is given in Section 3.4.2.

3.4 Examples

We provide examples for the usage of the different Energym features, starting with the basic usage, the definition of KPIs, to an MPC implementation.

3.4.1 Basic usage example

A simple example of the usage of the library is given below. It demonstrates the interaction with the simulation model for 100 timesteps, assuming a function `get_input()` has been implemented, that computes the control input for the current measured state of the model and a forecast for the next 10 timesteps. The chosen parameters are arbitrary and just fulfill demonstrative purposes.

```
import energym

env = energym.make("Apartments2Grid-v0")
out = env.get_output()
for i in range(100):
    forecast = env.get_forecast(forecast_length=10)
    inp = get_input(out, forecast)
    out = env.step(inp)
kpis = env.get_kpi()
env.close()
```

3.4.2 KPI example

Default KPIs are defined for each model, but the user can also define custom KPIs to be tracked. This is done by specifying a Python dictionary containing the information of the variables of interest and KPI computation method. An example dictionary for the KPIs looks as follows.

```
kpi_dict = {"kpi1": {"name": "Fa_Pw_All", "type": "avg"},
            "kpi2": {
                "name": "Z01_T",
                "type": "tot_viol",
                "target": [19,24],
            }
}
```

For more information on the KPI implementation, we refer to the documentation.

3.4.3 MPC example

As an illustration of the usage of Energym, we implement a simple MPC approach on the SimpleHouseRad-v0 model. Due to the low thermal mass and short simulation timestep of 5 minutes, a planning horizon of 2 hours, i.e. $H = 24$ steps, is chosen in the MPC.

System identification

We model the building using an AutoRegressive model with eXogenous input (ARX) with an input order of 1 and an output order of 3. In addition to the control variable u, we also consider the ambient temperature T and solar irradiance I as uncontrolled disturbances.

Denoting the room temperature at time t by y_t, we get a model of the following form.

$$y_{t+1} = a_1 y_t + a_2 y_{t-1} + a_3 y_{t-2} + b_1 u_t + b_2 T_t + b_3 I_t, \quad t \geq 0 \tag{3.6}$$

This functional dependence can be simplified to the state space representation

$$x_{t+1} = A x_t + B \tilde{u}_t, \quad t \geq 2 \tag{3.7}$$

with states $x_t = \begin{pmatrix} y_t \\ y_{t-1} \\ y_{t-2} \end{pmatrix}$, inputs $\tilde{u}_t = \begin{pmatrix} u_t \\ T_t \\ I_t \end{pmatrix}$, and matrices $A = \begin{pmatrix} a_1 & a_2 & a_3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$, $B = \begin{pmatrix} b_1 & b_2 & b_3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$.

One week of input-output data is sampled from the simulation model, containing phases of constant input u as well as input ramps. The Python package SIPPY (Armenise et al., 2018) is used to identify A and B from the data.

In the MPC evaluation, we additionally consider a simple integrator that removes possible constant offsets due to changing weather conditions.

MPC implementation

We implement an MPC with quadratic cost according to the following formulation, assuming given forecasts of $T_t, \ldots, T_{t+H-1}, I_t, \ldots, I_{t+H-1}$:

$$\begin{aligned} \min_{u_{t|t}, \ldots, u_{t+H-1|t}} \quad & \frac{1}{H} \sum_{k=t}^{t+H-1} \tilde{u}_{k|t}^\top \tilde{u}_{k|t} \\ \text{s.t.} \quad & x_{k+1|t} = A x_{k|t} + B \tilde{u}_{k|t} && k = t, \ldots, t+H-1 \\ & x_{k|t} \in X && k = t, \ldots, t+H \\ & \tilde{u}_{k|t} \in U && k = t, \ldots, t+H \end{aligned} \tag{3.8}$$

where the subscript $k|t$ denotes the prediction for timestep k, starting from timestep t, $x_{t|t}$ is the starting state, the feasible set for the states X is implemented through box constraints with the acceptable temperature range $[19, 24]$°C, and the feasible set for the inputs U indicates the range $[0, 1]$ for u. The state constraints are implemented as soft constraints to guarantee recursive feasibility in the experiments. The Python package CVXPY (Diamond & Boyd, 2016) is used for modeling the optimization problem (3.8).

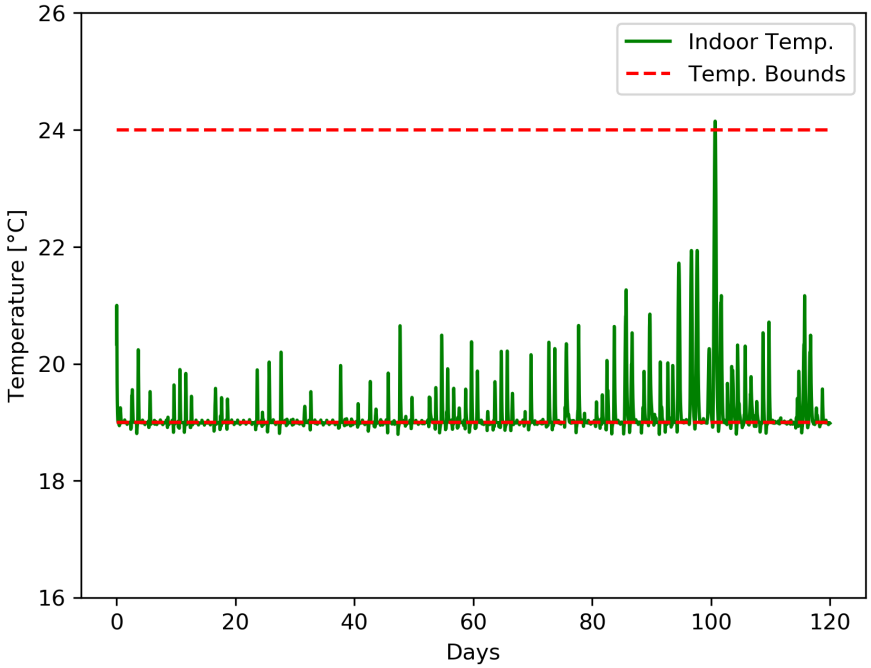


Figure 3.1 – Indoor temperature of the SimpleHouseRad-v0 model, achieved by the MPC over 4 months.

Results

We evaluate the performance of the MPC on the 4 evaluation months of the model (according to Table 3.2). The achieved indoor temperature and the temperature bounds are displayed in Figure 3.1. The strategy of the MPC is straightforward: to minimize power, it tries to keep the temperature close to the lower bound at all times. Especially later in the evaluation, higher external temperatures and high irradiance values lead to higher indoor temperatures and even a violation of the upper bound. This is due to the low thermal mass of the building, which can result in a quick increase of the temperature, despite the HP being turned off.

Over the whole evaluation period, the MPC achieves an average power demand of 1.327 kW. Due to the operation close to the lower temperature bound, many small violations are accumulated, which result in violations in about 64% of the timesteps. This clearly indicates the need to deal with different forms of uncertainty, here mostly in the form of modeling uncertainty, the simplest way of which would be a tightening of the lower temperature bound in the MPC problem.

3.5 Conclusion

The library Energym presented in this chapter aims at providing building models and standardized evaluation scenarios and metrics to develop, test, and benchmark controllers. The wide range of building models makes it an easy-to-use tool for reproducing controllers in different environments and testing their suitability and performance. As one of the first ready-to-use building simulation tools, Energym combines the benefits of EnergyPlus and Modelica models and allows for single-building control as well as multi-building coordination. Comparability is guaranteed through the implemented evaluation mode that fixes all variable conditions to a reference scenario. The presented KPIs include thermal comfort as well as metrics related to grid exchange, power demand, and CO_2 emissions. With the provision of different simulation wrappers and forecasting options for external conditions, the library is especially suited for testing advanced control methods.

The general usage of Energym was demonstrated together with the evaluation of the KPIs, and a basic MPC implementation. In the MPC, an ARX model is identified from training data and used for the predictions in the corresponding optimization problem. The results showed an operation of the model at a low power level, but also the need for more sophisticated schemes to limit the comfort-bound violations.

Different extensions of Energym and related work are envisioned. First, adding more representative models to the library is of interest. In this line of work, combining the strengths of EnergyPlus, the detailed envelope modeling, and Modelica, the realistic simulation of equipment, is of particular significance. Using, e.g., Spawn of EnerpyPlus (Wetter et al., 2020) or co-simulation of the two tools makes this possible. Second, the implementation of baseline control strategies, be it RBC, MPC, or RL, for all the models could provide benchmarks to test new controllers against. A coordination example of multiple Energym buildings is provided in the next chapter in the domain of consumption flexibility management.

3.6 Appendix

3.6.1 Building descriptions

In this part, we give a reference for the inputs and the outputs of the simulation models that are related to the KPIs (other outputs not entering the KPI calculation, like flow rate and flow temperature, are not listed). The common output variables for all EnergyPlus-based models are given in Table 3.3. The bounds given in the tables are not used to cut-off values (unless the specific cut-off wrapper is used), but are used by default by the input/output scaling wrappers to scale the signals to values close to/within the [0,1] interval.

Variable Name	Description	Bounds	Units
Ext_T	Current outdoor temperature	[-25,40]	°C
Ext_RH	Current outdoor relative humidity	[0,100]	%RH
Ext_Irr	Current direct normal irradiance	[0,1000]	$W.m^{-2}$

Table 3.3 – Common outputs for the EnergyPlus based models.

Apartments

Variable Name	Description	Bounds	Units	Model
	Inputs			
P1_T_Thermostat_sp ... P4_T_Thermostat_sp	Temperature setpoint per appartment	[16,26]	°C	1/2/3/4
Bd_T_HP_sp	Heat pump supply temperature setpoint	[35,55]	°C	1/2
P1_T_Tank_sp ... P4_T_Tank_sp	Bottom water tank temperature setpoint	[30,70]	°C	1/2
Bd_Pw_Bat_sp	Battery charging/discharging setpoint	[-1,1]	-	1/2/3/4
Bd_Ch_EVBat_sp	EV battery charging setpoint	[0,1]	-	1/2
Bd_Ch_EV1Bat_sp	EV battery charging setpoint	[0,1]	-	3/4
Bd_Ch_EV2Bat_sp	EV battery charging setpoint	[0,1]	-	3/4
HVAC_onoff_HP_sp	Heat pump on/off setpoint	{0,1}	-	1
P1_onoff_HP_sp ... P4_onoff_HP_sp	Heat pump on/off setpoint	{0,1}	-	3
	Outputs			
Fa_E_self	Energy exchanged with grid for timestep	[-2000,2000]	Wh	1/2/3/4
Z01_T ... Z08_T	Current zone temperature	[10,40]	°C	1/2/3/4

Table 3.4 – Inputs and outputs for the models ApartmentsThermal-v0 (1), ApartmentsGrid-v0 (2), Apartments2Thermal-v0 (3) and Apartments2Grid-v0 (4).

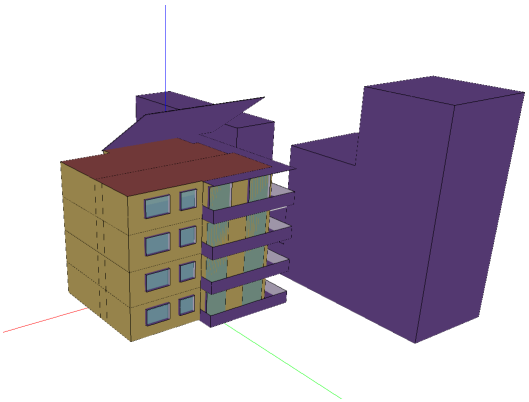


Figure 3.2 – Envelope visualization for the Apartments and Apartments2 buildings.

Offices

Variable Name	Description	Bounds	Units
	Inputs		
Z01_T_Thermostat_sp ... Z07_T_Thermostat_sp Z15_T_Thermostat_sp ... Z20_T_Thermostat_sp Z25_T_Thermostat_sp	Zone temperature setpoint	[16,26]	°C
Bd_Cooling_onoff_sp	Chiller on/off	{0,1}	-
Bd_Heating_onoff_sp	Boiler on/off	{0,1}	-
	Outputs		
Fa_Pw_All	Current power demand of whole facility	[0,10000]	W
Fa_Pw_PV	Current produced power	[0,2000]	W
Z01_T ... Z07_T Z15_T ... Z20_T Z25_T	Current zone temperature	[10,40]	°C

Table 3.5 – Inputs and outputs for the model OfficesThermostat-v0.

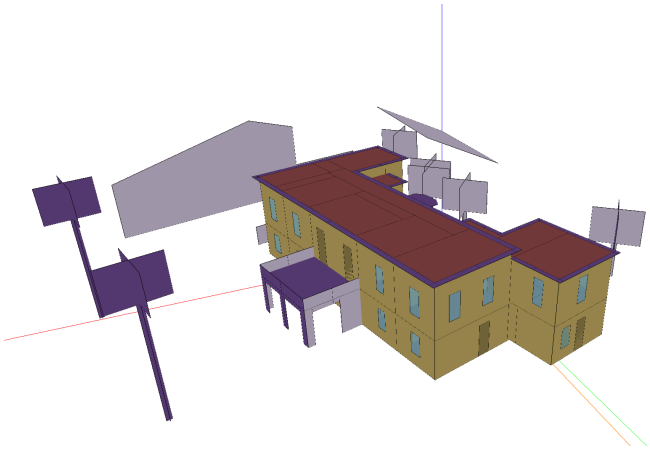


Figure 3.3 – Envelope of the Offices building.

MixedUse

Variable Name	Description	Bounds	Units
	Inputs		
Z02_T_Thermostat_sp ... Z05_T_Thermostat_sp Z08_T_Thermostat_sp ... Z11_T_Thermostat_sp	Zone temperature setpoint	[16,26]	°C
Bd_T_AHU1_sp Bd_T_AHU2_sp	AHU temperature setpoint	[10,30]	°C
Bd_Fl_AHU1_sp Bd_Fl_AHU2_sp	AHU flow rate setpoint	[0,1]	-
	Outputs		
Fa_Pw_All	Current power demand of whole facility	[0,50000]	W
Z02_T ... Z05_T Z08_T ... Z11_T	Current zone temperature	[10,40]	°C

Table 3.6 – Inputs and outputs for the model MixedUseFanFCU-v0.

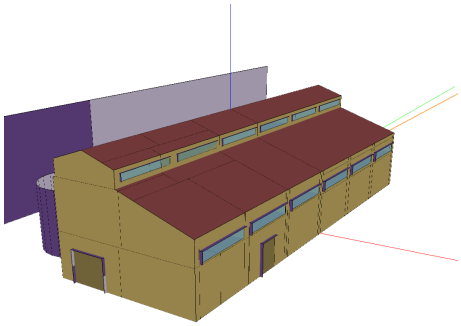


Figure 3.4 – Envelope of the MixedUse building.

Seminarcenter

Variable Name	Description	Bounds	Units	Model
	Inputs			
Z01_T_Thermostat_sp ... Z06_T_Thermostat_sp Z08_T_Thermostat_sp ... Z11_T_Thermostat_sp Z13_T_Thermostat_sp ... Z15_T_Thermostat_sp Z18_T_Thermostat_sp ... Z22_T_Thermostat_sp	Zone temperature setpoint	[16,26]	°C	1/2
Bd_onoff_HP1_sp ... Bd_onoff_HP4_sp	Heat pump on/off setpoint	{0,1}	-	2
Bd_T_HP1_sp ... Bd_T_HP4_sp	Heat pump temperature setpoint	[30,65]	°C	2
Bd_T_AHU_coil_sp	AHU water coil temperature setpoint	[15,40]	°C	2
Bd_T_buffer_sp	Buffer tank temperature setpoint	[15,70]	°C	2
Bd_T_mixer_sp	HPs water loop supply temperature setpoint	[20,60]	°C	2
Bd_T_HVAC_sp	AHU air supply temperature setpoint	[10,26]	°C	2
	Outputs			
Bd_CO2	Timestep equivalent CO_2 emission mass	[0,10]	kg	1/2
Fa_Pw_All	Current power demand of whole facility	[0,100000]	W	1/2
Z01_T ... Z06_T Z08_T ... Z11_T Z13_T ... Z15_T Z18_T ... Z22_T	Current zone temperature	[10,40]	°C	1/2

Table 3.7 – Inputs and outputs for the models SeminarcenterThermostat-v0 (1) and SeminarcenterFull-v0 (2).

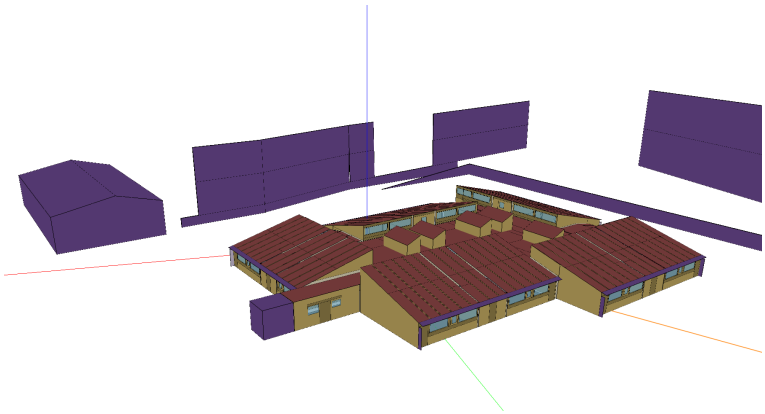


Figure 3.5 – Envelope of the Seminarcenter building.

SimpleHouse

Variable Name	Description	Bounds	Units
	Inputs		
u	Heat pump normalized power	[0,1]	-
	Outputs		
TOut.T	Outside Temperature	[253.15,343.15]	K
temRoo.T	Room Temperature	[263.15,343.15]	K
heaPum.P	Heat pump power	[0, 30]	kW
temRet.T	Heat pump return temperature	[273.15,353.15]	K
temSup.T	Heat pump supply temperature	[273.15,353.15]	K

Table 3.8 – Inputs and outputs for the models SimpleHouseRad-v0 and SimpleHouseRSla-v0.

SwissHouse

Variable Name	Description	Bounds	Units	Model
	Inputs			
u	Heat pump normalized power	[0,1]	-	1/2
uHP	Heat pump normalized power	[0,1]	-	3/4
uRSla	Hot water emitter flow fraction	[0,1]	-	3/4
uValveDHW	Hot water valve opening fraction	[0,1]	-	4
uFlowDHW	Hot water flow demand fraction	[0,1]	-	4
	Outputs			
TOut.T	Outside Temperature	[253.15,343.15]	K	1/2/3/4
temRoo.T	Room Temperature	[263.15,343.15]	K	1/2/3/4
heaPum.P	Heat pump power	[0, 30]	kW	1/2/3/4
temRet.T	Heat pump return temperature	[273.15,353.15]	K	1/2/3/4
temSup.T	Heat pump supply temperature	[273.15,353.15]	K	1/2/3/4

Table 3.9 – Inputs and outputs for the models SwissHouseRSlaA2W-v0 (1), SwissHouseRSlaW2W-v0 (2), SwissHouseRSlaTank-v0 (3), and SwissHouseRSlaTankDhw-v0 (4)

.

Chapter 4

Uncertainty-aware flexibility estimation, scheduling, and dispatch in buildings

4.1 Introduction

In light of increasingly renewable and distributed electricity generation, as well as increased electrification, e.g. through using EVs and electric heating systems like HPs, DR is a key technology in ensuring reliable grid operation and balancing while avoiding costly grid reinforcements (IEA, 2022).

Within the set of assets to provide flexibility services to the grid, buildings have been identified as promising candidates (e.g. (Vrettos et al., 2014) and (Lymperopoulos et al., 2015)). This is due to their inherent capabilities of changing or shifting their load and being responsible for a large share of the global final energy consumption and CO_2 emissions (about 34% and 37 % respectively in 2021 (Global Alliance for Buildings and Construction, 2022)).

To efficiently use the flexibility potential of buildings, an accurate estimation of the available flexibility of individual buildings or groups of buildings and a coordination mechanism for a pool of buildings are needed. Different ways to quantify flexibility are presented in (Reynders et al., 2018).

The available flexibility of a single building is estimated by a model-based approach in (Gasser et al., 2021). The authors use a flexibility envelope concept to describe flexibility as the availability times of discrete power levels, previously introduced in (D'hulst et al., 2015). An ML approach to learning and approximating the flexibility envelopes is demonstrated in (Hekmat et al., 2021). However, samples generated from

a model-based procedure are still needed in that work. In (Maasoumy et al., 2014), a flexibility envelope-like approach to quantifying the flexibility of fans is presented for individual commercial buildings. Additionally, a contract framework for flexibility usage in a receding horizon fashion is proposed. The New York day-ahead DR program is considered in (Qureshi et al., 2014) with the goal of maximizing the profit of individual commercial buildings. This approach applies MPC for the estimation and usage of the available flexibility.

To avoid potentially costly, difficult, and cumbersome modeling of the assets, data-driven approaches like (Hekmat et al., 2021) or (Brusokas et al., 2021), which uses ML methods to predict temperature trajectories for assets with HPs, have been proposed. Another popular approach, combining the availability of data with physical intuition, is the use of virtual battery models. In (Hughes et al., 2016), virtual battery modeling of flexible loads is combined with a direct control strategy. Still, a detailed model of the loads is used to identify the battery parameters. (F. L. Müller et al., 2017) determines switching times of the heating system in buildings from data to learn the parameters of a virtual battery model. Using this battery model to represent the feasible consumption profiles as a set, the approach is demonstrated in a large-scale study in (F. Müller & Jansen, 2019). With the goal of characterizing feasible input trajectories for constrained linear systems, a battery model parameterization of a trackable reference set is determined in (Gorecki et al., 2015). Virtual battery models have also been proposed to model the flexibility of aggregations of TCLs. In (Hao et al., 2013, 2015; Sanandaji et al., 2014), stochastic battery parameters and ramp-rate constraints are considered in the flexibility estimation, and a priority stack of the available assets is used to control the switching of operation modes. This control mechanism is also used in (Zhao & Zhang, 2016; Zhao et al., 2017), while flexibility of individual TCLs is described by polytopes and aggregated through the Minkowski sum. This aggregation is approximated and bounded by battery models.

