Simulation-based framework for design and optimization of wastewater treatment plants

Al, Resul

Publication date:
2020

Document Version
Publisher's PDF, also known as Version of record

Simulation-based framework for design and optimization of wastewater treatment plants

Resul Al

Ph.D. Thesis
April 2020

Department of Chemical and Biochemical Engineering
Technical University of Denmark

DTU Chemical Engineering
Department of Chemical and Biochemical Engineering

Preface

This thesis is submitted as a partial fulfillment of the requirements for the degree of Doctor of Philosophy at the Technical University of Denmark. The work has been carried out at the Process and Systems Engineering Centre (PROSYS) at the Department of Chemical and Biochemical Engineering from May 2017 to April 2020, and supervised by Assoc. Prof. Gürkan Sin, and co-supervised by Dr. Alexandr Zubov and Prof. Krist V. Gernaey.

This project received funding as part of the Marie Skłodowska-Curie Innovative Training Network "ModLife: Advancing Modeling for Process-Product Innovation, Optimization, Monitoring and Control in Life Science Industries" in the Horizon 2020 Program of the European Commission (H2020-MSCA-ITN-2015 call, Project No. 675251).

Østerbro, April 2020

Resul Al

Acknowledgments

First and foremost, I would like to gratefully acknowledge my supervisor, Gürkan Sin, and thank him for giving me the opportunity and the constant support over the last three years and also for placing his trust and confidence in my abilities. Special thanks go to my co-supervisors, Krist Gernaey and Alexandr Zubov, for providing me with their highly appreciated guidance and feedback. It has been a great pleasure to work together and witness the progress of this project.

In addition to my academic supervisors, I would like to also express my heartfelt gratitude to Prof. Costas Pantelides for welcoming me to PSE Enterprise during my secondment in London and providing the time and engaging discussions.

I have greatly enjoyed being part of the friendly work environment at PROSYS, and I would like to express my warmest gratitude to all the people I met here for making this Ph.D. journey a truly memorable experience. Special thanks go to my ModLife gang, Sasha, Łukasz, Mark, Frederico, and Hector for all the fun we had. Thanks to Carolina, Tiago, Rowan, Dasha, Eduardo, Chitta, Katrin, Franz, Francesco, Merve, Simoneta, Rasmus, Jess, Nikolaus, Ergys, Robert, Jérôme, Pau, Renata, Björn, and Manon for their generous and enjoyable companion during this time.

A huge thanks go to our administrative coordinators, Gitte Læssøe and Eva Mikkelsen, who were always ready to give their kind assistance whenever I needed it. Special thanks to Anja Jensen, whose kind welcome and support have been immensely helpful to me.

Last but not least, I would like to remember the unconditional support and encouragement I receive from my large, loving family.

Resul Al
Østerbro, April 2020

Abstract

Mathematical models have become increasingly vital tools in decision-making processes in engineering and beyond. Fueled by the continued growth in computing power and the recent boom in available data, these models have substantially grown in size and complexity, allowing for remarkably accurate simulations of highly complex process phenomena. Notwithstanding the progress, much of what is now achievable through these simulation models still remains not being fully utilized by industrial practitioners, especially during the early stage design of complex engineering systems. A good example is wastewater treatment plants, whose design problem presents a formidable challenge to engineers, and is often addressed through conventional design approaches that are based on industrial practices and experiences rather than mechanistic models. However, such approaches fall short of addressing the heightened level of complexity added to the design problem due to the recently shifting design objectives (e.g., increased focus on resources recovery), the rapidly growing number of emerging treatment technologies along with the uncertainties surrounding the influent compositions as well as the effluent quality requirements. Among the design professionals facing this challenge, there is, therefore, an increasingly pronounced need for integrating recent simulation and optimization techniques into the practiced design approaches.

The objective of this project was to address this need by investigating and developing systematic methodologies and tools to support simulation model-based design and optimization of wastewater treatment plants. The core question to be answered in this thesis is, how can we make the best use of the improved process understanding achieved in the form of high-fidelity process models in deciding a plant design and its operation in order to maximize its performance metrics? Through the use of fundamental process systems engineering tools and methods, the thesis systematically addresses this question by presenting a new comprehensive framework for simulation-based synthesis, design, uncertainty quantification, and optimization of wastewater treatment plants. This unified framework effectively integrates the first principle models of

alternative wastewater processes with its newly developed methods and algorithms for formulating and solving plant synthesis, design space exploration, and optimization under uncertainty problems. Furthermore, the generic features of the proposed framework have been implemented in a set of new software tools to facilitate their wider adoption by non-specialist practitioners in other domains. The thesis also presents these framework tools (along with their application examples), which include SPDLab (for assisting with plant layout selection), easyGSA (for machine learning-assisted design space exploration based on global sensitivity analysis), and MCSKopt (for stochastically constrained simulation-based optimization under uncertainty using Monte Carlo simulation). Finally, the thesis highlights the capabilities of the proposed framework and its tools in several case studies derived from different design and uncertainty scenarios typically encountered in wastewater treatment plants.

Resume

Matematiske modeller er blevet stadig vigtigere værktøjer i beslutningsprocesser inden for blandt andet ingeniørvidenskab. Drevet af væksten i computerkraft og med det nylige boom af tilgængelige data, er disse modeller vokset væsentligt i både størrelse og kompleksitet, hvilket muliggør bemærkelsesværdige nøjagtige simuleringer af meget komplekse procesfænomener. På trods af fremskridt er meget af det, der nu kan opnås gennem disse simuleringsmodeller, stadig ikke udnyttet fuldt ud af industrielle praktikere, især ikke under den tidlige designfase af komplekse ingeniørsystemer. Et godt eksempel er spildevandsrensningsanlæg, hvor designproblemer udgør en stor udfordring for ingeniører, og ofte behandles ved brug af konventionelle designmetoder, der er baseret på industriel praksis og erfaringer snarere end mekanistiske modeller. Sådanne tilgange kan dog ikke tackle det øgede kompleksitetsniveau der føjes til designproblemet på grund af de for nyligt skiftede designmål (f.eks. øget fokus på ressourcegenvinding), det hurtigt voksende antal nye behandlingsteknologier, samt usikkerheden omkring indstrømmende kompositioner såvel som kravene til spildevandskvalitet. Blandt designfagfolk, der står over for denne udfordring, er der derfor et stadig mere udtalt behov for at integrere nylige simulerings- og optimeringsteknikker i de praktiserede designmetoder.

Formålet med dette projekt var at imødekomme dette behov ved at undersøge og udvikle systematiske metoder og værktøjer til at understøtte simuleringsmodelbaseret design og optimering af spildevandsrensningsanlæg. Hovedspørgsmålet, der skal besvares i denne afhandling, er, hvordan kan vi bedst udnytte den forbedrede procesforståelse, der opnås i form af high-fidelity procesmodeller, når vi vælger et anlægsdesign og dets drift for at maksimere dets mål for ydeevne? Ved anvendelse af grundlæggende processystemtekniske værktøjer og metoder adresserer afhandlingen systematisk dette spørgsmål ved at præsentere et ny omfattende framework for simuleringsbaseret syntese, design, usikkerheds kvantificering og optimering af spildevandsrensningsanlæg. Det samlede framework

integrerer effektivt first principles modeller af alternative spildevandsprocesser med dets nyudviklede metoder og algoritmer til formulering og løsning af anlægs-syntese, afsøgning af designråderum og optimering ved tilstedeværelse af usikkerheder. Desuden er de generiske egenskaber ved det foreslåede framework blevet implementeret i et sæt af nye software-værktøjer, der faciliterer deres udbredelse blandt ikke-specialiserede praktikere fra andre områder. Afhandlingen præsenterer også disse framework værktøjer (sammen med deres eksempler på deres benyttelse), der inkluderer SPDLab (til at assistere ved valg af anlægs-layout), easyGSA (til maskinlæringsassisteret udforskning af designråderum baseret på global sensitivitetsanalyse) og MCSKopt (til stokastisk, begrænset, simuleringsbaseret optimering ved tilstedeværelse af usikkerhed ved hjælp af Monte Carlo-simuleringer). Endelig fremhæver afhandlingen kapaciteten af det foreslåede framework og dets værktøjer i flere casestudier, der stammer fra forskellige design- og usikkerhedsscenarier, der opleves i spildevandsrensningsanlæg.

