# Advanced Optimization Strategies for the Unification of Process Design and Predictive Control

A Thesis

Presented in Partial Fulfillment of the Requirements for the Degree Master of Science in the Graduate School of The Ohio State University

By

Naitik Choksi, BS

Graduate Program in Chemical & Biomolecular Engineering

The Ohio State University

2021

Master's Examination Committee:

Dr. Joel Paulson, Advisor

Dr. Bhavik Bakshi

# Abstract

A recent trend in the chemical process industries is the move toward designing safer, greener processes, while also trying to maximize profit. Profit is a strong function of the chosen design and the employed operation/control strategies, which have been traditionally treated separately. This sequential strategy can lead to unnecessarily rigid designs that cannot operate adequately under changes in the market, environment, or regulations that are sure to happen over time. A class of methods, commonly referred to as integrated design and control (IDC), have been developed to address this issue. The goal of IDC is to systematically account for the effect of the "here-and-now" design decisions (fixed throughout the process lifetime) on the "wait-and-see" control decisions (changed throughout operation) while attempting to maximize profit. Since the selected control structure can be an arbitrary function of the measurements, IDC is an extremely complicated multi-stage stochastic optimization problem that cannot be solved in most realistic applications. As such, most available IDC methods focus on simplified parameterizations of the control structure and a small set of disturbance realizations. Although the resulting optimization problem is simpler to solve, it comes at the cost of a (potentially large) drop in solution quality. The goal of this thesis is to develop a unified and tractable framework that can provide accurate solutions to IDC problems by combining high-quality controllers with state-of-the-art constrained black-box optimization methods. In particular, we propose the use of design-dependent model predictive controllers to select the control inputs at every operational period. This allows us

to consistently determine high-quality control actions, even when the system has important safety/quality constraints, significant multivariable interactions, and/or nonlinear/mixed-integer dynamics. Since the resulting closed-loop simulations are computationally expensive, we propose a Bayesian optimization (BO) method to sequentially select potential "good" designs using a probabilistic machine learning-based surrogate model. We demonstrate the effectiveness of the proposed BO framework on the design of a flexible building cooling system under weather and demand uncertainties. We also discuss strategies to further reduce the number of expensive closed-loop evaluations by taking advantage of computationally cheaper low-fidelity approximations to the IDC cost function. We further demonstrate how the multi-fidelity extension of BO allows us to much more efficiently handle year-long planning horizons with key disturbances evolving at the hourly scale. Lastly, motivated by these promising results, we discuss several potential directions for future work on *grey-box* BO, which are methods that exploit additional sources of information to improve upon the convergence rate of traditional BO methods, in the context of IDC problems.

*Dedicated to my parents, my extended family and all associated with The Ohio State University*

# Acknowledgments

First and foremost, I want to acknowledge and thank my advisor Dr. Joel A. Paulson for his huge contribution to the knowledge that I acquired during the journey of my Masters degree. Without his guidance, I would not have been able to accomplish this work. When I started my Masters, I was one of the first amongst the group of Dr. Paulson's students. He has been a constant source of support and has immensely given up his time to help me whenever required, especially with complex mathematical concepts and advanced control strategies. I would also like to thank my committee member Dr. Bakshi for taking the time to review my thesis and for his valuable comments.

The past two years would not have been smooth if not for these people's presence. I would like to thank all members of the Paulson group (Howie, Kevin, Farshud, Akshay) for being there, for all warm, silly and fun conversations we had along with constant words of encouragement during difficult times. I am eternally grateful to all faculty members whom I had the opportunity to learn and gain insights from their immense pool of knowledge. I would like to thank the entire chemical engineering family at Ohio State which made me feel privileged to be a part of the CBE community.

I would like to acknowledge the contribution of my friends Bhavya and Harsh towards my holistic development over the past two years. I also thank them for being constant pillars of support and making life much more fun in the United States.

Lastly, but most importantly, the ones without which I would not be the individual that I am today, my parents and my extended family. A special mention to my parents who made me understand the importance of education, encouraged me to pursue my dreams and supported me in all decision that I made at every important juncture of my life. Thank you for all that you have done for me. Thank you to my extended family for being so supportive and amazing.

# Vita

August 2021 ..... MS,
Chemical and Biomolecular Engineering,
The Ohio State University, USA.

May 2019 ..... B.Tech,
Chemical Engineering,
Nirma University, India

# Fields of Study

Major Field: Chemical & Biomolecular Engineering

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1: Introduction

Chemical process design is an important area of chemical engineering as it largely determines the operating conditions and the inherent safety of the process under different conditions. Process control is a method used to monitor and adjust any process to maintain quality and improve performance. Typically, process design and process control are procedures that are performed sequentially. The equipment sizes and the operating conditions are first estimated at steady state by optimizing the economics of the process. This procedure is followed by evaluating the controllability of the process, by assessing the closed-loop dynamic response of the process to various forms of uncertainty (e.g., external disturbances, unknown parameters, or process faults and failures). A certain control performance has to be achieved, which is accomplished by selecting control structures, algorithms and the tuning parameters of the structures.

The underlying issue with the sequential procedure is the rigidity of the design, which may result in dynamic constraint violations in presence of uncertainty. Any of these violations result in a significant increase in the operating cost of the process. Over-design results in a compromise with the capital cost to accommodate the uncertainty in the process. Figure 1.1 illustrates the concept. Therefore, a key challenge is to develop a systematic methodology to simultaneously address the process design and control in order to explicitly account for process dynamics while optimizing the design of the system. The indicated

approach results in designs (operating conditions and equipment sizes) expected to determine a process which has dynamic feasibility in presence of process disturbances or any parametric uncertainties, while being the most economically advantageous. This is because the method adjusts the design of the plant to operate the plant within feasible limits according to the system dynamics. The aforementioned procedure is called Integrated Design and Control (IDC) and represents a broad class of problems to simultaneously tackle design and control of a process under process disturbances and uncertainty.

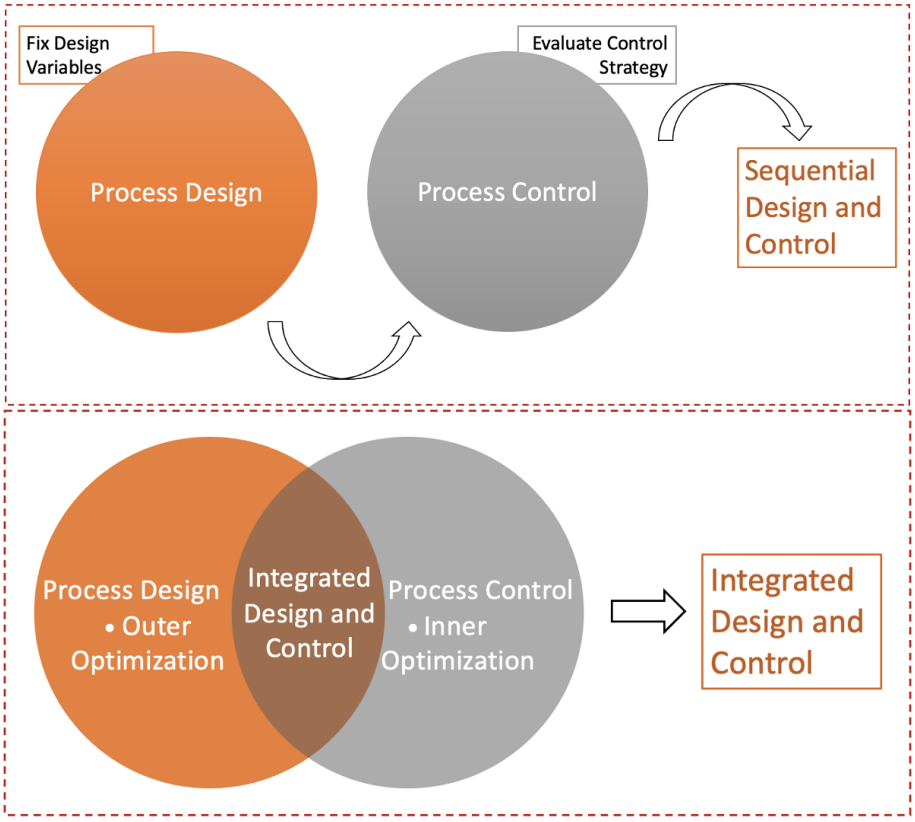


Figure 1.1: Sequential versus Integrated Design and Control

From this discussion, it is clear that IDC has a huge potential for overcoming the limitations of the conventional sequential approach, which is especially important in emerging next-generation processes that require a much higher degree of operational flexibility. This thesis moves towards the development of an efficient solution approach for the challenging IDC problem and showcases improvements over the current state of the art.

## 1.1 Review of Existing Solution Approaches

Typically, process design and process control are procedures that are performed sequentially. But it was discovered 80 years ago, by Ziegler and Nichols [1], that the control scheme of a system or a process is inherently dependent on the process design involved. Since then, a lot of work has been done to integrate control into process design optimization and a summary of a variety of different contributions is presented in Table 1.1. Initial efforts to integrate process design and control were aimed at understanding the dependencies of each mechanism on the other. One of the earliest concepts was to assess the controllability of the process design, termed as "dynamic resilience" by Morari et al. [2, 3], which was defined as "the maintenance of satisfactory performance despite adverse conditions", while flexibility was defined as "the ability to handle alternately desirable operating conditions". This work was a foundation for extensive research in the field. Development of dynamic resiliency indices were based on (i) plant-model mismatch, (ii) constrained control variables, (iii) presence of time delays, and (iv) presence of right-half place zeros in the system transfer function matrix. These indices helped identify more resilient designs in many different types of systems including reactor networks [4, 5], separation systems [6] and heat exchanger networks [7]. The main downside to these types of approaches is that they are only applicable for a limited range of operation and thus do not directly address flexibility

by considering many different operating conditions depending on the particular realizations of the uncertainties. To address this issue, the notion of a flexibility index was introduced by Swaney and Grossmann [8], which can be formulated as a two-stage robust optimization problem. When only a discrete set of uncertainty scenarios are considered, the flexible design problem reduces to a multiperiod design problem, which assumes perfect recourse to the uncertainties and instantaneous equilibration to a steady-state in each period.

To more directly handle the generally nonlinear process dynamics, Mohideen et al. [61] introduced a Mixed Integer Dynamic Optimization (MIDO) formulation for a multiperiod design and control problem accounting for dynamic variations due to uncertain parameters and disturbances. The uncertain parameters were picked from a range of discrete scenario sets or events, while disturbances followed a user defined function. To demonstrate the efficacy, the approach was tested on a mixing tank problem and a ternary distillation problem and then compared against the conventional sequential approach. Bansal et al. [28, 29] solved a dynamic optimization problem using the methodology with the intention to evaluate the disturbance profile producing the worst case scenario. A prominent contribution to the already burgeoning research interest in the simultaneous and design problem was integrating a Proportional Integral (PI) controller in the design optimization problem. Walsh and Perkins [23] proposed an integrated process design and PI control scheme, which was applied to a wastewater neutralization case study. It was effective for a single input single output (SISO) process, but it could not handle multiple input multiple output (MIMO) processes without large and complex changes in the feedback loop structure. Luyben and Floudas [19] developed a procedure to systematically assess the economic performance of every input-output pair on the basis of the controllability indices they developed before [62], within a multiple objective mixed integer non-linear programming (MINLP) problem. The

Table 1.1: Contributions to the field of integrated design and control (IDC).

| Topic | Control policy | Author (Year) | Contribution |
|---|---|---|---|
| Controllability and stability metrics | | Morari & co-workers (1982) [9], Perkins & co-workers (1991) [10], Bogle & co-workers (1989, 2000) [11, 12], Douglas & co-workers (1988) [13–15] | Controllability metrics included as constraints or objectives within the optimization formulation |
| Two-stage flexible process design | | Pistikopolous & co-workers (1994, 1997, 2001) [16–18], Floudas & co-workers (1994, 2000, 2001) [19–21], Romagnoli & co-workers (1996) [22], Perkins & co-workers (1994) [23], Floudas & co-workers (1998) [24], Exler & co-workers (2008) [25], Asteasuain & co-workers (2006) [26], Vega & co-workers (2009) [27]. | Flexibility, feasibility, and resilience considerations in steady state (MI)NLP optimization under scenario approximation of uncertainties |
| MIDO | PID | Pistikopoulos & co-workers (2000, 2002) [28, 29], Ricardez-Sandoval & co-workers (2008, 2012, 2016, 2017) [30–33], Swartz & co-workers (2014) [34], Biegler & co-workers (2008, 2018) [35, 36] | Substitutes explicit decision rule and nominal disturbance sequence and then converts (MI)DO into (MI)NLP using collocation or decomposition |
| | Linear MPC | Pistikopoulos & co-workers (2003, 2004, 2015g, 2017g) [37–41], Ricardez-Sandoval & co-workers (2011, 2013, 2014, 2015) [42–45], Engell & co-workers (2004) [46], | |
| DR-SO | Adaptive thresholding | Hakizimana (2019) [47], Zhang & co-workers (2016) [48], Evins & co-workers (2015) [49], Lambert & co-workers (2006) [50] | Substitutes decision rule for control policy to derive single-stage IDC problem tackled via simulation optimization method |
| | Nonlinear MPC | Paulson & co-workers (2021) [51], Li & co-workers (2017) [52] | |
| Review articles | | Sharifzadeh(2013) [53], Gani and co-workers(2012) [54], Vega and co-workers(2014) [55, 56], Ricardez Sandoval and co-workers(2009) [57], Pistikopolous and co-workers (2004, 2016, 2019) [58–60] | |

procedure was tested on a heat-integrated distillation system and a reactor-separator-recycle system. The aforementioned design and control frameworks using PI control can be used for non-linear processes. But, the range of problems that can be solved are limited due to the presence of plant-model mismatch (process is non-linear, control model is linear).

Ricardez-Sandoval et al. [63] combined robust control tools with the back-off approach for PI control integration in the design optimization problem. The Tennessee Eastman Process was solved using the proposed approach [64].

There are many drawbacks of using PI control in the design optimization problem, the biggest one being its inability to handle MIMO processes. PI controllers also cannot handle any constraints that arise in a process due to environmental, operational or safety considerations. Model Predictive Control (MPC) overcomes these limitations as it handles the constraints using an explicit model of the process that is subject to a constrained dynamic optimization problem. Brengel and Seider [65] were the first to integrate MPC in a non-linear design optimization problem, resulting in a bi-level optimization with an economic objective as the leader followed by the MPC formulation. This strategy was intractable for complex systems, due to numerical calculations of the second derivatives, which is a huge limitation of embedding MPC into the IDC problem. To tackle this limitation, Bemporad et al. [66] suggested a novel strategy to treat the initial conditions as parameters which allowed the derivation of piecewise affine control laws and called it multi-parametric MPC (mpMPC) or explicit MPC. Sakizlis et al. [37] proposed a bi-level MIDO problem formulation for design and control integration. The authors used the explicit nature of the mpMPC solution to reduce the complexity of the bi-level problem to a single level optimization. Sanchez-Sanchez and Ricardez-Sandoval [67] compared the MPC integrated framework to PI control based integration framework, using a system of CSTR's to assess the performance of each. The MPC formulation achieved superior results with lower investment and operating costs, proving its eminence. Vega and Lamanna [68] proposed a methodology for a multi-objective IDC and showcased its performance on a waste water treatment plant. However, the problem was still intractable for large-scale complex problems, as the MPC problem had to be

solved at every iteration which consisted of repetitive linearizations along with solving a multiparametric programming problem at every iteration. Diangelakis et al. [41] derived a "design dependent offline controller" and integrated the control law in the high fidelity model, which overcame the limitation of intractability by reducing the bi-level optimization to a single design and control optimization problem. This approach was tested on a CSTR, a mixing tank and a residential combined heat and power (CHP). Other works that use MPC for the integrated design and control problem can be found here [44, 69–72]. Reviews in IDC can be found here [53–55, 57–60].

## 1.2 Summary of Thesis Contributions

Although there has been a substantial amount of work on IDC as discussed in the previous section, a unified, tractable, and generally applicable framework for tackling optimal IDC problems has yet to be developed. The main aim of this thesis is to work toward the development of such a framework by combining advances in machine learning and optimization-based control strategies. The developed algorithms are tested on challenging applications, with the goal of showcasing the strengths of our initial framework relative to widely-used alternatives. We also highlight several important directions for future work that would further improve this framework when deployed in practice.

Before discussing the details of the proposed IDC framework, we first summarize the challenges with optimal IDC problems. As recently discussed in [47], optimal IDC problems can be generally formulated as multistage stochastic programs (MSPs) where the term “stage” refers to a single operational period. The following features, which are present in many emerging applications, result in an intractable MSP formulation of the IDC problem:

(i) dynamics and uncertainties occur on much shorter timescales than the system lifetime;

(ii) uncertainties are continuous random variables with potentially large variance;

(iii) key operational decisions are discrete, e.g., adaptive scheduling and unit commitment.

Features (i) and (ii) imply that a large number of operational periods and uncertainty scenarios, respectively, must be considered in the MSP model. These are significant challenges to overcome as MSPs are known to suffer from the curse-of-dimensionality [73] due to the fact that the control decisions can be different for different uncertainty realizations. Feature (iii), on the other hand, implies that the system model is nonlinear with mixed-integer decisions. As such, scenario-based approximations to MSPs, which consists of co-optimizing design decisions with control decisions for every stage and every scenario, easily results in extremely large-scale mixed-integer nonlinear programs (MINLPs) that are far beyond the capabilities of existing solution paradigms. Although the many different approximation methods (Table 1.1) can be employed to reduce complexity, they can degrade the accuracy of the IDC problem in exactly the ways that are important for realistically capturing the operational flexibility of nonlinear systems under possibly multi-scale uncertainty.

The proposed IDC framework in this thesis is composed of two main components: a high-quality *decision rule* (DR) that maps the measured data to the control decisions at every time step and an efficient *black-box* optimization method that is used to co-optimize the design variables and tuning parameters of the DR. The DR is essentially an offline parametrization of the control decisions and, in light of feature (iii), should be some type of an advanced controller that can flexibly handle scheduling and control decisions (that may need to be made at different timescales). As we will discuss in more detail later in the thesis, the model predictive control (MPC) paradigm is able to nicely serve this purpose since it adaptively optimizes the predictions of an underlying dynamic system model. This implies that MPC can straightforwardly handle multivariate nonlinear systems in the presence of

constraints, general objective functions, and uncertainty [39–41, 44, 45, 52]. Another key advantage of MPC, compared to other potential DRs, is that it can be parametrized by relatively few dimensions whenever a reasonable process model is available. As such, the MPC-based IDC problem is able to scale much more favorably with respect to the number of operational periods and uncertainty scenarios than the original MSP.

The main challenge with MPC-based DRs are that they are implicitly defined by the solution to an optimization problem at every time step, i.e., they are computationally intensive and non-differentiable. Since we cannot rely on gradient-based optimization algorithms to determine the optimal design variables, we must resort to black-box (or derivative-free) simulation optimization. We can think of these methods as being composed of an *outer* optimization over the relevant design variables and an *inner* stochastic closed-loop simulation to approximately evaluate the expected value of the long-term operating cost in the outer problem. Although many black-box simulation optimization methods are available, we require an approach that performs as few expensive closed-loop simulations as possible so that an adequate design can be identified within a fixed amount of computational resources. This precludes the use of many of the widely-used approaches, such as genetic algorithms [74], particle swarm optimization [75, 76], partition-based methods [77] and simulated annealing [78], as they require too many function evaluations to be practically useful in the context of MPC-based IDC problems. As such, we focus on Bayesian optimization (BO) that was specifically constructed to handle noisy observations of expensive functions. Not only has BO been successfully applied to many problems in the past few years including hyperparameter optimization in machine learning models [79], material design and discovery [80], aircraft design [81], and automated controller tuning [82, 83], but it has also been extended to handle fully black-box constraints that often appear in IDC

problems. Therefore, the main goal of this thesis is to demonstrate how MPC-based DRs and BO can be flexibly integrated to address IDC problems that exhibit the challenging features (i)–(iii) above.

The remainder of this thesis is organized as follows. Chapter 2 provides a comprehensive MSP formulation of optimal IDC problems and introduces the mathematical description of DRs. Chapter 3 provides a detailed overview of the BO algorithm as well as how it can be extended to accommodate black-box constraints[1]. The proposed framework is then applied to the design of a flexible heating and cooling system for a building wherein we show how both scheduling and control decisions can be made simultaneously with MPC. We also compare the proposed framework to sequential design and control methods and find that our method is able to find substantially better designs (in terms of lower cost and improved constraint satisfaction). Chapter 4 develops a more realistic model of the building heating and cooling system problem that incorporates solar power, battery storage, and grid support over a year-long planning horizon with realistic disturbance models based on data obtained from publicly available databases. Since the overall model is much more computationally intensive (i.e., each simulation requires multiple hours to complete), we further extend BO to take advantage of computationally cheaper approximations to better guide the search process. We refer to this process as multi-fidelity BO, and demonstrate its ability to further reduce the number of high-fidelity model evaluations to determine good and flexible designs. Lastly, we summarize the main conclusions of the thesis and describe several important avenues for future work in Chapter 5.

[1]Note that a comparison of BO to alternative black-box optimization methods is provided in Appendix A.

# Chapter 2: High-fidelity Approximations to Integrated Design and Control Problems using Predictive Control-based Decision Rules

In this chapter, we provide a detailed and general formulation of the optimal integrated design and control (IDC) problem. Even though the presented formulation is completely intractable, it is useful to treat this as an ideal baseline so we can better understand any employed approximations; as noted in Table 1.1 in the introduction, there are many different approximations available and so we must be careful not to lose out on the critical features of the problem needed to identify good designs. We then formally introduce the notion of a decision rule (DR) and highlight how that simplifies the IDC problem. Lastly, we describe the model predictive control (MPC) framework that is a particularly effective DR as it relies on repeated optimization of dynamic predictions of the process. As such, MPC can flexibly handle different types of dynamic systems, constraints, control objectives, and uncertainties. The effectiveness of MPC as a DR, relative to alternatives such as proportional-integral-derivative control, is demonstrated on a continuous reactor case study.

Note that some of the contents of this chapter have been adapted from a publication by the author in the Proceedings of the 2021 American Control Conference [51].

*Notation:* Throughout this chapter, the sets of non-negative and positive integers are denoted by $\mathbb{N}$ and $\mathbb{N}_+$ respectively. Non-negative real numbers are denoted by $\mathbb{R}_{\geq 0}$. Given column vectors $x$ and $y$, $(x,y) = [x^\top, y^\top]^\top$ denotes vector concatenation. Let $x_{i:j} =$

$(x_i, x_{i+1}, \ldots, x_j)$ be the subsequence of $\{x_i\}_{i\in\mathbb{N}}$ from $i$ to $j$. For $S \subset \mathbb{R}^{n_s}$, the set of all essentially bounded measurable maps from $S$ into $\mathbb{R}^m$ is denoted by $L^\infty(S, \mathbb{R}^m)$.

## 2.1 Multistage Stochastic Programming Formulation of IDC

Consider the following time-varying nonlinear system under uncertainty

$$x_{k+1} = f(k, x_k, u_k, w_k, d), \tag{2.1}$$

where $x_k \in X \subset \mathbb{R}^{n_x}$ are the system states, $u_k \in U \subset \mathbb{R}^{n_u}$ are the control inputs, $w_k \in W \subset \mathbb{R}^{n_w}$ are the uncertainties, and $d \in D \subset \mathbb{R}^{n_d}$ are the design variables. We are interested in a general state-space multistage stochastic program (MSP) with $T$ stages, which is affected by the disturbance sequence $\boldsymbol{\omega} = (w_0, \ldots, w_T)$ with random variables $w_k$ that are revealed after each stage $k \in \mathcal{T} = \{0, \ldots, T-1\}$. We assume the random sequence $\boldsymbol{\omega}$ has probability density $p_{\boldsymbol{\omega}} : \Omega \to \mathbb{R}$ defined on the finite support set $\Omega = W \times \cdots \times W$.