When dealing with systems that provide e.g. heating or cooling to buildings, considering uncertainty is vital to guarantee the comfort of the occupants (Oldewurtel et al., 2012). Sources for this uncertainty are e.g. inaccurate weather forecasts, and internal gains, but also varying levels of reliability of the systems when providing flexibility. Quantification of the impact of different sources of uncertainty, e.g. plant-model mismatch or forecasting errors, is presented in (Mathieu, Vayá, et al., 2013), where the authors use a battery modeling approach to estimate the flexibility of load aggregations. These sources of uncertainty are also addressed in (Amadeh et al., 2022), where a stochastic MPC approach is proposed to quantify available flexibility under uncertainty. The uncertain response of buildings to control signals is modeled with GPs in (Nghiem & Jones, 2017). The predictions of these GPs are used in an MPC framework to determine control signals for optimal demand tracking.

The previously mentioned approaches mostly focus on either flexibility estimation for single or multiple assets, or estimation and control for single assets. Coordination of a

pool of assets has been addressed in a number of works. Hierarchical control approaches have been proposed in e.g. (Borsche et al., 2014), (Vrettos et al., 2016), and (Qureshi & Jones, 2018), where the former uses a scenario-based approach and the latter two employ robust control methods. The principle of these hierarchical controllers is to have a high-level mechanism to plan available reserves in advance, e.g. in a day-ahead fashion, an intermediate level MPC scheme to optimize operation while keeping the planned reserves, and a low-level controller to track the reference determined by the MPC. However, these solutions require accessibility to low-level control, which can be prohibitive for some users. A priority-stack-based controller for switching TCLs, similar to the ones used in the previously mentioned works on battery modeling for aggregated TCLs, is used in (Mathieu, Koch, et al., 2013), considering different levels of available information in the control. Multiple optimization-based approaches to determine the activation of flexible assets and generation units are proposed for the unit commitment problem. An example of this is given in (Bertsimas et al., 2013), where a robust approach to dealing with uncertainty in large-scale mixed-integer programs is presented.

In this chapter, we present an estimation and coordination framework that addresses the previously mentioned issues in the following way. A data-driven virtual battery modeling approach is put forward to estimate flexibility of single assets. This approach eliminates the need for extensive modeling while being able to handle uncertainty from different sources. Furthermore, we present a MILP formulation that schedules a pool of assets according to their predicted flexibility and an aggregated consumption request. The actual dispatch of the incoming requests to the active assets is done by a heuristic algorithm. With this framework, the assets are controlled by their local controller most of the time and only occasionally receive consumption requests for a limited time. In large-scale simulations with up to 500 buildings from Energym, we show a clear advantage of maintaining comfort in the scenario of self-consumption maximization in comparison to a baseline approach. In the scenario of peak reduction, our approach delivers better results, while also being able to maintain the comfort bounds more consistently.

4.2 Risk-aware virtual battery modeling for flexibility estimation of buildings

In this section, we describe a method to estimate the available consumption flexibility of different assets in a data-driven way. For this, we learn the sample sets of uncertain battery parameters and handle this uncertainty via risk measures. Flexibility itself is characterized by different forms of flexibility envelopes, which we will introduce next.

4.2.1 Flexibility characterizations

To quantify the consumption flexibility of flexible assets, we first need to properly define it. We will use three different types of flexibility envelopes for this, the first two are the absolute and relative versions of the ones also used in (D'hulst et al., 2015) and (Gasser et al., 2021), characterized by methodology B in (Reynders et al., 2018). The third one has similar underlying principles, but instead considers a fixed time duration and characterizes the minimum and maximum available power in that period.

We consider assets with the following description of their state dynamics:

$$\boldsymbol{x}_{t+1} = h(\boldsymbol{x}_t, p_t, e_t, \omega_t) \tag{4.1}$$

where $\boldsymbol{x}_t \in \mathbb{R}^{n_x}$ denotes a collection of states and measured variables at time t, $p_t \in \mathbb{R}$ the power input, $e_t \in \mathbb{R}^{n_e}$ a collection of external conditions, and $\omega_t \in \mathbb{R}^{n_\omega}$ unmeasured disturbances. $h : \mathbb{R}^{n_x} \times \mathbb{R} \times \mathbb{R}^{n_e} \times \mathbb{R}^{n_\omega} \to \mathbb{R}^{n_x}$ denotes the function describing the state transition.

These assets are subject to constraints, e.g. bounding the temperature in thermal assets or the State of Charge (SoC) in batteries, where we consider box constraints of the form

$$\underline{\boldsymbol{x}} \leq \boldsymbol{x}_t \leq \bar{\boldsymbol{x}} \tag{4.2}$$

with the lower and upper bounds denoted by $\underline{\boldsymbol{x}}, \bar{\boldsymbol{x}} \in \mathbb{R}^{n_x}$ and the inequalities applied entry-wise. Additionally, we consider input constraints

$$\underline{p} \leq p_t \leq \bar{p}. \tag{4.3}$$

Furthermore, we consider controlled systems, so assets with a local controller installed. The controller tries to keep the state or a part of the state of the asset either at a certain setpoint or within a certain range and we assume that this controller behavior is reproducible, meaning that the same initial states and external states lead to the same controller responses. We denote the nominal state trajectory of length H resulting from this controller operation by $\boldsymbol{x}^{\mathrm{b}}_{0:H-1} = [x^{\mathrm{b}}_0, \ldots, x^{\mathrm{b}}_{H-1}] \in \mathbb{R}^{n_x \times H}$. The power inputs leading to this state trajectory are given by $\boldsymbol{p}^{\mathrm{b}}_{0:H-1} = [p^{\mathrm{b}}_0, \ldots, p^{\mathrm{b}}_{H-1}] \in \mathbb{R}^H$ in turn and are called nominal power inputs or baseline power inputs.

With this, we can define flexibility envelopes as follows:

Definition 5 (Flexibility envelope)**.** *Let $\boldsymbol{x}_{t+1} = h(\boldsymbol{x}_t, p_t, e_t, \omega_t)$ be the state dynamics of an asset, as in* (4.1)*. Given the nominal state trajectory $\boldsymbol{x}^b_{0:H-1}$, the trajectory of external conditions $\boldsymbol{e}_{0:H-1} = [e_0, \ldots, e_{H-1}]$, the trajectory of unmeasured disturbances $\boldsymbol{\omega}_{0:H-1} = [\boldsymbol{\omega}_0, \ldots, \boldsymbol{\omega}_{H-1}]$, and a vector of n_p discrete power levels $\boldsymbol{p} = [p_0, \ldots, p_{n_p-1}] \in [\underline{p}, \bar{p}]^{n_p}$ with*

$\underline{p} \leq p_0 < \cdots < p_{n_p} \leq \bar{p}$, we define the flexibility envelope $\boldsymbol{E} \in \mathbb{R}^{n_p \times H}$ by

$$\begin{aligned} \boldsymbol{E}_{i,t} = \max \quad & T \\ s.t. \quad & \boldsymbol{x}_{l+1} = h(\boldsymbol{x}_l, p_i, e_l, \omega_l), \quad l = t, \ldots, t+T-1 \\ & \boldsymbol{x}_t = \boldsymbol{x}_t^b \\ & \underline{\boldsymbol{x}} \leq \boldsymbol{x}_l \leq \bar{\boldsymbol{x}}, \quad l = t, \ldots, t+T \end{aligned} \tag{4.4}$$

This flexibility envelope specifies for how long certain power levels can be maintained without violating constraints while starting from the nominal state at each timestep. This can also be defined in a relative manner with respect to the nominal power inputs $\boldsymbol{p}^{\mathrm{b}}_{0:H-1}$. For this, we first introduce the notion of a relative consumption request.

Definition 6 (Relative Consumption Request)**.** *Given a baseline power $p_t^b \in \mathbb{R}$, we define a relative consumption request $r_t \in \mathbb{R}$ such that the desired total power at time t is $p_t = p_t^b + r_t$.*

Definition 7 (Relative flexibility envelope)**.** *Let $\boldsymbol{x}_{t+1} = h(\boldsymbol{x}_t, p_t, e_t, \omega_t)$ be the state dynamics of an asset, as in* (4.1)*. Given the nominal state trajectory $\boldsymbol{x}^b_{0:H-1}$, the trajectory of external conditions $\boldsymbol{e}_{0:H-1}$, the trajectory of unmeasured disturbances $\boldsymbol{\omega}_{0:H-1}$, the nominal power inputs $\boldsymbol{p}^b_{0:H-1}$, and a vector of n_p discrete power levels $\boldsymbol{r} = [r_0, \ldots, r_{n_p-1}] \in [\underline{p}, \bar{p}]^{n_p}$ with $\underline{p} \leq r_0 < \cdots < r_{n_p} \leq \bar{p}$, we define the relative flexibility envelope $\boldsymbol{F} \in \mathbb{R}^{n_p \times H}$ by*

$$\begin{aligned} \boldsymbol{F}_{i,t} = \max \quad & T \\ s.t. \quad & \boldsymbol{x}_{l+1} = h(\boldsymbol{x}_l, p_l^b + r_i, e_l, \omega_l), \quad l = t, \ldots, t+T-1 \\ & \boldsymbol{x}_t = \boldsymbol{x}_t^b \\ & \underline{\boldsymbol{x}} \leq \boldsymbol{x}_l \leq \bar{\boldsymbol{x}}, \quad l = t, \ldots, t+T \\ & \underline{p} \leq p_l^b + r_i \leq \bar{p}, \quad l = t, \ldots, t+T-1 \end{aligned} \tag{4.5}$$

As a last flexibility characterization, we consider a maximum activation time of $k \in \mathbb{N}$ timesteps. The flexibility is then given by the minimum and maximum available relative changes with respect to baseline consumption.

Definition 8 (Fixed-time flexibility envelope)**.** *Let $\boldsymbol{x}_{t+1} = h(\boldsymbol{x}_t, p_t, e_t, \omega_t)$ be the state dynamics of an asset, as in* (4.1)*. Given the nominal state trajectory $\boldsymbol{x}^b_{0:H-1}$, the trajectory of external conditions $\boldsymbol{e}_{0:H-1}$, the trajectory of unmeasured disturbances $\boldsymbol{\omega}_{0:H-1}$, the nominal power inputs $\boldsymbol{p}^b_{0:H-1}$, and a request duration $k \in \mathbb{N}$, we define the relative*

fixed-time flexibility envelope $\boldsymbol{R}^k_{0:H-1} = [\underline{\boldsymbol{R}}^k_{0:H-1}, \bar{\boldsymbol{R}}^k_{0:H-1}]$ *through*

$$\begin{aligned} \underline{\boldsymbol{R}}^k_t = \min \quad & r \\ s.t. \quad & \boldsymbol{x}_{l+1} = h(\boldsymbol{x}_l, p^b_l + r, e_l, \omega_l), \quad l = t, \dots, t+k-1 \\ & \boldsymbol{x}_t = \boldsymbol{x}^b_t \\ & \underline{\boldsymbol{x}} \leq \boldsymbol{x}_l \leq \bar{\boldsymbol{x}}, \quad l = t, \dots, t+k \\ & \underline{p} \leq p^b_l + r \leq \bar{p}, \quad l = t, \dots, t+k-1 \end{aligned} \tag{4.6}$$

$$\begin{aligned} \bar{\boldsymbol{R}}^k_t = \max \quad & r \\ s.t. \quad & \boldsymbol{x}_{l+1} = h(\boldsymbol{x}_l, p^b_l + r, e_l, \omega_l), \quad l = t, \dots, t+k-1 \\ & \boldsymbol{x}_t = \boldsymbol{x}^b_t \\ & \underline{\boldsymbol{x}} \leq \boldsymbol{x}_l \leq \bar{\boldsymbol{x}}, \quad l = t, \dots, t+k \\ & \underline{p} \leq p^b_l + r \leq \bar{p}, \quad l = t, \dots, t+k-1 \end{aligned} \tag{4.7}$$

This way of describing flexibility is similar to the power shifting capability described in (Reynders & Saelens, 2015), but uses a fixed time duration. Thus, it also relates to methodology F in (Reynders et al., 2018).

Remark 7. *The definitions 5, 7, and 8 assume perfect knowledge of the considered assets and all external conditions and disturbances. Therefore, these definitions give the exact quantification of flexibility but are not directly usable. A method to estimate these flexibility envelopes with virtual battery modeling and uncertainty quantification is presented next.*

4.2.2 Virtual battery modeling for buildings

In this section, we consider assets that behave like a generalized battery, meaning that their state can be described by a scalar, similar to the SoC of batteries. This holds in particular for thermal assets like buildings, and we denote the state at time t by $s_t \in [0, 1]$, in contrast with the more general state description x_t. The state can be understood as a measure of the stored energy in the system and the state bounds are an abstraction of the thermal bounds and operational constraints, meaning that $s_t = 0$ indicates that no energy can be extracted from the asset without violating constraints, and $s_t = 1$ indicates that no energy can be inserted. We begin by making an assumption about the local controller of the considered asset.

Assumption 4. *For each* $s_t \in [0, 1]$*, the controller is able to satisfy the comfort/operational constraints for all* $t' > t$*. When receiving flexibility requests, the controller follows them as closely as possible. Furthermore, we assume to either receive state measurements from the controller or measurements from which we can construct a state-like variable.*

This abstract state might have different characterizations, one example, which will also

be used in the numerical experiments, is the following:

$$s_t := \frac{\underline{\Delta}_t}{\underline{\Delta}_t + \bar{\Delta}_t} \tag{4.8}$$

where $\underline{\Delta}_t$ is the maximum runtime of the equipment at minimum power, and $\bar{\Delta}_t$ is the maximum runtime of the equipment at maximum power, without violating constraints. These quantities can be obtained in different ways, e.g., assuming the controller has access to either a detailed or simplified temperature model of the building (the latter of which is used in the simulations). For a building with a HP, $\underline{\Delta}_t$ and $\bar{\Delta}_t$ can be approximated by

$$\underline{\Delta}_t = \frac{T_t - T_{\min}}{P_{\text{loss},t}},\ \bar{\Delta}_t = \frac{T_{\max} - T_t}{P_{\max} - P_{\text{loss},t}}$$

where T_t is the temperature at time t, the upper and lower temperature limits are given by $T_{\max}$ and $T_{\min}$ respectively, the thermal power capacity of the heat pump is denoted by $P_{\max}$, and the average losses at time t are given by $P_{\text{loss},t}$. This description is e.g. used in (Gasser et al., 2021, Eqn. 8). Note that by communicating the state instead of quantities like the temperature and corresponding temperature bounds, possibly sensitive information does not need to be shared. On the other hand, we also have the option to compute a state ourselves, when receiving data of the mentioned type. Furthermore, note that the state does not need to be computed as in (4.8) as long as it fulfills Assumption 4. Therefore, it is possible to derive an approximate state from, e.g., measurements of the historical energy consumption along with measurements of the external conditions, or possibly other types of measurements for varying types of assets.

With this, we assume a general state difference equation, in accordance with (4.1). To simplify the following derivations, we assume additive uncertainty and obtain a general description with the following form:

$$s_{t+1} - s_t = h\left(s_t, e_t, p_t\right) + \omega_t, \tag{4.9}$$

Considering Assumption 4, we have a nominal state trajectory $\boldsymbol{s}^{\text{b}}_{0:H-1}$, similar to $\boldsymbol{x}^{\text{b}}_{0:H-1}$, given by

$$s^{\text{b}}_{t+1} - s^{\text{b}}_t = h(s^{\text{b}}_t, e_t, p^{\text{b}}_t) + \omega_t, \tag{4.10}$$

with the baseline power injected given by p^{b}_t.

Remark 8. *Different methods exist to predict the baseline consumption of various assets, an overview of data-driven methods to predict energy consumption in buildings can be found in (Sun et al., 2020). In this thesis, we will use kernel methods, especially KRR for this baseline prediction in the experiments. We will therefore assume to have accurate baseline predictions in the theoretical developments where necessary. An extension to considering uncertainty quantification with the approaches developed in Chapter 2 is*

envisioned for future work.

Considering the difference of (4.9) and (4.10), we get

$$s_{t+1} - s_t = s^{\mathrm{b}}_{t+1} - s^{\mathrm{b}}_t + h(s_t, e_t, p_t) - h(s^{\mathrm{b}}_t, e_t, p^{\mathrm{b}}_t). \tag{4.11}$$

To specify the state evolution under the assumed controller behavior of keeping the state at its nominal level and while responding to relative consumption requests given by Definition 6, we can distinguish two phases in the system operation:

1. The request phase where $p_t = p^{\mathrm{b}}_t + r_t$.
2. The recovery phase where s_t is driven towards s^{b}_t, with the injected power denoted by $p_{\mathrm{con},t}$.

Note that the recovery phase is a distinct feature of assets that return to an equilibrium, e.g., thermal assets with a fixed temperature setpoint. Moreover, in the request phase, we can distinguish between receiving positive or negative relative consumption requests, due to equipment or controller characteristics. Assuming a linear approximation of h, for simplicity, around the nominal operation point, we get that

$$h(s_t, \boldsymbol{e}_t, p_t) - h(s^{\mathrm{b}}_t, \boldsymbol{e}_t, p^{\mathrm{b}}_t) \approx \begin{cases} a^+ r_t \text{ if } r_t > 0 \\ a^- r_t \text{ if } r_t < 0 \\ b_f (s_t - s^{\mathrm{b}}_t) \text{ if } r_t = 0 \end{cases}. \tag{4.12}$$

Due to the stochasticity of s_t, notice that a^+, a^- and b_f are in general stochastic.

We make a few further assumptions on the nominal state evolution and the coefficients a^+, a^-, and b_f that will ease the rest of the analysis.

Assumption 5. *In* (4.12)*, we assume that*

(a) The request-free nominal state evolution s^b_t can be well approximated by a function $f : \mathbb{R}^m \to \mathbb{R}$ of the τ past weather variables, denoted hereafter by $\boldsymbol{e}_{t-\tau+1:t} := [e_{t-\tau+1}, \ldots, e_t] \in \mathbb{R}^m$, $e_i \in \mathbb{R}^{n_e}$, $i = t-\tau+1, \ldots, t$, $m = \tau n_e$,

(b) $b_f \in \mathbb{R}$ is a constant,

(c) a^+ and a^- are real-valued random variables on a finite probability space.

Assumption 5a) states that the request-free state evolution can be well-captured by a deterministic function that only depends on past and current weather variables. n_e

denotes the number of measured variables, and τ denotes the number of considered time steps. Despite being strong, this modeling assumption for thermal systems (in particular building assets) often leads to good results in practice because errors do not accumulate. This is due to the system being a controlled one, where similar external conditions lead to similar controller actions and therefore a predictable indoor temperature range. Since we are always considering τ past weather variables, we use the notational shorthand of $\boldsymbol{e}_{:t}$ for $\boldsymbol{e}_{t-\tau+1:t}$. Note that this assumption could be replaced by modeling the nominal state with a GP instead to take uncertainty into account, at the price of complicating further the analysis. Assumption 5b) is justified by the fact that the coefficient b_f has little influence on the flexibility quantification discussed here, as is highlighted in the following parameter identification explanation. Finally, Assumption 5c) is useful to extract the distributions of a^+ and a^- directly from data. The random variable assumption also captures possibly random state behavior and will be helpful in the uncertainty quantification explained in Section 4.2.3. We finally end up with the following state equation:

Definition 9 (Battery Model)**.** *Let $r_t \in \mathbb{R}$ denote a relative consumption request at time t with respect to a baseline, and let $r_t^+ = \max(r_t, 0), r_t^- = \min(r_t, 0)$ be the positive and negative part of the request. With the external influences given by $\boldsymbol{e}_{:t} \in \mathbb{R}^m$ and a function $f : \mathbb{R}^m \to \mathbb{R}$ to approximate the nominal state s^b in request-free operation, we model the state evolution as*

$$\hat{s}_{t+1} = \hat{s}_t + a^+ r_t^+ + a^- r_t^- + b_f(f(\boldsymbol{e}_{:t}) - \hat{s}_t)\chi_{r_t} + f(\boldsymbol{e}_{:t+1}) - f(\boldsymbol{e}_{:t}) \tag{4.13}$$

with $\chi_r = \begin{cases} 1, & \text{if } r = 0 \\ 0, & \text{if } r \neq 0 \end{cases}$. The state change depends on a parameter $b_f \in \mathbb{R}$, while a^+ and a^- are assumed to be real-valued random variables on a finite probability space.

Note that the approximated state $\hat{s}_t$ given by the battery model is no longer bounded between 0 and 1. Furthermore, $\hat{s}_t$ taking a value smaller than 0 or larger than 1 corresponds to a situation where the true state reaches its boundaries and the building controller is not able to fulfill the request. We will use this observation to quantify feasible and infeasible request trajectories and finally determine flexibility envelopes through this.

Parameter identification

The learning of the battery model is a two-step approach. First, $f(\boldsymbol{e}_{:t})$ is learned from data obtained during the nominal operation of the building's controller. Then the parameter b_f and the sample spaces of a^+ and a^- can be identified from request periods, followed by recovery periods.

For the learning approach, we use the following formulation that describes the dependence of the predicted state $\hat{s}_k$ on the starting state s_0 and the applied requests $\boldsymbol{r}_{0:k-1}$.

Lemma 5. *For a given state s_0 and a request trajectory $\boldsymbol{r} \in \mathbb{R}^k$, the state $\hat{s}_k$ is given by*

$$\hat{s}_k = (1-b_f)^{q_0^k} s_0 + \sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} (f(\boldsymbol{e}_{:l}) b_f \chi_{r_l} + a^+ r_l^+ + a^- r_l^- + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})), \quad (4.14)$$

with $q_l^k = \sum_{i=l}^{k-1} \chi_{r_i}$.

Proof. See Appendix 4.6.1. □

For identifying the sample spaces of a^+ and a^-, we consider request sequences $\boldsymbol{r}$, either strictly positive or strictly negative respectively (i.e. $r_i > 0$ or $r_i < 0, i = 0, \dots, k-1$). $\boldsymbol{r}$ is assumed to be followed by a request-free period, and we denote the corresponding state trajectory by $\boldsymbol{s}_{0:k}$. We either have that the requests are fulfillable, i.e. $0 < s_i < 1, i = 0, \dots, k$ which we denote by setting an index $l = k+1$, or not fulfillable at a certain point l, with $l = \arg\min q$ s.t. $s_q = 0$ or $s_q = 1$. Assuming a state evolution as given by (4.14), we have

$$s_{l-1} = s_0 + \sum_{i=0}^{l-2} a^{+/-} r_i + f(\boldsymbol{e}_{:l-1}) - f(\boldsymbol{e}_{:0}). \quad (4.15)$$

Therefore, a sample takes the form $a^{+/-} = (s_{l-1} - f(\boldsymbol{e}_{:l-1}) - (s_0 - f(\boldsymbol{e}_{:0})))/\sum_{i=0}^{l-2} r_i$.