Contents

Nomenclature

ANN	Artificial Neural Networks
AS	Activated Sludge
ASM	Activated Sludge Model
BSM	Benchmark Simulation Model
GPR	Gaussian Process Regression
PCE	Polynomial Chaos Expansion
SK	Stochastic Kriging
WWTP	Wastewater Treatment Plant
EDSS	Environmental Decision Support Systems
IFAS	Integrated Fixed-film Activated Sludge
RBF	Rotating Belt Filter
ERBF	Enhanced Rotating Belt Filter
PC	Primary Clarifier
SC	Secondary Clarifier
AD	Anaerobic Digester
ADM	Anaerobic Digester Model
SRT	Sludge Retention Time
HRT	Hydraulic Retention Time
MLE	Modified Ludzack-Ettinger
LHS	Latin Hypercube Sampling
PSE	Process Systems Engineering
MC	Monte Carlo
MCS	Monte Carlo Simulation
MINLP	Mixed-Integer Nonlinear Programming
DSE	Design Space Exploration
GSA	Global Sensitivity Analysis
OCI	Operational Cost Index
EQI	Effluent Quality Index (Chapter 4 and 5)
COD	Chemical Oxygen Demand

BOD	Biological Oxygen Demand
TSS	Total Suspended Solids
TKN	Total Kjeldahl Nitrogen
TN	Total Nitrogen
KPI	Key Performance Indicator
SNH	Ammonia
HPC	High-Performance Computing
PE	Person Equivalent
CHP	Combined Heat and Power
ELANM	Mainstream Granular-based Anammox Reactor
ELANS	Sidestream Granular-based Anammox Reactor
HRAS	High Rate Activated Sludge
OAT	One factor At a Time
SRC	Standardized Regression Coefficients
DNN	Deep Neural Networks
MLP	Multi-Layer Perceptron
FNN	Feedforward Neural Networks
RNN	Recurrent Neural Networks
EI	Expected Improvement
EQI	Expected Quantile Improvement (Chapter 6)
EGO	Efficient Global Optimization
AEI	Augmented Expected Improvement
cAEI	Constrained Augmented Expected Improvement
FEI	Feasibility Enhanced Constrained Expected Improvement
mcFEI	Multiple Constrained FEI

Chapter 1

Introduction

The chapter establishes the context for the research undertaken within this Ph.D. project and provides a general overview of the research questions and challenges in wastewater treatment plant design. The chapter also states the motivations and goals of the project and presents the overall structure of the thesis. Finally, the chapter closes with a list of dissemination activities carried out during the course of this project.

1.1 Background

To meet today's sustainable development goals (SDGs) [1], the design of modern process systems has to comply with ever more stringent economic, environmental, safety, and health-related regulations, which entails holistic thinking, especially at the early stages of process design. In the process systems engineering (PSE) field, we design computer-aided frameworks to navigate the complexities arising in the design, planning, operation, and control of these process systems, which encompasses a wide range of industries. Witnessing the advent of computers, PSE has been an active area of research for over 50 years, and by applying fundamental systems engineering principles, it has produced a remarkable number of systematic frameworks and paved the way for the development of relatively complex, poorly understood processes [2]. The core areas of research in this field include process synthesis (also referred to as conceptual design), process control, process intensification, process simulation, process optimization, among others. Process synthesis aims at fostering decision-making at the early stage of process design, often involving decisions on unit operations, and their interconnections, to transform raw materials into desired products in a processing flow diagram, usually under highly uncertain conditions with minimal available information.

Faced with this complex challenge, PSE researchers came up with numerous frameworks for process synthesis that can be primarily categorized in two classes: decomposition-based and optimization-based approaches [3]. The first class employs a hierarchical decomposition strategy [4] by dividing and sequentially solving the design problem into several decision layers. In contrast, the latter uses mathematical programming to search for the optimal set of design decisions rigorously, also allowing for interactions among different decision layers [5]. Despite having attracted significant research attention, both approaches have severe limitations, —the first yielding suboptimal designs due to the unconsidered interactions among the design decisions and the latter

imposing strict algebraic limitations on the structures of the models used to describe the relationships among the design decisions, objectives, and the constraints [6].

Fueled by the continued growth in computing power, simulation has emerged as an increasingly prominent tool in decision-making processes in engineering, science, and economics, among its other uses [7]. Using increasingly realistic simulations, engineers now foresee and thereby avoid faulty design alternatives and operational strategies, replacing otherwise costly yet necessary lab experiments. Such realistic simulations are made possible with process modeling and simulations tools. Significant accomplishments achieved in mechanistic process modeling have been successfully translated into commercially available process simulation tools, such as gPROMS, Aspen Plus, and PRO-II, etc. These tools provide engineers with state-of-the-art process understanding derived from a library of rigorous mechanistic process models, which have also grown increasingly complex, describing more process phenomena than ever at an increasingly refined resolution. In doing so, these models also require more information to be specified as their inputs (assumptions, parameters, etc.), which are at best known with uncertainty. When decisions are made based on such models, it becomes of fundamental concern to quantify the impact of these uncertainties on the outputs of the models [8], especially in life sciences industries.

A good example is a biological wastewater treatment plant (WWTP) from the life sciences industries. There is an established WWTP modeling community where intensive efforts have been devoted over the last decades to the development of mechanistic bioprocess models, such as the activated sludge models (ASM) family [9], which then led to the construction of widely-used plant-wide simulation models, such as Benchmark Simulation Model No. 2 (BSM2) [10]. Such models are employed to assist decision-making on the implementation of different control strategies and design configurations [11]. Besides the concerted modeling and simulation efforts, uncertainty is also widely recognized as

a central concept to wastewater systems since the results of those simulation models are significantly affected by the deviations in influent compositions as well as model parameters [12].

In the wastewater industry, recent years have also seen a significant paradigm shift in the way of thinking of wastewater, from a waste to treat towards an increasingly valuable source for energy production and resource recovery [13]. This gave rise to a vast number of available technologies tailored towards the treatment of various pollutants and recovery of scarce nutrients (such as phosphorus (P) and nitrogenous (N) compounds like ammonia and nitrate) [14]. Accounting for 3-4 % of overall electricity demand in the US [15], wastewater treatment is an extremely energy-intensive process. Thus, efforts have also been made to make the next generation of WWTPs energy neutral by maximizing energy recovery from waste sludge streams, mainly in the form of biogas produced in anaerobic digesters. The technological advancements in the industry are also driven by the increasingly stringent effluent quality limits, gradually being imposed by the regulatory authorities, especially on the discharged nitrogen and ammonia levels. The traditional ways of designing wastewater treatment plants rely primarily on expert knowledge, previous experiences, and well-established design guidelines such as US EPA guidelines, German ATV design rules and standards [6]. However, with the changing priorities in design objectives, tighter environmental regulations, and an ever-growing number of alternative treatment technologies (and severe uncertainties in data regarding their cost, performance, technology readiness levels, as well as in influent wastewater compositions), the conventional design approach becomes utterly impractical, often resulting in costly overdesign and falling short of addressing the exacerbated complexity in the conceptual design problem of next generation WWTPs. Therefore, the need for the use of simulation, optimization, and systematic process synthesis methodologies for engineering design of WWTPs is becoming more and more pronounced among the design professionals [16].

Similarly to the rise of simulation, recent years have also seen a surge in data-driven modelling and decision-making due to the abundance of available data. The rapid developments achieved in machine learning and artificial intelligence have now significantly expanded the capabilities of data-driven models (e.g., machine learning or deep learning models), allowing for inference of highly complex and previously poorly understood system behaviors. As an alternative to their mechanistic counterparts, these models are now increasingly put into powerful use in order to generate otherwise unavailable process insights [17]. Far-reaching implications and opportunities are expected in many industries, including water [18], pharmaceutical [19], and process systems [20,21]. However, much of this progress achieved in enriched process understanding (via either mechanistic or data-driven modeling) still remains mostly unexploited by the current conceptual design frameworks depending on mathematical optimization solvers, which limits the practitioners to mathematically tractable algebraic models. Hence, to fully leverage the potential offered by recent advances in machine learning, simulation, and optimization during the early stage process design, new frameworks integrating methods from such fields into the established PSE methodologies (e.g., superstructure optimization, global sensitivity analysis, and uncertainty quantification) are needed. As the early-stage process design in engineering is well-known to be subject to non-negligible uncertainties, robust decision-making in this phase could benefit from such an integrated framework of advanced process simulation, optimization, and systematic management of the inherent uncertainties.

1.2 Motivations and goals

From a broader perspective, the core objective of this project is to propose a framework to develop bridges between two separately evolved PSE areas: process simulation and process optimization, as illustrated in Figure 1.1. The framework needs to bring together the best of the two worlds: state-of-the-art process understanding via process simulations and informed design and operational decision making under uncertainty based upon simulation-based and data-driven optimization algorithms. To this end, efforts will be invested to achieve the following main goals:

- The framework needs to be able to effectively incorporate high-fidelity process models into the early-stage process design, synthesize (using the superstructure approach) and screen out alternative plant layouts, and find/suggest promising plant configurations.
- The framework needs to integrate techniques (e.g., global sensitivity analysis) and tools for how to comprehensively and efficiently explore process design and operational decision spaces (using both high-fidelity simulation and data-driven machine learning models) of complex systems.
- The framework needs to provide insights into decision-making under uncertainty by incorporating suitable simulation-based optimization under uncertainty algorithms along with Monte Carlo techniques for uncertainty quantification. Such algorithms need to be able to interface with process simulation models and optimize design spaces of systems subject to multiple uncertain constraints.

Moreover, efforts will also be allocated to achieve the following sub-goals:

- The framework algorithms need to be made available for public use and implemented as ready-to-employ software tools to facilitate their wider adoption.

- The framework needs to provide extensive parallelization capabilities to comply with modern computing architectures (e.g., high-performance computing clusters, clouds, etc.), as well as insightful data visualization workflows for assisting with decision making.

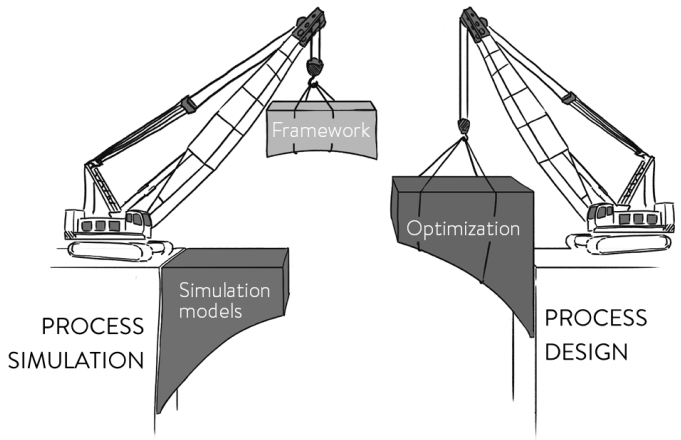

Figure 1.1: Illustration of the gap the proposed framework aims to bridge.