The design variables $d$ are the first-stage "here-and-now" decisions while the control inputs act as recourse "wait-and-see" decisions that can adapt over time to particular uncertainty realizations, i.e., $u_k(\boldsymbol{\omega})$. We make no assumption about $f : \mathcal{T} \times X \times U \times W \times D \to X$ or the recourse decisions $u_k(\boldsymbol{\omega})$, which can both contain a mixture of continuous and discrete states/inputs. Thus, $f$ can be interpreted as a *black-box* function for which we only need the ability to simulate (2.1) to compute $\{x_k(\boldsymbol{\omega})\}_{k=0}^T$ for a fixed design $d$, uncertainty sequence $\boldsymbol{\omega}$, and initial state $x_0(\boldsymbol{\omega}) = b_0$. We also assume the system is subject to joint chance constraints

$$\mathbb{P}_{\boldsymbol{\omega}}\{g(k, x_k(\boldsymbol{\omega}), u_k(\boldsymbol{\omega}), d) \leq 0\} \geq 1 - \varepsilon, \tag{2.2}$$

where $\mathbb{P}_{\boldsymbol{\omega}}\{A\}$ denotes the probability of event $A$ with respect to the random disturbance sequence $\boldsymbol{\omega}$ and $\varepsilon \in [0,1]$ denotes the allowed probability of constraint violation. The

function $g(\cdot)$ can again be anything, and will typically represent critical safety and/or quality constraints. Although we specify the constraints in the form of (2.2), many different other formulations could be considered including expectation-based constraints that are used in Chapter 4. Because these constraints are potentially time-varying, they also can be used to constraint the terminal state through proper selection of the $g(\cdot)$ functions.

Given the system dynamics (2.1) and constraints (2.2), we can formulate the integrated design and control (IDC) problem in terms of the following MSP optimization problem

$$\begin{aligned}
\min_{\substack{d\in D\\ u_k\in L^\infty(\Omega,U)}} \quad & C(d)+\mathbb{E}_{\boldsymbol{\omega}}\left\{\sum_{k=0}^{T-1}\ell(k,x_k(\boldsymbol{\omega}),u_k(\boldsymbol{\omega}),d)\right\},\\
\text{s.t.} \quad & x_{k+1}(\boldsymbol{\omega})=f(k,x_k(\boldsymbol{\omega}),u_k(\boldsymbol{\omega}),w_k,d),\\
& x_0(\boldsymbol{\omega})=b_0,\ \ \forall\boldsymbol{\omega}\in\Omega,\\
& \mathbb{P}_{\boldsymbol{\omega}}\{g(k,x_k(\boldsymbol{\omega}),u_k(\boldsymbol{\omega}),d)\le 0\}\ge 1-\varepsilon,\\
& u_k(\boldsymbol{\omega}) \text{ is non-anticipative},\\
& \forall k\in\{0,\ldots,T-1\},
\end{aligned} \tag{2.3}$$

where $C: D\to\mathbb{R}$ denotes the capital cost that is only a function of the design variables, $\ell:\mathcal{T}\times X\times U\times D\to\mathbb{R}$ is the operating cost at each stage, $\mathbb{E}_{\boldsymbol{\omega}}\{\cdot\}$ denotes the expected value with respect to the random disturbance sequence $\boldsymbol{\omega}$, and $u_k(\boldsymbol{\omega})$ satisfying *non-anticipativity constraints* implies that $u_k$ must be determined at time $k$ based on previous knowledge of $w_j$ with $j<k$, which is needed to model the fact that current decisions cannot depend on future measurements. This can be explicitly stated by the constraint $u_k(\boldsymbol{\omega})=u_k(\hat{\boldsymbol{\omega}})$ for all $\boldsymbol{\omega},\hat{\boldsymbol{\omega}}\in\Omega$ with $w_{0:k-1}=\hat{w}_{0:k-1}$. Therefore, the MSP (2.3) selects the first-stage decisions $d$ that simultaneously minimize the capital cost $C(\cdot)$ and expected value of the operating cost (defined as the sum of stage-wise costs $\ell_S(\cdot)$) under optimal operations in each stage $k\in\mathcal{T}$.

The MSP as stated in (2.3) is intractable for several reasons. First, the set of functions $L^{\infty}(\Omega, U)$ is infinite dimensional, as they can be arbitrary functions of $\boldsymbol{\omega}$. Second the probabilistic operators $\mathbb{E}_{\boldsymbol{\omega}}\{\cdot\}$ and $\mathbb{P}_{\boldsymbol{\omega}}\{\cdot\}$ cannot be computed exactly except for in very special and restrictive cases. Third, the relevant operational details and uncertainty realizations often occur at time-scales that are significantly shorter than the lifetime of the system (e.g., the value of a next-generation energy storage system with a 10 year lifetime may critically depend on its ability to respond to hourly variations in electricity pricing or renewable power generation). Lastly, the dynamics are likely to include discrete operational decisions, which results in a hybrid system with mixed-integer recourse decisions – leading to a very challenging large-scale, non-convex, and non-smooth optimization problem.

As recently noted in [47], a promising approach to address some of the aforementioned challenges is to approximate the recourse decisions $u_k(\boldsymbol{\omega})$ with a parametrized *decision rule* that directly enforces the non-anticipativity constraints and avoids the need to optimize over infinite dimensional spaces. We focus on state feedback for simplicity. The DR approximation is formally introduced next, which is followed by a discussion on the proposed high quality MPC-based DR.

**Remark 1.** *It is important to note that we have assumed that the disturbance sequence is measurable in* (2.1)*. This indirectly implies that the state is measurable as we could always use the model and disturbance realizations to reconstruct the state at any time $k \in \mathcal{T}$. Throughout this thesis, we focus on the full state feedback case for simplicity of exposition. However, the proposed framework can be straightforwardly extended to the more general output feedback case by incorporating state and/or disturbance estimators. This is a big advantage of the proposed framework, as alternative methods cannot easily be extended to cases where the state is not measured. Interested readers are referred to [82]*

*for initial results in this direction. In this case, however, additional tuning parameters in the estimator(s) are introduced into the DR, which does increase the complexity of the black-box optimization method discussed in Chapter 3.*

## 2.2 Decision Rule Approximation Methods

We can formally define a decision rule (DR) as follows

$$u_k(\boldsymbol{\omega}) = \kappa(k, x_k(\boldsymbol{\omega}), w_k, d, \gamma), \tag{2.4}$$

where $\kappa : \mathcal{T} \times X \times W \times D \times \Gamma \to U$ is a fixed structure function that can depend on parameters $\gamma \in \Gamma \subset \mathbb{R}^{n_\lambda}$ that can be optimized simultaneously with the first-stage design decisions $d$. Although the fixed structure of $\kappa$ potentially introduces some level of suboptimality, it eliminates the need to optimize over the infinite dimensional space $L^\infty(\Omega, U)$ since $\kappa$ ensures the value of $u_k(\boldsymbol{\omega})$ is fixed for each $(\boldsymbol{\omega}, d, \gamma) \in \Omega \times D \times \Gamma$. Moreover, $\kappa$ directly enforces non-anticipativity of $u_k(\boldsymbol{\omega})$ that eliminates the need to consider the infinite number of non-anticipativity constraints in (2.3).

Let us define $\theta = (d, \gamma) \in \Theta = D \times \Gamma$ as the concatenated vector of the design variables and the DR parameters. With a slight abuse of notation, we define the sequence of the state and control inputs by the following recursion

$$\begin{aligned} x_0(\boldsymbol{\omega}, \theta) &= b_0, \\ u_k(\boldsymbol{\omega}, \theta) &= \kappa(k, x_k(\boldsymbol{\omega}, \theta), w_k, \theta), \\ x_{k+1}(\boldsymbol{\omega}, \theta) &= f(k, x_k(\boldsymbol{\omega}, \theta), u_k(\boldsymbol{\omega}, \theta), w_k, \theta), \end{aligned} \tag{2.5}$$

for all admissible disturbance sequences and first-stage decisions $(\boldsymbol{\omega}, \theta) \in \Omega \times \Theta$. By substituting these equations into (2.3), we can define the *decision rule approximation* (DRA)

of the MSP as follows

$$\begin{aligned} \min_{\theta\in\Theta}\ & C(\theta)+\mathbb{E}_{\boldsymbol{\omega}}\{O(\boldsymbol{\omega},\theta)\}, \\ \text{s.t. }\ & \mathbb{P}_{\boldsymbol{\omega}}\{G_k(\boldsymbol{\omega},\theta)\le 0\}\ge 1-\varepsilon, \quad \forall k\in\{0,\dots,T-1\}, \end{aligned} \tag{2.6}$$

where the operating cost $O(\cdot)$ and chance constraint $G_k(\cdot)$ functions are given by

$$O(\boldsymbol{\omega},\theta)=\textstyle\sum_{k=0}^{T-1}\ell(k,x_k(\boldsymbol{\omega},\theta),u_k(\boldsymbol{\omega},\theta),\theta), \tag{2.7}$$

$$G_k(\boldsymbol{\omega},\theta)=g(k,x_k(\boldsymbol{\omega},\theta),u_k(\boldsymbol{\omega},\theta),\theta). \tag{2.8}$$

The optimization (2.6) is now a single-stage problem with a finite number of decisions $\theta$, which is a large simplification of (2.3), particularly for problems in which the model includes short-time scale operations over a long horizon. To ensure (2.6) provides a sufficiently good solution, we have to define a high-quality DR in terms of a relatively small number of parameters in order to make the problem remains tractable. We can broadly categorize DRs as either *explicit* or *implicit*. Explicit DRs are defined by (often simple) analytic expressions of the most recently measured state and uncertainty values. Two of the most commonly used explicit DRs are logic-based operational policies (e.g., energy management policies parametrized by thresholds in terms of the system state) and proportional-integral-derivative (PID) controllers that are parametrized by a set of gains. An important limitation of most explicit DRs is that they are not sufficiently expressive unless the parameters are adapted over time, which leads to a significant increase in problem size and, thus, somewhat defeats the purpose of using the DR in the first place. Implicit DRs, on the other hand, are typically model-based control strategies that solve an underlying optimization problem to make operational decisions – the word "implicit" here refers to the fact that the function $\kappa(\cdot)$ is not known in closed-form. Due to their flexibility and potential to provide very accurate

approximations to (2.3), we focus mostly on implicit DRs in this thesis, which are discussed in more detail in the next section.

## 2.3 Implicit DRs via Design-dependent Model Predictive Control

Model predictive control (MPC) is one of the most widely-used methods for advanced control of complex systems with multivariate interactions and safety-critical and/or quality constraints [84, 85]. Although MPC was originally developed for steady-state tracking problems, recent advances have enabled its application to a significantly broader class of systems including those with nonlinear models, constraints, and objectives that can be used to handle much more complicated physical equations as well as economic cost functions. Recent work [86] has also discussed the incorporation of discrete control inputs within MPC (which can be used to model higher level scheduling decisions such as turning a piece of equipment on or off) by taking advantage of state-of-the-art mixed integer programming methods. A general formulation of the finite-horizon MPC problem that must be solved at every time stage $k \in \mathcal{T}$ is given by

$$\min_{x_{k+i|k},u_{k+i|k}} \sum_{k=0}^{\theta_p-1} \hat{\ell}^{\theta_l}_{k+i}(x_{k+i|k},u_{k+i|k},\theta_{design}) + \hat{V}^{\theta_f}_{f,k+\theta_p}(x_{k+\theta_p|k},\theta_{design}), \tag{2.9a}$$

$$\text{s.t. } x_{k+i+1|k} = \hat{f}^{\theta_m}(k+i,x_{k+i|k},u_{k+i|k},\hat{w}_{k+i|k},\theta_{design}), \qquad \forall k \in \{0,\dots,\theta_p-1\}, \tag{2.9b}$$

$$\hat{w}_{k+i+1|k} = \hat{d}^{\theta_d}_{k+i}(\hat{w}_{k+i|k}), \qquad \forall k \in \{0,\dots,\theta_p-2\}, \tag{2.9c}$$

$$\{x_{k|k},\hat{w}_{k|k}\} = \{x_k(\boldsymbol{\omega},\theta_{design}),w_k\}, \tag{2.9d}$$

$$g(k+i,x_{k+i|k},u_{k+i|k},\hat{w}_{k+i|k},\theta_{design}) \leq -\theta_b, \qquad \forall k \in \{0,\dots,\theta_p-1\}, \tag{2.9e}$$

$$u_{k+i|k} \in \mathcal{U}, \qquad \forall k \in \{0,\dots,\theta_p-1\}, \tag{2.9f}$$

$$x_{k+\theta_p|k} \in \hat{\mathcal{X}}^{\theta_s}_{f,k+\theta_p}, \tag{2.9g}$$

where the notation $a_{k+i|k}$ denotes the predicted value of the variable $a \in \{x, u, \hat{w}\}$ at time $k+i$ given information up until time $k$; recall that the first stage decisions $\theta = \{\theta_{design}, \theta_{mpc}\}$ are composed of both the system design variables $\theta_{design} = d$ and the MPC-based DR parameters $\theta_{mpc} = \gamma = \{\theta_p, \theta_l, \theta_f, \theta_m, \theta_d, \theta_b, \theta_s\}$. Here, (2.9a) is the total cost to be minimized that is composed of a stage cost $\hat{\ell}^{\theta_l}_{k+i} : \mathcal{X} \times \mathcal{U} \times \Theta \to \mathbb{R}$ (parametrized by $\theta_l$) summed over the prediction horizon $\theta_p$ and terminal cost $\hat{V}^{\theta_f}_{f,k+\theta_p} : \mathcal{X} \times \Theta \to \mathbb{R}$ (parametrized by $\theta_f$). Often, one will select the MPC stage cost $\hat{\ell}^{\theta_l}_{k+i}(\cdot)$ to be closely related (or identical) to the IDC stage cost $\ell(\cdot)$, though additional parameters can be added to improve tuning. The constraints (2.9b) define the predicted state sequence as a function of the predicted input and estimated uncertainty values using approximate system model $\hat{f}^{\theta_m} : \mathcal{T} \times \mathcal{X} \times \mathcal{U} \times \mathcal{W} \times \Theta \to \mathcal{X}$ (parametrized by $\theta_m$). This approximate model can be derived in a variety of different ways including lumped parameter models derived from physics-based equations or black-box system identification methods using simulated data from (2.1). It is important to note that the approximate model should be constructed to depend on $\theta_{design}$ so that the MPC policy responds to changes in the system design. The controller's ability to automatically update to different design specifications is a big advantage of MPC compared to most alternative explicit DR representations. This design-dependent form of MPC is not traditionally considered in the literature since the design variables are assumed to be fixed. The recursion (2.9c) computes the predicted future disturbance sequence $\{\hat{w}_{t+k|t}\}$, while (2.9d) ensures the initial state and disturbance at time $t$ start from their most recently measured values. The set of constraints in (2.9e) approximately enforce the system constraints (2.2) for the predicted uncertainty values where $\theta_b \geq 0$ denote backoff parameters that are able to confer strong robustness properties when properly designed [87] (with larger values implying a

higher degree of conservatism). Hard input constraints are enforced via (2.9f) and terminal constraints are represented in (2.9g) in terms of the set $\mathcal{X}_f^{\theta_s}$ (parametrized by $\theta_s$).

The solution to the MPC optimization problem (2.9) is the optimal predicted state sequence $\{x^\star_{k+i|k}\}_{i=0}^{\theta_p}$ and control input sequence $\{u^\star_{k+i|k}\}_{i=0}^{\theta_p-1}$ given the most recent state $x_k(\boldsymbol{\omega},\theta)$ and disturbance $w_k$ measurement. The key idea in MPC is its receding horizon implementation, which implies only the first input is supplied to the system at every time $k \in \mathcal{T}$; then the next set of measurements are used to update the problem, which is solved again at $k+1$. Thus, we can represent the MPC law as an implicit DR as follows

$$u_k(\boldsymbol{\omega},\theta) = \kappa(k, x_k(\boldsymbol{\omega},\theta), w_k, \theta) = u^\star_{k|k}(x_k(\boldsymbol{\omega},\theta), w_k, \theta). \tag{2.10}$$

This receding horizon implementation is illustrated in Fig. 2.1. Note that the terminal cost $\hat{V}^{\theta_f}_{f,k+\theta_p}(\cdot)$ and constraint set $\hat{\mathcal{X}}^{\theta_s}_{f,k+\theta_p}$ can be chosen to ensure stability, constraint satisfaction, and recursive feasibility of the MPC problem (2.9), at least for the nominal model as discussed by Mayne [88]. In practice, however, these objects are difficult to construct for nonlinear and mixed-integer models, which implies a more practical approach is to search for optimal $\theta_{mpc}$ values that can more directly achieve the cost and constraint specifications that appear in the original IDC problem (2.3).

Next, we provide an illustrative example that compares MPC to a PID control on a nonlinear benchmark problem to highlight its advantages in terms of an enlarged feasible region and improved performance.

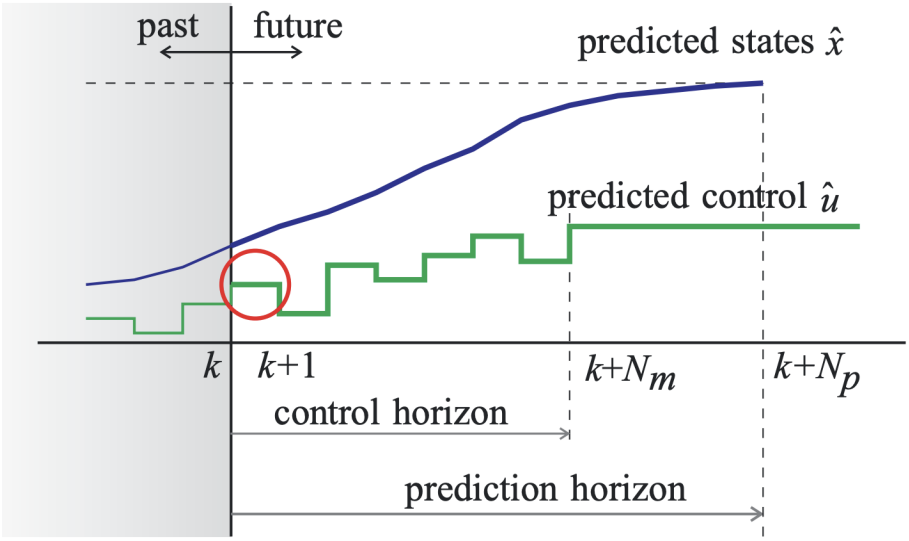


Figure 2.1: Receding horizon implementation of MPC. Here, the outputs are predicted over a finite horizon, while only the first set of optimal input values are actually implemented.

## 2.4 Example: Comparison of MPC- and PID-based Decision Rules

Consider an isothermal continuously stirred tank reactor (CSTR) in which an elementary exothermic second-order reaction is taking place that converts reactant $A$ to the desired product $B$, i.e.,

$$2A \longrightarrow B.$$

Under standard assumptions, the mass balance for species $A$ can be written as

$$\frac{dC_A}{dt} = \frac{F}{V_R}(C_{A0} - C_A) - k_0 e^{-\frac{E}{RT}} C_A^2, \tag{2.11}$$

where $C_A$ denotes the concentration of $A$ in the reactor, $F$ is the feed flow rate, $V_R$ is the reactor fluid volume, $k_0$ is the pre-exponential rate factor, $E$ is the activation energy for the reaction, $R$ is the gas constant, and $T$ is the reactor temperature. We assume a constant sampling time of $T_s = 0.01$ hour. The model parameters can be found in Table 2.1.

Table 2.1: Parameter values for CSTR case study

| Parameter | Value | Unit |
|---|---|---|
| $k_0$ | $2.115 \times 10^6$ | L mol$^{-1}$ s$^{-1}$ |
| $E/R$ | 6014 | K |
| $V_R$ | 1 | m$^3$ |
| $T$ | 515 | K |
| $C_{A0}$ | 3.5 | kmol m$^{-3}$ |

The single control input is the feed flow rate $F$, implying that (2.11) is of the form (2.1) with $x \leftarrow C_A$ and $u \leftarrow F$ (note there are no design or uncertain variables in this example). The initial state condition is assumed to be $b_0 = 0.1$ kmol m$^{-3}$. We also consider hard input and state constraints of the following form

$$0 \leq F \leq 10 \ \text{m}^3/s, \tag{2.12}$$

$$0 \leq C_A \leq 4 \ \text{kmol m}^{-3}. \tag{2.13}$$

These constraints can be straightforwardly cast as (2.2) with $\varepsilon = 1$; the probability operator can be trivially evaluated since there is no uncertainty considered in this example. We assume that the operating cost is defined in terms of the deviation from a desired steady state

$$O(\theta) = \sum_{k=0}^{T-1} \alpha(x_k(\theta) - x_s)^2 + \beta(u_k(\theta) - u_s)^2, \tag{2.14}$$

where $T = 100$, $\alpha = \$1.74 \times 10^4$, $\beta = \$1 \times 10^3$, and $(x_s, u_s) = (0.3376 \text{ kmol m}^{-3}, 5 \text{ m}^3/s)$. Since we do not consider any design variables in the CSTR system, $\theta$ only represents the controller tuning parameters in this case. Additionally, since there is no capital expenditure required, the total cost is directly equal to the operating cost.

First, we consider a PI controller for the DR structure (2.4). A discrete-time implementation of the PI controller can be expressed as

$$u_k(\theta) = u_s + K_c e_k + \frac{K_c}{\tau_I} \sum_{i=1}^{k} e_i T_s, \tag{2.15}$$

where $e_k = x_s - x_k$ is the error between the setpoint and the measured state at time step $k \in \mathcal{T}$, $K_c \in [1, 70]$ is the proportional gain, and $\tau_I \in [0.001, 1]$ is the integral time constant. The DR parameters $\theta = (K_c, \tau_I)$ in (2.15) are the proportional gain and integral time constant. Without performing any detailed analysis on the CSTR model, we cannot easily determine what range of parameters will lead to good performance, so we intentionally take a large range of values to test. The closed-loop operating cost for the CSTR system (2.14) controlled by the PI in (2.15) versus the tuning parameters $\theta \in \Theta = [1, 70] \times [0.001, 1]$ is shown in Fig. 2.2. Note that $\theta$ values that lead to violation of the state constraints (2.13), i.e., infeasible operation are depicted as white regions in Fig. 2.2. We see that the cost $O(\theta) \in \$[98.4, 1489.7]$ varies substantially over the feasible $\theta \in \Theta$ values. The feasible region is also fairly complicated (non-convex), with the closed-loop system resulting in constraint violations for all $K_c \geq 65$ and $\tau_I \leq 0.05$. This highlights some of the complications of PID controllers, even in the context of setpoint tracking, that can produce substantial variability in cost and constraint satisfaction.