This way of identifying the samples assumes constant a^+, a^- during a given request period that is often of at most a few hours. However, it takes into account potential value changes of a^+, a^- between two distinct request periods.

Remark 9. *The sample identification for a^+ and a^- as presented here assumes either measurements of the state or measurements of other quantities that allow the construction of a state, in line with Assumption 4. Like this, the sample identification works with any dataset with historical data of consumption requests. Note that by running dedicated identification experiments, state measurements could be made obsolete by sending requests that drive the assets to their lower and upper state bound alternatingly, and observing state saturation through the non-fulfillment of the sent request.*

To identify b_f, we consider sequences $\boldsymbol{s}_{0:k}$ that occur after a request period, so that $r_{-1} \neq 0$, $\boldsymbol{r}_{0:k-1} = \boldsymbol{0}$ and $r_k \neq 0$. Furthermore, we only use data from the recovery periods that fulfill $|s_t - f(\boldsymbol{e}_{:t})| > \delta$ for some threshold $\delta \in \mathbb{R}_+$, for identifying b_f, to capture the controller based recovery period and not small perturbations due to model mismatch.

As in the previous cases, we either have that $|s_i - f(\boldsymbol{e}_{:i})| > \delta, i = 0, \dots, k$ (thus $l = k+1$) or determine l as $l = \arg\min q$ s.t. $|s_q - f(\boldsymbol{e}_{:q})| \leq \delta$. Using the evolution of the battery

model from (4.14) for request-free periods, we can formulate the following least-squares problem, for each recovery period $j = 1, \dots, n$.

$$b_j = \arg\min_b \Big((1-b)^{l-1}(s_0 - f(\boldsymbol{e}_{:0})) + f(\boldsymbol{e}_{:l-1}) - s_{l-1} \Big)^2 \tag{4.16}$$

b_f is then chosen as $\max_j b_j$, but other choices like the average over the b_j's are also possible. Dealing with a fixed b_f is deliberate because treating b_f as stochastic and following through with the approach outlined in Section 4.2.3 introduces combinatorial issues and nonlinearities in the uncertainty set computation while having a minimal impact on the flexibility envelope computation, due to its influence in the recovery periods only.

In the following, we denote the finite sample spaces of a^+ and a^- as $\mathcal{P}^+, \mathcal{P}^-$ with $|\mathcal{P}^+| = n_1, |\mathcal{P}^-| = n_2$. Furthermore, we assume an ordering, such that $\mathcal{P}^+ = \{a_1^+, \dots, a_{n_1}^+ : a_i^+ \leq a_j^+ \text{ if } i < j\}, \mathcal{P}^- = \{a_1^-, \dots, a_{n_2}^- : a_i^- \leq a_j^- \text{ if } i < j\}$, which will be helpful in Section 4.2.4. Since these data are the only information we have about a^+ and a^-, it is natural to use them for constructing the sample spaces and therefore having finite sample spaces.

4.2.3 Uncertainty quantification

In this section, we consider the feasibility of request trajectories through the containment in a set with probabilistic constraints. Through the use of risk measures, a tightened version of this set is derived that is defined by deterministic constraints over a robust uncertainty set. This new set formulation is then used to derive data-driven and uncertainty-aware versions of the introduced flexibility envelopes.

In flexibility scenarios, the battery model (4.13) is used to determine the feasibility of request trajectories for building assets. As already stated, $\hat{s}_t$ taking a value smaller than 0 or larger than 1 corresponds to a situation where the true state saturates at its boundaries and the building controller is not able to fulfill the relative consumption request without violating constraints. This gives rise to the definition of the set of feasible request trajectories.

Definition 10. *Let $k \in \mathbb{N}$ be the specified trajectory length, $\tilde{\boldsymbol{e}}_{0:k,\tau} = [\boldsymbol{e}_{:0}, \dots, \boldsymbol{e}_{:k}]$ the trajectory of external conditions, s_0 the starting state of the asset, and $\alpha \in (0,1)$ the desired probability level. Then we define the set of feasible request trajectories of length k as*

$$\mathcal{R}_k^\alpha(s_0) = \Big\{ \boldsymbol{r}_{0:k-1} \in \mathbb{R}^k : \mathbb{P}\Big\{ \mathbf{0} \leq [\hat{s}_0, \cdots, \hat{s}_k]^\top \leq \mathbf{1} \Big\} \geq 1 - \alpha, \hat{s}_0 = s_0 \Big\}, \tag{4.17}$$

where the state dynamics are given by the battery model (4.13)

$$\hat{s}_{t+1} = \hat{s}_t + a^+ r_t^+ + a^- r_t^- + b_f(f(\boldsymbol{e}_{:t}) - \hat{s}_t)\chi_{r_t} + f(\boldsymbol{e}_{:t+1}) - f(\boldsymbol{e}_{:t}), t = 0, \dots, k-1$$

and the probability is taken element-wise.

The goal is to test the containment of request trajectories, that make up the different flexibility envelopes, in that set. To facilitate this, we propose a robust version of (4.17) with tightened constraints through the use of risk measures, thus avoiding the probabilistic state constraints.

A number of reformulations are needed for the risk measure approach. We will present them in the following, starting by rewriting (4.14) in the two following ways:

$$\hat{s}_k = c_k + [\boldsymbol{a}_k^+, \boldsymbol{a}_k^-] \begin{pmatrix} \boldsymbol{r}_k^+ \\ \boldsymbol{r}_k^- \end{pmatrix} \tag{4.18}$$

$$= c_k + \boldsymbol{R}_k \begin{pmatrix} a^+ \\ a^- \end{pmatrix}, \tag{4.19}$$

with $\boldsymbol{r}_k^+ = [r_0^+, \cdots, r_{k-1}^+]^\top, \boldsymbol{r}_k^- = [r_0^-, \cdots, r_{k-1}^-]^\top \in \mathbb{R}^k$, where

$$c_k = (1-b_f)^{q_0^k} s_0 + \sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} (f(\boldsymbol{e}_{:l}) b_f \chi_{r_l} + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})) \tag{4.20}$$

groups all the non-request parts and

$$\boldsymbol{a}_k^+ = [(1-b_f)^{q_1^k}, \dots, (1-b_f)^{q_k^k}] a^+ \in \mathbb{R}^k \tag{4.21}$$

$$\boldsymbol{a}_k^- = [(1-b_f)^{q_1^k}, \dots, (1-b_f)^{q_k^k}] a^- \in \mathbb{R}^k \tag{4.22}$$

$$\boldsymbol{R}_k = \left[\sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} r_l^+, \sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} r_l^- \right] \in \mathbb{R}^2 \tag{4.23}$$

group the request parts either depending on a^+, a^-, or the request trajectory, and we recall that $q_l^k = \sum_{i=l}^{k-1} \chi_{r_i}$. We can then alternatively write the set of feasible requests as

$$\mathcal{R}_k^\alpha(s_0) = \bigcap_{l=0}^{k} \left\{ \boldsymbol{r} \in \mathbb{R}^k : \mathbb{P}\left\{ \boldsymbol{b}_l \leq \boldsymbol{A}_l \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \right\} \geq 1 - \alpha \right\}, \tag{4.24}$$

with $\boldsymbol{b}_l = [-c_l, c_l - 1]^\top \in \mathbb{R}^2$ and $\boldsymbol{A}_l = \begin{pmatrix} \boldsymbol{a}_l^+ & \boldsymbol{a}_l^- \\ -\boldsymbol{a}_l^+ & -\boldsymbol{a}_l^- \end{pmatrix} \in \mathbb{R}^{2 \times 2l}$, by using (4.18).

The steps of the different reformulations are outlined in more detail in Appendix 4.6.2. In the following, we will consider a single set from the intersection in (4.24) and denote

$[\boldsymbol{a}_l^+, \boldsymbol{a}_l^-]$ as $\boldsymbol{a}_l$. From Assumption 5c), we have that the unknown $\boldsymbol{a}_l$ is a $\mathbb{R}^{2l}$-valued random variable on a finite probability space $(\Omega, \mathcal{F}, \mathbb{P})$ with $|\Omega| = N, \mathcal{F} = 2^{\Omega}$. We can construct the support by combining all possible a^+, a^- from the identified sets $\mathcal{P}^+, \mathcal{P}^-$. The sample set for $\boldsymbol{a}_l$ is denoted by $\mathcal{A}_l = \{\boldsymbol{a}_{l,1}, \dots, \boldsymbol{a}_{l,N}\}$ with $|\mathcal{A}_l| = n_1 n_2 =: N$. The data matrix is denoted as $\boldsymbol{D}_l = [\boldsymbol{a}_{l,1}^\top, \dots, \boldsymbol{a}_{l,N}^\top] \in \mathbb{R}^{2l \times N}$.

On the one hand, having $\boldsymbol{a}_l$ as a random variable on a finite probability space is restrictive, since the true sample space Ω might be larger or even continuous. On the other hand, since data is the only knowledge we have about $\boldsymbol{a}_l$, this assumption is aligned with the data-driven approach, and useful in practice (see (Bertsimas & Brown, 2009, Assumption 3.1)).

Robustness through risk measures

Utilizing the new formulation of the set of feasible trajectories (4.24), we will now exploit a specific risk measure, the Conditional Value at Risk (CVaR), as a way to specify how the uncertainty is dealt with. Concretely, we will consider user preferences to trade off conservativeness and the size of the feasible set, relying on results from (Bertsimas & Brown, 2009). The presented derivations are not unique for CVaR but also hold for general coherent risk measures. However, using CVaR, together with two straightforward assumptions, gives us a directly usable, tightened version of the set of feasible requests, which is why we focus our discussions on this specific risk measure.

Here, we will only present the main concepts necessary for our specific approach. For some additional insight, the reader is referred to e.g. (Delbaen, 2002) or (Uryasev, 2000).

Definition 11 (Conditional Value at Risk)**.** *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a finite probability space with $\Omega = \{\omega_1, \dots, \omega_N\}$ and let $\mathcal{X}$ be a linear space of random variables on Ω. We define the CVaR for $X \in \mathcal{X}$ with probability level α as*

$$\mathrm{CVaR}_\alpha(X) = \max_{\boldsymbol{q} \in \mathcal{Q}} \sum_{i=1}^{N} -q_i X(\omega_i), \tag{4.25}$$

with $\mathcal{Q}$ its family of generating measures, given by $\{\boldsymbol{q} \in \Delta^N : q_i \leq \frac{\mathbb{P}(\omega_i)}{\alpha}\}$.

An intuition about the meaning of CVaR can be drawn from its continuous probability space definition for atomless distributions (this intuition is inexact in the finite case, but nevertheless helpful). If we consider a constraint $\boldsymbol{a}^\top \boldsymbol{x} \geq b$ for a random variable $\boldsymbol{a} \in \mathbb{R}^l$, then $\mathrm{CVaR}_\alpha(\boldsymbol{a}^\top \boldsymbol{x} - b)$ gives the expected constraint violation in the α-% worst cases. This motivates the use of the risk-aversion constraint $\mathrm{CVaR}_\alpha(\boldsymbol{a}^\top \boldsymbol{x} - b) \leq 0$. Note that this constraint implies both constraint satisfaction in expectation and constraint satisfaction with probability $\geq 1 - \alpha$.

We will apply risk aversion constraints to the individual probabilistic constraints in (4.24)

and utilize the reformulation with robust uncertainty sets presented in (Bertsimas & Brown, 2009, Thm. 3.1). This is possible since CVaR is a coherent risk measure (i.e. it fulfills the properties of monotonicity, translation invariance, convexity, and positive homogeneity). Using risk aversion constraints instead of probabilistic constraints leads to a smaller feasible set for the same uncertainty level α since the former represent a tightened version of the latter.

Theorem 4. *We have*

$$\left\{ \boldsymbol{r} \in \mathbb{R}^k : \mathrm{CVaR}_\alpha \left(\boldsymbol{A}_l \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} - \boldsymbol{b}_l \right) \leq 0 \right\} \tag{4.26}$$

$$= \left\{ \boldsymbol{r} \in \mathbb{R}^k : [\boldsymbol{a}, -\boldsymbol{a}]^\top \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \geq \boldsymbol{b}_l \quad \forall \boldsymbol{a} \in \mathcal{U}_\alpha^l \right\}, \tag{4.27}$$

with a slight abuse of notation for the constraint-wise CVaR application, where $\mathcal{U}_\alpha^l = \mathrm{conv}(\{\boldsymbol{D}_l \boldsymbol{q} : \boldsymbol{q} \in \mathcal{Q}\})$*, and we recall that* $\boldsymbol{D}_l$ *is the data matrix, and* $\mathcal{Q}$ *the family of generating measures for* CVaR_α*.*

Proof. We can write the left-hand side as

$$\left\{ \boldsymbol{r} \in \mathbb{R}^k : \mathrm{CVaR}_\alpha \left(\boldsymbol{a}_l \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} + c_l \right) \leq 0 \right\} \tag{4.28}$$

$$\cap \left\{ \boldsymbol{r} \in \mathbb{R}^k : \mathrm{CVaR}_\alpha \left(-\boldsymbol{a}_l \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} - c_l + 1 \right) \leq 0 \right\}. \tag{4.29}$$

Using the robust uncertainty set reformulation from (Bertsimas & Brown, 2009) Theorem 3.1. for both sets in the intersection, we get

$$\left\{ \boldsymbol{r} \in \mathbb{R}^k : \boldsymbol{a}^\top \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \geq -c_l \quad \forall \boldsymbol{a} \in \mathcal{U}_\alpha^l \right\} \tag{4.30}$$

$$\cap \left\{ \boldsymbol{r} \in \mathbb{R}^k : -\boldsymbol{a}^\top \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \geq c_l - 1 \quad \forall \boldsymbol{a} \in \mathcal{U}_\alpha^l \right\}. \tag{4.31}$$

Since the same uncertainty sets are used, we can combine them in the form of (4.27). □

Theorem 4 is not limited to CVaR, but holds for general coherent risk measures. It provides a closed-form description of the set of request trajectories that fulfill the risk aversion constraint, by taking those that are robustly feasible for the uncertainty set $\mathcal{U}_\alpha^l$. For the feasibility of a given $\boldsymbol{r} \in \mathbb{R}^k$, this implies checking constraint satisfaction for all $\boldsymbol{a} \in \mathcal{U}_\alpha^l$ and $l = 1, \ldots, k+1$.

From Definition 11, we can directly observe the uncertainty set construction as in

Theorem 4 for CVaR_α, namely $\mathcal{U}_\alpha^l = \text{conv}(\{\boldsymbol{D}_l \boldsymbol{q} : \boldsymbol{q} \in \Delta^N, q_i \leq \frac{\mathbb{P}(\boldsymbol{a}_{l,i})}{\alpha}\})$, with the data matrix $\boldsymbol{D}_l$. The following theorem states a more practical form of this uncertainty set under certain assumptions.

Theorem 5. *Let the probabilities on the finite probability space be uniform (i.e. $\mathbb{P}(\boldsymbol{a}_{l,i}) = \frac{1}{N}$), and α chosen as $\frac{j}{N}$ for some $j \in \{1, \dots, N\}$. Then $\mathcal{U}_\alpha^l$ is the convex hull of all j-point averages in $\mathcal{A}_l$, i.e.*

$$\mathcal{U}_\alpha^l = \text{conv}\left(\left\{\frac{1}{j}\sum_{i \in J} \boldsymbol{a}_{l,i} : J \subset \{1, \dots, N\}, |J| = j\right\}\right). \tag{4.32}$$

Proof. The first direction, i.e. "$\supseteq$", follows directly from the definition of the family of generating measures.

For "$\subseteq$", pick an arbitrary $\boldsymbol{a} \in \mathcal{U}_\alpha^l$ and observe that we can write it as $\boldsymbol{a} = \boldsymbol{D}_l \boldsymbol{q}$ for some $\boldsymbol{q} \in \{\boldsymbol{q} \in \Delta^N : q_i \leq \frac{1}{j} \forall i\}$. Furthermore, we have $\text{conv}(Q) = \{\boldsymbol{q} \in \Delta^N : q_i \leq \frac{1}{j} \forall i\}$ for $Q := \{\boldsymbol{q} \in \mathbb{R}^N : q_i = \frac{1}{j} \forall i \in J, q_i = 0 \forall i \notin J, J \subset \{1, \dots, N\}, |J| = j\}$. Since each element $\bar{\boldsymbol{a}} = \boldsymbol{D}_l \bar{\boldsymbol{q}}$ with $\bar{\boldsymbol{q}} \in \text{conv}(Q)$ is in the right-hand-side, we conclude that $\boldsymbol{a}$ is an element of the right-hand-side. □

These uncertainty sets given by the j-point averages are the sets we will focus on in the following, for two reasons: Firstly, we do not consider a weighting of the samples, which makes the choice of uniform probabilities natural. Secondly, for N large enough, the choice of α as $\frac{j}{N}$ offers a fine discretization, while also providing a straightforward way of computing the uncertainty set. The advantage of computability, therefore, outweighs the limitation of choice through discretization.

We can then formulate the tightened set of feasible request trajectories, based on the CVaR uncertainty sets as

$$\mathcal{C}_k^\alpha(s_0) = \bigcap_{l=0}^{k} \left\{\boldsymbol{r} \in \mathbb{R}^k : [\boldsymbol{a}, -\boldsymbol{a}]^\top \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \geq \boldsymbol{b}_l \quad \forall \boldsymbol{a} \in \mathcal{U}_\alpha^l \right\}. \tag{4.33}$$

4.2.4 Flexibility estimation

To predict the available flexibility in the form of flexibility envelopes according to definitions 5, 7, and 8, we test the containment of request trajectories that make up the entries of the envelopes in $\mathcal{C}_k^\alpha(s_0)$. For predicting future flexibility, we assume the asset to be in its predicted nominal state, given by $[f(\boldsymbol{e}_{:0}), \dots, f(\boldsymbol{e}_{:k-1})]$, based on a forecast of the external conditions $\tilde{\boldsymbol{e}}_{0:k-1,\tau}$. Thus, we get the following uncertainty-aware flexibility envelope predictions:

Corollary 1 (Uncertainty-aware flexibility envelope)**.** *Let $\alpha \in (0, 1]$ be an uncertainty parameter that can be freely chosen. Given the trajectory of external conditions $\tilde{\boldsymbol{e}}_{0:H-1,\tau}$ or a forecast thereof and a vector of n_p discrete power levels $\boldsymbol{p} = [p_0, \dots, p_{n_p-1}] \in [\underline{p}, \bar{p}]^{n_p}$ with $\underline{p} \leq p_0 < \cdots < p_{n_p} \leq \bar{p}$, the uncertainty-aware flexibility envelope $\tilde{\boldsymbol{E}}^{\alpha} \in \mathbb{R}^{n_p \times H}$ is given by*

$$\begin{aligned} \tilde{\boldsymbol{E}}^{\alpha}_{i,t} = \max \quad & T \\ s.t. \quad & \boldsymbol{r}_{0:T-1} \in \mathcal{C}^{\alpha}_{T}(f(\boldsymbol{e}_{:t})) \\ & r_l = p_i - p^b_{t+l}, \quad l = 0, \dots, T-1 \end{aligned} \tag{4.34}$$

using the baseline consumption predictions p^b_t.

Similarly, we can derive uncertainty-aware relative flexibility envelopes, by considering requests of constant relative power.

Corollary 2 (Uncertainty-aware relative flexibility envelope)**.** *Let $\alpha \in (0, 1]$ be an uncertainty parameter that can be freely chosen. Given the trajectory of external conditions $\tilde{\boldsymbol{e}}_{0:H-1,\tau}$ or a forecast thereof and a vector of n_p discrete power levels $\boldsymbol{r} = [r_0, \dots, r_{n_p-1}] \in [\underline{p}, \bar{p}]^{n_p}$ with $\underline{p} \leq r_0 < \cdots < r_{n_p} \leq \bar{p}$, the uncertainty-aware relative flexibility envelope $\tilde{\boldsymbol{F}}^{\alpha} \in \mathbb{R}^{n_p \times H}$ is given by*

$$\begin{aligned} \tilde{\boldsymbol{F}}^{\alpha}_{i,t} = \max \quad & T \\ s.t. \quad & \mathbf{1}_T r_i \in \mathcal{C}^{\alpha}_{T}(f(\boldsymbol{e}_{:t})) \\ & \underline{p} \leq p^b_l + r_i \leq \bar{p}, \quad l = t, \dots, t+T-1 \end{aligned} \tag{4.35}$$

using the baseline consumption predictions p^b_t and where $\mathbf{1}_T$ denotes a vector of ones of length T.

In the experimental prediction of available flexibility, we will limit the maximum time T to 24h for each entry of the flexibility envelopes. In line with Corollaries 1 and 2, we can also determine the uncertainty-aware version for predicting fixed-time flexibility envelopes:

Corollary 3 (Uncertainty-aware fixed-time flexibility envelope)**.** *Let $\alpha \in (0, 1]$ be an uncertainty parameter that can be freely chosen and $k \in \mathbb{N}$ the maximum number of active timesteps for the asset. Given the trajectory of external conditions $\tilde{\boldsymbol{e}}_{0:H-1,\tau}$ or a forecast thereof, the uncertainty-aware fixed-time flexibility envelope is given by $\tilde{\boldsymbol{R}}^{\alpha,k}_{0:H-1} = [\underline{\boldsymbol{R}}^{\alpha,k}_{0:H-1}, \bar{\boldsymbol{R}}^{\alpha,k}_{0:H-1}]$ where*

$$\begin{aligned} \underline{\boldsymbol{R}}^{\alpha,k}_{t} = \min \quad & r \\ s.t. \quad & \mathbf{1}_k r \in \mathcal{C}^{\alpha}_{k}(f(\boldsymbol{e}_{:t})) \\ & \underline{p} \leq p^b_l + r \leq \bar{p}, \quad l = t, \dots, t+k-1 \end{aligned} \tag{4.36}$$

$$\begin{aligned} \bar{\boldsymbol{R}}_t^{\alpha,k} = \max \quad & r \\ s.t. \quad & \mathbf{1}_k r \in \mathcal{C}_k^{\alpha}(f(\boldsymbol{e}_{:t})) \\ & \underline{p} \leq p_l^b + r \leq \bar{p}, \quad l = t, \dots, t+k-1 \end{aligned} \tag{4.37}$$

using the baseline consumption predictions p_t^b and where $\mathbf{1}_k$ denotes a vector of ones of length k.