1.3 Structure of the Ph.D. thesis

This Ph.D. thesis is structured into the following chapters:

Chapter 1 Introduction

The first chapter briefly introduces the background and presents the foundation for the development of the framework of this thesis. The chapter also presents project motivations and objectives, as well as the structure of the Ph.D. thesis.

Chapter 2 Literature review

This chapter gives an overview of the state-of-the-art in three research fields: process synthesis and design, simulation-based optimization, and wastewater treatment plant design. Also, the chapter identifies the gaps that the proposed framework of the thesis aims to bridge.

Chapter 3 Simulation optimization-based framework

This chapter presents the simulation optimization-based framework proposed in this project. The chapter also includes the presentation of the developed workflows, components, and tools that are incorporated into the framework.

Chapter 4 WWTP synthesis and design space exploration

This chapter applies the proposed process synthesis approach to the WWTP synthesis problems and showcases the capabilities of the developed decision support tool, SPDLab.

Chapter 5 Plant-wide global sensitivity analysis

This chapter presents a methodological contribution for performing computationally efficient global sensitivity analysis of complex plant-wide models of WWTPs using both mechanistic and machine learning models. It also presents a new tool, easyGSA.

Chapter 6 WWTP optimization under uncertainty

This chapter introduces a novel simulation-based optimization under uncertainty methodology and applies it to the WWTP design optimization problem. The chapter also

presents an in-house developed stochastic black-box solver, MCSKopt, along with its novel infill criteria addressing systems under multiple stochastic constraints.

Chapter 7 Conclusion and future perspectives

This chapter summarizes the main achievements of this project, and finally concludes the thesis with the presentation of remaining challenges and the author's future perspectives.

Chapter 8 Appendix

This chapter provides supplementary information for the previous chapters.

1.4 Dissemination of the Ph.D. project results

The results obtained throughout this Ph.D. project have been disseminated as journal articles, book chapters, and peer-reviewed conference proceedings, as listed below. Besides, the results were also presented at 12 international conferences.

Journal articles

1. Al, R., Behera, C.R., Zubov, A., Gernaey, K. V., Sin, G., 2019. Meta-modeling based efficient global sensitivity analysis for wastewater treatment plants – An application to the BSM2 model. Comput. Chem. Eng. 127, 233–246. https://doi.org/10.1016/j.compchemeng.2019.05.015
2. Behera, C.R., Al, R., Gernaey, K. V., Sin, G., 2020. A process synthesis tool for WWTP - An application to design sustainable energy resource recovery facilities. Chem. Eng. Res. Des. 156, 353–370. https://doi.org/10.1016/j.cherd.2020.02.014
3. Al, R., Behera, C.R., Gernaey, K. V., Sin, G., 2020. Stochastic simulation-based superstructure optimization framework for process synthesis and design under uncertainty. Comput. Chem. Eng. (Revised version submitted.)
4. Hwangbo, S., Al, R., Sin, G., 2020. Development of AI-based and big data-driven framework for process modelling: A case study of the wastewater treatment plant. Comput. Chem. Eng. (Submitted.)
5. Hwangbo, S., Al, R., Chen, X., Sin, G., 2020. Integration of big data and deep learning models for understanding N_2O emissions from wastewater treatment plants. Water Res. (Submitted.)

Book chapter

1. Chen, X., Al, R., Behera, C.R., Sin, G., 2018. Process Synthesis, Design, and Control of Wastewater Treatment Plants, Reference

Module in Chemistry, Molecular Sciences and Chemical Engineering. Elsevier, pp. 1–14. https://doi.org/10.1016/B978-0-12-409547-2.14345-8

Peer-reviewed conference proceedings

1. Al, R., Behera, C.R., Zubov, A., Sin, G., 2018. Systematic framework development for the construction of surrogate models for wastewater treatment plants, Computer-Aided Chemical Engineering. pp. 1909–1914. https://doi.org/10.1016/B978-0-444-64241-7.50313-X
2. Al, R., Frutiger, J., Zubov, A., Sin, G., 2018. Prediction of Environmental Properties Using a Hybrid Group Contribution Approach, Computer-Aided Chemical Engineering. pp. 1723–1728. https://doi.org/10.1016/B978-0-444-64241-7.50282-2
3. Al, R., Behera, C.R., Gernaey, K. V., Sin, G., 2019. Towards development of a decision support tool for conceptual design of wastewater treatment plants using stochastic simulation optimization, 29th European Symposium on Computer-Aided Process Engineering. Elsevier Masson SAS, pp. 325–330. https://doi.org/10.1016/B978-0-12-818634-3.50055-2
4. Magnusson, A. F., Al, R., Sin, G., 2020. Development and application of simulation-based methods for engineering optimization under uncertainty, 30th European Symposium on Computer-Aided Process Engineering (Accepted).

Conference contributions

1. Al, R., Behera, C.R., Zubov, A., Gernaey, K. V., Sin, G., Optimization of wastewater treatment plant design using an early-stage techno-economic analysis under uncertainty, AIChE Annual Meeting, 2018, Pittsburgh, Pennsylvania.
2. Al, R., Behera, C.R., Zubov, A., Gernaey, K. V., Sin, G., Improving speed and efficiency of global sensitivity analysis using meta-

modeling-based approach: A case study on wastewater treatment plant modeling, AIChE Annual Meeting, 2018, Pittsburgh, Pennsylvania.

3. Behera, C.R., Al, R., Gernaey, K. V., Sin, G., Design optimization of wastewater treatment plants using surrogate models, IWA World Water Congress and Exhibition, 2018, Tokyo, Japan.
4. Al, R., Behera, C.R., Zubov, A., Gernaey, K. V., Sin, G., Towards Development of a Computational Framework for Sustainable Process Synthesis and Design, ModLife Conference, 2019, Kgs. Lyngby, Denmark.
5. Al, R., Behera, C.R., Zubov, A., Gernaey, K. V., Sin, G., Towards Development of a Computational Framework for Sustainable Process Synthesis and Design, KT-Consortium Annual Meeting, 2019, Elsinore, Denmark.
6. Al, R., Sin, G., A simulation-based optimization approach for process synthesis and design of wastewater treatment plants, Foundations of Computer-Aided Process Design (FOCAPD), 2019, Copper Mountain, Colorado.
7. Al, R., Gernaey, K. V., Sin, G., A simulation-based superstructure optimization approach for process synthesis and design under uncertainty, AIChE Annual Meeting, 2019, Orlando.
8. Behera, C.R., Al, R., Gernaey, K. V., Sin, G., An optimization-based decision support tool for sewage wastewater treatment process synthesis and design, 10th IWA Symposium on Modelling and Integrated Assessment (Watermatex), 2019, Copenhagen, Denmark.
9. Al, R., Behera, C.R., Sin, G., Towards the synthesis of wastewater resource recovery facilities using Monte Carlo-based simulation optimization, 3rd IWA Resource Recovery Conference (IWARR), 2019, Venice, Italy.
10. Hwangbo, S., Al, R., Sin, G., A deep learning-based framework for sensitivity analysis of N2O emission characteristics from

wastewater treatment plants, IWA World Water Congress and Exhibition, 2020, Copenhagen, Denmark.

11. Al, R., Hwangbo, S., Sin, G., deepGSA: Plant data-driven global sensitivity analysis using deep learning, AIChE Annual Meeting, 2020, San Francisco, California.
12. Al, R., Sin, G., A novel method for simulation-based optimization of stochastic black-box systems subject to multiple constraints, AIChE Annual Meeting, 2020, San Francisco, California.

Chapter 2

Literature review

This chapter provides a review of studies relating to the scope of this thesis, mainly in the fields of process systems engineering and wastewater treatment plant modeling. The methodological approaches for process synthesis and design problems in the PSE field are reviewed along with their applicability in WWTP design problems. Finally, the gaps and challenges in the current literature are identified, and the context for the thesis framework is established. Portions of this chapter have also been published in the following article.

X. Chen, R. Al, C.R. Behera, G. Sin, Process Synthesis, Design, and Control of Wastewater Treatment Plants, Ref. Modul. Chem. Mol. Sci. Chem. Eng., Elsevier, 2018: pp. 1–14. doi:10.1016/B978-0-12-409547-2.14345-8.

2.1 PSE approaches to process synthesis and design

Process synthesis is one of the most fundamental problems in chemical and biochemical engineering, and it aims to find the optimal flowsheet configuration among numerous alternatives given a set of raw materials, unit operations, and desired products. The past three decades in PSE have seen numerous contributions for solving process synthesis problems, made from two distinguishable schools of thought: hierarchical decomposition and superstructure optimization. Extensive reviews of recent contributions have been made by Cremaschi [22] and Chen & Grossmann [3].