We now analyze an MPC DR-based structure of the form (2.10), which is based on the solution to the optimization problem (2.9). We specify the stage cost as $\hat{\ell}_{k+i}^{\theta_l}(x_k, u_k, \theta) = (x_k - x_s)^2 + \theta_l (u_k - u_s)^2$ with ratio parameter $\theta_l \in [0.0005, 0.2]$. The terminal cost and constraints are neglected, and the system model is a discretized version of (2.11). We also treat the prediction horizon $\theta_p \in \{4, \ldots, 25\}$ as a parameter in the MPC-based DR. The optimization problem (2.9) is formulated in `CasADi` [89] in MATLAB R2019a and solved using the nonlinear programming solver `IPOPT` [90]. The closed-loop operating cost for the

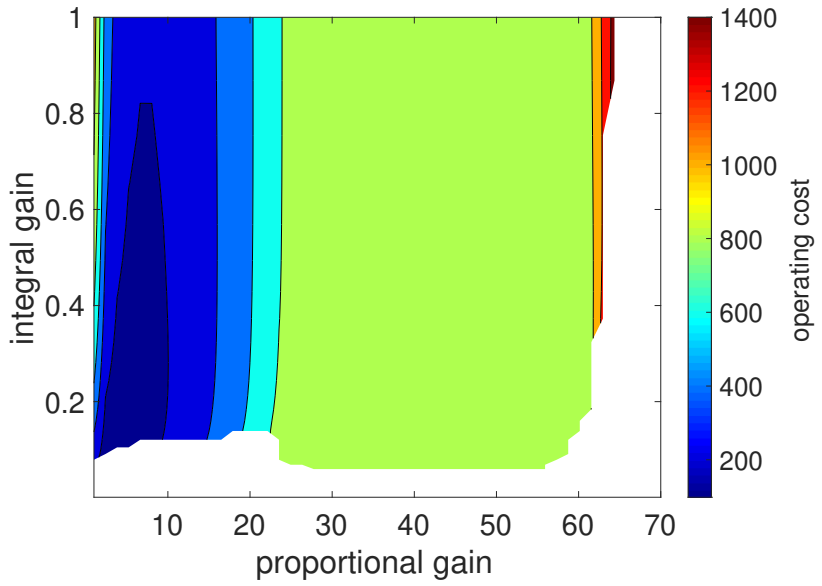


Figure 2.2: Comparison of closed-loop cost for the CSTR over PID tuning parameters $K_c$ and $\tau_I$. White regions denote parameters that lead to constraint violation.

CSTR system (2.14) controlled by MPC versus the tuning parameters $\theta = (\theta_l, \theta_p) \in \Theta = [0.0005, 0.2] \times \{4, \ldots, 25\}$ is shown in Fig. 2.3. Again, the white regions denote areas of infeasible operation. Note that the cost $O(\theta) \in \$[104.5, 121.4]$ varies significantly less over the feasible $\theta \in \Theta$ values of the MPC-based DR compared to that of the PID-based DR. Additionally, we see that only a relatively small part of the parameter region is infeasible for small prediction horizons, which is a fairly intuitive response that could have been forecast without running extensive simulations. These properties of MPC only become more important as the nonlinearity and dimensionality of the system increases, which clearly motivates the use of MPC as the DR throughout this thesis.

Although we have a way to simplify the IDC problem without introducing a substantial degree of suboptimality using MPC, the DRA-based IDC problem (2.6) remains a challenging optimization problem that requires tailored solution methods to be developed. The next

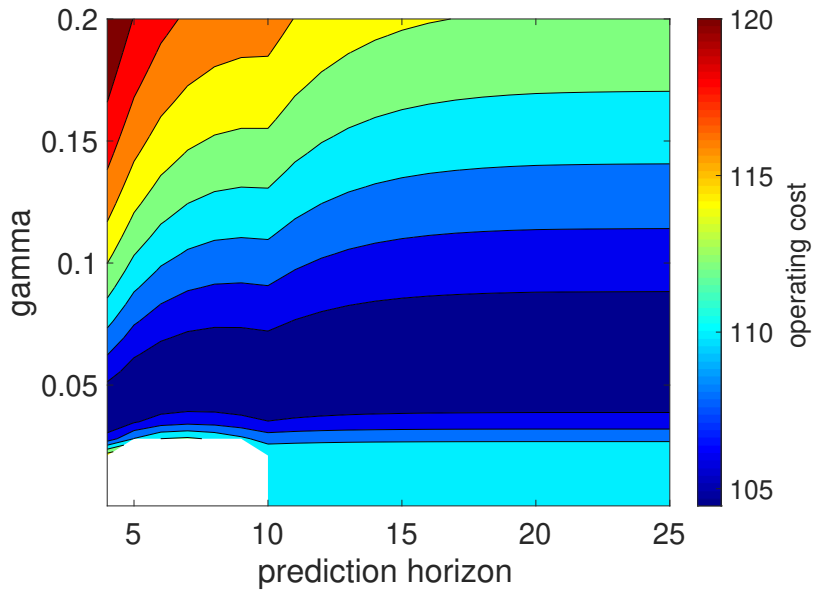


Figure 2.3: Comparison of closed-loop cost for the CSTR over MPC tuning parameters $\theta_l$ and $\theta_p$. White regions denote parameters that lead to constraint violation. Note that $\theta_p$ is technically only defined at integer values, though the plot is shown on a continuous scale for improved readability.

chapter introduces a particularly attractive algorithm for solving (2.6) that is mostly agnostic to the specific DR employed (i.e., can be flexibly applied to any explicit or implicit DR that does not add too many independent dimensions to $\theta$).

**Remark 2.** *The case study in Section 2.4 is meant to demonstrate the added value of MPC for constrained nonlinear systems. A key reason for why MPC performs so well is that we have assumed no plant-model mismatch (i.e., a highly accurate model is available) and a small number of states; this is often not the case in practice and can result in diminished MPC performance. In more complex systems, such as complete chemical processing plants, one cannot rely on a fully centralized MPC scheme. Thus, in practice, one often develops a hierarchical control structure that is a combination of lower-level regulatory PID controllers and a supervisory MPC layer that updates the setpoints to the fast-acting PID controllers*

*at a slow enough timescale to ensure closed-loop stability. Interested readers are referred to [91–93] for further details on plantwide control structures. Note that the black-box optimization procedure described in Chapter 3 is capable of handling such complex (but realistic) control structures whereas most alternative IDC methods (Table 1.1) would not be applicable.*

# Chapter 3: Practical Solutions to Decision Rule-based Integrated Design and Control using Constrained Bayesian Optimization

In Chapter 2, we introduced the DR-IDC problem (2.6) which can be generally formulated as a black-box optimization problem of the form

$$\min_{\theta\in\Theta} \ \mathcal{L}(\theta) \text{ s.t. } \mathcal{C}_k(\theta) \geq 0, \quad \forall k \in \{0,\ldots,T-1\}. \tag{3.1}$$

where $\mathcal{L}(\theta) = \mathbb{E}_{\boldsymbol{\omega}}\{C(\theta) + O(\boldsymbol{\omega},\theta)\}$, $\mathcal{C}_k(\theta) = \mathbb{P}_{\boldsymbol{\omega}}\{G_k(\boldsymbol{\omega},\theta) \leq 0\} - 1 + \varepsilon$. This DRA-based approximation to the IDC problem (2.6) is still intractable since the probabilistic operators $\mathbb{E}_{\boldsymbol{\omega}}\{O(\boldsymbol{\omega},\theta)\}$ and $\mathbb{P}_{\boldsymbol{\omega}}\{G_k(\boldsymbol{\omega},\theta) \leq 0\}$ are not finitely computable in general. Thus, a common approach is to evaluate both $\mathcal{L}(\theta)$ and $\mathcal{C}_k(\theta)$ via stochastic simulations. For fixed $\theta$, this consists of simulating (2.5) for randomly generated $\boldsymbol{\omega}$ to evaluate $O(\boldsymbol{\omega},\theta)$ and $G_k(\boldsymbol{\omega},\theta)$, which are then used to estimate the objective and constraints. Using this approach, we can then cast (2.6) as a simulation-based optimization problem involving an "outer" optimization over only $\theta$ and an "inner" or "embedded" stochastic simulation needed to evaluate $\mathcal{L}(\theta)$ and $\mathcal{C}_k(\theta)$. Monte Carlo (MC) sampling can then be used to approximate the objective and constraints from the inner stochastic simulations as follows:

$$\mathcal{L}(\theta) \approx \widehat{\mathcal{L}}_K(\theta) = K^{-1}\textstyle\sum_{j=1}^{K} C(\theta) + O(\boldsymbol{\omega}^j,\theta), \tag{3.2}$$

$$\mathcal{C}_k(\theta) \approx \widehat{\mathcal{C}}_{k,K}(\theta) = -1 + \varepsilon + K^{-1}\textstyle\sum_{j=1}^{K} \mathbb{1}_{(-\infty,0)}(G_k(\boldsymbol{\omega}^j,\theta)), \tag{3.3}$$

where $\mathbb{1}_{(a,b)} : \mathbb{R} \to \mathbb{R}$ is the indicator function of the range $(a,b)$, $K$ is the number of samples, and $\{\boldsymbol{\omega}^1, \ldots, \boldsymbol{\omega}^K\}$ are independent and identically distributed (i.i.d.) samples of the random vector $\boldsymbol{\omega} \sim p_{\boldsymbol{\omega}}$. The MC approximations in (3.2) and (3.3) generate stochastic/noisy observations for both the objective and the constraints since unique samples are drawn from the uncertainty set at every closed-loop simulation For any sample size $K$, $\mathbb{E}_{\boldsymbol{\omega}^1,\ldots,\boldsymbol{\omega}^K}\{\widehat{\mathcal{L}}_K(\theta)\} = \mathcal{L}(\theta)$ and $\mathbb{E}_{\boldsymbol{\omega}^1,\ldots,\boldsymbol{\omega}^K}\{\widehat{\mathcal{C}}_{k,K}(\theta)\} = \mathcal{C}_k(\theta)$, i.e. these estimators are unbiased [94]. According to the central limit theorem, as $K$ increases, the variance of these estimators decreases linearly in $K$ towards zero. As such, the objective (3.2) and constraint (3.3) estimates will be very accurate when a sufficiently large $K$ is selected. However, since the closed-loop simulations are often computationally expensive, we must keep $K$ small and thus it is crucial to select an optimization algorithm that can handle noisy observations.

In addition to expensive and noisy observations, another key feature of (3.1) is that we cannot easily estimate derivatives of $\mathcal{L}(\theta)$ and $\mathcal{C}_k(\theta)$ that could be exploited by an established gradient-based optimization algorithm. In fact, since $O(\boldsymbol{\omega}, \cdot)$ and $G_k(\boldsymbol{\omega}, \cdot)$ are potentially discontinuous functions with respect to $\theta$ for some $\boldsymbol{\omega}$ values, derivatives may not even exist such that we need a fundamentally different approach for solving (3.1). In the absence of gradient information, one usually must resort to so-called derivative-free optimization (DFO) methods. DFO can be broadly categorized into stochastic and deterministic approaches; an overview of this breakdown, along with example algorithms within each category, is provided in Fig. 3.1.

Although there are many DFO methods with abilities to handle some of these characteristics, most of them do not satisfy all of these requirements. For problems of low- to medium-dimensionality, Bayesian optimization (BO) [95] methods have been found to be particularly effective at handling these three important characteristics. The basic idea behind

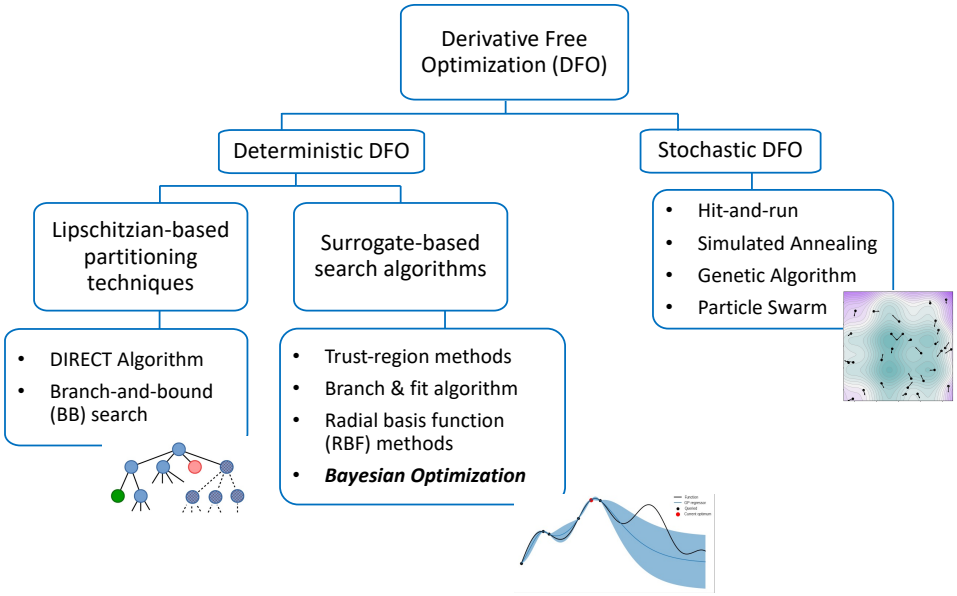


Figure 3.1: Derivative Free Optimization algorithm classes and examples

BO is to train a probabilistic surrogate model given observations of the objective/constraints and use this surrogate to optimize an acquisition function that is designed to tradeoff *exploration* of regions where the surrogate model is most uncertain and *exploitation* of the model's confidence in good solutions. An important challenge in these approaches is the selection of the "right" type of surrogate model when little is known about the structure of the objective. Gaussian process (GP) models [96] are a particularly attractive class of surrogates since they are both *probabilistic* and *non-parametric*. Given a set of function evaluations, a GP model can be easily derived by placing a prior over the set of possible objective functions and updating this prior with the available data using Bayes' rule.

The following section gives a detailed overview of the BO algorithm, along with relevant modifications to accommodate unknown black-box constraints. See Appendix A for a

detailed discussion on alternative DFO methods and their relationship to BO, as well as an example demonstrating the effectiveness of BO.

Note that the case study in this chapter has been adapted from a publication by the author in the Proceedings of the 2021 American Control Conference [51].

## 3.1 Overview of Gaussian Process Regression

This section provides a detailed overview of GP regression for general functions $f : \mathbb{R}^{n_x} \to \mathbb{R}$ from potentially noisy measurements

$$y = f(\theta) + \varepsilon \tag{3.4}$$

where $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$ is a Gaussian noise term with variance $\sigma_\varepsilon^2$. Effectively, GP's are an uncountable collection of random variables of which any finite subset have a joint Gaussian distribution, i.e., generalizes the notion of multivariate Gaussian distributions to "distributions over functions" [96]. A function $f(\cdot)$ is distributed as a GP in terms of their mean function $\mu^f(\cdot)$ and covariance kernel $k^f(\cdot,\cdot)$ such that $f(\cdot) \sim \mathcal{GP}(\mu^f(\cdot), k^f(\cdot,\cdot))$. A detailed overview of GP's is presented next. For any pairs of input points $\theta, \theta' \in \mathbb{R}^{n_\theta}$, the mean function and covariance kernel are defined as follows:

$$\mu^f(\theta) = \mathbb{E}_f\{f(\theta)\}, \tag{3.5}$$

$$k^f(\theta,\theta') = \mathbb{E}_f\{(f(\theta) - \mu^f(\theta))(f(\theta') - \mu^f(\theta'))\}, \tag{3.6}$$

where the expectation is taken over the function space. There are multiple covariance functions available to choose from; in this work, we selected the Matérn 5/2 function [97]

$$k^f(\theta,\theta') = \sigma^2 \left(1 + \sqrt{5r^2(\theta,\theta')} + \frac{5}{3}r^2(\theta,\theta')\right) \exp\left\{-\sqrt{5r^2(\theta,\theta')}\right\} \tag{3.7}$$

where, $\sigma$ is the standard deviation and $r$ is the Euclidean distance between $\theta$ and $\theta'$. To update the GP prior, the function has to be evaluated at particular values of $\theta$. Since

evaluating (3.2) results in noisy observations, we model them as (3.4). Let $y_{1:n} = \{y_1, \ldots, y_n\}$ denote $n$ noisy observations from (3.4) computed at corresponding inputs $\theta_{1:n} = \{\theta_1, \ldots, \theta_n\}$. Then, $y_{1:n}$ and $f(\theta)$ at any test point $\theta$ are jointly Gaussian under the GP prior assumption and can be represented with the following distribution

$$\begin{bmatrix} y_{1:n} \\ f(\theta) \end{bmatrix} \sim \mathcal{N}\left( \begin{bmatrix} \mu^f(\theta_{1:n}) \\ \mu^f(\theta) \end{bmatrix}, \begin{bmatrix} k^f(\theta_{1:n}, \theta_{1:n}) + \sigma_\varepsilon^2 I_n & k^f(\theta_{1:n}, \theta) \\ k^f(\theta, \theta_{1:n}) & k^f(\theta, \theta) \end{bmatrix} \right) \tag{3.8}$$

where the functions $\mu^f(\cdot)$ and $k^f(\cdot,\cdot)$ have been overloaded to include element-wise operations across their inputs. Therefore, the posterior distribution $p(f(\theta)|y_{1:n}, \theta_{1:n})$ of the objective under all noisy observations is Gaussian due to properties of joint Gaussian random variables, with the following expressions for mean and variance [98]:

$$\mu_n^f(\theta) = \mu^f(\theta) + k^f(\theta_{1:n}, \theta)\left(k^f(\theta_{1:n}, \theta_{1:n}) + \sigma_\varepsilon^2 I_n\right)^{-1}\left(y_{1:n} - \mu^f(\theta_{1:n})\right) \tag{3.9}$$

$$(\sigma_n^f)^2(\theta) = k^f(\theta, \theta) - k^f(\theta, \theta_{1:n})\left(k^f(\theta_{1:n}, \theta_{1:n}) + \sigma_\varepsilon^2 I_n\right)^{-1} k^f(\theta_{1:n}, \theta) \tag{3.10}$$

After obtaining the posterior distribution, the next step is to optimize the acquisition function $\alpha_n : \Theta \to \mathbb{R}$ to probe for the subsequent sampling point $\theta_{n+1}$. Intuitively, the acquisition function evaluates the utility of candidate points for the next evaluation of $f(\theta)$; therefore $\theta_{n+1}$ is selected by maximizing $\alpha_n$, where the index n indicates the implicit dependence on the currently available data. Here the "data" refers to previous locations where $f(\theta)$ has been evaluated, and the corresponding noisy outputs. Next, we discuss how this probabilistic model can be combined with an *acquisition function* that is capable of trading off between exploitation and exploration.

## 3.2 Expected Improvement with Constraints

Traditionally, BO has been limited to problems with known constraints; however, it has been recently extended to handle black-box constraints. Several constrained BO (cBO)

methods have been proposed in the literature (see, e.g., [99] for a detailed discussion). Here, we focus on an intuitive strategy that builds upon one of the most popular acquisition functions known as expected improvement (EI). Following [100], we define the constrained improvement for any candidate design $\theta$ as

$$\text{C-Improv}(\theta) = \text{Feas}(\theta)\max\{0, \eta - \mathcal{L}(\theta)\} = \text{Feas}(\theta)\text{Improv}(\theta) \tag{3.11}$$

where $\text{Feas}(\theta) \in \{0,1\}$ is a feasibility indicator function that is 1 if $\mathcal{C}_k(\theta) \geq 0$, and 0 otherwise, and $\eta$ is the current incumbent value (can be chosen in various ways as discussed in more detail below). Because $\mathcal{L}(\theta)$ and $\mathcal{C}_k(\theta)$ are computationally expensive to evaluate, we independently model them as GPs, i.e., $\mathcal{L}(\cdot) \sim \mathcal{GP}(\mu^{\mathcal{L}}(\cdot), k^{\mathcal{L}}(\cdot,\cdot))$ and $\mathcal{C}_k(\cdot) \sim \mathcal{GP}(\mu^{\mathcal{C}_k}(\cdot), k^{\mathcal{C}_k}(\cdot,\cdot))$ for all $k \in \mathcal{T}$. Due to these GP models, the constrained improvement (3.11) is a random quantity. Thus, we are interested in using the *expected* constrained improvement as our acquisition function that can be derived as follows

$$\begin{aligned}
\text{EIC}_n(\theta) &= \mathbb{E}_n\{\text{C-Improv}(\theta)\} \\
&= \mathbb{E}_n\{\text{Feas}(\theta)\text{Improv}(\theta)\} \\
&= \mathbb{E}_n\{\text{Feas}(\theta)\}\mathbb{E}_n\{\text{Improv}(\theta)\} \\
&= \prod_{k=0}^{T-1} \Pr_n\{\mathcal{C}_k \geq 0\}\text{EI}_n(\theta) = \prod_{k=0}^{T-1} \Phi\left(\frac{\mu_n^{\mathcal{C}_k}(\theta)}{\sigma_n^{\mathcal{C}_k}(\theta)}\right)\text{EI}_n(\theta)
\end{aligned} \tag{3.12}$$

where $\mathbb{E}_n\{\cdot\}$ denotes the conditional expectation with respect to $n$ observations of the objective and constraint functions, $\Phi(\cdot)$ is the standard Gaussian cumulative distribution function, $\mu_n^{\mathcal{C}_k}(\theta)$ and $\sigma_n^{\mathcal{C}_k}(\theta)$ are given by (3.9) for $n$ observations of $f = \mathcal{C}_k$, and $\text{EI}_n(\theta)$ is the standard (unconstrained) expected improvement function. As shown in Jones et al. [101], EI can be derived analytically for GP models

$$\text{EI}_n(\theta) = \sigma_n^{\mathcal{L}}(\theta)(z_n^{\mathcal{L}}(\theta)\Phi(z_n^{\mathcal{L}}(\theta)) + \phi(z_n^{\mathcal{L}}(\theta))), \tag{3.13}$$

where $z_n^{\mathcal{L}}(\theta) = (\eta - \mu_n^{\mathcal{L}}(\theta))/\sigma_n^{\mathcal{L}}(\theta)$; $\mu_n^{\mathcal{L}}(\theta)$ and $\sigma_n^{\mathcal{L}}(\theta)$ are given by (3.9) for $n$ observations of $f = \mathcal{L}$, and $\phi$ is the standard Gaussian probability density function.

In the unconstrained setting, the incumbent $\eta$ can be simply selected as the minimum current observation. When constraints are included, this definition must be updated to the be the minimum *feasible* observation

$$\eta = \begin{cases} \infty, & \text{if no points are feasible,} \\ \min_{i\in\{1,\dots,n\}} \mathcal{L}(\theta_i) \text{ s.t. } \mathcal{C}_k(\theta_i) \geq 0,\ \forall k \in \mathcal{T}, & \text{otherwise.} \end{cases} \tag{3.14}$$

In the presence of noise, however, this selection can be overly optimistic. A commonly used alternative that we pursue here is to select the incumbent based on the minimum predicted mean value subject to chance constraints

$$\eta = \min_{\theta\in\Theta} \ \mu_n^{\mathcal{L}}(\theta) \ \text{ s.t. } \prod_{k=0}^{T-1} \Phi\left(\frac{\mu_n^{\mathcal{C}_k}(\theta)}{\sigma_n^{\mathcal{C}_k}(\theta)}\right) \geq 1-\delta, \tag{3.15}$$

where $\delta \in (0,1)$ is a tuning parameter typically set to $\delta = 0.05$. This choice has been shown to be much less sensitive to noisy measurements and, in fact, can be thought of a way to filter out noise via the GP model. The complete cBO method is summarized in Algorithm 1, and a simple illustration is provided in Fig. 3.2 [102]. Next, we apply the cBO algorithm to a relevant case study to demonstrate its ability to handle the complex MPC-based DRs discussed at length in Chapter 2.