Note that for the computation of all the different types of uncertainty-aware flexibility envelopes, the set $\mathcal{C}_k^{\alpha}$ does not need to be computed explicitly, instead the constraints are checked for specific request trajectories. A way of doing this more efficiently is presented next.

Envelope computation

In the experiments, we will focus on flexibility envelopes specified by relative requests. To make the computation of these types of flexibility envelopes more efficient, we will use some observations based on the features of relative consumption requests that allow us to avoid specifying the whole robust uncertainty sets in (4.33). Furthermore, we will focus on choices of α as $\frac{j}{N}$ for some $j = 1, \dots, N$ due to the uncertainty set formulation in Theorem 5.

Due to the linearity of the $\boldsymbol{a}_l^{+/-}$ from (4.18) in $a^{+/-}$, we can consider the j-point averages of the (a^+, a^-) pairs in $\mathcal{P}^+ \times \mathcal{P}^-$, instead of their induced $\boldsymbol{a}_l$ samples, and test feasibility for (4.19) because of its equivalence to (4.18). We denote this set of j-point averages by $\mathcal{P}_j = \{\frac{1}{j}\sum_{i=1}^{j} \boldsymbol{a}_i : \boldsymbol{a}_i \in \mathcal{P}^+ \times \mathcal{P}^-, \boldsymbol{a}_l \neq \boldsymbol{a}_k \text{ for } l \neq k\}$.

However, since the relative requests are constant for each entry of the two types of flexibility envelopes considered here, either only a^+ or a^- has to be considered regarding the feasibility problem, depending on the sign of the request. Therefore, we can restrict ourselves to testing feasibility for the worst-case parameters in $\mathcal{P}_j$. Since the sets $\mathcal{P}^+$ and $\mathcal{P}^-$ are increasingly ordered, these parameters are given by $(a^+_{\max,j}, a^-_{\max,j})$ with

$$a^+_{\max,j} = \frac{1}{j}\left(n_2 \sum_{i=0}^{\lfloor \frac{j}{n_2} \rfloor} a^+_{n_1-i} + (j \bmod n_2) a^+_{n_1-\lfloor \frac{j}{n_2} \rfloor - 1} \right), \tag{4.38}$$

$$a^-_{\max,j} = \frac{1}{j}\left(n_1 \sum_{i=0}^{\lfloor \frac{j}{n_1} \rfloor} a^-_{n_2-i} + (j \bmod n_1) a^-_{n_2-\lfloor \frac{j}{n_1} \rfloor - 1} \right). \tag{4.39}$$

$a^+_{\max,j}$ represents the largest attainable value in the first element of a tuple, which is

constructed as a j-point average of elements in $\mathcal{P}^+ \times \mathcal{P}^-$. Similarly, $a^-_{\max,j}$ represents the largest value in the second element. Additional information on the derivation of the two is provided in Appendix 4.6.3.

The uncertainty-aware fixed-time flexibility envelopes can then be computed in the following way:

Corollary 4. *Let $\alpha = \frac{j}{N}$ for a $j \in \{1, \dots, N\}$. Then*

$$\underline{\boldsymbol{R}}_t^{\alpha,k} = \max \left\{ \max_{l=1,\dots,k} \frac{-f(\boldsymbol{e}_{:t+l})}{a^-_{\max,j} l}, \max_{l=0,\dots,k-1} \underline{p} - p^b_{t+l} \right\} \tag{4.40}$$

$$\bar{\boldsymbol{R}}_t^{\alpha,k} = \min \left\{ \min_{l=1,\dots,k} \frac{1 - f(\boldsymbol{e}_{:t+l})}{a^+_{\max,j} l}, \min_{l=0,\dots,k-1} \bar{p} - p^b_{t+l} \right\} \tag{4.41}$$

Proof. We will only focus on the reformulation of $\underline{\boldsymbol{R}}_t^{\alpha,k}$, the result for $\bar{\boldsymbol{R}}_t^{\alpha,k}$ follows with the same arguments.

Due to the equivalence of (4.18) and (4.19), we can expand the constraints in $\mathcal{C}_k^\alpha(s_0)$ using the unrolled state equation. Since the baseline consumption does not lead to a constraint violation, according to Assumption 4, $\underline{\boldsymbol{R}}_t^{\alpha,k} \leq 0$ and we only need to consider the request terms associated to a^-. Therefore, we have

$$\begin{aligned} \underline{\boldsymbol{R}}_t^{\alpha,k} = \min \quad & r \\ \text{s.t.} \quad & \mathbf{0} \leq \begin{pmatrix} 1 \\ \vdots \\ k \end{pmatrix} a^- r + \begin{pmatrix} f(\boldsymbol{e}_{:t+1}) \\ \vdots \\ f(\boldsymbol{e}_{:t+k}) \end{pmatrix} \leq \mathbf{1} \quad \forall (a^+, a^-) \in \mathcal{P}_j \\ & \underline{p} \leq p^{\mathrm{b}}_{l+t} + r \leq \bar{p}, \quad l = 0, \dots, k-1. \end{aligned} \tag{4.42}$$

Due to the same argument, we only need to consider the lower bounds in (4.42). This leads to

$$\begin{aligned} & \begin{pmatrix} -f(\boldsymbol{e}_{:t+1})/a^- \\ \vdots \\ -f(\boldsymbol{e}_{:t+k})/(ka^-) \end{pmatrix} \leq \mathbf{1} r \quad \forall (a^+, a^-) \in \mathcal{P}_j \\ & \underline{p} - p^{\mathrm{b}}_{l+t} \leq r, \quad l = 0, \dots, k-1. \end{aligned} \tag{4.43}$$

The left-hand side of (4.43) is maximized for $a^-_{\max,j}$. We can then determine the value of

r, and therefore $\underline{\boldsymbol{R}}_t^{\alpha,k}$, as the maximum over all constraints, so

$$\underline{\boldsymbol{R}}_t^{\alpha,k} = \max\left\{\max_{l=1,\dots,k} \frac{-f(\boldsymbol{e}_{:t+l})}{a^-_{\max,j} l}, \max_{l=0,\dots,k-1} \underline{p} - p^{\mathrm{b}}_{t+l}\right\}$$

which proves the claim. □

Using Corollary 4, the uncertainty-aware fixed-time flexibility envelope is computed with complexity $\mathcal{O}(kH)$, by taking the maximum over k numbers twice in (4.40) and (4.41) over H timesteps.

Experimental evaluation

Before using flexibility envelopes in the coordination of a pool of assets, we first demonstrate the prediction of relative flexibility envelopes for single assets. The SimpleHouseRad-v0 model from the simulation model library Energym presented in Chapter 3 is used as a flexible asset. The building model is controlled by a PI controller that measures and reports the state introduced in (4.8) based on a simplified model of the building. This PI controller is also used to track the request trajectories as closely as possible without violating the temperature bounds of 19 and 24 °C, while otherwise following a setpoint temperature of 21 °C.

Data collection for constructing the battery model is performed during the first 6 weeks of a year, using measurements of the external temperature and irradiance to construct the feature variable e_t, from the city of Basel, Switzerland. In the first three weeks, no requests are sent to the building, such that state data under nominal controller operation is collected for learning the nonlinear model $f(\boldsymbol{e}_{:t})$, using a KRR model with squared exponential kernel. During the second three weeks, random constant requests are sent to the building for random durations between 1 hour and 4 hours, alternating with request-free periods of 4 to 15 hours. Since the control input is the heat pump power fraction, we consider relative input requests instead of relative power requests. This data collection resulted in a total of 22 samples for a^+ and 20 samples for a^-, giving $N = 440$ as the size of the discrete sample space. Note that this enables α to be quantized with a precision of approximately 0.2%, corresponding to quantization levels of around 0.002.

We compute the relative flexibility envelopes of 10 days, starting from the 22nd of January, for a weather file from Lausanne, Switzerland. This second weather file allows us to cover the same time period in the evaluation as in training while making sure to not have exactly the same weather conditions. Different values of the uncertainty parameter α are used, and we compare the results with the true available flexibility. These true relative flexibility envelopes represent the perfect but unattainable flexibility predictions and are computed by running the relative requests on the simulation model itself and observing

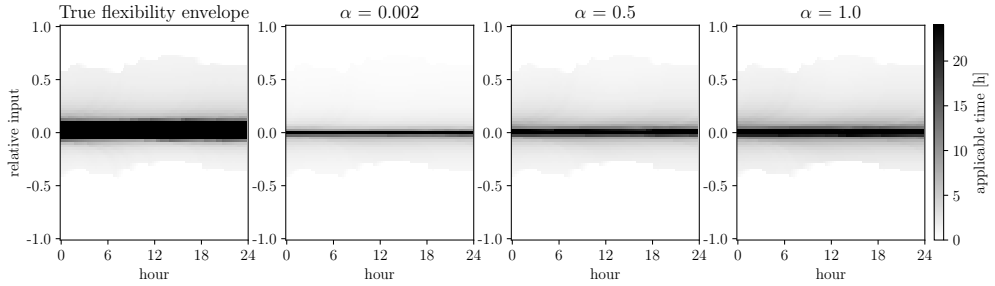


Figure 4.1 – Relative flexibility envelopes for day 5 of the test data. First: True flexibility envelope. Second to fourth: Flexibility envelope predictions for different α values.

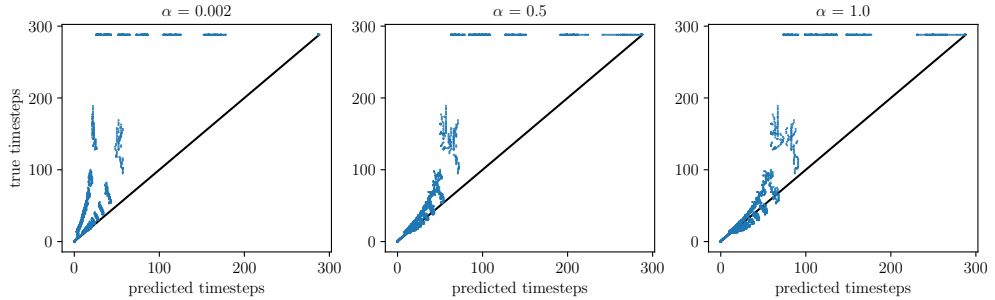


Figure 4.2 – Predicted vs. truly available timesteps in the relative flexibility envelopes for different values of α on day 5 of the test data.

violations of the temperature bounds.

An example of this evaluation is given in Fig. 4.1 for day 5 of the chosen 10 days in the test data. For how long a given relative input request (y-axis) can be held without violating constraints if started at a specific time (x-axis), is given by the shading in the plots. A decrease in conservativeness is observable for an increase in α.

The pointwise predicted availability (in number of timesteps) vs. the true availability of the requests, is shown in Fig. 4.2. Points that lie on the diagonal or slightly above are desirable, while points that are below the diagonal represent predictions that are more optimistic than the actual availability (and are therefore infeasible). For this specific day, we observe about 0.2% infeasible predictions for $\alpha = \frac{1}{440} \approx 0.002$, while for $\alpha = 0.5$ and $\alpha = 1.0$ the infeasible predictions make up about 5.1% and 10.7% respectively.

We get the following results regarding the percentage of infeasible predictions and mean absolute prediction error, displayed in Fig. 4.3. The percentage of infeasible predictions, over the course of the 10 days, is at about 0.16% for $\alpha = \frac{1}{440}$, and it rises up to about 6.09% for $\alpha = 1.0$. On the other hand, the mean absolute prediction error decreases from about 28 timesteps for $\alpha = \frac{1}{440}$, to about 15 for $\alpha = 1.0$, indicating a tradeoff between conservativeness and prediction error.

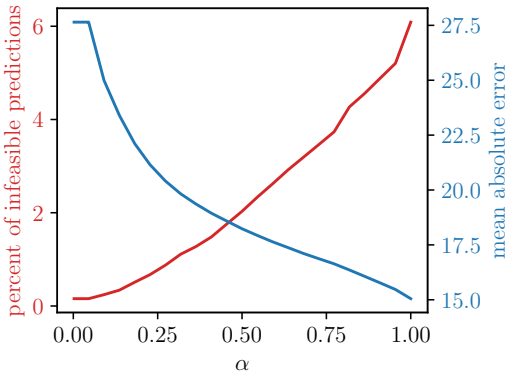


Figure 4.3 – Percentage of infeasible prediction vs mean absolute prediction error of the relative flexibility envelopes for different values of α over 10 days.

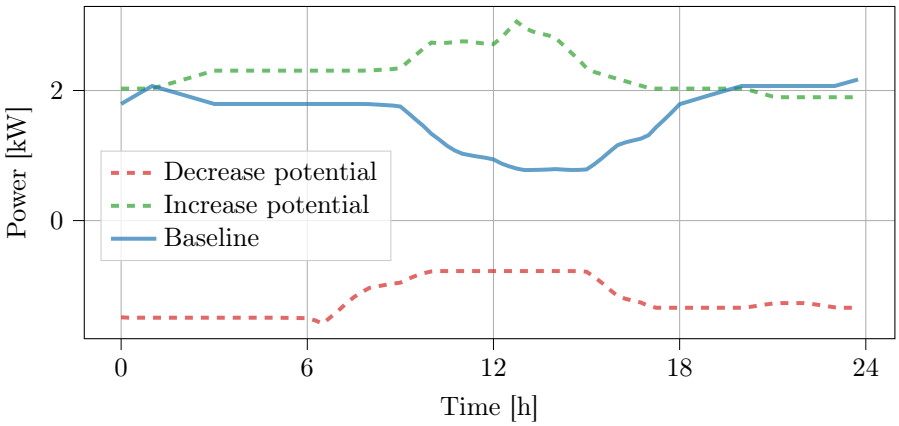


Figure 4.4 – Power increase (green dotted line) and decrease (red dotted line) potential for a duration of 3h with respect to a baseline consumption (blue line), computed with $\alpha = 1$.

We furthermore show the prediction of the other two types of flexibility envelopes, starting with the fixed-time flexibility envelopes. For this, we use the same battery model as described above, and compute the maximum increase and decrease potential with respect to a predicted baseline power. The prediction is done for a horizon of 24 hours with $\alpha = 1$, and we consider increases and decreases in power that can be held for up to 3 hours or $k = 36$ timesteps. An example of the resulting fixed-time flexibility envelope is shown in Figure 4.4.

Lastly, we demonstrate the computation of flexibility envelopes with absolute requests, in line with the example displayed in Figure 4.1. The resulting flexibility envelopes for different α values in comparison to the true flexibility envelope are displayed in Figure 4.5. Again, we observe increasing conservativeness for decreasing α, hinting at the tradeoff between conservativeness and infeasible predictions.

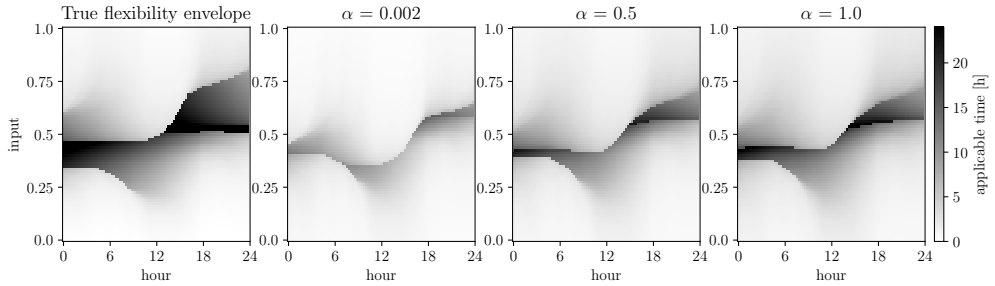


Figure 4.5 – Flexibility envelopes for day 3 of the test data. First: True flexibility envelope. Second to fourth: Flexibility envelope predictions for different α values.

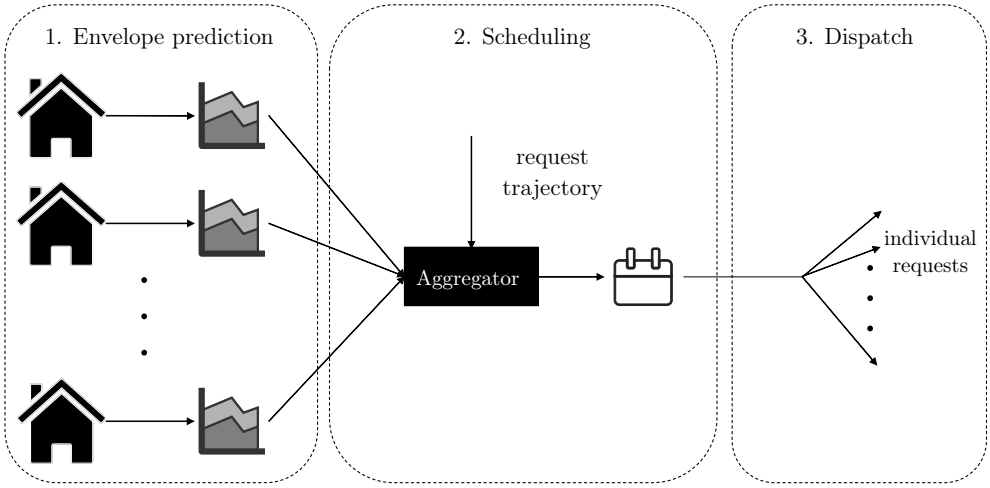


Figure 4.6 – The three phases of our proposed approach.

This tradeoff, together with the incentives for providing flexibility and penalties for not being able to provide the promised flexibility, can inform the selection of an uncertainty parameter α to be used in a flexibility scheme.

4.3 Coordinated scheduling and dispatch of flexibility in buildings

With the flexibility characterizations at hand, we will now turn to the problem of coordinating a pool of assets. For this, we will use the uncertainty-aware fixed-time flexibility envelope predictions of individual assets in a scheduling problem, formulated as a MILP, to determine the activation time of the individual assets. With this activation plan, incoming aggregated consumption requests are distributed between the active assets by a dispatch algorithm. A visualization of the complete approach is shown in Figure 4.6.

After specifying a general scheduling problem for request following, we introduce two

variations of it, tailored to the objective of fulfilling requests for self-consumption and peak power reduction. One feature of these variations is that they either determine a schedule to track the request if it is fulfillable or propose a new request, that maximizes the flexibility provided.

4.3.1 The scheduling problem

We consider the problem of tracking an aggregated relative consumption request $\boldsymbol{r}^{\text{agg}}_{0:H-1} = [r^{\text{agg}}_0, \dots, r^{\text{agg}}_{H-1}]$, where this request is known in advance. A MILP is formulated to determine the activation of consumption flexibility for M flexible assets, where we assume that each asset can be activated only once for a period of k timesteps. The flexibility envelopes of the different assets are given by $\tilde{\boldsymbol{R}}^{\alpha,k}_{i,0:H-1}, i = 1, \dots, M$.

Different objective functions are possible, here we minimize the number of activated assets to keep flexibility in reserve. The general scheduling problem then looks as follows.

$$\min_{u_{i,t}} \quad \sum_{i=1}^{M} \sum_{t=0}^{H-1} u_{i,t} \tag{4.44a}$$

$$\text{s.t.} \quad u_{i,t} \in \{0,1\} \qquad i = 1, \dots, M,\ t = 0, \dots, H-1 \tag{4.44b}$$

$$\sum_{t=0}^{H-1} u_{i,t} \leq 1 \qquad i = 1, \dots, M \tag{4.44c}$$

$$r^{\text{agg}}_t - \sum_{i=1}^{M} \sum_{l=t_s}^{t} u_{i,l} \underline{\boldsymbol{R}}^{\alpha,k}_{i,l} \geq \epsilon \qquad t = 0, \dots, H-1 \tag{4.44d}$$

$$r^{\text{agg}}_t - \sum_{i=1}^{M} \sum_{l=t_s}^{t} u_{i,l} \bar{\boldsymbol{R}}^{\alpha,k}_{i,l} \leq -\epsilon \quad t = 0, \dots, H-1 \tag{4.44e}$$

with $t_s = \max\{t-k-1, 0\}$. Constraint (4.44b) specifies the binary nature of the activation of building i at time t, (4.44c) states that each asset can only be activated once over the time horizon, and (4.44d) and (4.44e) describe the covering of the aggregated requests by the flexibility of the activated assets, which stay active for k timesteps.

The result $\boldsymbol{U} \in \{0,1\}^{M \times H}$ with $[\boldsymbol{U}]_{i,t} = u_{i,t}$ can be interpreted as follows. If $u_{i,t} = 1$, then asset i is activated at time t and can provide consumption flexibility within the range $[\underline{\boldsymbol{R}}^{\alpha,k}_{i,t}, \bar{\boldsymbol{R}}^{\alpha,k}_{i,t}]$ for k timesteps.

Note that problem (4.44) is infeasible if the aggregated request trajectory $\boldsymbol{r}^{\text{agg}}_{0:H-1}$ can not be followed with the available flexibility. Therefore, we introduce a scheduling formulation, that determines a committed request trajectory that is fulfillable.

Scheduling with request commitment

To circumvent the case of an infeasible scheduling problem, we reformulate the constraints of (4.44), using pointwise scaling factors $d_t \in [0,1], t = 0, \dots, H-1$. The constraints look as follows.

$$u_{i,t} \in \{0,1\} \qquad i = 1, \dots, M,\ t = 0, \dots, H-1 \tag{4.45}$$

$$\sum_{t=0}^{H-1} u_{i,t} \leq 1 \qquad i = 1, \dots, M \tag{4.46}$$

$$d_t r_t^{\text{agg}} - \sum_{i=1}^{M} \sum_{l=t_s}^{t} u_{i,l} \underline{\boldsymbol{R}}_{i,l}^{\alpha,k} \geq \epsilon \qquad t = 0, \dots, H-1 \tag{4.47}$$

$$d_t r_t^{\text{agg}} - \sum_{i=1}^{M} \sum_{l=t_s}^{t} u_{i,l} \bar{\boldsymbol{R}}_{i,l}^{\alpha,k} \leq -\epsilon \qquad t = 0, \dots, H-1 \tag{4.48}$$

$$0 \leq d_t \leq 1 \qquad t = 0, \dots, H-1 \tag{4.49}$$

with $t_s = \max\{t-k-1, 0\}$. The new request trajectory $\boldsymbol{r}_{0:H-1}^{\text{comm}} = [d_0 r_0^{\text{agg}}, \dots, d_{H-1} r_{H-1}^{\text{agg}}]$ is then referred to as the committed request trajectory. These constraints can in principle be combined with many different objective functions, aiming to maximize the provided flexibility through the committed requests, according to different metrics. In the following, we discuss the cases of self-consumption on a pool level and peak reduction.