The first school employed hierarchical approaches to process synthesis before computers became ubiquitous. Following from the earlier works of Siirola and Rudd [23], Douglas [24] made a seminal contribution along this line by introducing the hierarchical decomposition approach, which invited practitioners to divide the original synthesis problem into five separate sub-problems or decision levels. At each level of this hierarchical approach, available heuristics rules derived from conventional industrial practices are used to assist decisions, which, first and foremost, included a decision on whether a process should be operated continuously or in batch, taking into account the value and lifetime of the products. Level 2 included decisions on the input-output structure of the flowsheet considering raw materials, products, by-products, inert components, and their relations in the process. At this level, the process designer also establishes an overall material balance. At level 3, the process designer then decides on recycle flows, if any, of the flowsheet. Separation system synthesis is decided on level 4, whereas any opportunities for process optimization, such as resource recovery, are explored on level 5. This sequential nature of the decisions makes the initially formidable process design problem easier to solve, and this approach more implementable compared to the other process design approaches, especially in the context of industrial chemical process design [25]. However, as the decisions are made one at a time, interactions

between different decision levels are left unconsidered in the final design, which is, therefore, often sub-optimal [26]. The idea of simultaneous optimization of decisions on each level led the researchers into the second school of thought: the mathematical programming-based synthesis approaches.

Optimization has received tremendous research attention from the PSE researchers due to its promise of finding "truly" optimal designs for the fundamental synthesis problems. By using this approach, superstructures comprising of a large number of flowsheet alternatives are postulated to represent a search space (also referred to as design space) for the simultaneous optimization of design decisions. In doing so, the original process synthesis problem is cast as a constrained optimization problem (subject to process constraints), and the optimal design is sought by solving for minimizing/maximizing an objective (e.g., minimization of capital and operating costs). Application examples include superstructure optimization of process water networks [27], glycerol biorefinery [28], distillation sequence synthesis [29], heat exchanger network synthesis [30], carbon dioxide capture and utilization [31], bioethanol production [32], to name a few. To be able to integrate complex unit models in commercial simulators, attempts were also made to use simulation data-generated surrogate models in superstructure optimization [33,34]. Besides, hybrid approaches integrating hierarchical decomposition with superstructure optimization were also proposed for process synthesis [35,36]. A recent and comprehensive review of various superstructure optimization approaches in PSE is given by Mencarelli et al. [33].

Furthermore, to be able to allow for uncertain parameters in the formulation of superstructure optimization problems, researchers turned to stochastic programming [37], a scenario-based optimization under uncertainty framework. In stochastic programming, uncertainties are modeled as a set of discrete scenarios so as to account for all probable outcomes of uncertain parameters, and a separate deterministic optimization problem is solved for each scenario. Many contributions

were made to the area of process synthesis and design using stochastic programming-based approaches [38–41]. Aside from stochastic scenario-based approaches, robust optimization [42], a more recent optimization under uncertainty approach which focuses on returning a solution that remains feasible for any realization of the uncertainty in a pre-defined uncertainty set, is also explored in PSE applications [43–46], albeit not as extensively. A review of recent advances in mathematical programming-based techniques for the optimization of process systems under uncertainty is given in Grossmann et al. [47].

Despite overcoming the limitations of hierarchical decomposition, superstructure optimization-based approaches for process synthesis rely on mathematical programming solvers, which confine the modeler to mathematically tractable algebraic models for processing units. When those models contain any nonlinear expression, which is often the case in chemical and biochemical processes, the resulting optimization problem is a mixed-integer nonlinear program (MINLP), for which most of the commercially available solvers do not guarantee a global optimum solution without further compromising modelling rigor. Moreover, solutions obtained from those solvers are notoriously sensitive to the input data of the optimization problem, i.e., often greatly varying results can be expected when different realizations of input uncertainties are present. Although this uncertainty problem is partially addressed with the generation of uncertainty scenarios in stochastic programming, there remain computational tractability issues as the scenarios grow exponentially with the size of the problem and the number of uncertain parameters [48].

2.2 Simulation-based optimization

Many engineering systems of today make use of rigorous computer simulations that are increasingly trusted to simulate real-world consequences of various design and operating scenarios. Simulating multi-scale phenomena from molecular to enterprise-level, these models usually comprise of large systems of ordinary or partial differential equations in order to provide detailed simulations of temporal or spatial variations of system properties, which made them essential for assisting decision-making in design, operation, and control of these engineering systems. However, integration of these models with optimization-based decision support tools was impractical due to the closed nature of underlying optimization solvers, each of which requires users to use their own proprietary syntax for representing optimization problems, and uses specific model structure (such as information of derivatives, etc.) to be able to return an optimum [49]. This had led to a proactive interest in developing simulation data-driven optimization algorithms that can integrate rigorous process simulation models in search of the optimal solutions [50].

Driven by the recent advances in simulation and continued growth in computing power, a new optimization paradigm, termed stochastic simulation-based optimization (also referred to as *simulation optimization*), has recently been developed in order to incorporate both simulation models and data uncertainty considerations into optimization algorithms. Unlike deterministic optimization, simulation optimization does not assume that an algebraic expression between a target objective and the decision variables exists, treating the system to be optimized as a stochastic black-box whose objective and constraint values can rather be estimated with some variability, be it due to the simulation noise or the uncertainty [51]. This not only allows for the formulation of stochastic objectives and constraints for the optimization problem but also provides a means for taking into account available uncertainty information. Hence it is not surprising that simulation optimization is most commonly used

to optimize the expected value of a performance metric of a stochastic system, finding applications in fields as disparate as engineering, medicine, transportation, and logistics [52]. As also shown in Figure 2.1, a literature survey of documents for simulation optimization in Scopus database yields an almost steadily growing number of documents, especially in recent years, from a diverse array of subject areas.

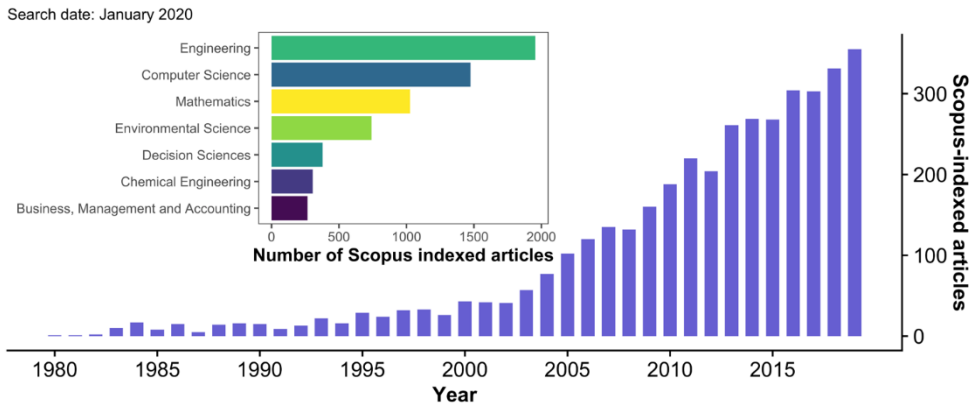

Figure 2.1: Number of Scopus-indexed documents containing "simulation optimization" in its title, abstract, or keywords. Also shown are subject areas of all the documents. Search date: January 2020.

Algorithms designed for the solution of simulation optimization problems can be broadly classified as falling into one of the following categories: ranking and selection, model-based methods, gradient-based methods, direct search methods, sample-path approximation, and multi-objective simulation optimization methods. A comprehensive review of these methods can be found in Amaran et al. [52]. Ranking and selection (R&S) methods can be directly applicable (albeit not scalable) when the number of alternatives is relatively small and finite, whereas gradient-based methods can be employed when the objective and constraint are differentiable or can be estimated via a mathematical framework such as stochastic approximation (SA). If the gradient is not readily available, one can resort to alternative derivative-free black-box search methods, such as pattern search, genetic algorithm, simulated annealing, scatter search, tabu search, etc. More details of these methods can be found in [53]. On the other hand, model-based methods, or surrogate-based methods, can

approximate a computationally expensive simulation model by constructing a surrogate from its observations and thereby generating insights into search directions towards an optimum. Extensive reviews of advances achieved in surrogate-based optimization can be found in [54,55].

In PSE literature, recent years have seen an increasing number of studies focusing both on the development and the application of simulation data-driven optimization methods to process synthesis, design, control, and optimization problems. In an initiative by the U.S. Department of Energy, Eslick et al. [56] developed a comprehensive framework for optimization and uncertainty quantification of carbon capture systems. Their framework relied on ALAMO surrogate models [57] for superstructure optimization and simulation-based derivative-free solvers to interface with the high-fidelity process models for design optimization. Hasan et al. [58] developed a simulation-based approach using Kriging models to investigate the optimal selection of CO2 capture technologies. Similarly, Boukouvala et al. [59] developed a constrained grey-box optimization approach using Kriging models to optimize the pressure swing adsorption column. Bajaj et al. [60] developed a trust region-based simulation optimization algorithm for black-box systems and applied it to a case study of designing an integrated carbon capture and conversion system. Using the same simulation-based algorithm, Arora et al. optimized the design parameters of hydrogen [61] and methanol [62] production. Wang and Ierapetritou [63] presented a Kriging-based constrained optimization framework for stochastic black-box systems and applied it to the optimization of a pharmaceutical manufacturing process.

2.3 Design of wastewater treatment plants

With the ever-growing number of alternative treatment technologies becoming economically viable and changing design priorities due to the ongoing paradigm shift in the industry towards resource recovery, the wastewater treatment plant synthesis/design problem is now more complicated than ever. Conventional design approaches used expert knowledge and experience-based engineering guidelines, such as EPA (US federal guidelines) and ATV (German design rules and standards), which are getting incapable of addressing the unprecedented complexity, especially given the extra design and operational degrees of freedom offered by the advanced treatment and separation technologies [64]. Moreover, in order to accommodate changing influent compositions and flows in conventional WWTPs, safety factors are typically used to ensure effluent quality requirements. Consequently, the conventional guidelines-based WWTP is an inevitably over-dimensioned design, translating to higher construction and operation costs [6]. To overcome these issues, researchers looked into systematic methods applying systems engineering approaches, which are discussed next.