**Algorithm 1** The EIC algorithm under the BO framework [103]

1: **Initialize:** Input space $\Theta$; GP prior $\mu(\cdot)$ and $k(\cdot,\cdot)$; number of MC evaluations $K$; and maximum number of iterations $N$.
2: **for** $n = 0$ to $N-1$ **do**
3: Construct GP surrogate models for $\mathcal{L}(\theta)$ and $\mathcal{C}(\theta)$ given all available data using the posterior mean and variance equations in (3.9) and (3.10).
4: Maximize the acquisition function to find $\theta_{n+1} = \text{argmax}_{\theta\in\Theta} \text{EIC}_n(\theta)$.
5: Perform expensive closed-loop simulations to evaluate the objective and constraints using (3.2) and (3.3)
6: **end for**

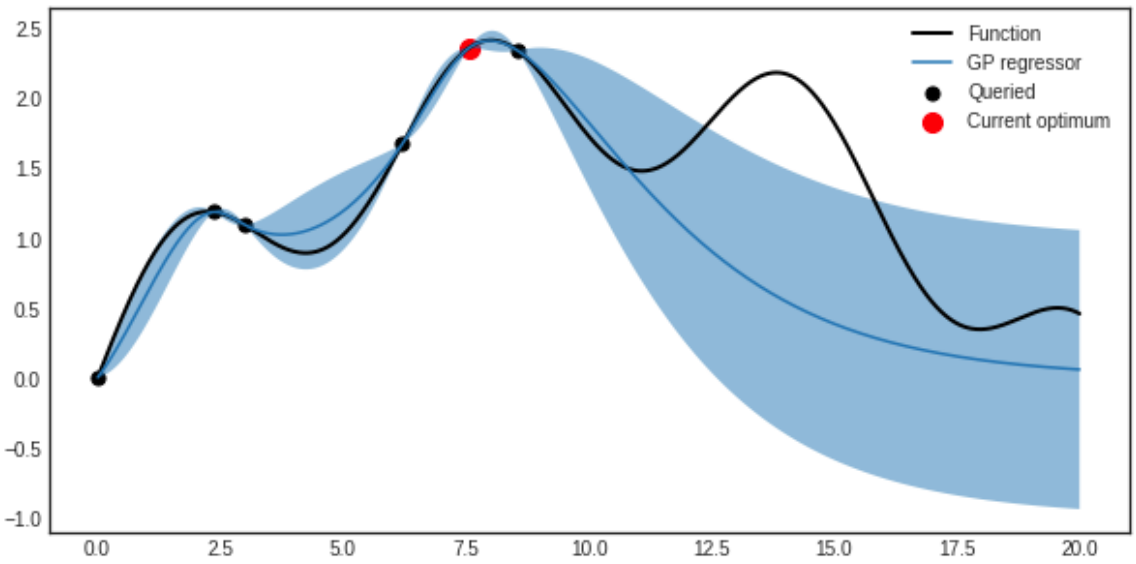


Figure 3.2: Illustration of the Bayesian Optimization algorithm

## 3.3 Case Study: Flexible Design of a Building Cooling System

### 3.3.1 System Description

The proposed IDC approach is applied to the design of a building cooling system, adapted from [104]. An illustration of the overall system is provided in Fig. 3.3. The system model (2.1) is the discretized version of the differential equation

$$m\frac{dx}{dt} = -k_b(x - T_{\text{amb}}) + q_{\text{amb}} - q + w, \tag{3.16}$$

where $x$ is the (average) building temperature, $T_{\text{amb}}$ is the ambient outside air temperature, $q_{\text{amb}}$ is the direct heating provided by the ambient, $q$ is the cooling supplied by the chillers, $m = 10$ is the constant building mass, $k_b$ is the building heat transfer coefficient, and $w$ is a random time-varying disturbance that is uniformly distributed between $[-1, 1]$. We assume a constant sample time of $T_s = 1$ hour and approximate the input $q$ and time-varying parameters $T_{\text{amb}}$ and $q_{\text{amb}}$ as piecewise constant.

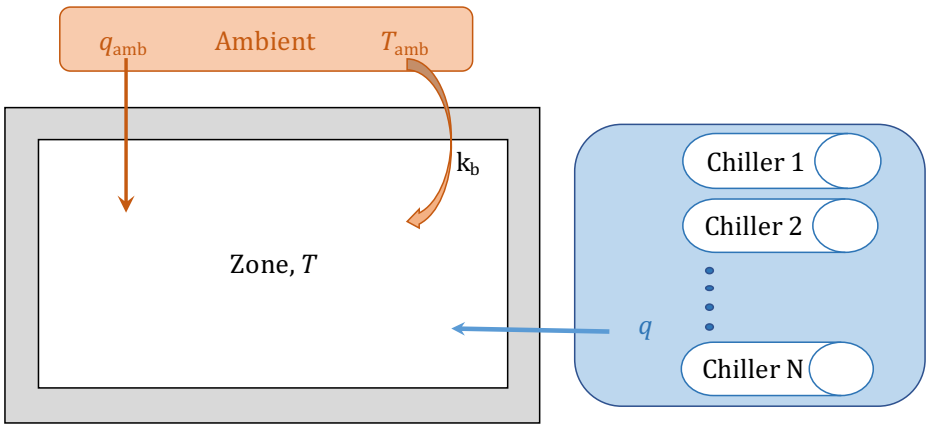


Figure 3.3: Illustration of simple building cooling system.

Let $N_{\text{chiller}}$ denote the number of chillers that can be either on or off. We include another discrete control input variable $v$ to select how many chillers to activate at each sample time. Since each active chiller must be within the range $[q_{\min}, q_{\max}]$, we can formulate the input constraints $u = (q, v) \in \mathcal{U}$ as

$$q_{\min} v \le q \le q_{\max} v, \quad v \in \{0, 1, \ldots, N_{\text{chiller}}\}, \tag{3.17}$$

where $q$ denotes the *total* cooling provided by all chillers. The number of chillers and heat transfer coefficient are the design variables, i.e., $\theta_{design} = (k_b, N_{\text{chiller}}) \in D = [0.1, 10] \times \{1, \ldots, 5\}$. The capital cost function is given by

$$C(\theta) = \alpha_{HT} k_b^{-0.8} + \alpha_{CH} N_{\text{chiller}}^{1.8}. \tag{3.18}$$

where $\alpha_{HT} = \$2 \times 10^5$ and $\alpha_{CH} = \$3.482 \times 10^2$ are the costs of building material and chiller equipment, respectively. Electricity is the main operating cost, which is given in terms of the stage cost function $\ell(k, x, u, d) = \beta \rho_k q$ where $\beta = \$2.734 \times 10^{-5}$ is a constant and $\rho_k$ is the (normalized) time-varying price of electricity. We are interested in 10 years of operation. Since the constraints and disturbances are periodic, we can compute the stage cost $\ell$ over a subset, e.g., $T = 72$ (3 days) and scale by $3650/3$ to approximate the total cost over 10 years. In addition, we would like to enforce comfort constraints on the state

$$\mathbb{P}\{T_{\min,k} \le x_k \le T_{\max,k}\} \ge 0.95, \tag{3.19}$$

where $T_{\min,k}$ and $T_{\max,k}$ are time-varying lower and upper bounds on the building temperature, respectively. The numerical values of the time-varying parameters $(T_{\text{amb}}, q_{\text{amb}}, \rho)$ are shown in Fig. 3.4, which are 24-hour periodic.

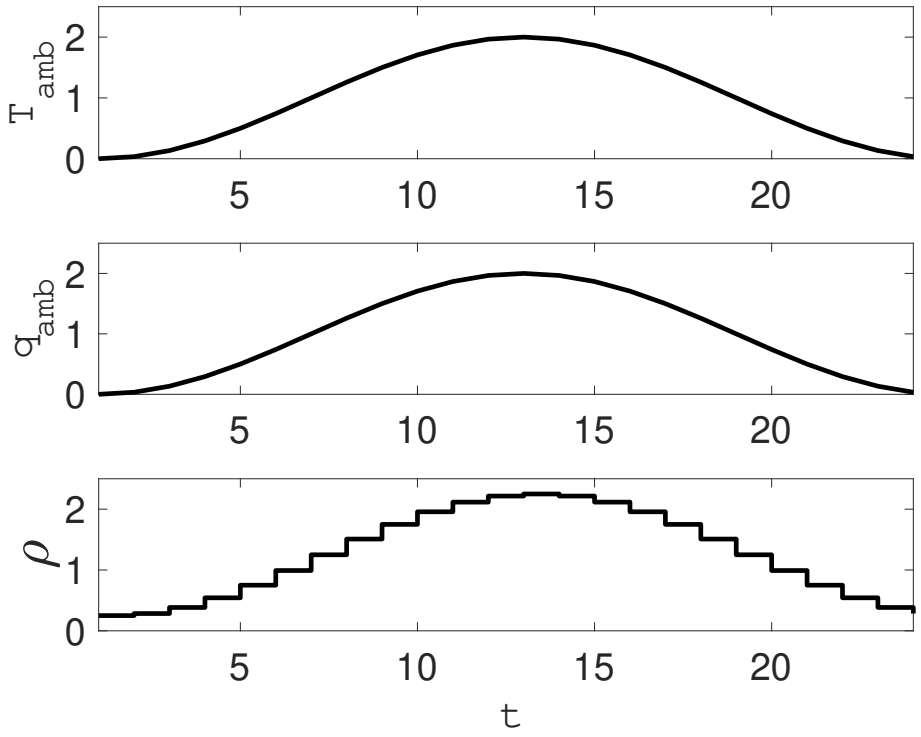


Figure 3.4: Nominal time-varying parameter values, which are 24-hr periodic.

### 3.3.2 Mixed Integer Model Predictive Control based decision rule

A particular case of interest in MPC is *discrete-valued* actuators, which allows for higher-level decisions, such as switching a piece of equipment on or off, to be made optimally as opposed to heuristically. The recent development of mixed-integer MPC (MIMPC) [104] provides a useful framework for addressing such problems that involve combinations of continuous-and discrete-valued actuators. Hence, the control inputs $u_k = (q_k, v_k)$ are selected using a high-quality MIMPC controller (2.10), where we partitioned the control inputs into their continuous $u^c(i) \in \mathbb{R}^{n_u^c}$ and discrete $u^d(i) \in \{0,1\}^{n_u^d}$ components above. The nominal model $\hat{f}^{\theta_m}$ used to derive this control law is the discretized version of (3.16) with $w = 0$. To avoid infeasibilities in (2.9), we implement the temperature bounds using soft constraints,

i.e.,

$$\max(x_k - T_{\max,k}, T_{\min,k} - x_k) + \theta_b \leq \varepsilon_k, \tag{3.20}$$

where $\varepsilon_k \in \mathbb{R}_{\geq 0}$ denotes the slack variables representing the magnitude of the constraint violations and $\theta_b \in \mathbb{R}_{\geq 0}$ is a backoff parameter in the DR that acts as an additional safety margin for disturbances. The stage cost is then the sum of electricity costs and the state constraint penalties

$$\hat{\ell}_k^{\theta_l} = \rho_k q + \theta_s \varepsilon_k, \tag{3.21}$$

where the penalty coefficient $\theta_s \in \mathbb{R}_{\geq 0}$ is a DR parameter. We assume a constant prediction horizon of $\theta_p = 10$. We set the terminal cost $V_f^{\theta_f}$ to zero and solve a single instance of the optimal control problem (2.9) to obtain an optimal periodic profile of the state and the input $(x_r, u_r)$ with initial conditions $x(0)$ free and added constraint $x(0) = x(T)$ for $T = 24$. We then enforce a terminal equality constraint $\mathcal{X}_f^{\theta_s}$ (for stability purposes) on the MPC wherein the state is required to converge to the optimal periodic cycle obtained.

### 3.3.3 Integrated design and control using constrained BO and comparison to various DFO solvers

Given the MIMPC-based DR with parameters $\theta_{mpc} = (\theta_s, \theta_b) \in [100, 1000] \times [0, 0.25]$, we can solve the DRA-based SO problem (3.1) using the constrained BO algorithm presented in Section 3.1. The objective and constraint functions are estimated using (3.2) and (3.3), respectively, with $M = 1$. Based on initial simulations, a feasible set of design decisions $\theta_{design}^0 = (k_b, N_{\text{chiller}}, \theta_s, \theta_b) = (2.5, 5, 500, 0.1)$ was found – $\mathcal{L}(\theta^0) = \$1.154 \times 10^5$ was estimated using 100 random $\boldsymbol{\omega}$ values. The constrained BO algorithm was executed using `bayesopt` in MATLAB R2019a for a total of 35 iterations, with the first 5 being randomly

selected in addition to $\theta^0$. Fig. 3.5 compares results obtained with the EIC acquisition function to that of probability of improvement and random search. Because these algorithms are stochastic, they were repeated for 10 different sets of random seed points. The EIC-based constrained BO approach terminated with $\mathcal{L}(\theta^\star) = \$7.597 \times 10^4$, which was the best point found by any method. All 10 runs converged within a few percent of this point. Although random search did find a similar point in 1 of the 10 runs, it resulted in significantly higher variance with several runs producing virtually no improvement over the feasible seed point $\theta^0$. This demonstrates the value of a sequential data-driven optimization strategy that can effectively use previous function evaluations to perform a targeted search of the design space, regardless of their complexity or convexity/smoothness properties.

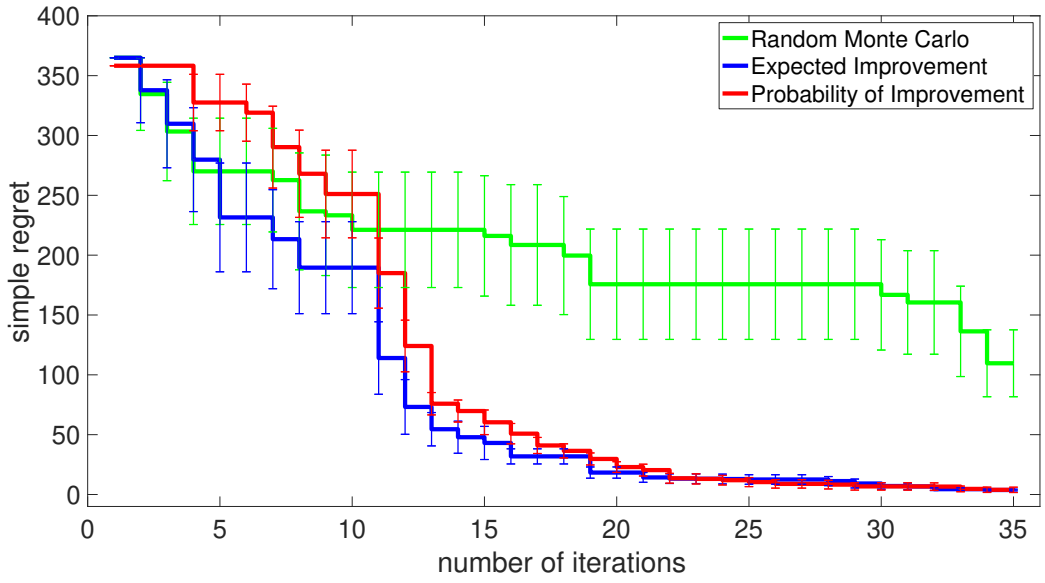


Figure 3.5: The simple regret versus number of iterations $n$ for three different acquisition functions. Each approach was repeated for 10 sets of random seed points. The mean of these 10 runs are shown in bold and the minimum and maximum values are shown with asymmetric error bars.

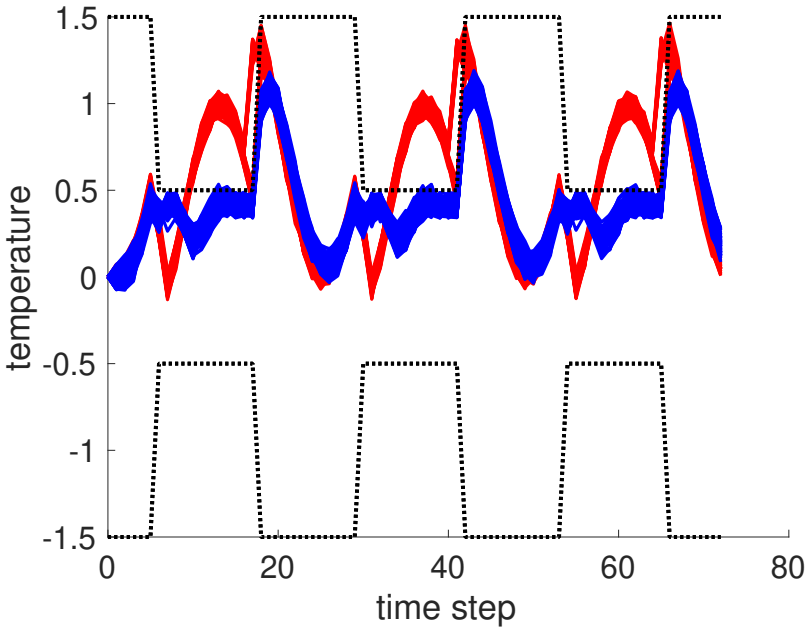


Figure 3.6: Closed-loop building temperature profiles over time for IDC (blue) and multi-period design (red) for 100 uncertainty realizations.

To highlight the importance of considering an integrated form of the design and control problem, we compare the results obtained from the constrained BO method above to that of a sequential design and control method. In particular, we solve a multiperiod design problem that can be thought of as a two-stage approximation of (2.3), as discussed in [105]. Since only two stages are considered, a steady-state version of the model is used with the control inputs being treated as "wait-and-see" decisions that are able to adapt to the set of time-varying uncertainty values (see Fig. 3.4). In addition, the multiperiod design problem cannot account for time-varying temperature constraints so that we fix the upper and lower bounds to $\min_{k\in\mathcal{T}} T_{\max,k}$ and $\max_{k\in\mathcal{T}} T_{\min,k}$, respectively. The restrictive constraints and varying uncertainties result in an infeasible multiperiod design problem such that the comfort constraints must be softened in order to obtain a solution. Under the resulting

multiperiod design solution, we were unable to find DR parameters that satisfy constraints (3.19). The building temperature profiles for the optimal IDC found using constrained BO and the multiperiod design (with MIMPC-based DR using $\theta_s = 1000$ and $\theta_b = 0.25$) for 100 random $\boldsymbol{\omega}$ realizations are shown in Fig. 3.6. We can clearly see that the IDC is able to consistently satisfy the comfort constraints with high probability whereas the sequential approach results in significant violation (regardless of the choice of $\theta$). This highlights the limitations of the sequential method. Thus, we can conclude that integrating the design in the control problem is very important in such problems with highly time-varying natures.

# Chapter 4: Multi-Fidelity Bayesian Optimization and Application to Next-Generation Energy Systems

With the significant increase in global energy demand and the advent of next-generation manufacturing and energy systems, such as combined heat and power (CHP) plants, smart grids, and multi-product chemical plants, the necessity for a paradigm shift in energy production and operations is paramount. This challenge is intensified by environmental concerns resulting from the commonplace use of traditional energy sources, like fossil fuels and its derivatives, which are large contributors to climate change. To enable this shift into the next-generation, we need to be able to design flexible energy production and manufacturing processes that are to respond/adapt to highly dynamic and uncertain conditions in an optimal manner. The design of such advanced energy and manufacturing systems requires an integrated design and control (IDC) approach.

In Chapter 3, we discussed a practical solution approach for the IDC problem using a novel black-box optimization strategy called constrained Bayesian optimization (cBO). We then showcased its performance on the flexible design of a building cooling system and compared it to the conventional sequential design and control approach. In this chapter however, we extend the notion of Bayesian optimization for IDC of large-scale advanced energy systems which requires a minor but important change in the DR-IDC problem formulation. In Chapter 2 we introduced the IDC problem formulation (2.3) which consisted

of an objective consisting of the capital cost and the operating cost at each stage subject to joint chance constraints (2.2). There are several ways to formulate these constraints, which is highly-dependent on the problem at hand. For certain classes of energy systems, we can instead formulate these constraints as expectation-type constraints

$$\mathbb{E}_{\boldsymbol{\omega}}\Big\{ \max\{G_k(\boldsymbol{\omega},\theta),0\}\Big\} = 0, \tag{4.1}$$

where $G_k(\boldsymbol{\omega},\theta) = g(k,x_k(\boldsymbol{\omega},\theta),u_k(\boldsymbol{\omega},\theta)$. These types of constraints are beneficial for a couple of reasons. They avoid the need to tune the probability of violation as a parameter and allow for an associated penalty whenever a violation occurs. An added advantage of formulating the constraints as expectation-type constraints is that they can be softened when they are not safety-critical for the process. Comfort constraints are prevalent in energy systems (e.g., temperature of a building, price of electricity, etc.), which can be modeled in the form of (4.1). In addition, when we soften these constraints, they can be easily pulled into the expectation as follows

$$\min_{\theta\in\Theta} \mathbb{E}_{\boldsymbol{\omega}}\left\{ \mathcal{L}(\boldsymbol{\omega},\theta) + \sum_{k=0}^{T-1} v_k^\top \max\{G_k(\boldsymbol{\omega},\theta),0\} \right\}, \tag{4.2}$$

where $\mathcal{L}(\boldsymbol{\omega},\theta) = C(\theta) + O(\boldsymbol{\omega},\theta)$ is the total capital plus operating cost and $v_k$ is the Lagrange multiplier for the $k^{\text{th}}$ constraint. By the weak duality theorem, the solution to (4.2) is a lower bound for the hard-constrained problem for any $v_k$. Since the expected constraint violation must be non-negative, we know that $v_k \geq 0$ for all $k = 0,\ldots,T-1$. The main advantage of this formulation is that we only need to develop surrogate models of the overall objective such that we can explore the use of more advanced multi-fidelity Bayesian optimization methods in this chapter.

The contents of this chapter have been adapted from a journal publication by the author that was submitted to a special issue in Optimal Control Methods and Applications [106].

## 4.1 Multi-Fidelity Bayesian Optimization

This section focuses on a more efficient extension of traditional BO (Chapter 3) when simpler approximate versions of the IDC problem can be developed. First, we reformulate (4.2) as a maximization problem for simplicity of presentation

$$F^\star = \max_{\theta\in\Theta} F(\theta) := -\mathbb{E}_{\boldsymbol{\omega}}\left\{\mathcal{L}(\boldsymbol{\omega},\theta) + \sum_{k=0}^{T-1} v_k^\top \max\{G_k(\boldsymbol{\omega},\theta),0\}\right\}, \tag{4.3}$$

Note that $F$ in this chapter will refer to our performance function that we want to maximize and should not be confused with the system dynamic or cost function $f$ from the previous chapters. The main bottleneck in the approach proposed in Chapter 3 is that the full closed-loop system must be simulated $M$ times at every iteration. An interesting idea for alleviating this bottleneck is the use of series of approximate models for the objective, as opposed to a single high-fidelity model. This is often referred to as *multi-fidelity* optimization in the literature and has been successfully applied in the context of automated machine learning [107–109]. Since the approximations are still correlated to the high-fidelity evaluation, we can reasonably expect that these models may provide valuable information at a fraction of the cost, which can be used to avoid "wasting" very expensive function evaluations on particularly poor designs. This discussion motivates the following two questions, which we look to (partially) answer in the next two sections:

1. Can we come up with a systematic procedure to decide at what fidelity and location we should sample?

2. How should we construct lower-fidelity representations of the IDC performance function in (4.3)?

### 4.1.1 Setting up the Multi-Fidelity Problem

We address the first question using a multi-fidelity extension of the BO framework, which we refer to as multi-fidelity BO (MFBO) for short. The main difference from the traditional setup in Chapter 3 is that the function is unconstrained and we assume access to a set of $M-1$ successively accurate approximations, which we denote by $F^{(1)}, F^{(2)}, \ldots, F^{(M-1)}$, to the true function of interest $F^{(M)} = F$. In other words, Chapter 3 assumed that $M = 1$. Following Kandasamy et al. [110], these approximations (also known as fidelities) must satisfy two important conditions:

1. The functions $F^{(1)}, \ldots, F^{(M-1)}$are approximations of $F^{(M)}$ with bounded error that successively improves, i.e.,

$$\|F^{(m)} - F^{(M)}\|_\infty \leq \zeta^{(m)}, \ \ \forall m \in \{1, \ldots, M\}, \tag{4.4}$$

   where the bounds $\zeta^{(1)} > \zeta^{(2)} > \cdots > \zeta^{(M)} = 0$ are known.