Scheduling for self-consumption and peak reduction

To adapt the scheduling problem to the scenarios of self-consumption and peak reduction, we first define these two types of requests, which represent the ideal scenario for the requesting party.

For self-consumption, the goal is to absorb excess power production by increasing the consumption of the flexible assets. This excess production is, e.g., due to high PV power production. Due to this increase of consumption, the request trajectory has only non-negative entries. In this work, we consider self-consumption on a pool level, meaning that we want to absorb the combined production from assets in the pool by the aggregated consumption. Therefore, it looks as follows.

Definition 12 (Self-consumption request trajectory)**.** *Given an aggregated baseline consumption trajectory $\boldsymbol{p}_{0:H-1}^{b,agg} = [p_0^{b,agg}, \dots, p_{H-1}^{b,agg}] \in \mathbb{R}^H$ and an aggregated production trajectory $\boldsymbol{g}_{0:H-1}^{agg} = [g_0^{agg}, \dots, g_{H-1}^{agg}] \in \mathbb{R}^H$, the self-consumption request trajectory is given by $\boldsymbol{r}_{0:H-1}^{self} = [r_0^{self}, \dots, r_{H-1}^{self}] \in \mathbb{R}^H$ with*

$$r_t^{self} := \max\{g_t^{agg} - p_t^{b,agg}, 0\} \quad t = 0, \dots, H-1. \tag{4.50}$$

This self-consumption request is motivated by definitions like the one given in Luthander et al., 2015 or H. Li et al., 2021 and references therein, which is also used as a metric in the evaluation, presented in 4.4.1. Tracking this request exactly would lead to a complete self-consumption according to that definition. The scheduling problem to maximize the self-consumption then looks as follows:

$$\begin{aligned} \max_{u_{i,t},\, d_t} \quad & \sum_{t=0}^{H-1} r_t^{\text{self}} d_t \\ \text{s.t.} \quad & (4.45) - (4.49) \end{aligned} \tag{4.51}$$

For peak reduction, on the other hand, the requests are non-positive, limiting the overall consumption to a desired level.

Definition 13 (Peak reduction request trajectory)**.** *Given an aggregated baseline consumption trajectory* $\boldsymbol{p}^{b,agg}_{0:H-1} = [p^{b,agg}_0, \dots, p^{b,agg}_{H-1}] \in \mathbb{R}^H$ *and a desired peak* $c \in \mathbb{R}$*, the peak reduction request trajectory* $\boldsymbol{r}^{peak}_{0:H-1} = [r^{peak}_0, \dots, r^{peak}_{H-1}] \in \mathbb{R}^H$ *is defined as*

$$r_t^{peak} := \min\{c - p_t^{b,agg}, 0\} \quad t = 0, \dots, H-1. \tag{4.52}$$

This can be interpreted as peak-clipping, already introduced in the 1980's, e.g., in Gellings, 1985, and also addressed in Hirmiz et al., 2019. In the scheduling, we want to determine a scaling of the original request that minimizes the new peak. This new peak is denoted by $\rho \in \mathbb{R}$ and lower bounded by the desired peak c if there is at least one non-zero request.

$$\begin{aligned} \min_{u_{i,t},\, d_t,\, \rho} \quad & \rho \\ \text{s.t.} \quad & (4.45) - (4.49) \\ & \rho \geq p_t^{\text{b,agg}} + d_t r_t^{\text{peak}} \quad t = 0, \dots, H-1 \end{aligned} \tag{4.53}$$

4.3.2 Responding to aggregated requests

Once a schedule $\boldsymbol{U}$ has been determined through solving (4.44), (4.51), or (4.53), request values still need to be computed for each activated asset. For a committed request r_t^{comm} at time t, this can be done via solving another optimization problem with the following constraints:

$$\sum_{i=1}^{M} r_{i,t} = r_t^{\text{comm}} \tag{4.54}$$

$$\sum_{l=t_s}^{t} u_{i,l} \underline{\boldsymbol{R}}_{i,l}^{\alpha,k} \leq r_{i,t} \leq \sum_{l=t_s}^{t} u_{i,l} \bar{\boldsymbol{R}}_{i,l}^{\alpha,k} \quad i = 1, \dots, M \tag{4.55}$$

with $t_s = \max\{t - k - 1, 0\}$. Different objectives are again possible in this case. For example, to achieve a balanced activation over all assets, it is possible to minimize $\sum_{i=1}^{M} r_{i,t}^2$.

Instead of solving an optimization problem, heuristic methods can be used to achieve a fast dispatch. For this, we use the shorthand notation

$$A_{i,t} = \sum_{l=t_s}^{t} u_{i,l} \in \{0, 1\} \tag{4.56}$$

with $t_s = \max\{t - k - 1, 0\}$, to indicate whether asset i is active at time t. Then we can formulate the heuristic dispatch through

$$r_{i,t} = A_{i,t} \frac{F_i}{\sum_{j=1}^{M} F_j A_{j,t}} r_t^{\text{comm}} \tag{4.57}$$

where F_i denotes a measure of the available flexibility of asset i. Possible choices of this are the available activated flexibility, by choosing

$$F_i = \sum_{t=0}^{H-1} u_{i,t} \bar{\boldsymbol{R}}_{i,t}^{\alpha,k} \tag{4.58}$$

or

$$F_i = \sum_{t=0}^{H-1} u_{i,t} \underline{\boldsymbol{R}}_{i,t}^{\alpha,k} \tag{4.59}$$

depending on the sign of the request r_t^{comm}. Another possibility is to use the average flexibility potential as a proxy, given by

$$F_i = \frac{1}{H} \sum_{t=0}^{H-1} (\bar{\boldsymbol{R}}_{i,t}^{\alpha,k} - \underline{\boldsymbol{R}}_{i,t}^{\alpha,k}). \tag{4.60}$$

The latter dispatch method is used for computational efficiency in the experiments.

For systems with a fixed setpoint temperature, the controller of each asset will try to steer the state of the asset back to its nominal value, after a request period. This can lead to undesirable rebound behavior, where, e.g., a new peak is reached during the rebound. To mitigate those effects, we limit the deviation from the baseline consumption during the rebound phase in the following way:

- If $r_t^{\text{agg}} \neq 0$, then all rebounding assets should consume according to their baseline, to guarantee request tracking.
- If $r_t^{\text{agg}} = 0$, then the allowed deviation from the baseline consumption is limited for

the rebounding assets, either by a fixed fraction or by a peak-dependent fraction.

For simplicity, the rebound-damping strategy is formulated for the aggregated requests r_t^{agg} only, but is applied in the same manner to self-consumption requests r_t^{self} and peak reduction requests r_t^{peak}. In the experiments, the deviation in the rebound phase is limited to 20% of the baseline consumption in the self-consumption scenario. For peak reduction, we continuously keep track of the actual peak during the day. The allowed deviation is then time-varying, depending on the aggregated baseline consumption and its difference to the actual peak, given by

$$\text{dev}_t = \frac{\text{peak} - p_t^{\text{b,agg}}}{p_t^{\text{b,agg}}}. \tag{4.61}$$

Like this, it is ensured that a rebound of all buildings at the same time does not lead to a new peak in consumption.

4.4 Numerical results

The full chain of flexibility estimation, scheduling, and dispatch is tested in simulation, using a pool of buildings from Energym. The scheduling problems are formulated with the linear programming toolkit PuLP (Mitchell et al., 2011) and solved with CBC (Forrest et al., 2023) or Gurobi (Gurobi Optimization, LLC, 2023).

4.4.1 Experimental setup

Building models and requests

We use the SimpleHouseRad-v0 model from Energym as a flexible asset. We sample the building parameters of thermal capacity, thermal conductance, and nominal COP of the HP uniformly at random from pre-specified intervals, furthermore, we scale the maximum HP power according to the sampled thermal conductance. This is done to ensure slightly varying characteristics in the overall pool of buildings.

The HP power fraction is controlled by a PID controller with a control timestep of 5 minutes. A temperature setpoint of 21 °C is followed by the PID controller, and we set the acceptable temperature to the range $[19, 24]$ °C. Flexibility predictions and requests are on the other hand sent with a 15-minute timestep.

Experiments of the coordination of 100 to 500 buildings for the scenarios of self-consumption and peak reduction are performed, as well as scalability experiments for solving the scheduling problem with up to 1000 buildings.

Self-consumption requests are generated as follows. An aggregated production curve is

computed, using the PVSystem class of the Python library pvlib (Holmgren et al., 2018) as a single system, with a capacity scaled to the overall pool of buildings. For this, perfect forecasts of the irradiance and temperature are used, provided by Energym. The request is then given by the difference of the production and the aggregated baseline consumption prediction, as stated in Definition 12.

Peak reduction requests are based on real consumption data from the canton of Neuchâtel, Switzerland, provided by the grid operator Swissgrid (Swissgrid, 2023). This consumption data is scaled down to match the magnitude of the consumption of the pool of buildings. The difference of the scaled real consumption data and the aggregated baseline consumption of the buildings gives the non-shiftable baseline consumption. We define a desired new peak c, in the experiments chosen as 1 to 1.1 times the average consumption for that particular day, and then compute the request based on Definition 13 with respect to the scaled real consumption.

Baseline approach

We compare our approach to a simple greedy strategy for responding to requests. For this, assets are grouped into three sets: available assets, active assets, and inactive assets. Upon receiving a relative consumption request, it is checked if this request is fulfillable with the available flexibility of all buildings in the active group. If it is not fulfillable, buildings from the available group are activated until the request is either fulfillable or no buildings are left in the available group. This activation is done at random. Buildings stay active for at most k timesteps, after that they are set as inactive. Violating the comfort bounds while responding to requests also leads to a building being set as inactive.

To mitigate the rebound of the inactive assets and to guarantee comparability in the experiments, the approach outlined in Section 4.3.2 is used.

Evaluation metrics

In the experiments, we distinguish between performance metrics and comfort metrics. As performance metrics for the peak reduction case, we use the absolute peak power reduction, defined as follows.

Definition 14. *Given the aggregated baseline consumption* $\boldsymbol{p}^{b,agg}_{0:H-1}$ *and the aggregated actual consumption* $\boldsymbol{p}^{agg}_{0:H-1}$*, we define the absolute peak power reduction as*

$$\Delta P_a = \max(\boldsymbol{p}^{b,agg}_{0:H-1}) - \max(\boldsymbol{p}^{agg}_{0:H-1}). \tag{4.62}$$

For the case of self-consumption, we consider the metric of self-consumed power fraction, defined as follows.

Definition 15. *Given the aggregated production* $\boldsymbol{g}^{agg}_{0:H-1}$*, the production sum* $g_{sum} = \sum_{t=0}^{H-1} g_t^{agg}$*, and the aggregated actual consumption* $\boldsymbol{p}^{agg}_{0:H-1}$*. Then the self-consumed power fraction is given by*

$$\Delta S_r = \frac{g_{sum} - \sum_{t=0}^{H-1} \max\{g_t^{agg} - p_t^{agg}, 0\}}{g_{sum}}. \tag{4.63}$$

This metric specifies how much of the overall production was directly consumed. These performance metrics are, e.g., presented in (H. Li et al., 2021).

To measure comfort, we specify the percentage of temperature-bound violations for a specific acceptable temperature interval. This is defined as follows.

Definition 16. *Given a temperature trajectory* $\boldsymbol{T}_{0:H-1} = [T_0, \dots, T_{H-1}]$ *and temperature bounds* $\underline{T}, \bar{T}$*, we define the percentage of temperature bound violations as*

$$\Delta T_r = \frac{\sum_{t=0}^{H-1} I_{\underline{T}}^{\bar{T}}(T_t)}{H} 100\% \tag{4.64}$$

$$\text{with } I_a^b(c) = \begin{cases} 0, \text{ if } c \in [a,b] \\ 1, \text{ if } c \notin [a,b] \end{cases}.$$

This metric is similar to the prediction interval coverage percentage, frequently used in statistical forecasting (González-Sopeña et al., 2021).

For the scalability experiments, we determine the solving times of the scheduling problem up to a predefined gap and report the average time for multiple runs as a metric for scalability.

4.4.2 Results

In this section, we collect the results of the different simulation experiments. The experiments include the application of our approach to the scenarios of self-consumption and peak reduction, a comparison with the baseline approach explained in Section 4.4.1, a variation of the allowed activation timesteps k for a fixed number of buildings, a variation of the number of buildings for a fixed request, and the scalability experiments.

For all building models, we collect data from the first 21 days of the year to fit the virtual battery models described in Section 4.2.2. The test period covers the 50 following days. To evaluate the results for peak reduction, the metric in (4.62) is computed for each day and then averaged over the 50 days. For the self-consumed power fraction and the percentage of temperature violations, (4.63) and (4.64) are computed over the whole 50 days.

Self-consumption experiments

The results of running the self-consumption experiments for the 50 test days are displayed in Figure 4.7. Figure 4.7a shows the first three days of the test period with the baseline consumption forecast given in green, the actual consumption in red, the baseline plus relative request in blue, and the baseline plus committed request in orange. Due to the perfect tracking during the request periods, the orange line is covered by the red one. The first day shows a large request which is not fulfillable with the estimated available flexibility and therefore leads to a lower committed request. On days two and three, the requests are fulfillable and thus tracked exactly. An example of a computed schedule for a self-consumption request is given in Appendix 4.6.4.

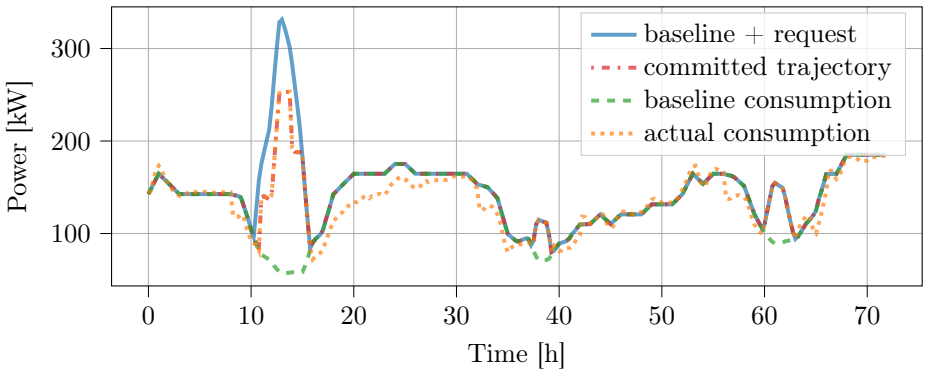
Figure 4.7b shows the self-consumed power fraction and the percentage of temperature-bound violations for α values of 0.001, 0.5, and 1, and compares them to the baseline approach and the nominal controller operation without receiving flexibility requests. As expected, a lower α leads to more conservative predictions of the available flexibility, and therefore both to a lower self-consumption ΔS_r and a lower percentage of violations ΔT_r, with about 0.548 and 0.06% respectively. With increasing α, both of the metrics increase as well, up to a ΔS_r of about 0.631 and a ΔT_r of about 1.11% for $\alpha = 1$. The baseline approach achieves the highest self-consumption with $\Delta S_r \approx 0.686$, but also a high percentage of violations with $\Delta T_r \approx 5.17\%$. In comparison, the nominal controller operation does not result in any temperature-bound violations and has a self-consumption of $\Delta S_r \approx 0.391$.

We also run the experiments for 500 buildings with our approach and $\alpha = 1$. Since the requests are scaled to the number of buildings, a comparable result of $\Delta S_r \approx 0.629$ and $\Delta T_r \approx 1.17\%$ is achieved. The results for three of the test days are shown in Appendix 4.6.5.

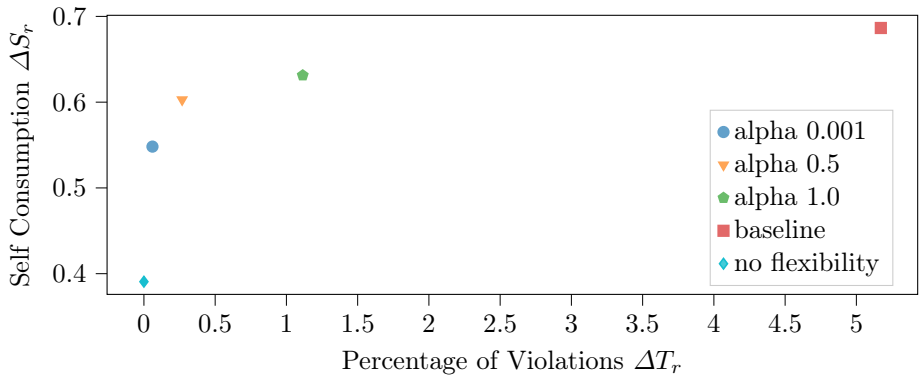
Peak reduction experiments

Figure 4.8 presents the results of the peak reduction experiments for the 50 test days. Again, the results for the first three days with $\alpha = 1$ are shown on the left-hand side, Figure 4.8a. In this example, the request of the first day is fulfillable and tracked accordingly. For days two and three, the available flexibility is not sufficient to decrease the overall consumption to the desired level (blue line), instead, a smaller decrease is committed to and followed (orange line, covered by red line). An example of a computed schedule for a peak reduction request is given in Appendix 4.6.4.

As in the self-consumption case, a smaller α leads to a lower absolute peak power reduction and a lower percentage of violations ($\Delta P_a \approx 40.71$kW, $\Delta T_r \approx 0.005\%$ for $\alpha = 0.001$), whereas a higher α leads to better performance, but higher violations ($\Delta P_a \approx 46.49$kW, $\Delta T_r \approx 0.691\%$ for $\alpha = 1$). An intermediate α seems to deliver a good tradeoff between



(a) Self-consumption request following for the first three days of the test period with $\alpha = 1$.



(b) Cumulated metrics for the test period with varying α values and comparison to the baseline and nominal controller operation.

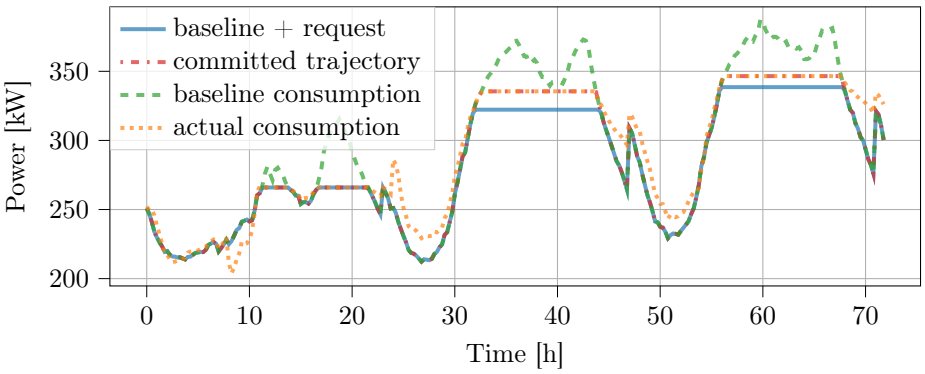
Figure 4.7 – Results of the self-consumption experiments with 100 buildings and a maximum activation time of 3h per building.

performance and violations for both self-consumption and peak reduction. The baseline approach performs less well in the peak reduction case, having both the lowest absolute peak power reduction of about 29.36kW and the highest percentage of violations of about 1.89%.

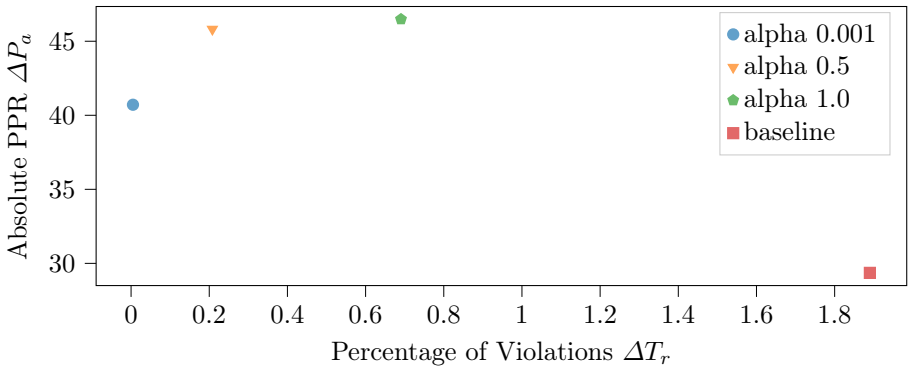
In the corresponding experiments with 500 buildings and $\alpha = 1$, a peak power reduction of about 229.80kW is achieved with about 0.608% bound violations. See Appendix 4.6.5 for a visualization of these results.

Additionally, we run experiments to quantify how the metrics change when we vary the number of available timesteps k. Similarly, we run experiments to quantify how the number of buildings influences the resulting metrics.

For the first part, we run the experiments for 100 buildings and compute flexibility envelopes for $k = 12$ (i.e., 1 hour availability), $k = 36$ (i.e., 3 hours availability), and $k = 60$ (i.e., 5 hours availability) with $\alpha = 1$. The results are shown in Figure 4.9a. A short availability time leads to no temperature violations in this case, but also a lower



(a) Peak reduction request following for the first three days of the test period with $\alpha = 1$.