Early attempts to use systematic methods were made for the development of expert knowledge-based environmental decision support systems (EDSS) [65], for the selection of appropriate wastewater treatment systems for small communities, acknowledging the need for tools to assist the decision-making process. Attempts were also made for the water network design problem in process plants [66,67], for which a mixed-integer nonlinear program (MINLP) was derived to solve the synthesis problem using the superstructure optimization-based approach borrowed from the PSE field. For the design of municipal WWTPs, Flores et al. [68] proposed a systematic procedure to explore critical tradeoffs during the conceptual design of activated sludge plants using a weighted sum approach involving multi-objective design criteria based on mathematical modeling and qualitative knowledge. Similarly, Garrido-Baserba et al. [69] developed a knowledge-based methodology

based on a hierarchical decision approach, which was also implemented in an environmental decision support system (Novedar EDSS), and later applied [70] to several design scenarios to select among different WWTP configurations. Alasino et al. [71,72] used nonlinear programming to optimize both process configuration and equipment dimensions of an activated sludge plant simultaneously by using net present value (NPV) as the objective function. On the other hand, Hakanen et al. [73] investigated the interplay among multiple conflicting design objectives and proposed a multi-objective optimization-based decision support tool for the design optimization of WWTPs. Bozkurt et al. [74] proposed a superstructure-based optimization framework to address the optimal WWTP network selection problem along with considerations of the robustness of the solutions against uncertainty. Their methodology relied on the simplified algebraic description of wastewater processes in a generic process model and the use of mixed-integer linear programming solvers for solving the resulting optimization problem. Later, Bozkurt et al. [75] applied this framework to a realistic full-scale WWTP case study and found out high variations in the objective values due to the considerations of uncertainties. Castillo et al. [76] integrated this framework with the knowledge-based Novedar EDSS in order to further assist with the optimal network selection problem of WWTPs. Similarly, Puchongkawarin et al. [77] also proposed a superstructure optimization-based methodology for the synthesis of WWTP networks. While recognizing the importance of full-fledged process simulation during the early stage design, their methodology made use of simple regression models for performance and cost calculations and formulation of an NPV-based objective function that is solved with deterministic optimization solvers. More recently, Behera et al. [78] proposed a process synthesis tool for the design of municipal WWTPs using simulation optimization. By interfacing with a model library of high fidelity process models, their tool allows decision-makers to explore multiple design objectives under various uncertainty scenarios. Although their approach represents the

latest attempt to directly employ rigorous process models to address the optimal plant design selection problem in WWTPs, exploration of design spaces in each plant layout does not utilize data-driven optimization techniques but Monte Carlo sampling-based exploration. On the whole, recent studies show a clear trend towards more systematic approaches being proposed for addressing the design problems in WWTPs, as opposed to the previously employed design guidelines. However, there still remain several critical knowledge gaps hindering the application of such approaches in design of WWTPs. In the next section, those gaps and challenges are identified.

2.4 Gaps and challenges

Despite the significant progress made in mechanistic process modeling and simulation of wastewater treatment processes over the last three decades, the integration of those state-of-the-art models in decision support tools for selecting the optimal plant design remains limited. The literature review reveals that there are apparent gaps between currently achieved process understanding and currently employed process design practices in wastewater treatment plants. More specifically, this project aims to address the following three critical gaps.

The first critical gap identified in the literature is that the current design practices of WWTPs harness neither the power of improved process understanding achieved by the high-fidelity process modeling nor the potential of simulation optimization techniques for fostering design decision-making. There is a lack of an early-stage design tool for systematically and effectively incorporating high-fidelity process models to address the optimal WWTP layout problem by synthesizing and evaluating a large number of alternative layouts (e.g., via superstructures) and suggesting the optimal ones among them. There is also a need for how to effectively explore/optimize the design and operational decision spaces of each alternative layout using simulation models. Therefore, decision making at the early-stage design can significantly benefit from such a decision support tool providing practical algorithms targeted at addressing the needs mentioned above.

The second critical gap identified in the literature is that the WWTP design practice does not leverage the potential of machine learning algorithms (e.g., Gaussian process regression, artificial neural networks, etc.) for gaining a deeper understanding of the most significant design and operational decisions as well as the most crucial uncertainty drivers of the plant performance indicators. Although methods exist in PSE for addressing such needs, such as global sensitivity analysis methods, the application of those methods to plant-wide WWTP models is hampered

by the lack of an enabling tool that would alleviate the high computational costs associated with performing such useful analysis by systematically constructing and employing the machine learning-based surrogate models. Therefore, such a tool would prove useful by providing decision-makers with readily available quantitative measures for assessing the worthiness of exploring and further optimizing WWTP design decision variables.

The third critical gap identified in the literature is that even though attempts have been made to integrate rigorous simulation data into optimization-based design frameworks, none of the studies above addressed the combined challenge of how to use directly first principles simulation models for optimization while also quantifying and taking into account uncertainties surrounding the plant design objective and constraints. Especially in the context of WWTPs, those uncertainties are hardly negligible, stemming from diverse sources (e.g., model parameters, influent pollutant compositions, regulatory limits for effluent quality, cost and performance data, to name a few.). Conventional scenario-based optimization under uncertainty frameworks, such as stochastic programming, are not directly applicable along with detailed process models as they rely on restrictive solvers discouraging integration with such detailed models [49]. Alternatively, Monte Carlo techniques for uncertainty quantification allow for natural and non-intrusive integration with process simulation models as well as a comprehensive analysis and assessment of a large number of uncertainty parameters [79]. Hence, there is a need for a new simulation-based optimization under uncertainty framework incorporating simulation-data driven optimization techniques with Monte Carlo techniques for uncertainty quantification. There is also a need for a software tool implementing such a framework applicable to the specific needs of WWTP design under uncertainty problems,—such as how to handle multiple uncertain effluent quality constraints while searching for a better design.

Chapter 3

Simulation optimization-based framework for process design

This chapter presents the proposed framework for simulation-based process synthesis and design of wastewater treatment plants. The framework consists of three major stages, namely process synthesis, process design, and optimization under uncertainty. Each stage has its own workflows and tools, which are also discussed generically. The proposed framework was also presented in the following submitted article.

R. Al, C.R. Behera, K. V. Gernaey, G. Sin, Stochastic simulation-based superstructure optimization framework for process synthesis and design under uncertainty, Comput. Chem. Eng. (under review).

3.1 Framework overview: A 3-stage approach

In order to manage the complexity associated with the overall process design problem, the proposed framework adopts a 3-stage approach, each of which sequentially addresses different sub-problems of the previously introduced process design problem. Each stage produces an output that serves as an input to the further stages. By decomposing the problem into three stages, the framework aims to refine the search space for the optimal design and allocate more computational resources towards those designs that are worthy.

In the first stage, *process synthesis*, a superstructure is postulated in order to represent a large number of plant layout alternatives, which are synthesized using both combinatorial and expert knowledge-based synthesis approaches. The inputs to this stage include design problem definition along with objectives, constraints, and technologies of interest. The second stage, *process design*, performs simulation-based design space exploration of previously generated layouts using sampling-based approaches. The core objective at this stage is to identify top k layouts that contain the most promising designs with respect to the design objective, hence worthy of further consideration. The final stage, *optimization*, investigates further opportunities for optimization of key plant performance indicators of the previously selected top designs under uncertainty considerations, hence providing insights into decision-making under uncertainty. Every stage uses high-fidelity model-based process simulations in order to generate rigorous insights and also to validate the effectiveness of solver-proposed designs. A graphical representation of the 3-stage approach is shown in Figure 3.1, which illustrates the inputs and outputs of each stage. The overall workflow of the 3-stage-approach is also shown in Figure 3.2 and will be detailed in the subsequent sections.

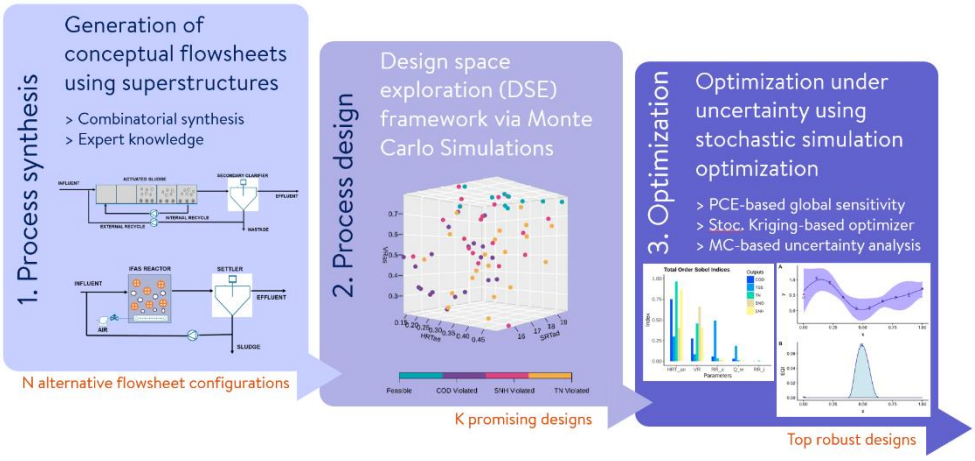

Figure 3.1: Overview of the 3-stage-approach adopted in the proposed framework.