2. The functions $F^{(1)}, \ldots, F^{(M-1)}$ are cheaper to evaluate than $F^{(M)}$, i.e., $0 < \lambda^{(1)} < \lambda^{(2)} < \cdots < \lambda^{(M)}$ where $\lambda^{(m)}$ denotes the computational cost of querying at fidelity $m \in \{1, \ldots, M\}$.

Roughly speaking, these two conditions state that the approximations should become both more accurate and more costly, as the level $m$ increases. As opposed to just sampling $\{x_n\}_{n \geq 0}$, the multi-fidelity version of the algorithm must determine a sequence of query-fidelity pairs $\{(x_n, m_n)\}_{n \geq 0}$ where, at any given time $n$, the algorithm can use information from the previous $n-1$ query-observation-fidelity triples, i.e., $\{(\theta_i, m_i, y_i)\}_{i=1}^{n-1}$. Note that, similarly to (3.2), the observations are modeled as

$$y_i = F^{(m_i)}(\theta_i) + \varepsilon_i, \tag{4.5}$$

where $\varepsilon_i$ are independent noise realizations at every iteration $i$, with $\mathbb{E}\{\varepsilon_i\} = 0$. Note that the cases of interest are for $M$ values that are fixed and relatively small (in the range of 2 to 4) and $\lambda^{(1)}$ values that are comparable to $\lambda^{(M)}$ – this implies that the approximations to $F(\theta)$ remain fairly expensive and still require an intelligent BO-like procedure to optimize.

Let $\Lambda$ denote the maximum allowed resources that can be used by the multi-fidelity optimization procedure. The number of iterations taken until the resources have been exhausted can be inferred from the evaluation cost of each fidelity above as follows

$$N = \max\{n \geq 1 : \textstyle\sum_{i=1}^{n} \lambda^{(m_i)} \leq \Lambda\}, \tag{4.6}$$

for a given set of data $\{(\theta_i, m_i, y_i)\}_{i\geq 0}$ produced by some algorithm. It is important to note that $N$ is an implicit function of the initial data in this case. This means $N$ cannot be computed *a priori* and will be a *random* variable whenever the algorithm is seeded with some randomly selected initial function evaluations.

### 4.1.2 MFBO using Upper Confidence Bounds

Given the posterior distribution (3.9), the next step is to choose an acquisition function to probe for the subsequent sampling point, as discussed in Chapter 3. A popular choice is Expected Improvement (EI) as the acquisition, but we choose a recent alternative that is particularly relevant in this work, called GP-based upper confidence bound (GP-UCB), that can be defined as follows [103]

$$\alpha_n(\theta) = \mathrm{UCB}_n(\theta) = \mu_n(\theta) + \sqrt{\beta_n}\sigma_n(\theta), \tag{4.7}$$

where $\beta_n > 0$ is some user-specified constant at each iteration $n$ (typically chosen to be $\beta_n = 0.2 n_\theta \log(2(n-1))$ [111]). The main intuition behind UCB is that the mean $\mu_n(\theta)$ encourages querying where we know $F(\theta)$ is high, while the standard deviation $\sigma_n(\theta)$

encourages querying at regions that we are most uncertain about $F(\theta)$. As such, the factor $\beta_n$ directly controls the trade-off between exploration and exploitation.

The reason why GP-UCB is chosen over EI is that we cannot compute a single GP when there are bound constraints between models and hence, we must condition on data only at one fidelity to simplify the model. This restricts the computation of the expectation for the full model, which makes it limiting to use EI. However, each GP is able to bound the response at each fidelity, so it is then easy to use them to compute a set of M upper bounds. We want to take the lowest of these M upper bounds as our acquisition, hence UCB is the preferred choice of acquisition. Now that we have established the choice of the desired acquisition, we can discuss the extension of the GP-UCB algorithm to the multi-fidelity setting based on the algorithm presented by Kandasamy et al. [110]. The main idea is to maintain an upper confidence bound for $F^{(M)}$ using the data available at *all* fidelity levels. Due to the constraints (4.4), the posterior for any $F^{(m)}$ conditioned on all available data is not Gaussian. Let $\mu_n^{(m)}(\theta)$ and $\sigma_n^{(m)}(\theta)$ denote the posterior GP mean and standard deviation for $F^{(m)}$ conditioned on *only* the previous data points available at the $m^{\text{th}}$ fidelity. For a reasonably chosen $\beta_n$ value, we know that $\mu_n^{(m)}(\theta) + \sqrt{\beta_n}\sigma_n^{(m)}(\theta)$ will upper bound $F^{(m)}(\theta)$ with high probability. Combining this with the bounds in (4.4), we know

$$\varphi_n^{(m)}(\theta) = \mu_n^{(m)}(\theta) + \sqrt{\beta_n}\sigma_n^{(m)}(\theta) + \zeta^{(m)}, \quad \forall m \in \{1,\dots,M\}, \tag{4.8}$$

represent a set of $M$ upper bounds for $F^{(M)}$. The best upper bound is then given by

$$\varphi_n(\theta) = \min_{m\in\{1,\dots,M\}} \varphi_n^{(m)}(\theta). \tag{4.9}$$

We use this upper confidence bound for our acquisition function, in place of the traditional single-fidelity GP-UCB in (4.7), meaning that our next query point is at $\theta_{n+1} = \operatorname{argmax}_{\theta\in\Theta} \varphi_n(\theta)$. To determine which fidelity to query, we find the smallest fidelity such

that the following inequality holds

$$\sqrt{\beta_n}\sigma_n^{(m)}(\theta_{n+1}) \geq \gamma^{(m)}, \tag{4.10}$$

where $\gamma^{(m)} > 0$ is a threshold value for all $m \in \{1,\ldots,M-1\}$. If this is not satisfied for any $m$, then we query at $m_{n+1} = M$. The intuition behind (4.10) is that it is not worth spending resources in a region where the function $F^{(m)}$ has a small amount of uncertainty since the bound $\zeta^{(m)}$ caps off how much we can learn about the true function. As such, smaller values for $\gamma^{(m)}$ result in a larger number of queries at fidelity $m$ to reduce the variance below this threshold. A summary of the multi-fidelity GP-UCB (MF-GP-UCB) method is provided in Algorithm 2.

**Algorithm 2** The MF-GP-UCB algorithm [110]

**Initialize:** Input space $\Theta$; specify GP priors $\{\mu^{(m)}(\cdot), k^{(m)}(\cdot,\cdot)\}_{m=1}^{M}$; bounds $\{\zeta^{(m)}\}_{m=1}^{M}$; thresholds $\{\gamma^{(m)}\}_{m=1}^{M}$; initial datasets $\mathcal{D}_0^{(m)} = \emptyset$ for all $m = 1,\ldots,M$; and maximum allowed resources $\Lambda$.
**for** $n = 0$ to $N-1$ **do** ▷ $N$ is defined implicitly based on spent resources according to (4.6).
    **for** $m = 1$ to $M$ **do**
        Construct a GP surrogate model for $F^{(m)}(\theta)$ given available data $\mathcal{D}_n^{(m)}$ similar to (3.9) ▷ only needed if new data was added in previous iteration, otherwise can reuse previous mean and covariance functions.
    **end for**
    Maximize the MF acquisition function to find $\theta_{n+1} = \text{argmax}_{\theta\in\Theta}\, \varphi_n(\theta)$.
    Select fidelity level based on $m_{n+1} = \min\{m : \sqrt{\beta_n}\sigma_n^{(m)}(\theta_{n+1}) \geq \gamma^{(m)} \text{ or } m = M\}$.
    Query the function $F^{(m_{n+1})}(\theta_{n+1})$ to get observation $y_{n+1}$.
    Update $\mathcal{D}_{n+1}^{(m)} \leftarrow \mathcal{D}_n^{(m)} \cup \{(\theta_{n+1}, y_{n+1})\}$ and set $\mathcal{D}_{n+1}^{(m)} \leftarrow \mathcal{D}_n^{(m)}$ for all $m \neq m_{n+1}$.
**end for**

The $\gamma^{(1)},\ldots,\gamma^{(M-1)}$ values are tuning parameters of the algorithm, which are needed to ensure too much effort is not spent at the lower fidelities. This is achieved in practice by setting $\gamma^{(m)}$ to small values for all $m \in \{1,\ldots,M-1\}$; however, if the algorithm does not

query above fidelity $m$ for more than $\lambda^{(m+1)}/\lambda^{(m)}$ iterations, then $\gamma^{(m)} \leftarrow 2\gamma^{(m)}$. All of the $\gamma^{(m)}$ values were initialized to 1% of the range of the observations from the initial queries. Additionally, Algorithm 2 assumes that the bounds $\zeta^{(1)}, \dots \zeta^{(M-1)}$ are given, which is hardly the case in most practical applications. In the available open-source implementation of MF-GP-UCB, these $M-1$ values are converted into a single bound by making the following stronger assumption

$$\|F^{(m)} - F^{(m-1)}\|_\infty \leq \zeta, \quad \forall m \in \{2, \dots, M\}. \tag{4.11}$$

Note that this satisfies (4.4) by setting $(\zeta^{(1)}, \dots, \zeta^{(M)}) = ((M-1)\zeta, \dots, \zeta)$. The value of $\zeta$ is initialized to 1% of the range of the observations from the initial queries. In addition, whenever we query at any fidelity $m > 1$, we check if thee following condition holds $|F^{(m)}(\theta_{n+1}) - \mu_n^{(m-1)}(\theta_{n+1})| > \zeta$; if so, then we also query at fidelity $m-1$. If the difference between the evaluation at the two fidelity levels exceeds the current bound, i.e., $|F^{(m)}(\theta_{n+1}) - F^{(m-1)}(\theta_{n+1})| > \zeta$, then the bound is doubled $\zeta \leftarrow 2\zeta$. Lastly, we note that the required resources $\lambda^{(1)}, \dots, \lambda^{(M)}$ may not be known exactly, so that they must also be estimated as the average computational cost for the initial set of queries at each fidelity level.

Simple regret bounds have been established for the multi-fidelity case shown in Algorithm 2 [110]. The definition of the simple regret is as follows

$$S(\Lambda) = \begin{cases} \min\limits_{n: m_n = M, \forall n \in \{1,\dots,N\}} F^\star - F^{(M)}(\theta_n) & \text{if we have queried at the } M^{\text{th}} \text{ fidelity,} \\ +\infty & \text{otherwise.} \end{cases} \tag{4.12}$$

The main remaining ingredient of the MF-GP-UCB algorithm is how to select the different approximation methods to derive $F^{(1)}, \dots, F^{(M-1)}$. Several different strategies for doing this in the context of DR-based IDC problems are discussed next.

## 4.2 Proposed Lower-Fidelity Models for DR-based IDC Problems

In this section, we discuss three broad approaches for deriving lower fidelity representations of the cost and constraint functions in the high-fidelity problem given by (4.3). Let $\lambda_x$ and $\lambda_u$ denote the cost of determining the successor state in (2.1) and optimal control input in (2.10), respectively, which are assumed to be roughly constant at each time step $t \in \mathcal{T}$. Then, the total cost of a the high-fidelity simulation is then approximately

$$\lambda^{(M)} \sim (\lambda_x + \lambda_u)T. \tag{4.13}$$

The first approach involves the development of dynamic reduced models (D-RM's) for (2.1) to speed up dynamic simulation of the overall system by reducing $\lambda_x$. Modern simulation models can cover a wide range of length- and time-scales depending on the system of interest. For example, physics-based building energy and micogrid simulators, such as EnergyPlus [112] and HOMER Pro [113], require the solution to complex systems of differential algebraic equations (DAEs) that can involve thousands to millions of state variables. If we have access to the underlying DAE-based representation, then we are able to apply physics-based reductions which yields a DAE with many fewer unknown variables, often referred to as *reduced-order* D-RM where "order" refers to the number of effective states variables. If we do not have access to an equation-oriented representation of (2.1), then the main alternative is to construct a *data-driven* D-RM from transient input-output data. This data can be generated from repeated simulation of the system under multiple step changes in the input – system identification methods can then be used to build a surrogate model for (2.1) in the form of, e.g., a Volterra series, neural network, or nonlinear autoregressive moving average with exogenous input (NARMAX) model.No matter how

the D-RM is built, we expect significant gains in simulation time whenever $\lambda_x$ is large since the evaluation of these simplified models can be done in a very efficient manner.

The second approach is based on reducing the complexity of the DR (i.e., lowering $\lambda_u$) which, in this work, involves the repeated solution of the MPC optimization problem (2.10) at every time step of the simulation. In cases that $\lambda_u$ is large, there are several strategies that can be employed to approximate (2.10) in order to develop a lower-fidelity representation of the DR-based IDC problem (4.3). One of the most obvious strategies is to reduce the prediction horizon $\theta_p$, which directly reduces the number of variables that need to be considered in the MPC optimization. Other general approximation strategies include increasing the solver tolerance (or limiting the maximum number of iterations) as well as replacing the cost, system, and/or constraints with convex approximations, so that the resulting approximate optimization can be solved to global optimality using state-of-the-art convex programming solvers.

The third and final approach derives a simplified representation of the time grid (i.e., reducing $T$) using machine learning-based methods. As mentioned previously, one of the key advantages of IDC is considering the transient response of the system to disturbances that are inherently multi-scale in nature. This is an especially large challenge in the context of energy systems where the design decisions last for years (or even decades), while the operational decisions occur on the order of minutes to hours. As renewable energy integration continues to increase in scope, the coupling between these different time scales will only increase in importance. An emerging strategy to include short time scale phenomena in long-term planning problems is to aggregate time-series data into *representative* periods, which directly reduces the number of time steps $T$ needed to complete one closed-loop simulation. It is not uncommon to be able to represent an entire year (365 days) with a set

of 5-10 days, which easily produces 1-2 orders of magnitude reduction in computational cost [114]. Representative periods can be created using time series-based clustering methods that have been developed within the machine learning community – the basic idea is to group periods (usually days) into a small number of groups that are similar. A wide-variety of clustering methods have been used for deriving representative periods, which includes $k$-means, $k$-medoids, hierarchical, and dynamic time warping barycenter averaging (DBA) clustering.

Note that the discussions provided in this section are not intended to be a comprehensive list of all possible reductions. The main goal is to highlight the many different approximation avenues that are available in practically relevant DR-based IDC problems. Furthermore, these different strategies can easily be combined to develop any sequence of models that satisfy bounds above in (4.4). As such, we are not advocating any particular approximation or sequencing strategy in this paper – we intend to study this topic more in our future work. Here, we mainly want to highlight the value of lower-fidelity approximations and their impact in the context of Algorithm 2 on a case study defined in Section 4.3.

## 4.3 Case Study: Integrated Design and Control of a Building Heating-Cooling System with Photovoltaic Power Generation, Battery Storage and Grid Support

### 4.3.1 Description of System Model and IDC problem

We consider the design of a building heating and cooling (HC) system that is connected to a photovoltaic (PV) array and battery energy storage device, with grid support, as depicted in Figure 4.1. The main system design variables of interest are the battery capacity $\theta^{B}_{design}$ and the PV area $\theta^{PV}_{design}$. We also have three key control inputs $u_t = \{u^{HC}_t, u^{B}_t, u^{G}_t\}$ where

$u_t^{HC}$ is the net heating energy supplied to the building (which will be positive when heating and negative when cooling), $u_t^B$ is the energy sourced from the battery, and $u_t^G$ is the energy sourced from the grid. As depicted in Figure 4.1, these variables must always satisfy an energy balance $u_t^{HC} = u_t^G + u_t^B$. We model the grid as an infinite reservoir, meaning that energy may purchased and stored in the battery, used directly for the HC load, or drawn from the battery and sold for profit. Positive values $u_t^G > 0$ indicate electricity has been purchased from the grid, while negative values $u_t^G < 0$ indicate electricity is being sold to the grid. All electricity lines are assumed to be limited to 1500 kWh.

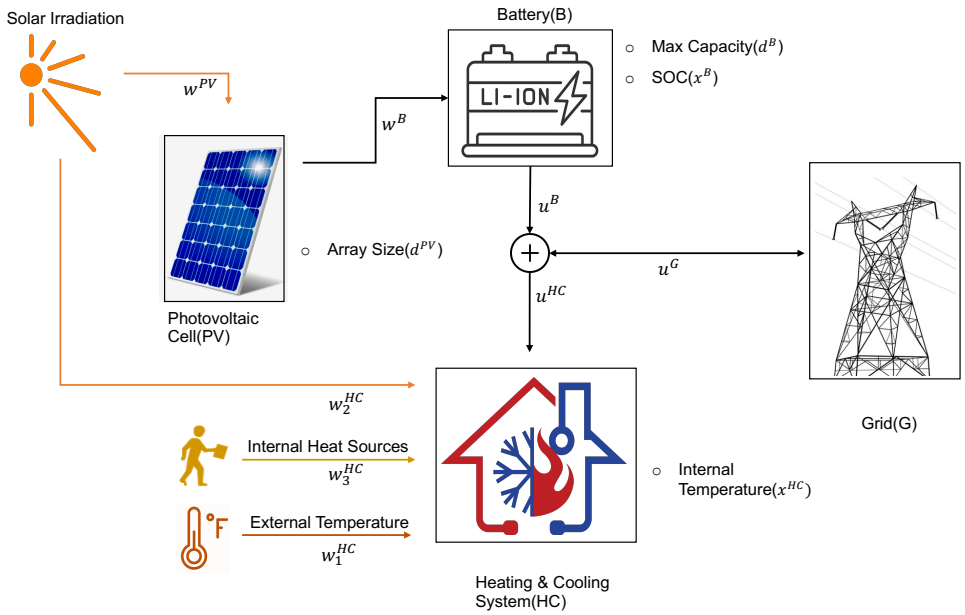


Figure 4.1: Schematic overview of solar-powered building heating and cooling system with battery storage and grid support.

The building model is adapted from Gondhalekar et al. [115] that describes the dynamic evolution of temperature inside of a single room in a larger office building and is composed

of three states $x_t^{HC} = \{x_{t,1}^{HC}, x_{t,2}^{HC}, x_{t,3}^{HC}\}$ that represent the indoor building temperature, average interior wall temperature and exterior wall temperature, respectively.

$$\begin{aligned}
\dot{x}_1^{HC} &= \frac{1}{C_1}[K_3(x_2^{HC} - x_1^{HC}) + K_1(w_1^{HC} - x_1^{HC}) + K_4(x_3^{HC} - x_1^{HC}) \\
&\quad + \tau_1 w_2^{HC} + \eta_h u^h + \eta_c u^c + w_3^{HC}], \\
\dot{x}_2^{HC} &= \frac{1}{C_2}[K_2(w_1^{HC} - x_2^{HC}) + K_3(x_1^{HC} - x_2^{HC}) + w_2^{HC}], \\
\dot{x}_3^{HC} &= \frac{1}{C_3}[K_4(x_1^{HC} - x_3^{HC})],
\end{aligned} \tag{4.14}$$

The battery model is composted of a single state $x_t^B$ that represents the state-of-charge (SOC) of the battery.

$$\dot{x}^B = \frac{100}{\theta_{design}^B}\left(-u^{HC} + u^G + w^B\right), \tag{4.15}$$

where $x^B$ is the SOC in percent. All parameters and variables for these models are defined in Table 4.1. Although a simplified battery model was used here to reduce computational burden in the extensive testing performed, the proposed DR-based IDC and MF-GP-UCB approaches are generally applicable to sophisticated physics-based models including those available in the Simulink toolbox that explicitly model chemistry-specific degradation rates. All relevant parameters in the building and battery models, including the bounds on the design variables and control inputs, are summarized in Table 4.2.

Several disturbances also enter into different components of the model . The PV can only absorb so much energy from the sunlight based on the direct horizontal irradiance (DHI), denoted by $w_t^{PV}$. Only a fraction of the DHI can then be stored by the battery in every time instance (assuming capacity is available), which is given by $w_t^B = \theta_{design}^{PV} w_t^{PV} \eta^{PV}$ where $\eta^{PV}$ denotes the PV efficiency. The external temperature $w_{t,1}^{HC}$, direct normal irradiance (DNI) $w_{t,2}^{HC}$, and internal heat sources $w_{t,3}^{HC}$ also impact the system. Historical weather data

Table 4.1: System model states, inputs, disturbances, and parameters

| Symbol | Description | Units | Value or Range |
|---|---|---|---|
| $\theta^{PV}_{design}$ | PV array size | [m$^2$] | [0, 540] |
| $\theta^{B}_{design}$ | battery capacity | [kWh] | [0, 1300] |
| $x^{HC}_1$ | room air temperature | [°C] | - |
| $x^{HC}_2$ | exterior wall temperature | [°C] | - |
| $x^{HC}_3$ | interior wall temperature | [°C] | - |
| $x^{B}$ | state of charge | [%] | - |
| $u^{HC}$ | net heating load | [kW] | [-1500,1500] |
| $u^{G}$ | energy from grid | [kW] | [-1500,1500] |
| $u^{B}$ | energy from battery | [kW] | [-1500,1500] |
| $\bar{x}^{B}$ | SOC upper bound | [%] | 95 |
| $\underline{x}^{B}$ | SOC lower bound | [%] | 10 |
| $v^{B}$ | SOC violation penalty | [$] | 0 |
| $\hat{v}^{B}$ | SOC violation penalty | [$] | $2 \times 10^6$ |
| $w^{HC}_1$ | outside air temperature | [°C] | [-23,35] |
| $w^{HC}_2$ | horizontal solar radiation | [kW/m$^2$] | [0,1033] |
| $w^{HC}_3$ | internal heat sources | [kW] | [25,35] |
| $\eta^{h}$ | heating efficiency | - | 4 |
| $\eta^{c}$ | cooling efficiency | - | 2 |
| $\tau_1$ | window radiation coefficient | - | 20 |
| $K_1$ | heat conductivity | [kW/° C] | 16.48 |
| $K_2$ | heat conductivity | [kW/° C] | 108.5 |
| $K_3$ | heat conductivity | [kW/° C] | 5 |
| $K_4$ | heat conductivity | [kW/° C] | 30.5 |
| $C_1$ | heat capacity | [kJ/° C] | $9.356 \times 10^5$ |
| $C_2$ | heat capacity | [kJ/° C] | $2.970 \times 10^6$ |
| $C_3$ | heat capacity | [kJ/° C] | $6.695 \times 10^5$ |
| $\pi^{PV}$ | PV price per square meter | [$/m$^2$] | 44 |
| $\pi^{B}$ | battery price per kWh | [$/kWh] | 13.6 |

for Columbus, Ohio was used to represent $w^{PV}_t$, $w^{HC}_{t,1}$, and $w^{HC}_{t,2}$. The internal heat sources $w^{HC}_{t,3}$, on the other hand, were modeled as a uniform random variable between the bounds given in Table 4.1 whenever the time period satisfies $t \bmod 24 \in \{8, \ldots, 18\}$ (represents

Table 4.2: Time-varying constraint, price, and penalty parameter values

| **Time-varying parameters** | **Values in terms of daily time index**, $q = t \bmod 24$ | | | | **Units** |
|---|---|---|---|---|---|
| | $q \in \{0,\ldots,7\}$ | $q \in \{8,\ldots,18\}$ | $q \in \{19,20\}$ | $q \in \{21,\ldots,24\}$ | |
| $\overline{x}_t^{HC}$ | 30 | 26 | 30 | 30 | [°C] |
| $\underline{x}_t^{HC}$ | 19 | 21 | 19 | 19 | [°C] |
| $v_t^{HC}$ | $10^{-4}$ | $10^{-2}$ | $10^{-4}$ | $10^{-4}$ | [\$ / °C] |
| $\hat{v}_t^{HC}$ | $10^2$ | $10^3$ | $10^2$ | $10^2$ | [\$ / °C] |
| $\pi_t^G$ | 0.01 | 0.025 | 0.025 | 0.01 | [\$ / kWh] |

business hours from 8am to 6pm) and zero otherwise. The overall system model can then be cast in the form of (2.1) with states $x_t = \{x_t^{HC}, x_t^B\}$, control inputs $u_t = \{u_t^{HC}, u_t^B, u_t^G\}$, disturbances $w_t = \{w_t^{PV}, w_t^B, w_{t,1}^{HC}, w_{t,2}^{HC}, w_{t,3}^{HC}\}$, and design variables $\theta = \{\theta_{design}^B, \theta_{design}^{PV}\}$ using a forward Euler discretization scheme with a 1 hour sampling time. We also assume a fixed initial condition $b_0(\theta) = \{21°\text{C},\ 20°\text{C},\ 4°\text{C},\ 50\%\}$.