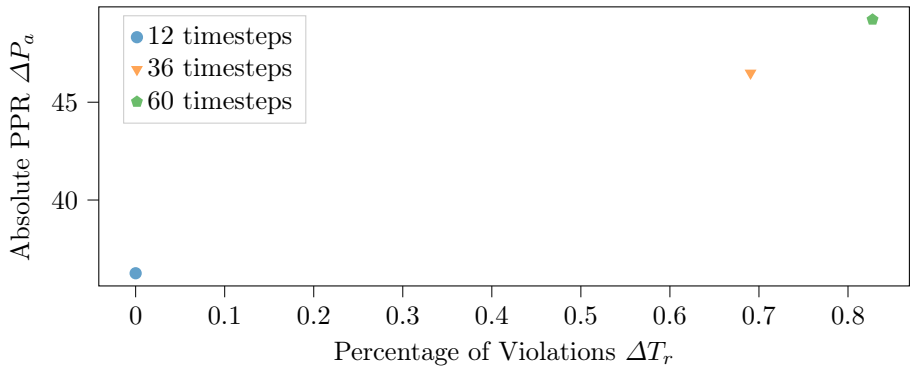


(b) Averaged metrics for the test period with varying α values and comparison to the baseline.

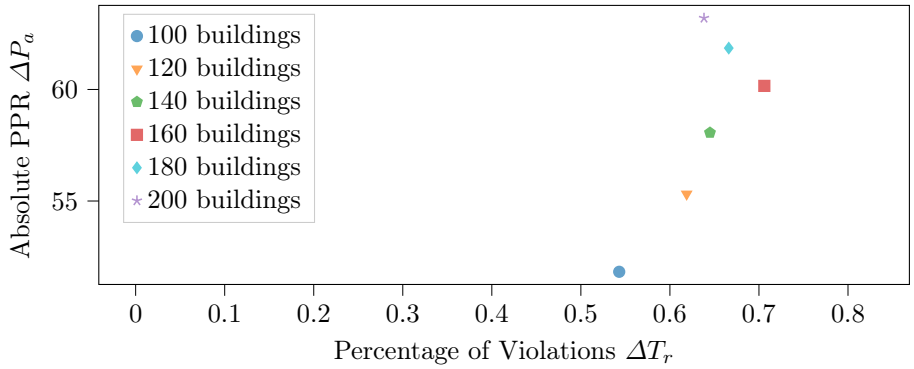
Figure 4.8 – Results of the peak reduction experiments with 100 buildings and a maximum activation time of 3h per building.

absolute peak power reduction of about 36.26kW. With increasing activation duration, both of these metrics increase to about 0.828% of violations and ΔP_a of 49.23kW. This might be due to either a duration of 1 hour not being enough for the building to saturate its temperature bounds, or to a longer prediction horizon leading to an accumulation of errors, and therefore an overestimation of the available flexibility.

For the second part, we generate requests according to the description in Section 4.4.1, for a pool of 120 buildings. The request following is then attempted with 100 to 200 buildings. The results are shown in Figure 4.9b. As expected, having more buildings increases the capabilities in peak reduction, which can be observed with an absolute peak power reduction of 51.83kW for 100 buildings to a ΔP_a of 63.20kW for 200 buildings. Interestingly, no clear trend is visible in the percentage of violations, being in the range of 0.543% to 0.706%.



(a) Averaged metrics for the test period with $\alpha = 1$ and varying availability times.



(b) Averaged metrics for the test period with $\alpha = 1$, a fixed request, and a varying number of buildings.

Figure 4.9 – Impact of varying parameters in the peak reduction experiments with $\alpha = 1$.

Scalability experiments

To test the scalability of our approach to the number of included assets, we measure the wall-clock solving time of the scheduling problem with different numbers of flexibility envelopes. The flexibility envelopes are chosen at random from a set of 25000 envelopes, generated from the operation of 500 buildings during 50 days. The request trajectory is generated for a random day of the year, distinguishing between a high request (new desired peak c as the average consumption of that day), a medium request (1.05 times the average), and a low request (1.1 times the average) scenario. This is done for 50, 100, 250, 500, 1000, and 2000 envelopes and the solving times are averaged over 20 runs. The problem is solved up to an absolute gap of $0.01c$, using Gurobi as a solver. These experiments were run on a laptop with Intel i7-8565U processor running at 1.8 GHz and 16 GB of RAM. The results of the experiments are displayed in Figure 4.10.

The solving times scale about linearly in the number of considered envelopes, the peak being reached at an average of 24 seconds for the high request scenario for 2000 envelopes, and the minimum at an average of about 0.4 seconds for the medium request scenario for 50 envelopes. When repeating the experiments with a fixed absolute gap of 1, meaning

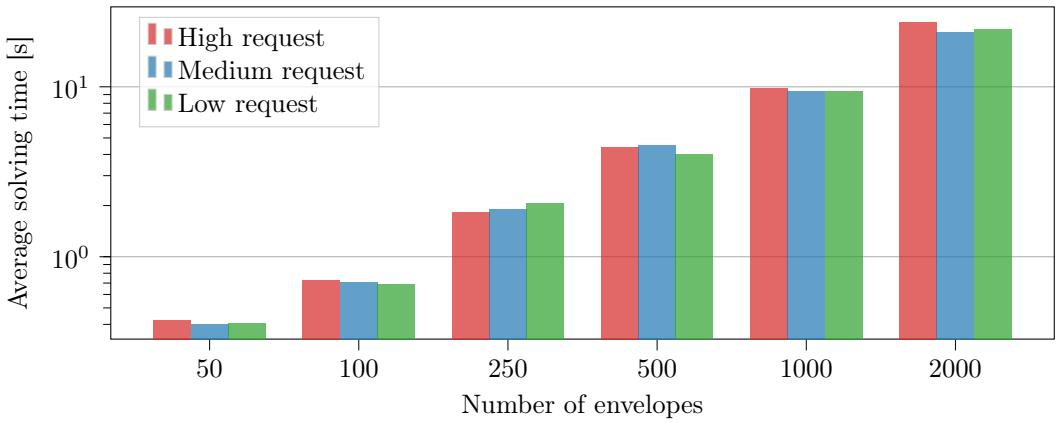


Figure 4.10 – Average solving times of the peak reduction scheduling problem for varying numbers of assets in three request cases.

that the promised peak would be at most 1kW away from the optimally achievable one, the results lie in a comparable range. Considering that solving this problem would usually be done once a day for scheduling the activation of assets, these runtimes suggest the feasibility of our approach for an even larger number of assets. Also, depending on the exact time requirements, the approach could be used in a receding horizon framework for a medium to high number of assets.

4.5 Conclusion

In this chapter, we presented a full pipeline for estimating and coordinating the consumption flexibility of flexible assets, in particular buildings. Starting from different definitions of flexibility in the form of flexibility envelopes, a virtual battery modeling approach was introduced to model the state transition of these flexible assets. Considering uncertainty in the response to flexibility requests, feasible request trajectories, and in turn flexibility envelopes, were characterized in a probabilistic fashion and reformulated in a deterministic way with the help of risk measures. This allowed us to adjust the conservativeness of the flexibility predictions by choosing an uncertainty parameter. Additional assumptions were introduced to facilitate the computation of these flexibility envelopes. Flexibility predictions were then used in the formulation of a MILP for scheduling a pool of assets. The formulation was flexible to the use of different objectives and request types, like maximizing self-consumption or peak reduction. With the schedule in place, we discussed a heuristic algorithm to do the dispatch of aggregated requests to the individual assets, as well as a simple strategy to mitigate rebound effects. In simulation, the approach was effective in balancing the comfort and performance metrics, depending on the chosen uncertainty parameter. The scalability of solving the scheduling problem with up to 2000 assets was demonstrated with a solving time of about 21 seconds on a conventional laptop, making it feasible for the deployment on a large number of assets.

Many extensions are possible for this line of work. One clear connection to the results discussed in Chapter 2 is the usage of uncertainty quantification techniques for the predictions of baseline consumption and state. How this additional uncertainty affects the flexibility envelope predictions would need to be investigated, first in a theoretical and second in a practical analysis. The extension of the scheduling and dispatch approach to a receding horizon formulation seems promising for dealing with imperfect knowledge of future requests and seems feasible as well due to the reasonable solving times of the scheduling problem. On the practical side, the examination of the impact of heterogeneous assets in the considered pool is of interest, since the simulation study presented here deals with the same type of asset with varying parameters. Making the step from simulation to deployment on real systems is another goal for the future of this approach.

4.6 Appendix

4.6.1 Proof of Lemma 5

We use (4.13) to prove (4.14) by induction. For this, we repeatedly use the fact that $q_k^k = 0$ and therefore $(1 - b_f)^{q_k^k} = 1$.

Let $k = 1$. We distinguish the cases $r_0 = 0$ and $r_0 \neq 0$. For $r_0 = 0$, we have $\chi_{r_0} = 1$ and therefore $q_0^1 = 1$.

We have

$$\begin{aligned}\hat{s}_1 &= s_0 + b_f(f(\boldsymbol{e}_{:0}) - s_0) + f(\boldsymbol{e}_{:1}) - f(\boldsymbol{e}_{:0}) \\ &= (1 - b_f)s_0 + b_f f(\boldsymbol{e}_{:0}) + f(\boldsymbol{e}_{:1}) - f(\boldsymbol{e}_{:0}) \\ &= (1 - b_f)^{q_0^1} s_0 + \sum_{l=0}^{0} (1 - b_f)^{q_{l+1}^1} (f(\boldsymbol{e}_{:l}) b_f \chi_{r_l} + a^+ r_l^+ + a^- r_l^- + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})).\end{aligned}$$

Similarly, for $r_0 \neq 0$, we have $\chi_{r_0} = 0$ and therefore $q_0^1 = 0$, resulting in

$$\begin{aligned}\hat{s}_1 &= s_0 + a^+ r_t^+ + a^- r_t^- + f(\boldsymbol{e}_{:1}) - f(\boldsymbol{e}_{:0}) \\ &= (1 - b_f)^{q_0^1} s_0 + \sum_{l=0}^{0} (1 - b_f)^{q_{l+1}^1} (f(\boldsymbol{e}_{:l}) b_f \chi_{r_l} + a^+ r_l^+ + a^- r_l^- + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})).\end{aligned}$$

In both of those reformulations we used adding terms that evaluate to zero or multiplying by terms that are one to arrive at the formulation in (4.14).

Assume that (4.14) holds for a fixed $k \in \mathbb{N}$. Then we observe the following for $k + 1$:

In the case that $r_k = 0$, we have $\chi_{r_k} = 1$ and therefore $q_l^{k+1} = q_l^k + 1$ for $l = 0, \ldots, k$.

From (4.13), we have

$$\begin{aligned}
\hat{s}_{k+1} &= \hat{s}_k + b_f(f(\boldsymbol{e}_{:k}) - \hat{s}_k) + f(\boldsymbol{e}_{:k+1}) - f(\boldsymbol{e}_{:k}) \\
&= (1-b_f)\hat{s}_k + b_f f(\boldsymbol{e}_{:k}) + f(\boldsymbol{e}_{:k+1}) - f(\boldsymbol{e}_{:k}) \\
&= (1-b_f)\Bigg((1-b_f)^{q_0^k} s_0 + \sum_{l=0}^{k-1}(1-b_f)^{q_{l+1}^k}\Big(f(\boldsymbol{e}_{:l})b_f\chi_{r_l} + a^+ r_l^+ + a^- r_l^- \\
&\quad + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})\Big)\Bigg) + b_f(f(\boldsymbol{e}_{:k}) - \hat{s}_k) + f(\boldsymbol{e}_{:k+1}) - f(\boldsymbol{e}_{:k}) \\
&= (1-b_f)^{q_0^k+1} s_0 + \sum_{l=0}^{k-1}(1-b_f)^{q_{l+1}^k+1}\Big(f(\boldsymbol{e}_{:l})b_f\chi_{r_l} + a^+ r_l^+ + a^- r_l^- \\
&\quad + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})\Big) + (1-b_f)^{q_{k+1}^{k+1}}\left(b_f(f(\boldsymbol{e}_{:k}) - \hat{s}_k) + f(\boldsymbol{e}_{:k+1}) - f(\boldsymbol{e}_{:k})\right) \\
&= (1-b_f)^{q_0^{k+1}} s_0 + \sum_{l=0}^{k}(1-b_f)^{q_{l+1}^{k+1}}(f(\boldsymbol{e}_{:l})b_f\chi_{r_l} + a^+ r_l^+ + a^- r_l^- + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})).
\end{aligned}$$

In the case that $r_k \neq 0$, we have $\chi_{r_k} = 0$, $q_l^{k+1} = q_l^k$ for $l = 0, \ldots, k$, and therefore

$$\begin{aligned}
\hat{s}_{k+1} &= \hat{s}_k + a^+ r_k^+ + a^- r_k^- + f(\boldsymbol{e}_{:k+1}) - f(\boldsymbol{e}_{:k}) \\
&= (1-b_f)^{q_0^k} s_0 + \sum_{l=0}^{k-1}(1-b_f)^{q_{l+1}^k}\Big(f(\boldsymbol{e}_{:l})b_f\chi_{r_l} + a^+ r_l^+ + a^- r_l^- \\
&\quad + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})\Big) + a^+ r_k^+ + a^- r_k^- + f(\boldsymbol{e}_{:k+1}) - f(\boldsymbol{e}_{:k}) \\
&= (1-b_f)^{q_0^k+1} s_0 + \sum_{l=0}^{k-1}(1-b_f)^{q_{l+1}^k+1}\Big(f(\boldsymbol{e}_{:l})b_f\chi_{r_l} + a^+ r_l^+ + a^- r_l^- \\
&\quad + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})\Big) + (1-b_f)^{q_{k+1}^{k+1}}\left(a^+ r_k^+ + a^- r_k^- + f(\boldsymbol{e}_{:k+1}) - f(\boldsymbol{e}_{:k})\right) \\
&= (1-b_f)^{q_0^{k+1}} s_0 + \sum_{l=0}^{k}(1-b_f)^{q_{l+1}^{k+1}}(f(\boldsymbol{e}_{:l})b_f\chi_{r_l} + a^+ r_l^+ + a^- r_l^- + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})).
\end{aligned}$$

Again, we used adding terms that evaluate to zero or multiplying by terms that are one. The final equations give (4.14), which proves the claim for general $k \in \mathbb{N}$.

4.6.2 Reformulations

We start by recalling the formulation of the state $\hat{s}_k$ as in (4.14):

$$\hat{s}_k = (1-b_f)^{q_0^k} s_0 + \sum_{l=0}^{k-1}(1-b_f)^{q_{l+1}^k}(f(\boldsymbol{e}_{:l})b_f\chi_{r_l} + a^+ r_l^+ + a^- r_l^- + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l})),$$

with $q_l^k = \sum_{i=l}^{k-1} \chi_{r_i}$. We first group all the non-request parts to get

$$\hat{s}_k = \underbrace{(1-b_f)^{q_0^k} s_0 + \sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} (f(\boldsymbol{e}_{:l}) b_f \chi_{r_l} + f(\boldsymbol{e}_{:l+1}) - f(\boldsymbol{e}_{:l}))}_{c_k} + \sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} (a^+ r_l^+ + a^- r_l^-).$$

The latter part can then be reformulated in the two following ways:

$$\begin{aligned}
\hat{s}_k &= c_k + \sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} a^+ r_l^+ + \sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} a^- r_l^- \\
&= c_k + \underbrace{[(1-b_f)^{q_1^k} a^+, \dots, (1-b_f)^{q_k^k} a^+]}_{\boldsymbol{a}_k^+} \underbrace{\begin{pmatrix} r_0^+ \\ \vdots \\ r_{k-1}^+ \end{pmatrix}}_{\boldsymbol{r}_k^+} \\
&\quad + \underbrace{[(1-b_f)^{q_1^k} a^-, \dots, (1-b_f)^{q_k^k} a^-]}_{\boldsymbol{a}_k^-} \underbrace{\begin{pmatrix} r_0^- \\ \vdots \\ r_{k-1}^- \end{pmatrix}}_{\boldsymbol{r}_k^-} \\
&= c_k + [\boldsymbol{a}_k^+, \boldsymbol{a}_k^-] \begin{pmatrix} \boldsymbol{r}_k^+ \\ \boldsymbol{r}_k^- \end{pmatrix}
\end{aligned}$$

which is the same as (4.18) and

$$\begin{aligned}
\hat{s}_k &= c_k + \left(\sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} r_l^+ \right) a^+ + \left(\sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} r_l^- \right) a^- \\
&= c_k + \underbrace{\left[\sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} r_l^+, \sum_{l=0}^{k-1} (1-b_f)^{q_{l+1}^k} r_l^- \right]}_{\boldsymbol{R}_k} \begin{pmatrix} a^+ \\ a^- \end{pmatrix} \\
&= c_k + \boldsymbol{R}_k \begin{pmatrix} a^+ \\ a^- \end{pmatrix}
\end{aligned}$$

which is the same as (4.19). Using this, we can reformulate the set of feasible request trajectories (4.17)

$$\mathcal{R}_k^\alpha(s_0) = \left\{ \boldsymbol{r}_{0:k-1} \in \mathbb{R}^k : \mathbb{P}\left\{ \boldsymbol{0} \leq [\hat{s}_0, \cdots, \hat{s}_k]^\top \leq \boldsymbol{1} \right\} \geq 1-\alpha, \hat{s}_0 = s_0 \right\}.$$

Since the probability is taken element-wise, we can write the set as an intersection over sets of the individual elements:

$$\mathcal{R}_k^\alpha(s_0) = \bigcap_{l=0}^{k} \left\{ \boldsymbol{r}_{0:k-1} \in \mathbb{R}^k : \mathbb{P}\left\{0 \leq \hat{s}_l \leq 1\right\} \geq 1-\alpha, \hat{s}_0 = s_0 \right\}.$$

Considering an individual element of the intersection, we can substitute $\hat{s}_l$ by

$$\left\{ \boldsymbol{r}_{0:k-1} \in \mathbb{R}^k : \mathbb{P}\left\{ 0 \leq c_l + [\boldsymbol{a}_l^+, \boldsymbol{a}_l^-] \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \leq 1 \right\} \geq 1-\alpha, \hat{s}_0 = s_0 \right\}.$$

For the inequality in the probability, we have

$$\begin{aligned}
& 0 \leq c_l + [\boldsymbol{a}_l^+, \boldsymbol{a}_l^-] \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \leq 1 \\
\Leftrightarrow \quad & 0 \leq c_l + [\boldsymbol{a}_l^+, \boldsymbol{a}_l^-] \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \\
& -1 \leq -c_l - [\boldsymbol{a}_l^+, \boldsymbol{a}_l^-] \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \\
\Leftrightarrow \quad & \begin{pmatrix} 0 \\ -1 \end{pmatrix} \leq \begin{pmatrix} c_l \\ -c_l \end{pmatrix} \begin{pmatrix} \boldsymbol{a}_l^+ & \boldsymbol{a}_l^- \\ -\boldsymbol{a}_l^+ & -\boldsymbol{a}_l^- \end{pmatrix} \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \\
\Leftrightarrow \quad & \underbrace{\begin{pmatrix} -c_l \\ c_l - 1 \end{pmatrix}}_{\boldsymbol{b}_l} \leq \underbrace{\begin{pmatrix} \boldsymbol{a}_l^+ & \boldsymbol{a}_l^- \\ -\boldsymbol{a}_l^+ & -\boldsymbol{a}_l^- \end{pmatrix}}_{\boldsymbol{A}_l} \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix}
\end{aligned}$$

which results in the updated form of the set of feasible request trajectories

$$\mathcal{R}_k^\alpha(s_0) = \bigcap_{l=0}^{k} \left\{ \boldsymbol{r}_{0:k-1} \in \mathbb{R}^k : \mathbb{P}\left\{ \boldsymbol{b}_l \leq \boldsymbol{A}_l \begin{pmatrix} \boldsymbol{r}_l^+ \\ \boldsymbol{r}_l^- \end{pmatrix} \right\} \geq 1-\alpha \right\}.$$

4.6.3 j-point averages

We recall that $\mathcal{P}^+ = \{a_1^+, \ldots, a_{n_1}^+ : a_i^+ \leq a_j^+ \text{ if } i < j\}, \mathcal{P}^- = \{a_1^-, \ldots, a_{n_2}^- : a_i^- \leq a_j^- \text{ if } i < j\}$, and $\mathcal{P}_j = \{\frac{1}{j}\sum_{i=1}^{j} \boldsymbol{a}_i : \boldsymbol{a}_i \in \mathcal{P}^+ \times \mathcal{P}^-, \boldsymbol{a}_l \neq \boldsymbol{a}_k \text{ for } l \neq k\}$. To get the parameters in $\mathcal{P}_j$ with the maximum first or second component, we look at the sorted list of tuples in $\mathcal{P}^+ \times \mathcal{P}^-$. We restrict ourselves to the case of sorting by the first component,

i.e. finding $a^+_{\max,j}$, the result for $a^-_{\max,j}$ follows analogously. We have

$$\underbrace{\boldsymbol{a}_1 = (a^+_{n_1}, a^-_{n_2}), \boldsymbol{a}_2 = (a^+_{n_1}, a^-_{n_2-1}), \ldots, \boldsymbol{a}_{n_2} = (a^+_{n_1}, a^-_1)}_{n_2 \text{ tuples}}$$
$$\underbrace{\boldsymbol{a}_{n_2+1} = (a^+_{n_1-1}, a^-_{n_2}), \boldsymbol{a}_{n_2+2} = (a^+_{n_1-1}, a^-_{n_2-1}), \ldots, \boldsymbol{a}_{2n_2} = (a^+_{n_1-1}, a^-_1)}_{n_2 \text{ tuples}}$$
$$\vdots$$
$$\underbrace{\boldsymbol{a}_{(n_1-1)n_2+1} = (a^+_1, a^-_{n_2}), \boldsymbol{a}_{(n_1-1)n_2+2} = (a^+_1, a^-_{n_2-1}), \ldots, \boldsymbol{a}_{n_1 n_2} = (a^+_1, a^-_1)}_{n_2 \text{ tuples}},$$

so n_2 tuples with the maximum first component, n_2 tuples with the second-largest first component, and so on. To get the j-point average with the largest first component, we need to take the average of $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_j$. Like this, we get that

$$a^+_{\max,j} = \frac{1}{j}\left(n_2 \sum_{i=0}^{\lfloor \frac{j}{n_2} \rfloor} a^+_{n_1-i} + (j \bmod n_2) a^+_{n_1-\lfloor \frac{j}{n_2} \rfloor - 1}\right).$$

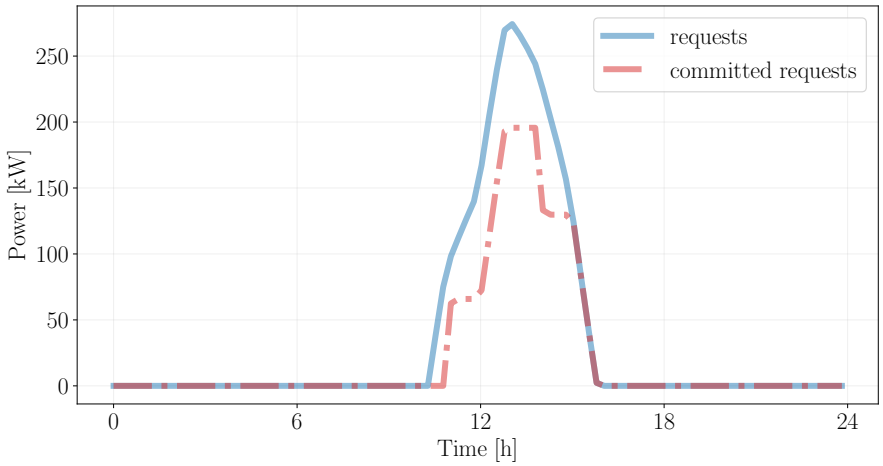
Equivalently, we get

$$a^-_{\min,j} = \frac{1}{j}\left(n_1 \sum_{i=0}^{\lfloor \frac{j}{n_1} \rfloor} a^-_{n_2-i} + (j \bmod n_1) a^-_{n_2-\lfloor \frac{j}{n_1} \rfloor - 1}\right).$$

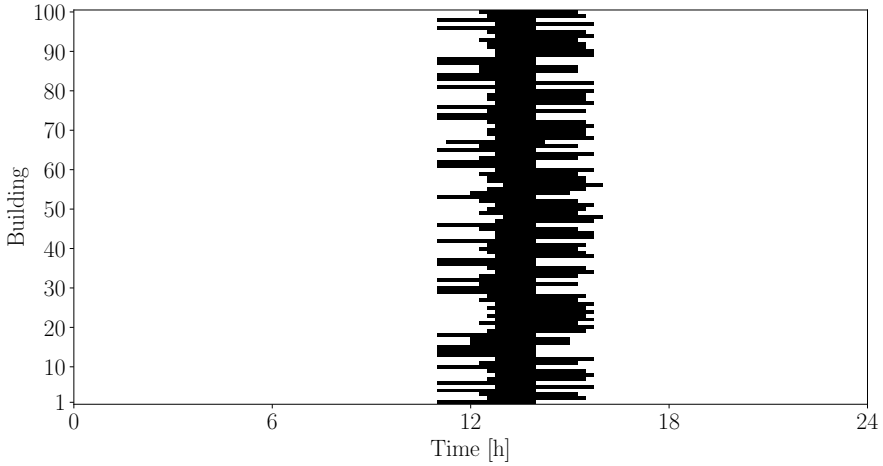
4.6.4 Example schedules

Figure 4.11 shows an example of the scheduling result for a self-consumption request. The overall request is not fulfillable, and the resulting schedule together with the committed request shows a clear pattern: buildings are either activated right at the start of the request period or with a delay, such that the highest self-consumption is achieved in the middle of the request period when all buildings are active at the same time.