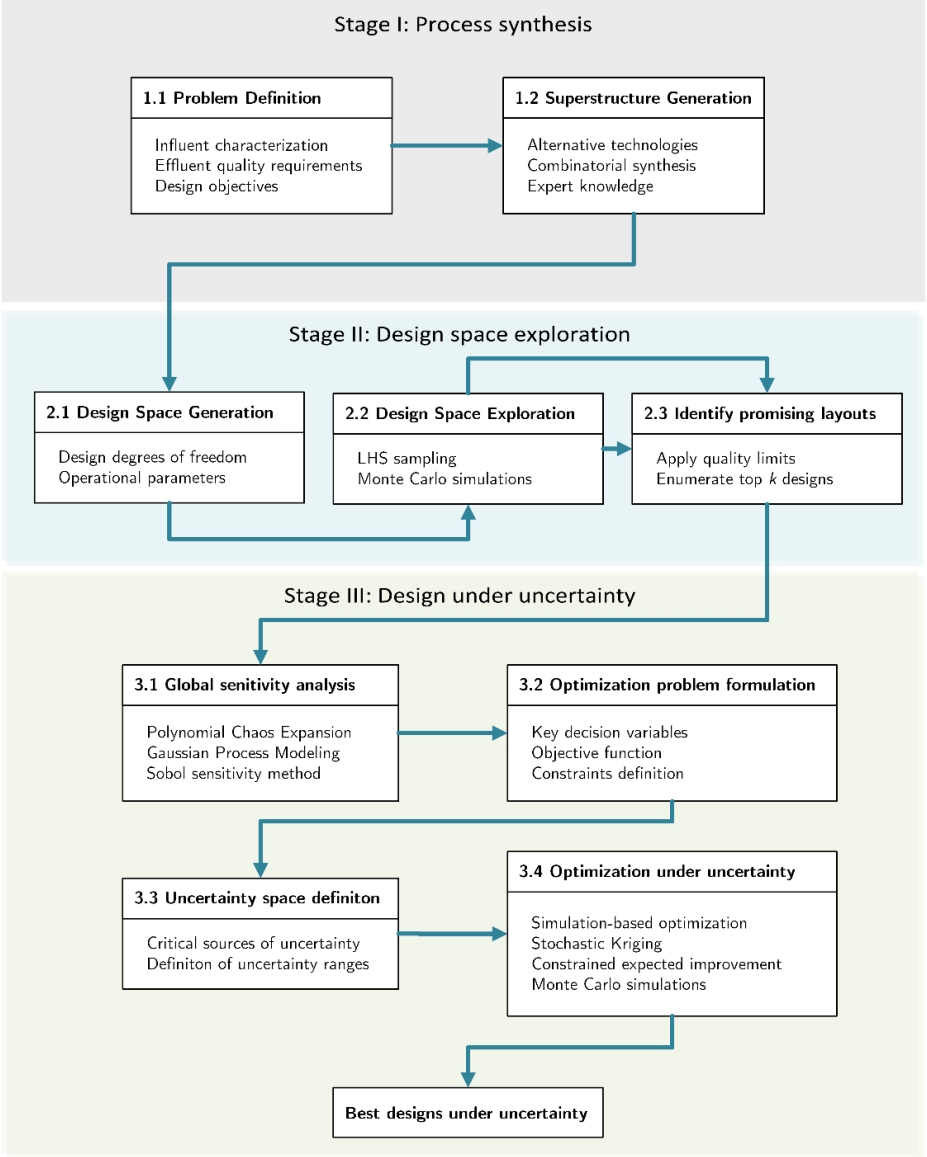

Figure 3.2: The overall workflow of the proposed framework [80].

3.2 Stage 1: Process synthesis

In the first stage of the proposed framework, a process synthesis problem is formulated and solved using the concept of superstructures. First and foremost, a statement of the overall design problem is needed in order to set the scope and the boundaries for the design problem at hand. A generic process design problem statement outlines all the options for feedstock, processing technologies, and products in order to find the optimal process configuration according to a design objective subject to various operational, design, economic, and environmental constraints. For wastewater processes, this includes specification of the influent wastewater flow and compositions, treatment technologies, expected effluent quality concentrations, and a primary design objective (i.e., better effluent quality, minimum operating costs, etc.). Generally, the primary design objective for a wastewater treatment project is to treat the contaminants with the lowest possible cost in order to satisfy the effluent quality limits set by the regulatory authorities. Alternatively, the design problem statement for a wastewater treatment project can also encompass several more aspects, such as the maximum possible recovery of energy or scarce materials [81].

Secondly, a superstructure is postulated to represent the technological design space of the problem, within which an optimal processing flow diagram will be sought after. A superstructure, as generically shown in Figure 3.3, is a matrix-like compact structure whose columns represent processing steps (or treatment tasks) needed to transform raw materials into desired products. For wastewater processes, raw materials are wastewater sources, whereas products are treated wastewater and recovered resources. In each column of the superstructure, there are usually several different alternative technologies, among which any can be selected to perform the necessary treatment task. Finally, there are also interconnections among different technologies, which also needs to be defined in order to complete the definition of a superstructure.

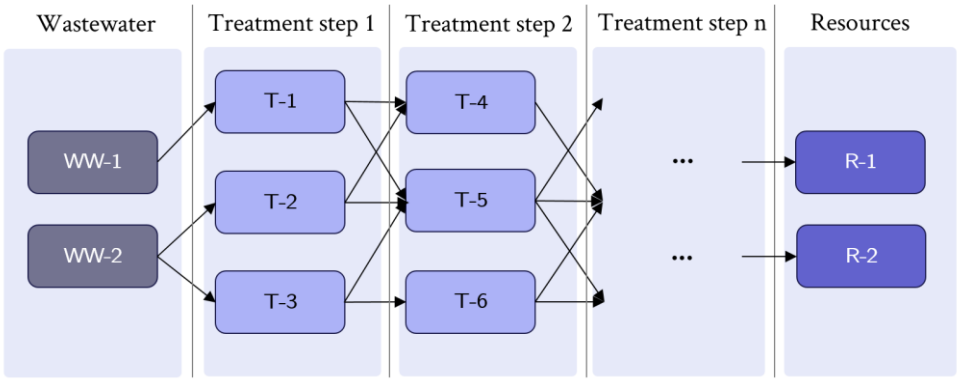

Figure 3.3: A generic superstructure for a wastewater treatment plant design problem.

Once the superstructure is created, the framework generates a large number of alternative processing flow diagrams using both combinatorial synthesis and insight-based approaches [82]. The former uses a full factorial combination of all possible technologies and their interconnections in order to identify all different flowsheets in a given superstructure. This can provide an exhaustive search for the optimal layout problem. However, the framework also employs an insight-based approach in order to eliminate flowsheets that contain an infeasible sequence of unit operations or connections. This helps to allocate computational efforts to potentially promising flowsheets, rather than to unfeasibly connected flowsheets, and thereby avoid obtaining samples of rigorous simulations, which might prove computationally demanding.

One also has to decide on process models of each technology before proceeding with stage 2. In this framework, unlike many other conceptual design frameworks [81–83], we advocate the use of high fidelity simulation models, if available, in the superstructure. Contrary to simple mass balance based input-output black-box models and pre-set conversion factors, such simulation models allow for the formulation of detailed design objectives and reliable estimations of key plant performance indicators, taking into account many more process phenomenal considerations. These models also allow for better management of uncertainties, which might otherwise become harder to quantify using simple mass balance models. Besides technical

performance models, one might also need to decide on costing models of different treatment and separation technologies if a cost-based design objective is to be optimized. The results of rigorous simulation models can also provide reliable estimates of the operational costs of a plant. For instance, in wastewater treatment plants, the anaerobic digestion process can be rigorously simulated to better account for the energy recovery potential via biogas production in the digesters, hence the overall operational costs of the plant [84].

3.3 Stage 2: Design space exploration

In the second stage of the framework, previously generated plant layouts are further explored for their performance indicators under different design and operational conditions. For each treatment and separation technology, there exist several different design degrees of freedom, be it operational parameters such as the hydraulic residence time or the design decision variables such as the volume ratio of the anoxic tanks to aerobic tanks in an activated sludge system. Taking such kind of decisions a priori is a challenging task for engineers, especially when the design space is vast, giving rise to the field of design space exploration (DSE). DSE is an engineering practice of comparative performance evaluation of different design alternatives in order to determine which set of decisions is optimal with respect to the design objective and constraints. Plenty of DSE algorithms exist [85]. The second stage of the framework employs an exhaustive Monte Carlo sampling-based DSE algorithm, which provides a reliable performance comparison using high-fidelity simulation models.

In the Monte Carlo-sampling based DSE algorithm, design and operational decisions of a single flowsheet are gathered into a global parametric space, by combining both individual unit-based and flowsheet-wide decisions. The resulting space is then sampled with a uniform space-filling sampling design, such as Latin hypercube sampling (LHS) [86]. Each sample in this space contains a set of randomly generated values for the design decisions in question. By using the high-fidelity simulation models, design objectives and constraints at each sample are evaluated with (potentially parallelized) Monte Carlo simulations. The results are then analyzed and exposed to effluent quality limits in order to filter out infeasible solutions and quantify the number of feasible design solutions contained in the flowsheet. Figure 3.4 shows an example of these solutions in a three-dimensional design space of the BSM2 layout. The procedure is repeated for every flowsheet in the superstructure, and feasible solutions of each flowsheet are subsequently ranked according to the design objective in order to identify the top k flowsheets whose design

spaces should be more rigorously explored. In doing so, the framework reduces the size of the design problem from a large number of alternative flowsheet candidates to only a select few.