The cost function in the IDC problem (2.3) is composed of capital and operating cost. The capital cost function is the sum of the battery and PV costs, which is given by

$$C(\theta) = \pi^B \theta_{design}^B + \pi^{PV} \theta_{design}^{PV}, \tag{4.16}$$

where $\pi^B$ and $\pi^{PV}$ are the per unit costs of the battery and PV, respectively, reported in Table 4.1. The operating costs, on the other hand, are computed according to a fixed time-varying electricity market structure

$$\ell_s(k, x_k, u_k, w_k, \theta) = \pi_t^G u_t^G, \tag{4.17}$$

where $\pi_t^G$ is a time-varying price. We consider a one-year planning horizon that corresponds to $T = 8760$ steps. Not only do we wish to minimize cost, but we also want to regulate the internal temperature of the building to satisfy time-varying temperature constraints and

respect physical SOC constraints. We can cast these as follows

$$g(k,x_k,u_k,w_k,\theta) = \begin{bmatrix} x_{t,1}^{HC} - \overline{x}_{k,1}^{HC} \\ \underline{x}_{k,1}^{HC} - x_{k,1}^{HC} \\ x_t^B - \overline{x}^B \\ \underline{x}^B - x_k^B \end{bmatrix}, \tag{4.18}$$

where $\overline{x}_{k,1}^{HC}$ and $\underline{x}_{k,1}^{HC}$ are the time-varying upper and lower bounds for the internal building temperature and $\overline{x}^B$ and $\underline{x}^B$ are the fixed upper and lower bounds for the SOC. As shown in (4.2), the constraints are penalized in the objective function with weights $v$. Here, we select the weights $v$ to be time-varying for the temperature constraints, with higher penalties incurred during business hours, and 0 for the SOC constraints. All relevant price and constraint values are reported in Tables 4.1 and 4.2. Note that we have implicitly assumed that we have access to an HC system that offers continuous modes of operation between the maximum and minimum net energy input in our formulation of the system model. In practice, several smaller HC units may be required to satisfy the load – if each unit has a fixed duty cycle, then we would need to include discrete/integer decisions in the model that represent, for example, turning on or off certain units in certain time periods. This assumption was made for simplicity, but our framework can easily accommodate such decision as shown in the previous case study.

### 4.3.2 Comparing MF-GP-UCB to Alternative Black-box Optimizers

We test out MF-GP-UCB on a deterministic version of (2.3). In particular, we assume only nominal disturbance sequence $\hat{\boldsymbol{\omega}} = \{\hat{w}_0, \ldots, \hat{w}_{T-1}\}$ and relax the non-anticipativity constraints such that the control input profile can be optimized under this specific disturbance realization. Under these assumptions, the ideal IDC problem (2.3) simplifies to the following

finite-dimensional optimization

$$\min_{\theta\in\Theta,u_k\in\mathcal{U}} \; C(\theta)+\sum_{k=0}^{T-1}\ell_s(k,x_k(\theta),u_k,\hat{w}_k,\theta)+\ell_T(x_T(\theta),\theta), \tag{4.19}$$

$$\text{s.t. } x_{k+1}(\theta)=f(k,x_k(\theta),u_k,\hat{w}_k,\theta), \qquad \forall k\in\mathcal{T}, \tag{4.20}$$

$$x_0(\theta)=b_0(\theta), \tag{4.21}$$

$$g(k,x_k(\theta),u_k,\hat{w}_k,\theta)\leq 0, \qquad \forall k\in\mathcal{T}, \tag{4.22}$$

where $\theta = d$ since there is no need for a DR in this case as the control inputs are optimized directly. For any fixed value of $\theta$, (4.19) can be written as a large-scale linear program based on the assumed system models (4.14) and (4.15) that can easily be represented in the Yalmip [116] modeling environment and efficiently solved using Gurobi [117]. One high-fidelity evaluation of (4.19) with fixed $\theta \in \Theta$ and $T = 8760$ took approximately 1 minutes on a MacBook Pro with a 2.3 GHz Intel Core i9 and 16 GB of RAM.

To develop a low-fidelity model, we focus on $k$-means clustering approach for simplicity. We consider one year worth of ambient temperature and solar irradiation data for Columbus, Ohio from January 2015 to December 2015. Solar irradiation was represented by the direct normal irradiance (DNI) and direct horizontal irradiance (DHI) values. Temperature, DNI, and DHI profiles over this time period were obtained from the publicly available National Solar Radiation Data Base [118] maintained by by the National Renewable Energy Laboratory. Daily profiles for these quantities over this two-year period are shown in Figure 4.2. To preserve any existing correlation between these quantities, the three datasets are combined together into a 72-element vector before applying the $k$-means clustering algorithm. To determine the "best" number of representative days, we successively run the algorithm for increasing number of clusters $k = 2,3,\ldots,20$ and calculate the normalized sum of squared errors between the clustered and actual days, as shown in Figure 4.3. In

this case, we see that 5 representative days reduces the error to acceptable levels (below 10%). The resulting temperature, DNI, and DHI profiles for the 5 representative days (corresponding to the centroid of the 5 clusters) are shown in Figures 4.2, along with the fraction of the year that each day represents. We are now able to use these representative days, weighted by their fraction of occurrence, as a lower-fidelity approximation of (4.3).

Evaluating the low-fidelity model, which is equivalent to (4.19) with $T = 120$ and $\hat{\boldsymbol{\omega}}$ replaced with the clustered disturbance sequences, reduced the computational cost by a factor of 50 (estimated as the average over ten separate runs at each fidelity level). As such, we were able to set the cost values to $\boldsymbol{\lambda}^{(1)} = 0.02$ and $\boldsymbol{\lambda}^{(2)} = 1$, with $\boldsymbol{\lambda}^{(2)}$ representing the high-fidelity cost that we normalize to 1. We set our total budget to $\Lambda = 12$, which corresponds to 12 equivalent high-fidelity evaluations. Note that 4 out of the 12 maximum budget units are allocated to a set of evaluations at randomly sampled points, which are needed to construct an initial GP as well as estimate the bounds in (4.4).

We compare the MF-GP-UCB algorithm to three alternatives; note that we focus on the BO framework since this has already been demonstrated to be effective in reasonably low-dimensional spaces compared to alternatives (see Appendix A for further discussion). The three alternatives are single-fidelity GP-UCB, expected improvement (EI), and random search. The EI approach was implemented using `bayesopt` [119] within the Matlab Optimization Toolbox, while the others were executed in an open-source implementation of MF-GB-UCB available at: `https://github.com/kirthevasank/mf-gp-ucb`. We use simple regret (4.12) as our performance metric to compare these different methods. Due to the random initialization, simple regret is a random variable and thus it is not informative to show results for a single initialization. Instead, we repeat each experiment 10 times to estimate the average simple regret for each algorithm – error bars are computed by estimating

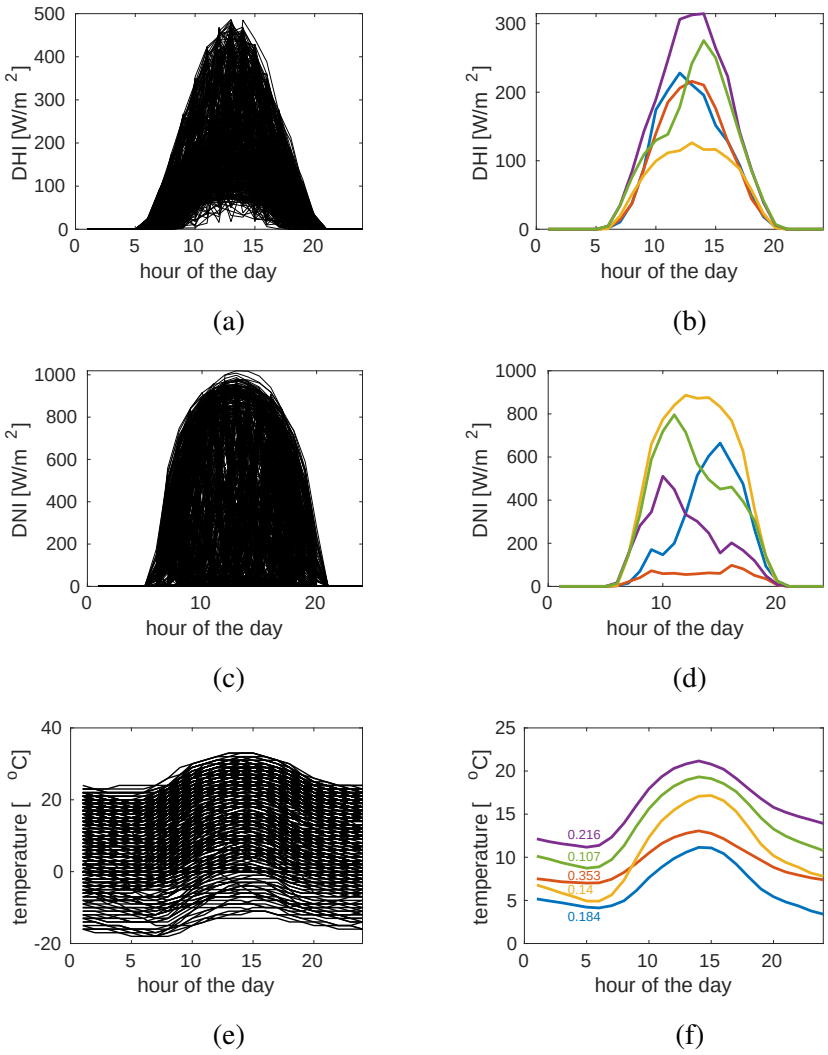


Figure 4.2: Clustering one year of weather data into 5 representative days using k-means clustering. The top, middle, and bottom rows represent DHI, DNI, and temperature, respectively, while the left and right columns represent the year-long and clustered data, respectively. The fraction of occurrence for each representative day in the year is shown next to each curve in (f).

the confidence intervals as 1.96 times the standard deviation divided by the square root of the number of repeats.

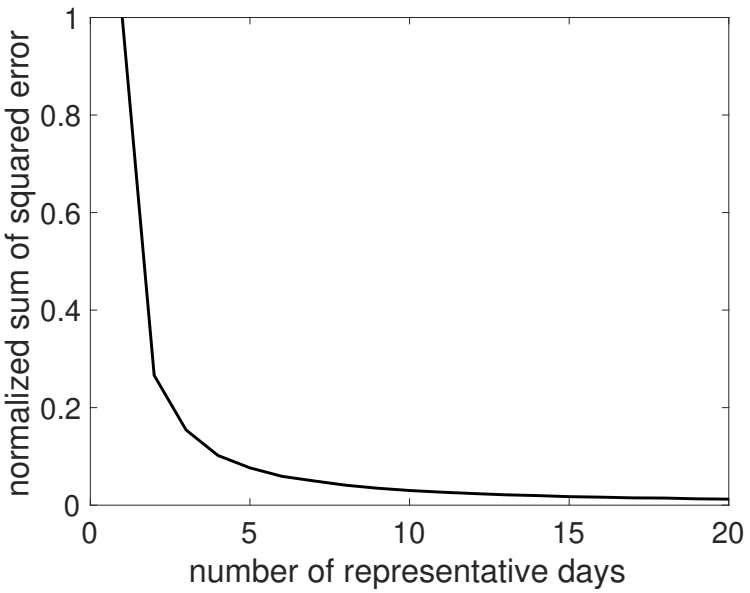


Figure 4.3: Elbow plot showing how well the disturbance dataset for the case study in Section 4.3 is represented as a function of the number of clusters.

The expected simple regret versus fraction of the maximum budget spent for all four algorithms is shown in Figure 4.4. We clearly see that MF-GB-UCB converges faster than the other methods and does particularly well for budget fractions between 0.5 and 0.8. It is interesting to note that random search actually outperformed the single-fidelity GP-UCB approach in this case. We believe this is due to a relatively small sample size of 10 repeats and the tendency for GP-UCB to over explore. EI, on the other hand, does result in similar quality solutions at the end of the budget; however, it does considerably worse at smaller budgets, which suggests significantly worse anytime performance. To demonstrate that MF-GP-UCB was able to identify good designs, we analyze the solution in more detail for the median of the 10 repeats. The year-long sequence of the day-averaged control trajectories are shown in Figure 4.5. Although we see that the average net heating is higher during the summer than winter, there is no clear trend in these results, which highlights the need to

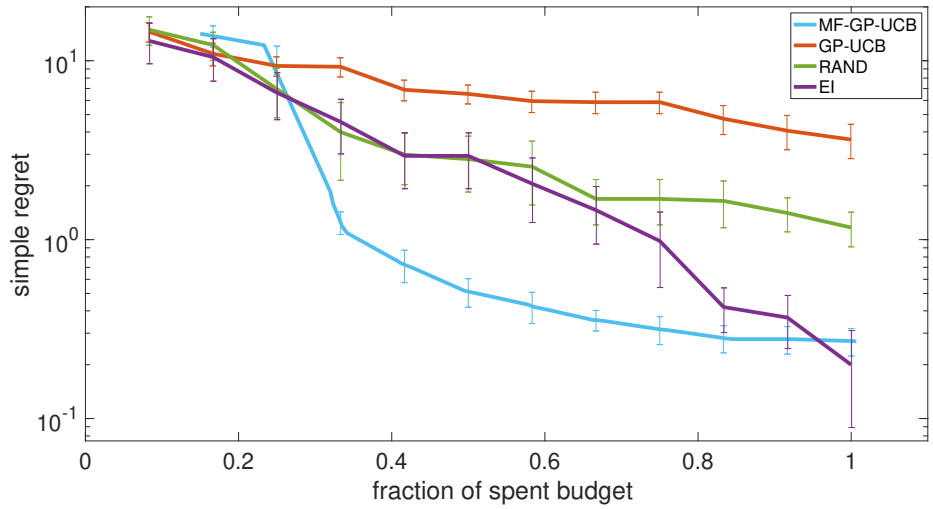


Figure 4.4: The simple regret $S(\Lambda)$ versus fraction of budget spent for four different optimizers. Each approach was repeated for 10 sets of random seed points. The mean of these 10 runs are shown in bold and confidence intervals shown with error bars.

consider short time-scale phenomena when dealing with highly variable disturbances. Based on these daily averaged profiles, we selected three consecutive summer and winter days to plot at the hourly scale in Figure 4.6. From this figure, we can see the cyclic nature of the state and control profiles from day to day as well as strong seasonal effects that result in vastly different control strategies.

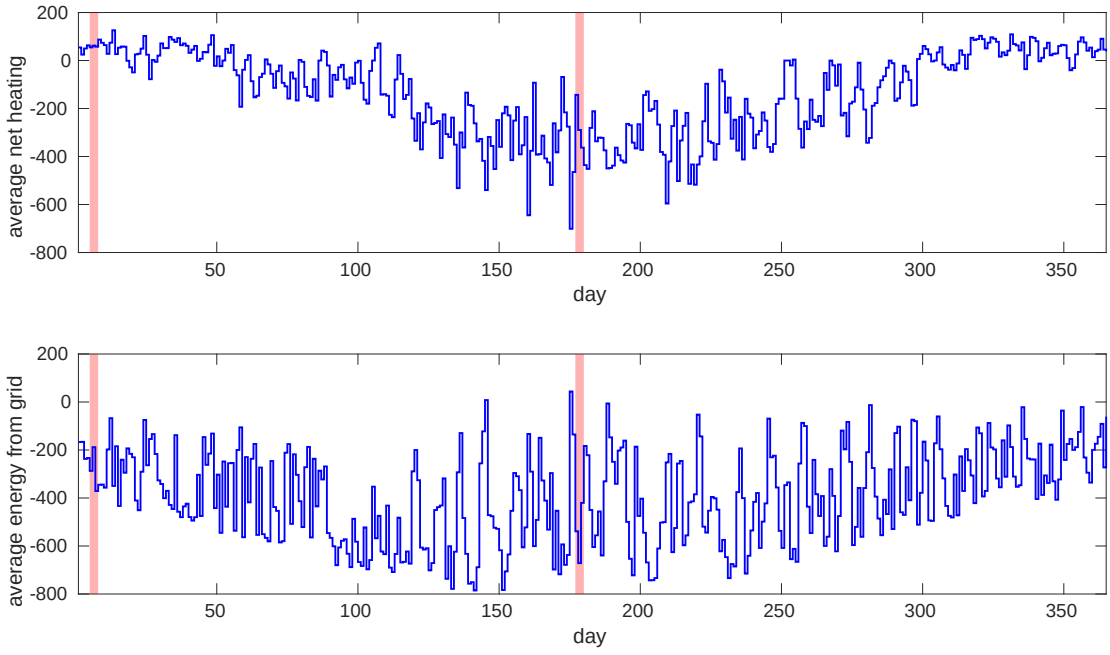


Figure 4.5: Year-long trajectory of the daily average control strategy during business hours (8am to 6pm) for the MF-GP-UCB solution to (4.19). Days 5-8 and 177-180 are highlighted in red, and are shown in detail in Figure 4.6 to highlight the differences in winter versus summer operation

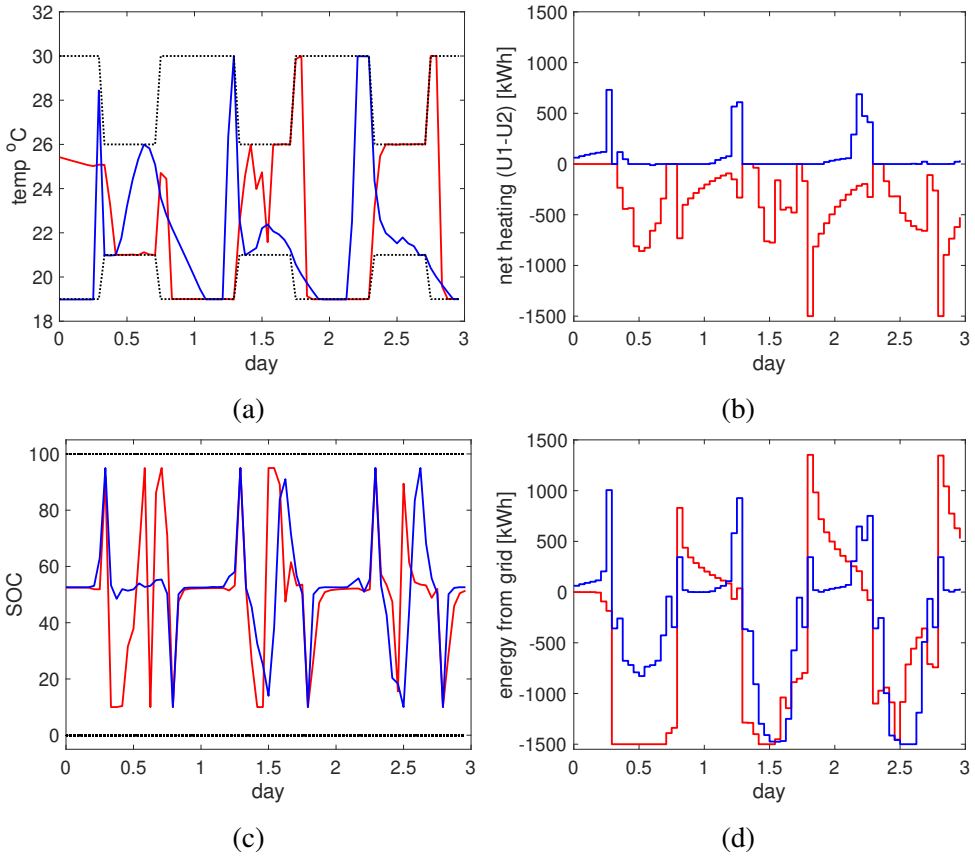


Figure 4.6: State and control trajectories for three consecutive days in winter (blue) and summer (red) corresponding to Figure 4.5.

# Chapter 5: Conclusions and Future Work

## 5.1 Summary of Contributions

This thesis focuses on the development of practical and efficient approaches for solving complex and expensive optimization problems derived from integrated design and control (IDC) models under uncertainty. Such IDC formulations are needed to help enable the shift to next-generation manufacturing and energy systems that can flexibly respond to highly dynamic and uncertain conditions arising from a variety of sources including the increased integration with renewable energy technologies. However, realistic representations of IDC problems are challenging to solve because they are naturally formulated as large-scale, non-convex, and non-smooth optimization problems that cannot be tractably solved using currently available methods. These properties will occur whenever the following details are considered in the IDC problem: (i) relevant dynamics and uncertainties occur on much shorter timescales than the system's lifetime, (ii) many uncertainties are best described by continuous variables with large variance, and (iii) key operational decisions are discrete such as unit commitment or adaptive scheduling. Although several approximation methods that result in more tractable solution methods have been developed, as outlined in Table 1.1, they may lead to highly sub-optimal designs that miss out on key IDC features.

In this thesis, we propose an efficient, flexible, and accurate IDC solution framework that is able to overcome the aforementioned issues. The proposed framework is composed of two

important concepts: (i) approximating the complex recourse decisions using a high-quality decision rule (DR) derived from an advanced nonlinear model predictive control (MPC) law and (ii) applying an efficient constrained simulation-based optimization approach to co-optimize the design variables and DR parameters. Relative to existing solution methods, this thesis does not make any simplifications to the operational details of the IDC problem and uses a nonlinear (potentially mixed-integer) MPC-based DR that can make very accurate operational decisions at every time step. Furthermore, we take advantage of Bayesian optimization (BO) strategies that can be used to optimize objective functions (subject to unknown constraints) that take a long time to evaluate. Through the use of BO, we can flexibly handle any choice of (non-differentiable) DR as well as any type of system model including expensive high-fidelity simulators with multiple interacting components that can be written and stored in different programming languages and/or computing platforms.

Chapter 2 considers the exact formulation of the IDC problem as a multi-stage stochastic program (MSP) that is intractable in realistic cases. In particular, the MSP looks to minimize the sum of the capital cost and the expected operating cost computed over the system lifetime subject to probabilistic constraints on the system state. The evaluation of the expected operating cost requires specific choices of the uncertainty realizations along with control decisions that are specified by a DR. Roughly speaking, a DR is a function that maps the measured data to the control inputs at every operational period and thus provides an offline parametrization of the control decisions at every operational period. Although certain DRs, such as logic-based or proportional-integeral-derivative (PID) controllers, may result in a simpler optimization problem, they can result in highly sub-optimal control actions. We demonstrated this loss in performance on a continuously stirred tank reactor (CSTR) case study that showed significantly higher operational cost and a reduced feasible region when

compared to an MPC controller. Lastly, we discuss how to incorporate MPC into the IDC problem by developing a *design-dependent* MPC law that can adapt to any choice of the design variables by updating the prediction model and relevant tuning parameters.