In Figure 4.12, an example of the scheduling result for a peak reduction request is displayed. Again, the overall request is not fulfillable, so a scaled-down version of this request is determined as the committed request. Building activation is more scattered in this example, as shown in Figure 4.12b, but centered around the two request peaks at about 12 h and 18 h.



(a) Relative self-consumption request (blue solid line) and committed request (red dashed line).

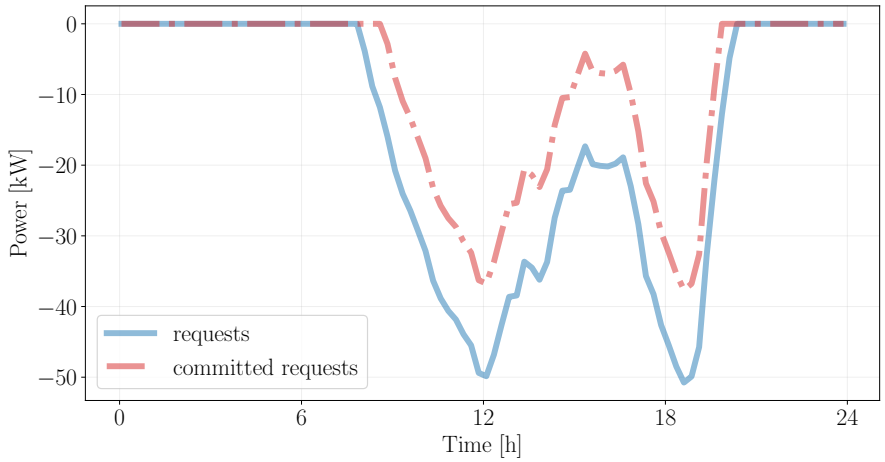


(b) Building activation times (black bars) for each of the 100 buildings over one day.

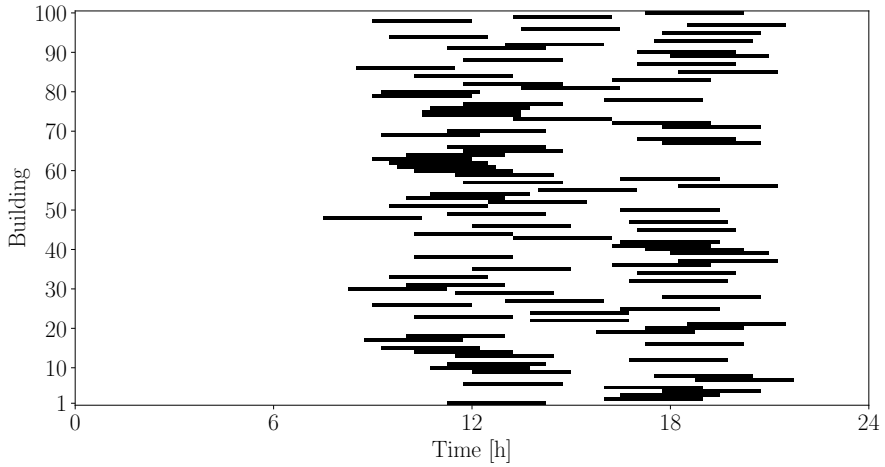
Figure 4.11 – Scheduling example for a self-consumption request with 100 buildings and $\alpha = 1$.

4.6.5 Request following with 500 buildings

Examples of the scheduling and dispatch with 500 buildings and $\alpha = 1$ are shown in Figure 4.13 for the case of self-consumption, and in Figure 4.14 for the case of peak reduction.



(a) Relative peak reduction request (blue solid line) and committed request (red dashed line).



(b) Building activation times (black bars) for each of the 100 buildings over one day.

Figure 4.12 – Scheduling example for a peak reduction request with 100 buildings and $\alpha = 1$.

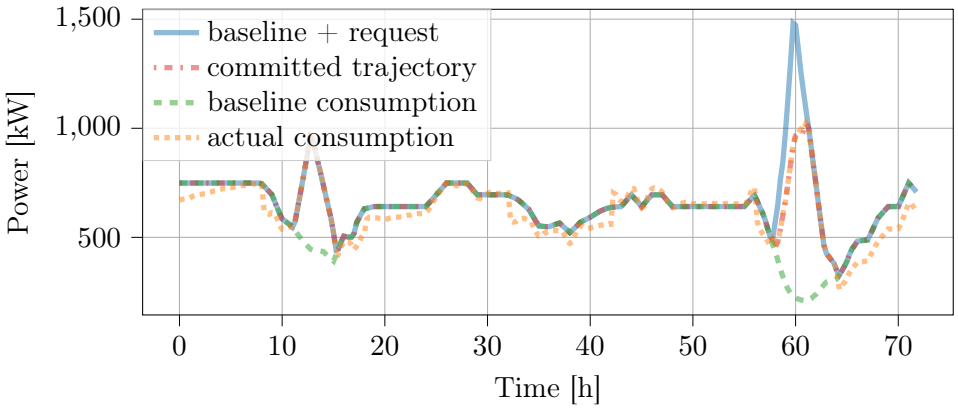


Figure 4.13 – Self-consumption request following for three days of the test period with 500 buildings and $\alpha = 1$.

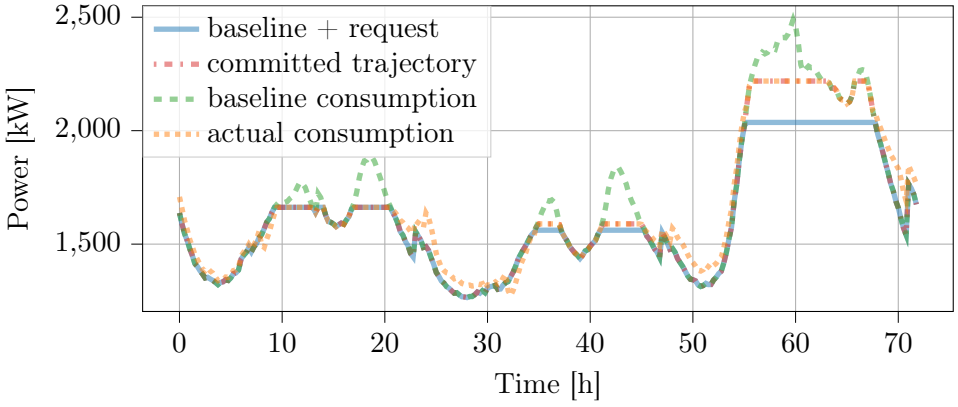


Figure 4.14 – Peak reduction request following for three days of the test period with 500 buildings and $\alpha = 1$.

Bibliography

Amadeh, A., Lee, Z. E., & Zhang, K. M. (2022). Quantifying demand flexibility of building energy systems under uncertainty. *Energy*, *246*, 123291. https://doi.org/https://doi.org/10.1016/j.energy.2022.123291

Armenise, G., Vaccari, M., Capaci, R. B. D., & Pannocchia, G. (2018). An open-source system identification package for multivariable processes. *2018 UKACC 12th International Conference on Control (CONTROL)*, 152–157.

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American mathematical society*, *68*(3), 337–404.

Arroyo, J., Manna, C., Spiessens, F., & Helsen, L. (2021). An open-ai gym environment for the building optimization testing (boptest) framework. *Proceedings of Building Simulation 2021: 17th Conference of IBPSA*, *17*, 175–182. https://doi.org/https://doi.org/10.26868/25222708.2021.30380

Aubin-Frankowski, P.-C., & Szabo, Z. (2020). Hard shape-constrained kernel machines. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (pp. 384–395). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2020/file/03fa2f7502f5f6b9169e67d17cbf51bb-Paper.pdf

Azuatalam, D., Lee, W.-L., de Nijs, F., & Liebman, A. (2020). Reinforcement learning for whole-building hvac control and demand response. *Energy and AI*, *2*, 100020. https://doi.org/https://doi.org/10.1016/j.egyai.2020.100020

Beck, A. (2015). On the convergence of alternating minimization for convex programming with applications to iteratively reweighted least squares and decomposition schemes. *SIAM Journal on Optimization*, *25*(1), 185–209.

Berkenkamp, F., Turchetta, M., Schoellig, A., & Krause, A. (2017). Safe model-based reinforcement learning with stability guarantees. *Advances in neural information processing systems*, *30*.

Bertsimas, D., & Brown, D. B. (2009). Constructing uncertainty sets for robust linear optimization. *Operations Research*, *57*(6), 1483–1495. https://doi.org/10.1287/opre.1080.0646

Bertsimas, D., Litvinov, E., Sun, X. A., Zhao, J., & Zheng, T. (2013). Adaptive robust optimization for the security constrained unit commitment problem. *IEEE Trans-*

actions on Power Systems, *28*(1), 52–63. https://doi.org/10.1109/TPWRS.2012.2205021

Bhattacharyya, S. (2017). Robust control under parametric uncertainty: an overview and recent results. *Annual Reviews in Control*, *44*, 45–77. https://doi.org/https://doi.org/10.1016/j.arcontrol.2017.05.001

Blochwitz, T., Otter, M., Åkesson, J., Arnold, M., Clauss, C., Elmqvist, H., Friedrich, M., Junghanns, A., Mauss, J., Neumerkel, D., Olsson, H., & Viel, A. (2012). Functional mockup interface 2.0: the standard for tool independent exchange of simulation models. *Proceedings of the 9th International Modelica Conference*, 173–184. https://doi.org/10.3384/ecp12076173

Blum, D., Arroyo, J., Huang, S., Drgoňa, J., Jorissen, F., Walnum, H. T., Chen, Y., Benne, K., Vrabie, D., Wetter, M., & Helsen, L. (2021). Building optimization testing framework (boptest) for simulation-based benchmarking of control strategies in buildings. *Journal of Building Performance Simulation*, *14*(5), 586–610. https://doi.org/10.1080/19401493.2021.1986574

Boodi, A., Beddiar, K., Benamour, M., Amirat, Y., & Benbouzid, M. (2018). Intelligent systems for building energy and occupant comfort optimization: a state of the art review and recommendations. *Energies*, *11*(10). https://doi.org/10.3390/en11102604

Borsche, T., Oldewurtel, F., & Andersson, G. (2014). Scenario-based mpc for energy schedule compliance with demand response [19th IFAC World Congress]. *IFAC Proceedings Volumes*, *47*(3), 10299–10304. https://doi.org/https://doi.org/10.3182/20140824-6-ZA-1003.01284

Boyd, S., Parikh, N., & Chu, E. (2011). *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. (2016). Openai gym.

Brusokas, J., Pedersen, T. B., Šikšnys, L., Zhang, D., & Chen, K. (2021). Heatflex: machine learning based data-driven flexibility prediction for individual heat pumps. *Proceedings of the Twelfth ACM International Conference on Future Energy Systems*, 160–170. https://doi.org/10.1145/3447555.3464866

Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N., Patzlaff, H., Naumann, F., & Harmouch, H. (2022). The effects of data quality on machine learning performance.

Burt, D. R., Rasmussen, C. E., & van der Wilk, M. (2020). Convergence of sparse variational inference in gaussian processes regression. *Journal of Machine Learning Research*, *21*, 1–63.

Capone, A., Lederer, A., & Hirche, S. (2022). Gaussian process uniform error bounds with unknown hyperparameters for safety-critical applications. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, & S. Sabato (Eds.), *Proceedings of the 39th international conference on machine learning* (pp. 2609–2624). PMLR. https://proceedings.mlr.press/v162/capone22a.html

Chachuat, B., Srinivasan, B., & Bonvin, D. (2009). Adaptation strategies for real-time optimization. *Computers & Chemical Engineering*, *33*(10), 1557–1567.

Chowdhury, S. R., & Gopalan, A. (2017). On kernelized multi-armed bandits. In D. Precup & Y. W. Teh (Eds.), *Proceedings of the 34th international conference on machine learning* (pp. 844–853). PMLR. https://proceedings.mlr.press/v70/chowdhury17a.html

Coraci, D., Brandi, S., Piscitelli, M. S., & Capozzoli, A. (2021). Online implementation of a soft actor-critic agent to enhance indoor temperature control and energy efficiency in buildings. *Energies*, *14*(4). https://doi.org/10.3390/en14040997

Crawley, D. B., Pedersen, C. O., Lawrie, L. K., & Winkelmann, F. C. (2000). Energyplus: energy simulation program. *ASHRAE Journal*, *42*, 49–56.

Darwazeh, D., Duquette, J., Gunay, B., Wilton, I., & Shillinglaw, S. (2022). Review of peak load management strategies in commercial buildings. *Sustainable Cities and Society*, *77*, 103493. https://doi.org/https://doi.org/10.1016/j.scs.2021.103493

Delbaen, F. (2002). Coherent risk measures on general probability spaces. *Advances in Finance and Stochastics (Essays in honour of Dieter Sondermann)*, *35*. https://doi.org/10.1007/978-3-662-04790-3_1

D'hulst, R., Labeeuw, W., Beusen, B., Claessens, S., Deconinck, G., & Vanthournout, K. (2015). Demand response flexibility and flexibility potential of residential smart appliances: experiences from large pilot test in belgium. *Applied Energy*, *155*, 79–90. https://doi.org/https://doi.org/10.1016/j.apenergy.2015.05.101

Di Natale, L., Svetozarevic, B., Heer, P., & Jones, C. (2022). Physically consistent neural networks for building thermal modeling: theory and analysis. *Applied Energy*, *325*, 119806. https://doi.org/https://doi.org/10.1016/j.apenergy.2022.119806

Diamond, S., & Boyd, S. (2016). CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, *17*(83), 1–5.

Dong, B., Prakash, V., Feng, F., & O'Neill, Z. (2019). A review of smart building sensing system for better indoor environment control. *Energy and Buildings*, *199*, 29–46. https://doi.org/https://doi.org/10.1016/j.enbuild.2019.06.025

d'Onofrio, A. (2013). *Bounded noises in physics, biology, and engineering*. Springer.

Drgoňa, J., Arroyo, J., Cupeiro Figueroa, I., Blum, D., Arendt, K., Kim, D., Ollé, E. P., Oravec, J., Wetter, M., Vrabie, D. L., & Helsen, L. (2020). All you need to know about model predictive control for buildings. *Annual Reviews in Control*, *50*, 190–232. https://doi.org/https://doi.org/10.1016/j.arcontrol.2020.09.001

Fasshauer, G. E. (2011). Positive definite kernels: past, present and future. *Dolomites Research Notes on Approximation*, *4*, 21–63.

Fiedler, C., Scherer, C. W., & Trimpe, S. (2021). Practical and rigorous uncertainty bounds for gaussian process regression. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(8), 7439–7447. https://doi.org/10.1609/aaai.v35i8.16912

Fiedler, C., Scherer, C. W., & Trimpe, S. (2022). Learning functions and uncertainty sets using geometrically constrained kernel regression. *2022 IEEE 61st Conference on*

Decision and Control (CDC), 2141–2146. https://doi.org/10.1109/CDC51059.2022.9993144

Findeis, A., Kazhamiaka, F., Jeen, S., & Keshav, S. (2022). Beobench: a toolkit for unified access to building simulations for reinforcement learning, 374–382. https://doi.org/10.1145/3538637.3538866

Fogel, E., & Huang, Y. (1982). On the value of information in system identification—bounded noise case. *Automatica*, *18*(2), 229–238. https://doi.org/https://doi.org/10.1016/0005-1098(82)90110-8

Forrest, J., Ralphs, T., Santos, H. G., Vigerske, S., Forrest, J., Hafer, L., Kristjansson, B., jpfasano, EdwinStraver, Lubin, M., Jan-Willem, rlougee, jpgoncal1, Brito, S., h-i-gassmann, Cristina, Saltzman, M., tosttost, Pitrus, B., ... to-st. (2023). *Coin-or/cbc: release releases/2.10.10* (Version releases/2.10.10). Zenodo. https://doi.org/10.5281/zenodo.7843975

Galataud, A. (2023). Rllib-energyplus [Accessed on 7 September 2023].

Gao, H., Koch, C., & Wu, Y. (2019). Building information modelling based building energy modelling: a review. *Applied Energy*, *238*, 320–343. https://doi.org/https://doi.org/10.1016/j.apenergy.2019.01.032

Gao, Y., Miyata, S., & Akashi, Y. (2023). Energy saving and indoor temperature control for an office building using tube-based robust model predictive control. *Applied Energy*, *341*, 121106. https://doi.org/https://doi.org/10.1016/j.apenergy.2023.121106

Gasser, J., Cai, H., Karagiannopoulos, S., Heer, P., & Hug, G. (2021). Predictive energy management of residential buildings while self-reporting flexibility envelope. *Applied Energy*, *288*, 116653. https://doi.org/10.1016/j.apenergy.2021.116653

Gellings, C. (1985). The concept of demand-side management for electric utilities. *Proceedings of the IEEE*, *73*(10), 1468–1470. https://doi.org/10.1109/PROC.1985.13318

Global Alliance for Buildings and Construction. (2022). *2022 global status report for buildings and construction: towards a zero-emission, efficient and resilient buildings and construction sector* (tech. rep.). International Energy Agency and the United Nations Environment Programme.

González-Sopeña, J., Pakrashi, V., & Ghosh, B. (2021). An overview of performance evaluation metrics for short-term statistical wind power forecasting. *Renewable and Sustainable Energy Reviews*, *138*, 110515. https://doi.org/https://doi.org/10.1016/j.rser.2020.110515

Gorecki, T. T., Bitlislioglu, A., Stathopoulos, G., & Jones, C. N. (2015). Guaranteeing input tracking for constrained systems: Theory and application to demand response. *2015 American Control Conference (ACC)*, 232–237. https://doi.org/10.1109/ACC.2015.7170741

Gurobi Optimization, LLC. (2023). Gurobi Optimizer Reference Manual. https://www.gurobi.com

Hao, H., Sanandaji, B. M., Poolla, K., & Vincent, T. L. (2013). A generalized battery model of a collection of thermostatically controlled loads for providing ancillary

service. *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 551–558. https://doi.org/10.1109/Allerton.2013.6736573

Hao, H., Sanandaji, B. M., Poolla, K., & Vincent, T. L. (2015). Aggregate flexibility of thermostatically controlled loads. *IEEE Transactions on Power Systems*, *30*(1), 189–198. https://doi.org/10.1109/TPWRS.2014.2328865

Hashimoto, K., Saoud, A., Kishida, M., Ushio, T., & Dimarogonas, D. V. (2022). Learning-based symbolic abstractions for nonlinear control systems. *Automatica*, *146*, 110646. https://doi.org/https://doi.org/10.1016/j.automatica.2022.110646

Hekmat, N., Cai, H., Zufferey, T., Hug, G., & Heer, P. (2021). Data-driven demand-side flexibility quantification: prediction and approximation of flexibility envelopes.