It is important to underline that the DSE algorithm employed in the framework decouples the identification of promising layouts problem from the optimal design under uncertainty problem. In other words, the performance comparisons of different flowsheets are made without the considerations of uncertainties, but rather based on the performance values obtained at the nominal values of the uncertain parameters. This is mainly to ensure efficient allocation of computational resources while serving the sole purpose of this stage,—to identify k promising plant configurations.

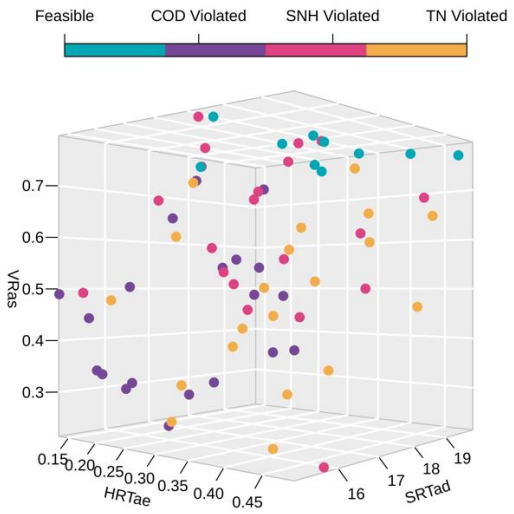

Figure 3.4: Design space exploration for an activated sludge system. Design decisions include the volume ratio of the anoxic tank to the aerobic tank (VR_{as}), the hydraulic residence time in the aerobic tank (HRT_{ae}), sludge retention time in the anaerobic digester (SRT_{ad}). Colors represent violations of different effluent quality requirements, details of which are discussed in Chapter 4.

With the advances in high-performance computing and cloud computing architectures, more and more computing threads, such as CPUs or GPUs, are now made accessible for use by engineers and scientists. This paves the way for parallel computing, i.e., simultaneous execution of computational tasks in order to speed up total execution

times. As Monte Carlo simulations are inherently independent of one another and rather computationally intensive, they are also perfectly suitable for parallel computing. Therefore, design space exploration algorithms of the proposed framework were designed with parallelization in mind, allowing them to be smoothly run on modern computing architectures without the need for further modifications of the code. In doing so, one can explore design spaces of as many flowsheets as there are parallel cores within the same computational time as needed for one flowsheet. Along with previously presented motivations, this also lays the foundation for the simulation-based approach embraced in this conceptual design framework.

3.4 Stage 3: Design optimization under uncertainty

For each of the previously identified promising plant configurations, the third stage of the framework solves a design optimization under uncertainty problem using simulation optimization. The overall workflow at this stage consists of four distinct steps. Firstly, as the results of Monte Carlo simulations are available from the previous stage, the framework proposes to perform a global sensitivity analysis of the design space to better understand the factors affecting the design objectives and constraints. This helps to identify significant design decisions that are worthy of further optimization efforts. Global sensitivity analysis is readily achievable with the use of advanced surrogate models, such as polynomial chaos expansions (PCE), Gaussian process models, artificial neural networks, etc. Chapter 5 will present a methodological contribution regarding how to effectively employ such surrogate models for global sensitivity applications. Figure 3.5 shows an example of global sensitivity analysis results of the operational cost index of several WTTP configurations with respect to their design and operational decisions. Secondly, the design optimization problem is reformulated with the updated key decision variables as well as the design objectives and constraints. Thirdly, the critical sources of uncertainties affecting the design performances are identified. The design optimization problem is extended to a design optimization under uncertainty problem by incorporating these uncertainties into the optimization. This requires quantification of the effects of uncertainties on the key plant performance indices. To that end, the framework employs a Monte Carlo-based uncertainty quantification approach in order to propagate uncertainties in the inputs to the outputs. Lastly, the resulting optimization under uncertainty problem is solved using a simulation-based approach.

The simulation-based approach for optimization has its own internal workflow, which is shown in Figure 3.6. Firstly, surrogate models are constructed from a dataset of initial simulations. An uncertainty analysis is also performed on each design in the initial dataset. The surrogate

models are later used to find the next best candidate design points in an adaptive sampling stage. The designs proposed by this stage are validated by performing rigorous simulations, and the data obtained from these simulations are used to improve surrogate models of the objective, constraints, and their uncertainties. Lastly, after a certain computational budget is exhausted, the best feasible design among the simulated design points is returned as the near-optimal design. Since this design is obtained by taking into account the effects of uncertainties on the optimization objective and constraints, it is also the near-optimal design under uncertainty. A more detailed presentation of this simulation-based optimization approach will be discussed in Chapter 6.

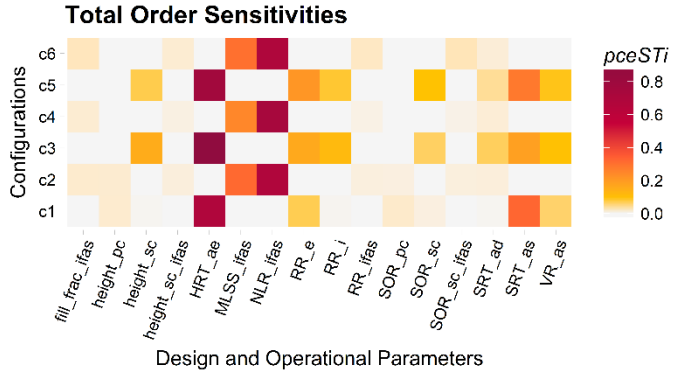

Figure 3.5: Global sensitivity analysis of operation cost index (OCI) of different plant configurations to their design and operational parameters. Polynomial chaos expansion type surrogate models are used to perform the otherwise computationally demanding analysis. A more detailed explanation of this result will be given in Chapter 6.

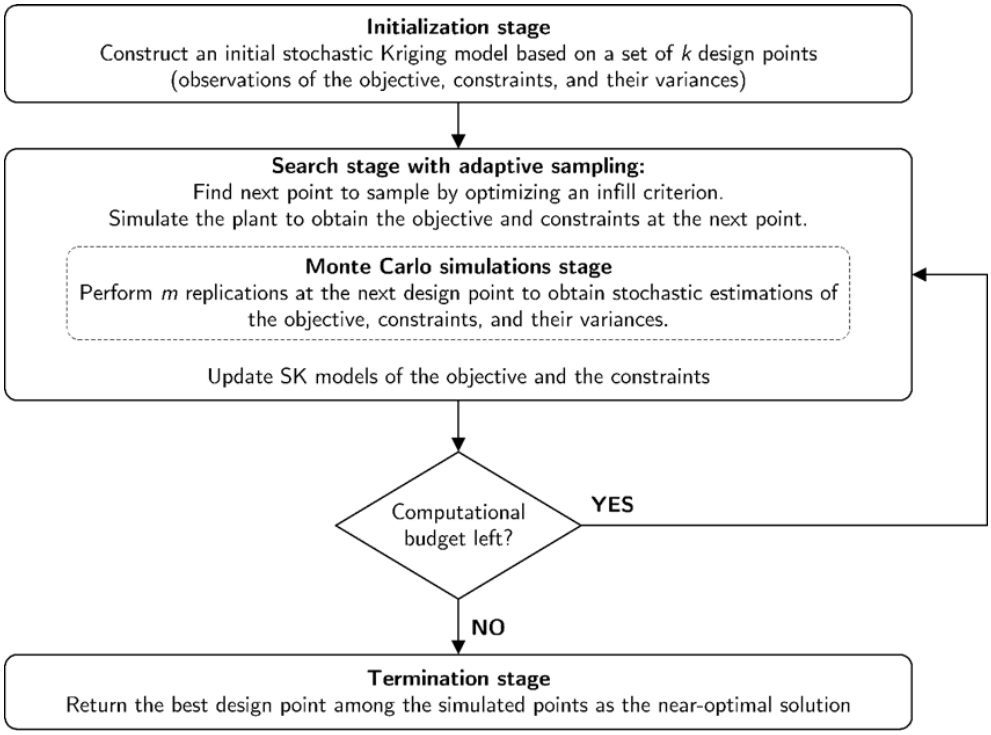

Figure 3.6: Stages of stochastic Kriging-based simulation optimization with embedded Monte Carlo simulations aimed at quantifying the effects of uncertainty on the objective and constraints. A more detailed explanation of this workflow will be given in Chapter 6.

3.5 Workflows and tools

Throughout this Ph.D. project, several computational tools have been developed in order to support the computer-aided design of processes. These tools, as depicted in Figure 3.7, were developed in a generic manner, allowing them to be extendable to new processes. By using these tools, some of the tasks that need to be performed by the process design engineer are automated to provide easy implementation of the overall workflow. These tools include SPDLab (for synthesizing different wastewater treatment plant flowsheets using first principle models), easyGSA (for performing efficient global sensitivity analysis), and MCSKopt (for simulation-based optimization under uncertainty).

SPDLab provides a user-friendly interface to readily plug-in wastewater process models into its model library, which follows a superstructure-like layout to allow for automatic generation of complete WWTP flowsheets using its process synthesis algorithms. Also provided by this tool are algorithms for automatically generating design spaces for each flowsheet, algorithms for calculating key plant performance indicators, algorithms for effectively initializing WWTP flowsheets for simulation and optimization, databases of cost, design, and operational parameters of the process technologies, workflows for advanced visualizations of feasible solutions contained in each flowsheet, to name a few. More details of this tool will be introduced in Chapter 4.

easyGSA is a generic global sensitivity analysis tool, developed for quantifying sensitivities of plant performance indicators to the inputs in the models and the influent compositions. The tool uses either mechanistic models or advanced machine learning models to perform both variance decomposition-based and regression-based global sensitivity analysis. A more detailed presentation of the capabilities of this tool will be given in Chapter 5.