In Chapter 3, we address the remaining challenge of how to optimize the DR-based IDC problem, which remains intractable due to the presence of probabilistic operators (e.g., expected value) and the absence of derivative information. These features, along with the computationally expensive closed-loop simulations needed to evaluate the objective and constraints, lead us to the choice of BO to tackle the DR-based IDC problem in a fully simulation-based manner. BO constructs a surrogate model for the objective and constraints and quantifies the uncertainty in these surrogates using a Bayesian machine learning algorithm known as Gaussian process (GP) regression. It then defines an acquisition function using the GP surrogate models to decide the next optimal sample location – this process is repeated sequentially until the available budget has been exhausted. We provide a detailed discussion on the implementation of BO and how it can be extended to handle unknown constraint functions, which appear in the IDC problem. We demonstrate the effectiveness of the proposed method for design of a flexible building cooling system. Since the system involves several chiller sub-units, we develop a mixed-integer MPC control strategy that simultaneously selects the number and amount of cooling in each chiller, which is needed to reject time-varying weather, price, and occupation disturbances. Not only does the BO method quickly find designs near the globally optimal solution, we observe significant improvements over sequential optimization-based design and control strategies.

Chapter 4 discusses a novel *multi-fidelity* extension of BO that is able to leverage a series of computationally cheaper (lower-fidelity) approximations to the "exact" IDC problem. The multi-fidelity BO (MFBO) looks to further reduce the time needed to identify good designs

by using bounds on the quality and estimates of the costs of the approximations. Using these bounds, MFBO sequentially selects the next design and fidelity level that should be evaluated. In the chapter, we discuss three major ways to derive low-fidelity approximations of the DR-based IDC problem including simplifications to the system model, DR, and number of simulation time steps. To demonstrate the effectiveness of MFBO, we compare it to traditional BO on the design of a building heating and cooling system with solar power generation, battery storage, and grid support. We consider several sources of uncertainty, including weather and demand conditions, that can vary at the hour scale over a year-long planning horizon. We found that MFBO consistently found better solutions with fewer expensive function evaluations than alternative methods, especially when the budget is very limited (10 or less total high-fidelity runs). Additionally, we found that MFBO was able to suitably handle random forecast errors in the key disturbances and reduce constraint violations by tuning backoff values in the MPC-based DR. Lastly, we analyzed the impact on the sequence of approximations impacted the convergence of MFBO. We found that the relative accuracy and computational cost of the fidelities play an important role in the performance of MFBO in the early iterations (with less accurate, cheaper models being preferred), while the differences in performance tended to shrink as the budget increased.

## 5.2 Suggestions for Future Work

The proposed IDC framework in this thesis, which is based on a combination of MPC-based decision rules and simulation-based BO, showed promising results on multiple case studies. Due to its flexibility, the framework could be directly applied to complex system models arising in many different application areas. However, it is important to note that the underlying black-box assumption of BO does fundamentally limit the rate of convergence

to a global solution. Thus, this thesis leaves room for improvement in the efficiency of the "outer" BO algorithm by exploiting additional information relevant to the structure of IDC problems. We broadly refer to algorithms that supplement the assumption of black-box functions as *grey-box* BO. We discuss specific examples of grey-box BO below, which represent some of the interesting and important directions for future work. Note that the MFBO approach presented in Chapter 4 can be thought of as a type of grey-box BO method.

**Extending BO to high-dimensional problems:** As we move toward IDC of realistic systems that involve a collection of many interacting sub-units, the number of independent design variables can become very large. Additionally, as the complexity of the system increases, the complexity of the MPC controller must also increase, which could result in a large number of tuning parameters. Even when the number of MPC tuning parameters remains small, the overall size of the problem can greatly increase when these parameters are not fixed at all times but are allowed to adapt to different external conditions (e.g., seasonal changes throughout the year). In these cases, the dimensionality of the BO problem can easily become on the order of hundreds to thousands of variables, which is beyond the capabilities of the traditional BO approaches implemented in this thesis (most successful BO applications consider less than 20 independent design variables [120]). Thus, a critical direction for future work is enabling BO to work in high-dimensional inputs spaces $\theta \in \Theta \subset \mathbb{R}^{n_\theta}$. This is a very difficult problem, especially under the assumption that all $n_\theta$ dimensions are important/sensitive, since exponentially more evaluations of $\theta$ would be needed to ensure good coverage of $\Theta$ as $n_\theta$ increases.

Two directions that have been pursued in the literature to overcome this challenge are: low-dimensional embeddings and trust regions. The basic motivation for low-dimensional

embedding methods comes from the simple observation that, in various engineering problems, many of the dimensions do not significantly change the objective function. These types of problems can be said to have a *low effective dimensionality* (i.e., $\delta = T(\theta)$ with $T : \mathbb{R}^{n_\theta} \to \mathbb{R}^{n_\delta}$ and $n_\delta \ll n_\theta$) that can be exploited by the BO procedure by working within a low-dimensional space $\delta \in \Delta$. The key open challenge is how to select the embedding transformation $T$, dimensionality $n_\delta$, and constraint set $\Delta$. This problem is addressed in [121] by using random linear embeddings $\delta = A\theta$ where $A \in \mathbb{R}^{n_\delta \times n_\theta}$ is a random matrix with independent Gaussian elements and the effective dimension $n_\delta$ is assumed to be known. As opposed to selecting a fixed embedding, one can also update the choice of $T$ at every iteration using supervised dimensionality reduction techniques. One such example of this is the SEGOKPLS algorithm [122] that uses a Kriging partial least squares (KPLS) model in place of the standard GP, which was capable of solving a 50 dimensional problem with only $\sim 100$ objective function evaluations. Trust region methods, on the other hand, focus on constructing a local surrogate model and so do not need to make any assumption about the effective dimensionality of the problem. The size of the trust region either shrinks or expands at each iteration depending on the progress made in terms of quality of the objective function. Although established *local* convergence results exist for trust region methods, they rely on fully linear models constructed using some form of interpolation that means the number of evaluations at each iteration scales with the input dimension. Thus, future work could attempt to combine the powerful local guarantees of trust regions with the global exploration power of BO. One example in this direction is TuRBO [123], which is a recently developed trust region BO algorithm that runs several independent models in parallel. An implicit multi-arm bandit approach is then used to decide which local model should be

allocated samples at each iteration. Incorporating these ideas within the IDC framework proposed in this thesis will be an important step for further widening its applicability.

**Improving constraint handling methods in BO:** As is the case in much of the DFO literature, BO was originally formulated for problems with simple (known) constraints. However, as discussed in Chapter 3, BO has been extended to handle black-box constraints using the notion of expected improvement with constraints (EIC). Several alternative constraint handling methods have been developed, as discussed in [99], which categorized them as either *implicit* or *explicit*. In implicit methods, a new merit-based objective function is defined that simultaneously accounts for the effects of the unknown objective and constraints. This includes EIC [100] and the augmented Lagrangian BO (ALBO) method in [124]. Explicit methods, on the other hand, develop a *constrained* acquisition optimization wherein the feasible region is constructed in terms of the GP model of the constraints. This includes the SEGO algorithm [125] and the more recent extension using upper trust bounds [126]. Although we observed good performance with EIC on our IDC case studies, it is an open question as to which method is best-suited to this class of problems. Additional simulation-based experiments are needed to characterize the performance of these approaches on realistic IDC problems – with the hope of identifying a strategy that works "best" in this context. In addition to experiment-driven exploration, it would also be useful to understand the convergence properties of these different constrained BO methods, as this would help practitioners make good choices depending on the problem characteristics.

**Integrating input-dependent noise models with GP regression:** As discussed in Chapter 3, the Monte Carlo-based estimators in (3.2) and (3.3) generate noisy observations for the objective and constraints. For any function $f(\theta) = \mathbb{E}_{\boldsymbol{\omega}}\{F(\theta, \boldsymbol{\omega})\}$ with MC estimator

$y_K = \frac{1}{K}\sum_{i=1}^{K} F(\theta, \boldsymbol{\omega}^{(i)})$, we know that $y_K$ satisfies the central limit theorem (CLT):

$$\sqrt{K}(y_K - f(\theta)) \Rightarrow \mathcal{N}(0, \sigma_f^2(\theta)),$$

where $\sigma_f^2(\theta)$ denotes the variance of the function $F(\theta, \boldsymbol{\omega})$ for any fixed $\theta \in \Theta$ and $\Rightarrow$ denotes convergence in distribution. We see that the measurement error process is *asymptotically* normal with variance that converges to zero as $K \to \infty$. Due to the expensiveness of the closed-loop simulations, however, we must select $K$ to be very low, meaning we are not able to invoke the central limit theorem in practice. As long as the effect of the uncertainty $\boldsymbol{\omega}$ is not too large, we have found that the assumption $y_K = f(\theta) + \varepsilon$ where $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$ works well in practice. Future work is needed to more systematically address this issue by developing more detailed input-dependent (also known as heteroscedastic) noise models, as discussed in [127]. The main idea is to develop another GP model for the noise variance that can be simultaneously estimated with the GP model for $f$. Since this representation no longer remains a GP, it can be challenging to incorporate it into a BO framework. An interesting alternative is to use the so-called *most likely* heteroscedastic GP model in [128], which develops a simplified procedure that retains the overall GP structure for $f$.

**Exploiting composite functions:** Throughout this thesis, we have considered the objective and constraints to be black-box (fully unknown). Although this allowed our framework to be generally applied, this generality does come at the cost of performance (in the form of convergence rate). In most practical engineering problems, only a portion of the model is unknown and, in such cases, it is more accurate to represent the model in a *grey-box* form. A particularly relevant representation of grey-box models is in the form of *composite* functions that are of the form $f(\theta) = g(\theta, h(\theta))$ where $g(\cdot)$ represents the known part of the model and $h(\cdot)$ represents the unknown part of the model (and could provide a vector of outputs). Recent work [99] has shown that significant improvements in regret can be

achieved by exploiting this composite structure whenever available. In the context of IDC problems, $\mathbf{y} = (y_1, \ldots y_T) = h(\theta)$ could represent the set of closed-loop simulation outputs over $t \in \{1, \ldots, T\}$ needed to evaluate the operating cost for a given set of design values $\theta$ and $g(\theta, \mathbf{y}) = C(\theta) + \sum_{t=1}^{T} \ell(y_t)$ could be the total cost composed of the capital cost $C(\theta)$ and sum of the stage costs $\ell(y_t)$ over time with the functions $C$ and $\ell$ being known. One challenge with this representation would be the need to construct a large number of GP models, i.e., one for each component of $\mathbf{y}$, so an interesting open question is how best to exploit this structure in a computationally efficient manner.

# Appendix A: Derivative Free Optimization

## A.1 Introduction & Classification

Advancements in black box optimization or DFO algorithms haven been fueled by increasing interest in applications ranging from problems in science [129–131] to engineering/process design [132–135]. Lets define a function $f : \mathbb{R}^n \to \mathbb{R}$ with constraints $c(x) \geq 0$ with $c : \mathbb{R}^n \to \mathbb{R}^m$ bounded in a domain of interest. This function is computationally expensive to evaluate and may or may not be accompanied by noise. With the assumption that the derivatives of $f$ and $c$ are neither numerically computable nor distinctively available due to discontinuity, DFO algorithms and their computational implementations are the desired choice for optimizing such functions. DFO algorithms automate the process of naively searching for better solutions with trial and error with mathematical convergence theory to guide the exploration to an optimal solution.

DFO algorithms can be broadly broken into 2 approaches; deterministic and stochastic. Stochastic DFO methods can be broadly classified into evolutionary or population-based algorithms like hit and run algorithms [136,137], simulated annealing [78], genetic algorithm (GA) [74] and particle swarm (PSO) [75, 76]. We will not focus on these methods here as they inherently rely on substantial objective function calls, which are expensive to evaluate, leading to prohibitively large computation times along with lacking convergence

results. Deterministic derivative free optimization algorithms can be broken down into two subcategories, namely, Lipschitzian-based partitioning techniques and surrogate-based search algorithms.

### A.1.1 Lipschitzian-based partitioning techniques

Lipschitzian-based methods build and drive a pseudo-function of the original objective function to its optimal to construct an underestimate. If the underestimator is constructed in a piecewise fashion, it is possible to find the global optimum of the original problem. Lets define $L > 0$ as a Lipschitz constant of $f$. By definition, $\left|f(x) - f(y)\right| \leq L\|x - y\|$ for all $x$, $y$ in the domain of $f$. With knowledge of $L$, we can evaluate extreme points of the domain and then construct linear underestimators [77]. Then, we can evaluate the underestimator at the minimum point and further partition the search space to construct another piecewise underestimator. The two major drawbacks of implementing this algorithm are: unknown Lipschitz constant for black-box functions and the exponential increase in number of function evaluations with partitions, as there are $2^n$ extreme points of a $n-$dimensional hypercube. The two algorithms based on Lipschitz partitioning, discussed next, address these challenges.

*The DIRECT algorithm* The first extension to Shubert's algorithm [77] was the DIRECT algorithm (DIvide a hyperRECTange), proposed by Jones et al. [101], where the function evaluations were computed at the centre of the partitions instead of the extreme points. This method also modified the selection criteria of the partitions, hyperrectangles in this case, depending on the rate of decrease of the objective function value. The DIRECT algorithm also gave convergence guarantees for general constrained problems.

*Branch-and-bound (BB) search* BB determines the upper and lower extremes of the optimum by sequentially partitioning the search space [138]. The inferior partitions of the search space are iteratively eliminated during the search.

### A.1.2 Surrogate-based search algorithms

An alternative to partition-based DFO methods are surrogate-based optimization algorithms that look to replace the true (unknown) functions with data-driven models that can be efficiently optimized.. The surrogate-based search algorithms initially sample the search space and build a rudimentary surrogate model, since a high-fidelity surrogate is unavailable for the given black box functions. The surrogate model is iteratively optimized, evaluated and then updated to steer/direct the optimization of the original function. These methods can be further divided into local search and global search methods.

Trust-region methods are popular local search methods that utilize a simple, smooth surrogate model with the assumption that it is accurate around a region about the current iterate, denoted as the 'trust' region. A linear surrogate model was proposed by Powell [139] within a trust-region method. This method considered a radius parameter for the trust region that decreased monotonically with every iteration, but included only those which satisfied geometric conditions of the interpolation points. Further progress in trust-region methods was using a quadratic model as the surrogate [140, 141], which had an added advantage of being able to capture the curvature of the underlying function, as opposed to the linear model. Although, local search methods are effective for local optimization with strong convergence properties, we are interested in methods that evaluate the global optimum. Some global search methods are outlined next.

Global search methods overcome the limitation of reaching only a local optimum of the function. They utilize a similar approach of optimizing a surrogate model and then evaluating and updating it, but the surrogate model is developed initially for the entire domain and then dynamically refined over iterations through partitioning. *Radial basis functions* use an interpolating radial function based model to approximate $f$. Radial basis function were conceptualized for DFO applications by Powell [142]. There are multiple algorithms based on radial basis functions presented and analyzed in literature [143–145] along with extensions to constrained problems in optimization [146] and proof of global convergence.

*Branch-and-fit* optimization methods combine randomization with surrogate modeling for global optimization. Huyer and Neumaier [147] proposed this algorithm where quadratic models are fitted around the best known feasible solution and linear models are used for other evaluated points. On optimizing these models, we obtain candidate points in the search space to evaluate. If the number of candidates are insufficient to fit the models, then random points are generated.

Although all the methods outlined above are effective for optimizing various black box functions, they have a few drawbacks: (1) local search methods are not able to attain global optimum, (2) rate of convergence and number of function evaluations are dependent on the initialization, and (3) they cannot account for noisy surrogate models - assume only deterministic models and hence cannot handle uncertainty. These drawbacks motivate the utilization of *Bayesian Optimization (BO)*, a global surrogate-based search method for black box optimization. the main advantages of BO are the ability to optimally select the next sample point that systematically tradeoffs exploration and exploitation and is able to handle noisy function evaluations.

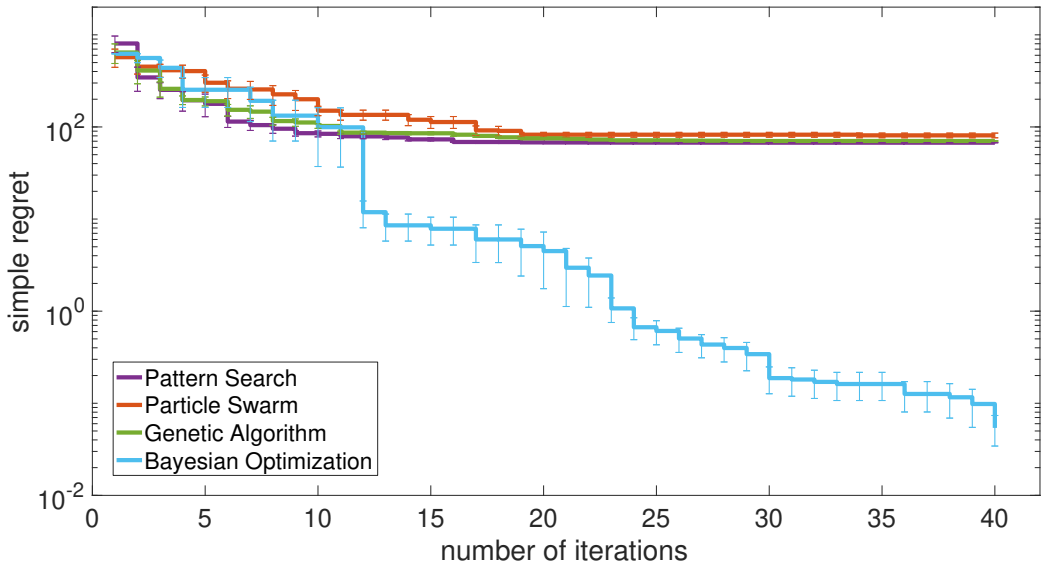


Figure A.1: Expected simple regret for the 2d Himmelblau test function, with approximate confidence region shown via error bars, estimated from 10 independent realizations.

It is useful to compare the performance of BO to several alternative methods mentioned previously to highlight its advantages with respect to a reduced number of function evaluations on a benchmark global optimization problem. We consider a highly nonlinear and multi-modal composite objective function with only box-constraints on the decision variables, called the Himmelblau function [148]:

$$\begin{aligned} \min_{x} \;\; & f(x) = x^2 + (x_1 + x_2^2 - 7)^2 \\ \text{s.t.} \;\; & -5 \leq x_1, x_2 \leq 5, \end{aligned} \tag{A.1}$$

We optimize this function using all four DFO methods mentioned above and use simple regret as the performance metric, defined as

$$S_n = (l_n^\star - l_{true}^\star), \tag{A.2}$$

where $\ell_n^\star = \min_{i=1,\ldots,n} f(x^{(i)})$, the incumbent solution which is the minimum function evaluation observed up until iteration $n$ and $l_{true}^\star = \min_{x \in X} f(x)$ is the global optimum of the optimization problem. The regret acts as a measure of difference between the current best sample at each iteration $n$ of the algorithm versus the global optimum. The simple regret is a non-negative number that will only equal zero whenever the global minimum has been exactly identified. Since simple regret is a random variable due to the random initialization, it is not informative to show results of a single experiment. Hence, we repeat each experiment 10 times for 40 iterations each and use the average simple regret to showcase the performance of each algorithm. The BO algorithm was executed using `bayesopt` in MATLAB R2019a, and compared against GA, PSO and DIRECT algorithms that are representative algorithms in the categories mentioned previously, executed using the Global Optimization Toolbox in MATLAB R2019a. Fig. A.1 shows the expected simple regret over the 10 replications for the Himmelblau function. Error bars showcase the confidence region estimated as the standard deviation divided by the square root of the number of replications. We see that BO outperforms all other tested DFO algorithms by upto 2 orders of magnitude. It converges consistently to the best point across all the methods while the others result in several runs producing virtually no improvement after certain number of iterations. This strongly motivates the choice of BO as the DFO algorithm for the IDC problem.

# Bibliography


[1] J. Ziegler and N. Nichols. Process lags in automatic control circuits. *Trans. ASME*, 65(5):433–443, 1943.

[2] M. Morari. Design of resilient processing plants-iii: A general framework for the assessment of dynamic resilience. *Chemical Engineering Science*, 38(11):1881–1891, 1983.

[3] M. Morari. Flexibility and resiliency of process systems. *Computers & Chemical Engineering*, 7(4):423–437, 1983.

[4] M. Morari, W. Grimm, M. Oglesby, and I. Prosser. Design of resilient processing plants—vii. design of energy management system for unstable reactors—new insights. *Chemical Engineering Science*, 40(2):187–198, 1985.

[5] A. Palazoglu and Y. Arkun. A multiobjective approach to design chemical plants with robust dynamic operability characteristics. *Computers & Chemical Engineering*, 10(6):567–575, 1986.

[6] S. Skogestad and M. Morari. Design of resilient processing plants-ix. effect of model uncertainty on dynamic resilience. *Chemical Engineering Science*, 42(7):1765–1780, 1987.

[7] R. Colberg, M Morari, and D Townsend. A resilience target for heat exchanger network synthesis. *Computers & Chemical Engineering*, 13(7):821–837, 1989.

[8] R. E. Swaney and I. E. Grossmann. An index for operational flexibility in chemical process design. Part I: Formulation and theory. *AIChE Journal*, 31(4):621–630, 1985.

[9] A.M. Lenhoff and M. Morari. Design of resilient processing plants—i process design under consideration of dynamic aspects. *Chemical Engineering Science*, 37(2):245–258, 1982.

[10] L. Narraway, J. Perkins, and G. Barton. Interaction between process design and process control: economic analysis of process dynamics. *Journal of Process Control*, 1(5):243–250, 1991.

[11] I. Bogle and M. Rashid. An assessment of dynamic operability measures. *Computers & Chemical Engineering*, 13(11-12):1277–1282, 1989.

[12] E. Fraga, J. Hagemann, A. Estrada-Villagrana, and I. Bogle. Incorporation of dynamic behaviour in an automated process synthesis system. *Computers & Chemical Engineering*, 24(2-7):189–194, 2000.

[13] W.R. Fisher, M.F. Doherty, and J.M. Douglas. The interface between design and control. 1. Process controllability. *Industrial & Engineering Chemistry Research*, 27(4):597–605, 1988.

[14] W.R. Fisher, M.F. Doherty, and J.M. Douglas. The interface between design and control. 2. Process operability. *Industrial & Engineering Chemistry Research*, 27(4):606–611, 1988.

[15] W.R. Fisher, M.F. Doherty, and J.M. Douglas. The interface between design and control. 3. Selecting a set of controlled variables. *Industrial & Engineering Chemistry Research*, 27(4):611–615, 1988.

[16] K. Papalexandri and E. Pistikopoulos. Synthesis and retrofit design of operable heat exchanger networks. 1. Flexibility and structural controllability aspects. *Industrial & Engineering Chemistry Research*, 33(7):1718–1737, 1994.

[17] M. Mohideen, J. Perkins, and E. Pistikopoulos. Robust stability considerations in optimal design of dynamic systems under uncertainty. *Journal of Process Control*, 7(5):371–385, 1997.

[18] M. Georgiadis, M. Schenk, R. Gani, and E. Pistikopoulos. The interactions of design, control and operability in reactive distillation systems. In *Computer Aided Chemical Engineering*, volume 9, pages 997–1002. Elsevier, 2001.