Hewing, L., Kabzan, J., & Zeilinger, M. N. (2020). Cautious model predictive control using gaussian process regression. *IEEE Transactions on Control Systems Technology*, *28*(6), 2736–2743. https://doi.org/10.1109/TCST.2019.2949757

Hirmiz, R., Teamah, H., Lightstone, M., & Cotton, J. (2019). Performance of heat pump integrated phase change material thermal storage for electric load shifting in building demand side management. *Energy and Buildings*, *190*, 103–118. https://doi.org/https://doi.org/10.1016/j.enbuild.2019.02.026

Holmgren, W. F., Hansen, C. W., & Mikofski, M. A. (2018). Pvlib python: a python package for modeling solar energy systems. *Journal of Open Source Software*, *3*(29), 884. https://doi.org/10.21105/joss.00884

Huang, S., Lutes, R., Faulkner, C. A., Vrabie, D. L., Katipamula, S., & Zuo, W. (2023). An open-source framework for simulation-based testing of buildings control strategies. *Journal of Building Performance Simulation*, *0*(0), 1–13. https://doi.org/10.1080/19401493.2023.2191220

Hughes, J. T., Domínguez-García, A. D., & Poolla, K. (2016). Identification of virtual battery models for flexible loads. *IEEE Transactions on Power Systems*, *31*(6), 4660–4669. https://doi.org/10.1109/TPWRS.2015.2505645

IEA. (2022). *World energy outlook 2022* (tech. rep.). IEA. Paris.

Jiménez-Raboso, J., Campoy-Nieves, A., Manjavacas-Lucas, A., Gómez-Romero, J., & Molina-Solana, M. (2021). Sinergym: a building simulation and control framework for training reinforcement learning agents. *Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 319–323. https://doi.org/10.1145/3486611.3488729

Katipamula, S., Haack, J., Hernandez, G., Akyol, B., & Hagerman, J. (2016). Volttron: an open-source software platform of the future. *IEEE Electrification Magazine*, *4*(4), 15–22.

Khayatian, F., Cai, H., Bojarski, A., Heer, P., & Bollinger, A. (2023). Benchmarking hvac controller performance with a digital twin. *Energy Proceedings: Clean Energy Technologies towards Carbon Neutrality*, *31*. https://doi.org/10.46855/energy-proceedings-10382

Kim, S., & Lim, H. (2018). Reinforcement learning based energy management algorithm for smart energy buildings. *Energies*, *11*(8). https://doi.org/10.3390/en11082010

Koller, T., Berkenkamp, F., Turchetta, M., & Krause, A. (2018). Learning-based model predictive control for safe exploration. *2018 IEEE Conference on Decision and Control (CDC)*, 6059–6066. https://doi.org/10.1109/CDC.2018.8619572

Lederer, A., Conejo, A. J. O., Maier, K. A., Xiao, W., Umlauft, J., & Hirche, S. (2021). Gaussian process-based real-time learning for safety critical applications. In M. Meila & T. Zhang (Eds.), *Proceedings of the 38th international conference on machine learning* (pp. 6055–6064). PMLR. https://proceedings.mlr.press/v139/lederer21a.html

Lederer, A., Umlauft, J., & Hirche, S. (2019). Uniform error bounds for gaussian process regression with application to safe control. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems*. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2019/file/fe73f687e5bc5280214e0486b273a5f9-Paper.pdf

Li, H., Wang, Z., Hong, T., & Piette, M. A. (2021). Energy flexibility of residential buildings: a systematic review of characterization and quantification methods and applications. *Advances in Applied Energy*, *3*, 100054. https://doi.org/https://doi.org/10.1016/j.adapen.2021.100054

Li, Y., & Tong, Z. (2021). Model predictive control strategy using encoder-decoder recurrent neural networks for smart control of thermal environment. *Journal of Building Engineering*, *42*, 103017. https://doi.org/https://doi.org/10.1016/j.jobe.2021.103017

Luthander, R., Widén, J., Nilsson, D., & Palm, J. (2015). Photovoltaic self-consumption in buildings: a review. *Applied Energy*, *142*, 80–94. https://doi.org/https://doi.org/10.1016/j.apenergy.2014.12.028

Lymperopoulos, I., Qureshi, F. A., Nghiem, T., Khatir, A. A., & Jones, C. N. (2015). Providing ancillary service with commercial buildings: the swiss perspective [9th IFAC Symposium on Advanced Control of Chemical Processes ADCHEM 2015]. *IFAC-PapersOnLine*, *48*(8), 6–13. https://doi.org/https://doi.org/10.1016/j.ifacol.2015.08.149

Maasoumy, M., Rosenberg, C., Sangiovanni-Vincentelli, A., & Callaway, D. S. (2014). Model predictive control approach to online computation of demand-side flexibility of commercial buildings hvac systems for supply following. *2014 American Control Conference*, 1082–1089. https://doi.org/10.1109/ACC.2014.6858874

Maddalena, E. T., Lian, Y., & Jones, C. N. (2020). Data-driven methods for building control — a review and promising future directions. *Control Engineering Practice*, *95*, 104211. https://doi.org/https://doi.org/10.1016/j.conengprac.2019.104211

Maddalena, E. T., Scharnhorst, P., Jiang, Y., & Jones, C. N. (2021). KPC: learning-based model predictive control with deterministic guarantees. *Learning for Dynamics and Control*, 1015–1026.

Mařík, K., Rojíček, J., Stluka, P., & Vass, J. (2011). Advanced hvac control: theory vs. reality [18th IFAC World Congress]. *IFAC Proceedings Volumes*, *44*(1), 3108–3113. https://doi.org/https://doi.org/10.3182/20110828-6-IT-1002.03085

Marzullo, T., Dey, S., Long, N., Vilaplana, J. L., & Henze, G. (2022). A high-fidelity building performance simulation test bed for the development and evaluation of advanced controls. *Journal of Building Performance Simulation*, *15*(3), 379–397. https://doi.org/10.1080/19401493.2022.2058091

Mathieu, J. L., Koch, S., & Callaway, D. S. (2013). State estimation and control of electric loads to manage real-time energy imbalance. *IEEE Transactions on Power Systems*, *28*(1), 430–440. https://doi.org/10.1109/TPWRS.2012.2204074

Mathieu, J. L., Vayá, M. G., & Andersson, G. (2013). Uncertainty in the flexibility of aggregations of demand response resources. *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, 8052–8057. https://doi.org/10.1109/IECON.2013.6700479

Mattsson, S. E., & Elmqvist, H. (1997). Modelica - an international effort to design the next generation modeling language [7th IFAC Symposium on Computer Aided Control Systems Design (CACSD '97), Gent, Belgium, 28-30 April]. *IFAC Proceedings Volumes*, *30*(4), 151–155. https://doi.org/https://doi.org/10.1016/S1474-6670(17)43628-7

Mesbah, A. (2016). Stochastic model predictive control: an overview and perspectives for future research. *IEEE Control Systems Magazine*, *36*(6), 30–44. https://doi.org/10.1109/MCS.2016.2602087

Micchelli, C. A., Xu, Y., & Zhang, H. (2006). Universal kernels. *Journal of Machine Learning Research*, *7*(Dec), 2651–2667.

Mitchell, S., O'Sullivan, M. J., & Dunning, I. (2011). PuLP: a linear programming toolkit for python.

Moriyama, T., De Magistris, G., Tatsubori, M., Pham, T.-H., Munawar, A., & Tachibana, R. (2018). Reinforcement learning testbed for power-consumption optimization. *Methods and Applications for Modeling and Simulation of Complex Systems*, 45–59.

Müller, F. L., Jansen, B., & Sundström, O. (2017). Autonomous estimation of the energetic flexibility of buildings. *2017 American Control Conference (ACC)*, 2713–2718. https://doi.org/10.23919/ACC.2017.7963362

Müller, F., & Jansen, B. (2019). Large-scale demonstration of precise demand response provided by residential heat pumps. *Applied Energy*, *239*, 836–845. https://doi.org/https://doi.org/10.1016/j.apenergy.2019.01.202

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. The MIT Press.

Nagy, Z., Henze, G., Dey, S., Arroyo, J., Helsen, L., Zhang, X., Chen, B., Amasyali, K., Kurte, K., Zamzam, A., Zandi, H., Drgoňa, J., Quintana, M., McCullogh, S., Park, J. Y., Li, H., Hong, T., Brandi, S., Pinto, G., ... Bernstein, A. (2023). Ten questions concerning reinforcement learning for building energy management.

Building and Environment, *241*, 110435. https://doi.org/https://doi.org/10.1016/j.buildenv.2023.110435

Nghiem, T. X., & Jones, C. N. (2017). Data-driven demand response modeling and control of buildings with gaussian processes. *2017 American Control Conference (ACC)*, 2919–2924. https://doi.org/10.23919/ACC.2017.7963394

O'Grady, T., Chong, H.-Y., & Morrison, G. M. (2021). A systematic review and meta-analysis of building automation systems. *Building and Environment*, *195*, 107770. https://doi.org/https://doi.org/10.1016/j.buildenv.2021.107770

Oldewurtel, F., Parisio, A., Jones, C. N., Gyalistras, D., Gwerder, M., Stauch, V., Lehmann, B., & Morari, M. (2012). Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy and Buildings*, *45*, 15–27. https://doi.org/https://doi.org/10.1016/j.enbuild.2011.09.022

Oldewurtel, F., Sturzenegger, D., Esfahani, P. M., Andersson, G., Morari, M., & Lygeros, J. (2013). Adaptively constrained stochastic model predictive control for closed-loop constraint satisfaction. *2013 American Control Conference*, 4674–4681. https://doi.org/10.1109/ACC.2013.6580560

Oldewurtel, F., Sturzenegger, D., & Morari, M. (2013). Importance of occupancy information for building climate control [Sustainable Development of Energy, Water and Environment Systems]. *Applied Energy*, *101*, 521–532. https://doi.org/https://doi.org/10.1016/j.apenergy.2012.06.014

O'Neill, Z., Li, Y., & Williams, K. (2017). Hvac control loop performance assessment: a critical review (1587-rp). *Science and Technology for the Built Environment*, *23*(4), 619–636. https://doi.org/10.1080/23744731.2016.1239466

Petersen, I. R., & Tempo, R. (2014). Robust control of uncertain systems: classical results and recent developments. *Automatica*, *50*(5), 1315–1335. https://doi.org/https://doi.org/10.1016/j.automatica.2014.02.042

Qureshi, F. A., Gorecki, T. T., & Jones, C. N. (2014). Model predictive control for market-based demand response participation [19th IFAC World Congress]. *IFAC Proceedings Volumes*, *47*(3), 11153–11158. https://doi.org/https://doi.org/10.3182/20140824-6-ZA-1003.02395

Qureshi, F. A., & Jones, C. N. (2018). Hierarchical control of building hvac system for ancillary services provision. *Energy and Buildings*, *169*, 216–227. https://doi.org/https://doi.org/10.1016/j.enbuild.2018.03.004

Rasmussen, C. E., & Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning*. The MIT Press. https://doi.org/10.7551/mitpress/3206.001.0001

Reynders, G., Amaral Lopes, R., Marszal-Pomianowska, A., Aelenei, D., Martins, J., & Saelens, D. (2018). Energy flexible buildings: an evaluation of definitions and quantification methodologies applied to thermal storage. *Energy and Buildings*, *166*, 372–390. https://doi.org/https://doi.org/10.1016/j.enbuild.2018.02.040

Reynders, G., & Saelens, D. (2015). Quantifying the impact of building design on the potential of structural storage for active demand response in residential buildings.

Salakij, S., Yu, N., Paolucci, S., & Antsaklis, P. (2016). Model-based predictive control for building energy management. i: energy modeling and optimal control. *Energy and Buildings*, *133*, 345–358. https://doi.org/https://doi.org/10.1016/j.enbuild.2016.09.044

Sanandaji, B. M., Hao, H., Poolla, K., & Vincent, T. L. (2014). Improved battery models of an aggregation of thermostatically controlled loads for frequency regulation. *2014 American Control Conference*, 38–45. https://doi.org/10.1109/ACC.2014.6858956

Schaback, R., & Wendland, H. (2006). Kernel techniques: from machine learning to meshless methods. *Acta numerica*, *15*, 543.

Schölkopf, B., Herbrich, R., & Smola, A. J. (2001). A generalized representer theorem. *International conference on computational learning theory*, 416–426.

Schölkopf, B., & Smola, A. J. (2018). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press. https://doi.org/10.7551/mitpress/4175.001.0001

Sha, L., Jiang, Z., & Sun, H. (2023). A control strategy of heating system based on adaptive model predictive control. *Energy*, *273*, 127192. https://doi.org/https://doi.org/10.1016/j.energy.2023.127192

Smith, R. C. (2013). *Uncertainty quantification: theory, implementation, and applications*. Society for Industrial; Applied Mathematics. https://doi.org/10.1137/1.9781611973228

Soudbakhsh, D., Annaswamy, A. M., Wang, Y., Brunton, S. L., Gaudio, J., Hussain, H., Vrabie, D., Drgona, J., & Filev, D. (2023). Data-driven control: theory and applications. *2023 American Control Conference (ACC)*, 1922–1939. https://doi.org/10.23919/ACC55779.2023.10156081

Srinivas, N., Krause, A., Kakade, S. M., & Seeger, M. W. (2012). Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, *58*(5), 3250–3265. https://doi.org/10.1109/TIT.2011.2182033

Sun, Y., Haghighat, F., & Fung, B. C. (2020). A review of the-state-of-the-art in data-driven approaches for building energy prediction. *Energy and Buildings*, *221*, 110022. https://doi.org/https://doi.org/10.1016/j.enbuild.2020.110022

Svetozarevic, B., Baumann, C., Muntwiler, S., Di Natale, L., Zeilinger, M., & Heer, P. (2022). Data-driven control of room temperature and bidirectional ev charging using deep reinforcement learning: simulations and experiments. *Applied Energy*, *307*, 118127. https://doi.org/https://doi.org/10.1016/j.apenergy.2021.118127

Swissgrid. (2023). Energy statistic switzerland 2022 [Avaiable at https://www.swissgrid.ch/en/home/customers/topics/energy-data-ch.html, accessed on 05.04.2023].

Terzi, E., Bonetti, T., Saccani, D., Farina, M., Fagiano, L., & Scattolini, R. (2020). Learning-based predictive control of the cooling system of a large business centre. *Control Engineering Practice*, *97*, 104348. https://doi.org/https://doi.org/10.1016/j.conengprac.2020.104348

Touzani, S., Prakash, A. K., Wang, Z., Agarwal, S., Pritoni, M., Kiran, M., Brown, R., & Granderson, J. (2021). Controlling distributed energy resources via deep reinforcement learning for load flexibility and energy efficiency. *Applied Energy*, *304*, 117733. https://doi.org/https://doi.org/10.1016/j.apenergy.2021.117733

Uryasev, S. (2000). Conditional value-at-risk: optimization algorithms and applications. *Proceedings of the IEEE/IAFE/INFORMS 2000 Conference on Computational Intelligence for Financial Engineering (CIFEr) (Cat. No.00TH8520)*, 49–57. https://doi.org/10.1109/CIFER.2000.844598

Van Thillo, L., Verbeke, S., & Audenaert, A. (2022). The potential of building automation and control systems to lower the energy demand in residential buildings: a review of their performance and influencing parameters. *Renewable and Sustainable Energy Reviews*, *158*, 112099. https://doi.org/https://doi.org/10.1016/j.rser.2022.112099

Vandenbogaerde, L., Verbeke, S., & Audenaert, A. (2023). Optimizing building energy consumption in office buildings: a review of building automation and control systems and factors influencing energy savings. *Journal of Building Engineering*, *76*, 107233. https://doi.org/https://doi.org/10.1016/j.jobe.2023.107233

Vázquez-Canteli, J. R., Kämpf, J., Henze, G., & Nagy, Z. (2019). Citylearn v1.0: an openai gym environment for demand response with deep reinforcement learning. *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 356–357. https://doi.org/10.1145/3360322.3360998

Vrettos, E., Oldewurtel, F., & Andersson, G. (2016). Robust energy-constrained frequency reserves from aggregations of commercial buildings. *IEEE Transactions on Power Systems*, *31*(6), 4272–4285. https://doi.org/10.1109/TPWRS.2015.2511541

Vrettos, E., Oldewurtel, F., Zhu, F., & Andersson, G. (2014). Robust provision of frequency reserves by office building aggregations [19th IFAC World Congress]. *IFAC Proceedings Volumes*, *47*(3), 12068–12073. https://doi.org/https://doi.org/10.3182/20140824-6-ZA-1003.01655

Wang, Z., Chen, B., Li, H., & Hong, T. (2021). Alphabuilding rescommunity: a multi-agent virtual testbed for community-level load coordination. *Advances in Applied Energy*, *4*, 100061.

Wang, Z., & Hong, T. (2020). Reinforcement learning for building controls: the opportunities and challenges. *Applied Energy*, *269*, 115036. https://doi.org/https://doi.org/10.1016/j.apenergy.2020.115036

Wendland, H. (2004). *Scattered data approximation*. Cambridge University Press.

Wetter, M., Benne, K. S., Gautier, A., Nouidui, T. S., Ramle, A., Roth, A., Tummescheit, H., Mentzer, S. G., & Winther, C. (2020). Lifting the garage door on spawn, an open-source bem- controls engine. *BPAC & Simbuild Conference, Virtual event.* https://api.semanticscholar.org/CorpusID:221857040

Wetter, M., Haves, P., Coffey, B., & USDOE. (2008). Building controls virtual test bed. https://doi.org/10.11578/dc.20210416.27

Yang, S., Wan, M. P., Chen, W., Ng, B. F., & Dubey, S. (2021). Experiment study of machine-learning-based approximate model predictive control for energy-efficient building control. *Applied Energy*, *288*, 116648. https://doi.org/https://doi.org/10.1016/j.apenergy.2021.116648

Yang, S., Wan, M. P., Chen, W., Ng, B. F., & Zhai, D. (2019). An adaptive robust model predictive control for indoor climate optimization and uncertainties handling in buildings. *Building and Environment*, *163*, 106326. https://doi.org/https://doi.org/10.1016/j.buildenv.2019.106326

Zhang, C., Shi, Y., & Chen, Y. (2023). Bear: physics-principled building environment for control and reinforcement learning, 66–71. https://doi.org/10.1145/3575813.3576877

Zhang, T., & Ardakanian, O. (2020). Cobs: comprehensive building simulator. *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 314–315. https://doi.org/10.1145/3408308.3431119

Zhang, Y., Duchi, J., & Wainwright, M. (2013). Divide and conquer kernel ridge regression. *Conference on Learning Theory (COLT)*, 592–617.

Zhang, Z., & Lam, K. P. (2018). Practical implementation and evaluation of deep reinforcement learning control for a radiant heating system, 148–157. https://doi.org/10.1145/3276774.3276775

Zhao, L., & Zhang, W. (2016). A geometric approach to virtual battery modeling of thermostatically controlled loads. *2016 American Control Conference (ACC)*, 1452–1457. https://doi.org/10.1109/ACC.2016.7525121

Zhao, L., Zhang, W., Hao, H., & Kalsi, K. (2017). A geometric approach to aggregate flexibility modeling of thermostatically controlled loads. *IEEE Transactions on Power Systems*, *32*(6), 4721–4731. https://doi.org/10.1109/TPWRS.2017.2674699

Paul Scharnhorst

Rue Edmond de Reynier 2
2000 Neuchâtel, Switzerland

+41767491184
paul.scharnhorst@csem.ch
www.linkedin.com/in/paul-scharnhorst

Research Experience

PhD candidate in data-driven control for energy systems Oct. 2019 – Present
CSEM S.A. and EPFL *Neuchâtel, Switzerland*

- Co-developed the Python library Energym for controller performance benchmarking on building systems
- Developed a virtual battery model for predicting consumption flexibility of building systems
- Derived deterministic error bounds for function approximation with kernel methods

Education

Mathematics M.Sc. Dec. 2015 – Jun. 2019
Otto-von-Guericke University Magdeburg *Magdeburg, Germany*

- Specialisations: Numerical Optimization, Numerical Analysis
- Master thesis: "Optimal Control Methods for Deep Learning"

Mathematics B.Sc. Oct. 2011 – Dec. 2015
Otto-von-Guericke University Magdeburg *Magdeburg, Germany*

- Bachelor thesis: "Endliche Automaten zum Verständnis von Multi-Level AV Block" (Simulation of heart arrhythmia using finite automata with MatLab)

Publications

PS, Schubnel, B., Carrillo, R.E., Alet, P.-J., Jones, C.N. (Accepted for ACC 2023). "Uncertainty-aware Flexibility Envelope Prediction in Buildings with Controller-agnostic Battery Models".

PS*, Maddalena, E.T.*, Jiang, Y., Jones, C.N. (2023). "Robust Uncertainty Bounds in Reproducing Kernel Hilbert Spaces: A Convex Optimization Approach". *IEEE Transactions on Automatic Control*, vol.68, no. 5, pp. 2848-2861.

Maddalena, E.T., **PS**, Jiang, Y., Jones, C.N. (2021). "KPC: Learning-Based Model Predictive Control with Deterministic Guarantees". *Learning for Dynamics and Control. PMLR*, pp. 1015–1026.

Maddalena, E.T*, **PS***, Jones, C.N. (2021). "Deterministic error bounds for kernel-based learning techniques under bounded noise". *Automatica*, vol. 134, p. 109896.

PS, Schubnel, B., Bandera, C.F., Salom, J., Taddeo, P., Boegli, M., Gorecki, T., Stauffer, Y., Peppas, A., Politi, C. (2021). "Energym: A Building Model Library for Controller Benchmarking". *Applied Sciences*, vol. 11, no. 8.

Skills

Programming Languages: Python, basic knowledge in MatLab and C++
Languages: native German, full professional English, Elementary French

Other Experience

Student assistant Nov. 2018 – Feb. 2019
Institute of Material and Joining Technology, OvGU Magdeburg *Magdeburg, Germany*
- Implemented the automatised evaluation of measurement data with Python

Intern Oct. 2017 – Jan. 2018
IAV GmbH *Gifhorn, Germany*
- Implemented GUI-features for a simulation tool with PyQt

Student tutor Sep. 2013 – Jan. 2017
OvGU Magdeburg *Magdeburg, Germany*
- Held weekly tutorials for economics and engineering students

Extracurricular Activities

University ball co-organiser Nov. 2014 – Jul. 2015
University ball of the OvGU Magdeburg *Magdeburg, Germany*
- Co-organised a ball for 400 university members. Managed the finances of the project.

Financial Manager Nov. 2014 – Jul. 2016
Cultural department, OvGU student council *Magdeburg, Germany*
- Decided over financial support for cultural projects.

Student council member Dec. 2013 – Jul. 2015
Faculty of Mathematics, OvGU Magdeburg *Magdeburg, Germany*
- Represented the students of the faculty and organised events.