[19] M.L. Luyben and C.A. Floudas. Analyzing the interaction of design and control—1. a multiobjective framework and application to binary distillation synthesis. *Computers & Chemical Engineering*, 18(10):933–969, 1994.

[20] C. Floudas. Global optimization in design and control of chemical process systems. *Journal of Process Control*, 10(2-3):125–134, 2000.

[21] C. Floudas, Z. Gümüş, and M. Ierapetritou. Global optimization in design under uncertainty: feasibility test and flexibility index problems. *Industrial & Engineering Chemistry Research*, 40(20):4267–4282, 2001.

[22] J.L. Figueroa, P.A. Bahri, J.A. Bandoni, and J.A. Romagnoli. Economic impact of disturbances and uncertain parameters in chemical processes—a dynamic back-off analysis. *Computers & Chemical Engineering*, 20(4):453–461, 1996.

[23] S. Walsh and J. Perkins. Application of integrated process and control system design to waste water neutralisation. *Computers & Chemical Engineering*, 18:S183–S187, 1994.

[24] C.A. Schweiger and C.A. Floudas. Interaction of design and control: Optimization with dynamic models. In *Optimal Control*, pages 388–435. Springer, 1998.

[25] O. Exler, L.T. Antelo, J.A. Egea, A.A. Alonso, and J.R. Banga. A tabu search-based algorithm for mixed-integer nonlinear problems and its application to integrated process and control system design. *Computers & Chemical Engineering*, 32(8):1877–1891, 2008.

[26] M. Asteasuain, A. Bandoni, C. Sarmoria, and A. Brandolin. Simultaneous process and control system design for grade transition in styrene polymerization. *Chemical Engineering Science*, 61(10):3362–3378, 2006.

[27] D. Dominguez, S. Revollar, M. Francisco, R. Lamanna, and P. Vega. Simultaneous synthesis and integrated design of chemical processes using IMC PID tuning methods. *Chemical Engineering Transactions*, 17:1371–1376, 2009.

[28] V. Bansal, J.D. Perkins, E.N. Pistikopoulos, R. Ross, and J.M.G. van Schijndel. Simultaneous design and control optimisation under uncertainty. *Computers & Chemical Engineering*, 24(2):261 – 266, 2000.

[29] V. Bansal, J. D. Perkins, and E. N. Pistikopoulos. A Case Study in Simultaneous Design and Control Using Rigorous, Mixed-Integer Dynamic Optimization Models. *Industrial & Engineering Chemistry Research*, 41(4):760–778, 2002.

[30] L.A. Ricardez-Sandoval, H.M. Budman, and P.L. Douglas. Simultaneous design and control of processes under uncertainty: A robust modelling approach. *Journal of Process Control*, 18(7-8):735–752, 2008.

[31] L.A. Ricardez-Sandoval. Optimal design and control of dynamic systems under uncertainty: A probabilistic approach. *Computers & Chemical Engineering*, 43:91–107, 2012.

[32] S. Mehta and L.A. Ricardez-Sandoval. Integration of design and control of dynamic systems under uncertainty: a new back-off approach. *Industrial & Engineering Chemistry Research*, 55(2):485–498, 2016.

[33] M. Rafiei-Shishavan, S. Mehta, and L.A. Ricardez-Sandoval. Simultaneous design and control under uncertainty: A back-off approach using power series expansions. *Computers & Chemical Engineering*, 99:66–81, 2017.

[34] I. D. Washington and C. L. E. Swartz. Design under uncertainty using parallel multiperiod dynamic optimization. *AIChE Journal*, 60:3151–3168, 2014.

[35] A. Flores-Tlacuahuac and L.T. Biegler. Integrated control and process design during optimal polymer grade transition operations. *Computers & Chemical Engineering*, 32(11):2823–2837, 2008.

[36] R.W. Koller, L.A. Ricardez-Sandoval, and L.T. Biegler. Stochastic back-off algorithm for simultaneous design, control, and scheduling of multiproduct systems under uncertainty. *AIChE Journal*, 64(7):2379–2389, 2018.

[37] V. Sakizlis, J. D. Perkins, and E. N. Pistikopoulos. Parametric controllers in simultaneous process and control design optimization. *Industrial & Engineering Chemistry Research*, 42(20):4545–4563, 2003.

[38] V. Sakizlis, J. D. Perkins, and E.N. Pistikopoulos. Simultaneous process and control design using mixed integer dynamic optimization and parametric programming. *Computer Aided Chemical Engineering*, 17:187–215, 2004.

[39] E.N. Pistikopoulos, N.A. Diangelakis, R. Oberdieck, M.M. Papathanasiou, I. Nascu, and M. Sun. PAROC—an integrated framework and software platform for the optimisation and advanced model-based control of process systems. *Chemical Engineering Science*, 136:115–138, 2015.

[40] S. Avraamidou, N.A. Diangelakis, and E.N. Pistikopoulos. Mixed integer bilevel optimization through multi-parametric programming. *Foundations of Computer Aided Process Operations/Chemical Process Control*, pages In–Press, 2017.

[41] N.A. Diangelakis, B. Burnak, J. Katz, and E.N. Pistikopoulos. Process design and control optimization: A simultaneous approach by multi-parametric programming. *AIChE Journal*, 63(11):4827–4846, 2017.

[42] G. Gutierrez, L.A. Ricardez-Sandoval, H Budman, and C Prada. Integration of Design and Control using an MPC-based superstructure for control structure selection. *IFAC Proceedings Volumes*, 44(1):7648–7653, 2011.

[43] K.B. Sanchez-Sanchez and L.A. Ricardez-Sandoval. Simultaneous design and control under uncertainty using model predictive control. *Industrial & Engineering Chemistry Research*, 52(13):4815–4833, 2013.

[44] S.S. Bahakim and L.A. Ricardez-Sandoval. Simultaneous design and MPC-based control for dynamic systems under uncertainty: A stochastic approach. *Computers & Chemical Engineering*, 63:66–81, 2014.

[45] B.P. Patil, E. Maia, and L.A. Ricardez-Sandoval. Integration of scheduling, design, and control of multiproduct chemical processes under uncertainty. *AIChE Journal*, 61(8):2456–2470, 2015.

[46] C. Noeres, K. Dadhe, R. Gesthuisen, S. Engell, and A. Górak. Model-based design, control and optimisation of catalytic distillation processes. *Chemical Engineering and Processing: Process Intensification*, 43(3):421–434, 2004.

[47] A. Hakizimana. Novel optimization approaches for integrated design and operation of smart manufacturing and energy systems. 2019.

[48] J. Zhang, K. Li, M. Wang, W. Lee, H. Gao, C. Zhang, and K. Li. A bi-level program for the planning of an islanded microgrid including caes. *IEEE Transactions on Industry Applications*, 52(4):2768–2777, 2016.

[49] R. Evins. Multi-level optimization of building design, energy system sizing and operation. *Energy*, 90:1775–1789, 2015.

[50] T. Lambert, P. Gilman, and P. Lilienthal. Micropower system modeling with HOMER. *Integration of alternative sources of energy*, 1(1):379–385, 2006.

[51] N.A. Choksi and J.A. Paulson. Simulation-based Integrated Design and Control with Embedded Mixed-Integer MPC using constrained Bayesian Optimization. In *Proceedings of the American Control Conference*, 2021.

[52] B. Li, R. Roche, and A. Miraoui. Microgrid sizing with combined evolutionary algorithm and MILP unit commitment. *Applied Energy*, 188:547–562, 2017.

[53] M. Sharifzadeh. Integration of process design and control: A review. *Chemical Engineering Research and Design*, 91(12):2515–2549, 2013.

[54] Z. Yuan, B. Chen, G. Sin, and R. Gani. State-of-the-art and progress in the optimization-based simultaneous design and control for chemical processes. *AIChE Journal*, 58(6):1640–1659, 2012.

[55] P. Vega, R. L. De Rocco, S. Revollar, and M. Francisco. Integrated design and control of chemical processes–Part I: Revision and classification. *Computers & Chemical Engineering*, 71:602–617, 2014.

[56] P. Vega, R Lamanna, S. Revollar, and M. Francisco. Integrated design and control of chemical processes–Part ii: An illustrative example. *Computers & Chemical Engineering*, 71:618–635, 2014.

[57] L.A. Ricardez-Sandoval, H.M. Budman, and P.L. Douglas. Application of robust control tools to the simultaneous design and control of dynamic systems. *Industrial & Engineering Chemistry Research*, 48(2):801–813, 2009.

[58] V. Sakizlis, J. D. Perkins, and E. N. Pistikopoulos. Recent advances in optimization-based simultaneous process and control design. *Computers & Chemical Engineering*, 28(10):2069–2086, 2004.

[59] E. N. Pistikopoulos and N. A Diangelakis. Towards the integration of process design, control and scheduling: Are we getting closer? *Computers & Chemical Engineering*, 91:85–92, 2016.

[60] B. Burnak, N.A. Diangelakis, and E.N. Pistikopoulos. Towards the grand unification of process design, scheduling, and control—utopia or reality? *Processes*, 7(7):461, 2019.

[61] M. Mohideen, J.D. Perkins, and E.N. Pistikopoulos. Optimal design of dynamic systems under uncertainty. *AIChE Journal*, 42(8):2251–2272, 1996.

[62] M. Luyben and C. Floudas. A multiobjective optimization approach for analyzing the interaction of design and control. In *Interactions between process design and process control*, pages 101–106. Elsevier, 1992.

[63] L.A. Ricardez-Sandoval, H.M. Budman, and P.L. Douglas. Simultaneous design and control of processes under uncertainty: A robust modeling approach. *Journal of Process Control*, 18(7-8):735–752, 2008.

[64] L.A. Ricardez-Sandoval, H.M. Budman, and P.L. Douglas. Simultaneous design and control of chemical processes with application to the Tennessee Eastman process. *Journal of Process Control*, 19(8):1377–1391, 2009.

[65] D.D. Brengel and W.D. Seider. Coordinated design and control optimization of nonlinear processes. *Computers & Chemical Engineering*, 16(9):861–886, 1992.

[66] A. Bemporad, M. Morari, V. Dua, and E.N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

[67] K. B. Sanchez-Sanchez and L. A Ricardez-Sandoval. Simultaneous design and control under uncertainty using model predictive control. *Industrial & Engineering Chemistry Research*, 52(13):4815–4833, 2013.

[68] P. Vega, R. Lamanna, S. Revollar, and M. Francisco. Integrated design and control of chemical processes–Part II: An illustrative example. *Computers & Chemical Engineering*, 71:618–635, 2014.

[69] R. Baker and C.L. Swartz. Interior point solution of integrated plant and control design problems with embedded MPC. In *AIChE Annual Meeting (San Francisco),(2006)*, 2006.

[70] M. Francisco and P. Vega. Diseño integrado de procesos de depuración de aguas utilizando control predictivo basado en modelos. *Revista Iberoamericana de Automática e Informática Industrial*, 3(4):87–97, 2006.

[71] M. Francisco, P. Vega, and H. Alvarez. Robust integrated design of processes with terminal penalty model predictive controllers. *Chemical Engineering Research and Design*, 89(7):1011–1024, 2011.

[72] G. Gutierrez, L.A. Ricardez-Sandoval, H. Budman, and C. Prada. An MPC-based control structure selection approach for simultaneous process and control design. *Computers & Chemical Engineering*, 70:11–21, 2014.

[73] M. V. F. Pereira and L. M. V. G. Pinto. Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, 52:359–375, 1991.

[74] J.H. Holland. Adaptation in natural and artificial systems, University of Michigan press. *Ann Arbor, MI*, 1(97):5, 1975.

[75] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43. IEEE, 1995.

[76] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-International Conference on Neural Networks*, volume 4, pages 1942–1948. IEEE, 1995.

[77] B.O. Shubert. A sequential method seeking the global maximum of a function. *SIAM Journal on Numerical Analysis*, 9(3):379–388, 1972.

[78] S. Kirkpatrick. Optimization by simulated annealing: Quantitative studies. *Journal of Statistical Physics*, 34(5):975–986, 1984.

[79] J. Bergstra, D. Yamins, and D. Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International Conference on Machine Learning*, pages 115–123, 2013.

[80] P.I. Frazier and J. Wang. Bayesian optimization for materials design. In *Information Science for Materials Discovery and Design*, pages 45–75. Springer, 2016.

[81] M. Meliani, N. Bartoli, T. Lefebvre, M. Bouhlel, J.R. Martins, and J. Morlier. Multi-fidelity efficient global optimization: Methodology and application to airfoil shape design. In *AIAA Aviation 2019 forum*, page 3236, 2019.

[82] F. Sorourifar, G. Makrygirgos, A. Mesbah, and J.A. Paulson. A data-driven automatic tuning method for mpc under uncertainty using constrained bayesian optimization. In *Proceedings of the Advanced Control of Chemical Processes Symposium*, 2020.

[83] J.A. Paulson and A. Mesbah. Data-Driven Scenario Optimization for Automated Controller Tuning with Probabilistic Performance Guarantees. *IEEE Control Systems Letters*, 5(4):1477–1482, 2020.

[84] J. B. Rawlings and D. Q. Mayne. *Model predictive control: Theory and design*. Nob Hill Publishing, Madison, Wisconsin, 2009.

[85] D. Q. Mayne. Model predictive control: Recent developments and future promise. *Automatica*, 50(12):2967–2986, 2014.

[86] M. J. Risbeck, C. T. Maravelias, and J. B. Rawlings. Unification of closed-loop scheduling and control: State-space formulations, terminal constraints, and nominal theoretical properties. *Computers & Chemical Engineering*, 129:106496, 2019.

[87] J. A. Paulson and A. Mesbah. Nonlinear model predictive control with explicit backoffs for stochastic systems under arbitrary uncertainty. *IFAC-PapersOnLine*, 51(20):523–534, 2018.

[88] D.Q. Mayne. Control of constrained dynamic systems. *European Journal of Control*, 7(2):87–99, 2001.

[89] J. Andersson, J. Åkesson, and M. Diehl. Casadi: A symbolic package for automatic differentiation and optimal control. In *Recent Advances in Algorithmic Differentiation*, pages 297–307. Springer, 2012.

[90] A. Wächter and L.T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.

[91] T. Larsson and S. Skogestad. Plantwide control-a review and a new design procedure. 2000.

[92] S. Skogestad. Economic plantwide control. *Plantwide Control*, pages 229–251, 2012.

[93] J.J. Downs and S. Skogestad. An industrial and academic perspective on plantwide control. *Annual Reviews in Control*, 35(1):99–110, 2011.

[94] A. J. Kleywegt, A. Shapiro, and T. Homem-de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, 12:479–502, 2002.

[95] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104:148–175, 2015.

[96] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. MIT Press, Cambridge, MA, 2006.

[97] J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, pages 2951–2959, 2012.

[98] C.E. Rasmussen. Gaussian processes in machine learning. In *Summer School on Machine Learning*, pages 63–71. Springer, 2003.

[99] J.A. Paulson and C. Lu. Cobalt: Constrained bayesian optimization of computationally expensive grey-box models exploiting derivative information. *arXiv preprint arXiv:2105.04114*, 2021.

[100] J.R. Gardner, M.J. Kusner, Z.E. Xu, K.Q. Weinberger, and J.P. Cunningham. Bayesian Optimization with Inequality Constraints. In *ICML*, volume 2014, pages 937–945, 2014.

[101] D.R. Jones, M. Schonlau, and W.J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998.

[102] Bayesian optimization. `https://modal-python.readthedocs.io/en/latest/content/examples/bayesian_optimization.html`. Accessed: 05-10-2021.

[103] N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

[104] M. J. Risbeck. *Mixed-Integer Model Predictive Control with Applications to Building Energy Systems*. PhD thesis, University of Wisconsin–Madison, 2018.

[105] W. C. Rooney and L. T. Biegler. Optimal process design with model parameter uncertainty and process variability. *AIChE Journal*, 49:438–449, 2003.

[106] F. Sorourifar, N.A. Choksi, and J.A. Paulson. Practical integrated design and predictive control of flexible energy systems using multi-fidelity bayesian optimization. *Optimal Control Applications and Methods*, 2021, under review.

[107] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, 18(185):1–52, 2018.

[108] J. Wu, S. Toscano-Palmerin, P. Frazier, and A. Wilson. Practical multi-fidelity Bayesian optimization for hyperparameter tuning. *arxiv*, abs/1903.04703, 2019.

[109] S. Falkner, A. Klein, and F. Hutter. BOHB: Robust and efficient hyperparameter optimization at scale. In *International Conference on Machine Learning*, pages 1437–1446, 2018.

[110] K. Kandasamy, G. Dasarathy, J.B. Oliva, J. Schneider, and B Poczos. Multi-fidelity Gaussian process bandit optimisation. *Journal of Artificial Intelligence Research*, 66:151–196, 2019.

[111] K. Kandasamy, J. Schneider, and B. Póczos. High dimensional Bayesian optimisation and bandits via additive models. In *International Conference on Machine Learning*, pages 295–304, 2015.

[112] D. B. Crawley, L. K. Lawrie, F. C. Winkelmann, W. F. Buhl, J. Huang, Y, C. O. Pedersen, R. K. Strand, R. J. Liesen, D. E. Fisher, and M. J. Witte. EnergyPlus: Creating a new-generation building energy simulation program. *Energy and Buildings*, 33(4):319–331, 2001.

[113] S. Bahramara, M. P. Moghaddam, and M. R. Haghifam. Optimal planning of hybrid renewable energy systems using HOMER: A review. *Renewable and Sustainable Energy Reviews*, 62:609–620, 2016.

[114] H. Teichgraeber and A.R. Brandt. Clustering methods to find representative periods for the optimization of energy systems: An initial framework and comparison. *Applied Energy*, 239:1283–1293, 2019.

[115] R. Gondhalekar, F. Oldewurtel, and C.N. Jones. Least-restrictive robust MPC of periodic affine systems with application to building climate control. In *49th IEEE Conference on Decision and Control (CDC)*, pages 5257–5263. IEEE, 2010.

[116] J. Lofberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No. 04CH37508)*, pages 284–289. IEEE, 2004.

[117] LLC Gurobi Optimization. Gurobi optimizer reference manual, 2020.

[118] M. Sengupta, Y. Xie, A. Lopez, A. Habte, G. Maclaurin, and J. Shelby. The national solar radiation data base (nsrdb). *Renewable and Sustainable Energy Reviews*, 89:51–60, 2018.

[119] R. Martinez-Cantin. Bayesopt: A Bayesian Optimization Library for Nonlinear Optimization, Experimental Design and Bandits. *Journal of Machine Learning Research*, 2014.

[120] P.I. Frazier. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

[121] Z. Wang, M. Zoghi, F. Hutter, D. Matheson, and N. De Freitas. Bayesian optimization in high dimensions via random embeddings. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

[122] N. Bartoli, T. Lefebvre, S. Dubreuil, R. Olivanti, R. Priem, N. Bons, J.R. Martins, and J. Morlier. Adaptive modeling strategy for constrained global optimization with application to aerodynamic wing design. *Aerospace Science and Technology*, 90:85–102, 2019.

[123] D. Eriksson, M. Pearce, J. Gardner, R. D. Turner, and M. Poloczek. Scalable global optimization via local Bayesian optimization. In *Advances in Neural Information Processing Systems*, pages 5496–5507, 2019.

[124] V. Picheny, R.B. Gramacy, S. Wild, and S.L. Digabel. Bayesian optimization under mixed constraints with a slack-variable augmented Lagrangian. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 1443–1451, 2016.

[125] M.J. Sasena, P. Papalambros, and P. Goovaerts. Exploration of metamodeling sampling criteria for constrained global optimization. *Engineering optimization*, 34(3):263–278, 2002.

[126] R. Priem, N. Bartoli, and Y. Diouane. On the use of upper trust bounds in constrained bayesian optimization infill criteria. In *AIAA Aviation 2019 forum*, page 2986, 2019.

[127] P.W. Goldberg, C.K. Williams, and C.M. Bishop. Regression with input-dependent noise: A Gaussian process treatment. *Advances in Neural Information Processing Systems*, 10:493–499, 1997.

[128] K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard. Most likely heteroscedastic Gaussian process regression. In *Proceedings of the 24th International Conference on Machine learning*, pages 393–400, 2007.

[129] M.A. Abramson, T.J. Asaki, J.E. Dennis Jr, K.R. O'Reilly, and R.L. Pingel. Quantitative object reconstruction using Abel transform x-ray tomography and mixed variable optimization. *SIAM Journal on Imaging Sciences*, 1(3):322–342, 2008.

[130] D. Di Serafino, S. Gomez, L. Milano, F. Riccio, and G. Toraldo. A genetic algorithm for a global optimization problem arising in the detection of gravitational waves. *Journal of Global Optimization*, 48(1):41–55, 2010.

[131] G.A. Gray, T.G. Kolda, K. Sale, and M.M. Young. Optimizing an empirical scoring function for transmembrane protein structure determination. *INFORMS Journal on Computing*, 16(4):406–418, 2004.

[132] M.A. Abramson. Pattern search algorithms for mixed variable general constrained optimization problems. Technical report, 2002.

[133] M.C. Bartholomew-Biggs, S.C. Parkhurst, and S.P. Wilson. Using DIRECT to solve an aircraft routing problem. *Computational Optimization and Applications*, 21(3):311–323, 2002.

[134] V. Béchard, J. Chaouki, and C. Audet. *Robust optimization of chemical processes using a MADS algorithm*. Groupe d'études et de recherche en analyse des décisions, 2005.

[135] R.E. Hayes, F. H. Bertrand, C. Audet, and S.T. Kolaczkowski. Catalytic Combustion Kinetics: Using a Direct Search Algorithm to Evaluate Kinetic Parameters from Light-Off Curves. *The Canadian Journal of Chemical Engineering*, 81(6):1192–1199, 2003.

[136] A. Boneh and A. Golan. Constraints' redundancy and feasible region boundedness by random feasible point generator (RFPG). In *3rd European Congress on Operations Research (EURO III), Amsterdam*, pages 9–11, 1979.

[137] R.L. Smith. Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*, 32(6):1296–1308, 1984.

[138] A.H. Land and Doig A.G. An automatic method for solving discrete programming problems. *Econometrica*, 28(3):497, 1960.

[139] M.J. Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in Optimization and Numerical Analysis*, pages 51–67. Springer, 1994.

[140] M.J. Powell. UOBYQA: unconstrained optimization by quadratic approximation. *Mathematical Programming*, 92(3):555–582, 2002.

[141] M.J. Powell. The NEWUOA software for unconstrained optimization without derivatives. In *Large-scale nonlinear optimization*, pages 255–297. Springer, 2006.

[142] M Powell. Recent advances in Cambridge on radial basis functions. In *The 2nd Int. Dortmunt Meeting on Approximation Theory*, 1998.

[143] D.R. Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization*, 21(4):345–383, 2001.

[144] R. Oeuvray. Trust-region methods based on radial basis functions with application to biomedical imaging. Technical report, EPFL, 2005.

[145] K. Holmström. An adaptive radial basis algorithm (ARBF) for expensive black-box global optimization. *Journal of Global Optimization*, 41(3):447–464, 2008.

[146] R.G. Regis and C.A. Shoemaker. Constrained global optimization of expensive black box functions using radial basis functions. *Journal of Global optimization*, 31(1):153–171, 2005.

[147] W. Huyer and A. Neumaier. SNOBFIT–stable noisy optimization by branch and fit. *ACM Transactions on Mathematical Software (TOMS)*, 35(2):1–25, 2008.

[148] David M Himmelblau. Applied Nonlinear Programming, McGraw-Hill Book Company. *New York*, 1972.