MCSKopt is an in-house developed simulation-based optimization solver, which uses advanced surrogate models to direct the search for an

optimum design with an embedded uncertainty quantification capability. The solver can interface with complex simulation models in order both to validate the designs and also to quantify the effects of critical sources of uncertainties. More details of this solver and its application to wastewater treatment plant design optimization problems will be discussed in Chapter 6.

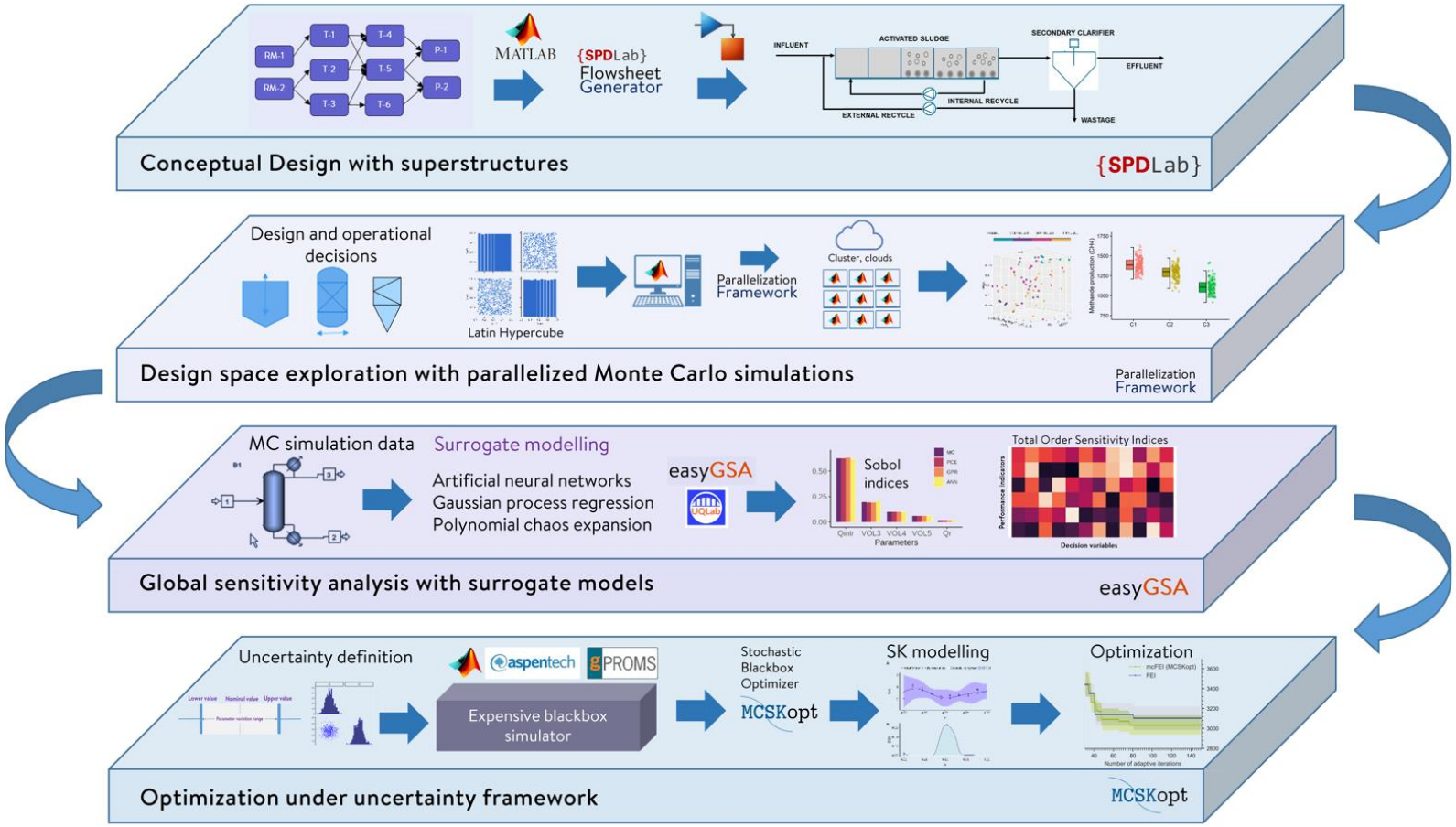

Figure 3.7: An illustration of the algorithmic components and the set of tools employed within the proposed framework, which aims at addressing the process synthesis and design under uncertainty problems in a systematic way using techniques and concepts like superstructure generation, parallel computing, global sensitivity analysis, and stochastic simulation-based optimization [80].

Chapter 4

WWTP synthesis and design space exploration

This chapter illustrates the application of the proposed framework to process synthesis and design problems in wastewater treatment plants. Two case studies are presented. The first one addresses a design problem in a benchmark plant taken from literature, whereas the second one investigates design alternatives of two full-scale municipal WWTPs. Also presented is a decision support tool, SPDLab, which was developed to assist decision-making at the early stage design of WWTPs in collaboration with Chitta Ranjan Behera. Portions of this chapter are based on the following articles.

R. Al, C.R. Behera, K. V. Gernaey, G. Sin, Stochastic simulation-based superstructure optimization framework for process synthesis and design under uncertainty, Comput. Chem. Eng. (Under review)

R. Al, C.R. Behera, K. V. Gernaey, G. Sin, Towards development of a decision support tool for conceptual design of wastewater treatment plants using stochastic simulation optimization, 29th Eur. Symp. Comput. Aided Process Eng., 2019: pp. 325–330. doi:10.1016/B978-0-12-818634-3.50055-2.

In this chapter, two case studies are formulated and presented in order to elucidate the application of the proposed framework for wastewater treatment plant design problems. Firstly, a widely studied benchmark wastewater treatment plant is taken from literature, and plant synthesis and design space exploration are performed. Secondly, design spaces of two full-scale WWTPs are explored using the simulation-based framework.

4.1 Benchmark Plant Design

The plant layout synthesis and design problem is solved for a simple yet illustrative case study using the proposed framework. A widely studied wastewater influent is taken from [11], whose major pollutant compositions are shown in Table 4.1. This influent characterizes a medium-sized domestic wastewater treatment plant load with a capacity of 100,000 persons equivalent and a flowrate of 20,648 m^3d^{-1}.

Table 4.1: Wastewater characterization of the BSM2 influent.

Characterization	Unit	Value
Chemical Oxygen Demand (COD)	g COD/m^3	592
Biochemical Oxygen Demand (BOD)	g COD/m^3	305
Ammonia (NH_4-N)	g N/m^3	24
Total Kjeldahl Nitrogen (TKN)	g N/m^3	56
Flowrate (Q) (m^3/d)	m^3/d	20,648
Capacity	PE	100,000

4.1.1 Design problem definition

The primary design objective in WWTPs is to treat the pollutants in the water, which would otherwise cause severe environmental quality degradation of the limited water resources. However, as discussed earlier, reducing operational costs of this very energy-intensive process is of growing importance to the water utilities. Although the capital costs also matter, acquiring such costs reliably, especially for emerging technologies, is not readily achievable due to the lack of reported cost data [78]. Therefore, with this caveat in mind, a design problem is defined as follows: Given the composition of the BSM2 influent, design a WWTP

that will treat the pollutants in the wastewater to satisfy the regulated organic matter and nitrogen discharge limits with the lowest plant operational cost index possible. Key elements of this plant operational cost index include aeration cost, sludge disposal cost, mixing energy cost, heating cost, costs for chemical and polymer addition as well as costs reductions achieved due to the utilization of biogas produced in the digesters of the plant. The desired effluent quality concentrations are set according to the European Urban Wastewater Treatment Directive (91/271/EEC), which sets the maximum allowable effluent concentrations of the total nitrogen as 15 g N/m^3 and the organic matter (COD) as 100 g COD/m^3. Besides these EU-regulated limits, a maximum limit of 5 g N/m^3 for the effluent ammonia concentration is also introduced for this case study since it is already being adopted in many western countries [87].

4.1.2 WWTP alternative layout synthesis with superstructures

As shown in Figure 4.1, a superstructure is postulated for the BSM2 case study. The layout of the standard Benchmark simulation model 2 (BSM2) plant is populated with two more alternative technologies for the standard primary clarifier, namely rotating belt filter (RBF) and enhanced rotating belt filter (ERBF), as well as with a biofilm-based secondary treatment technology based on fixed-film activated sludge (IFAS) process, as an alternative to the conventional Modified Ludzack-Ettinger (MLE) process. The sludge treatment line is kept the same as the original BSM2 plant, which has a thickener and an anaerobic digester producing biogas and disposable digested sludge, which is also dewatered before landfill. The full factorial combinatorial synthesis approach is applied to consider each different combination of primary and secondary treatment technologies with a de facto sludge treatment line. The resulting superstructure had six alternative plant layout configurations, which are tabulated in Table 4.2. The superstructure in this case study is intentionally kept relatively simple in order to allow for an illustrative

application. More extensive coverage of alternative treatment technologies is done for the second case study and also carried out by Behera et al. [78].

Table 4.2: Different plant configurations contained in the superstructure of the BSM2 case study [80].

Configuration ID	Alternative units connected
1	PC – MLE – TH – AD – DW
2	PC – IFAS – TH – AD – DW
3	RBF – MLE – TH – AD – DW
4	RBF – IFAS – TH – AD – DW
5	ERBF – MLE – TH – AD – DW
6	ERBF – IFAS – TH – AD – DW

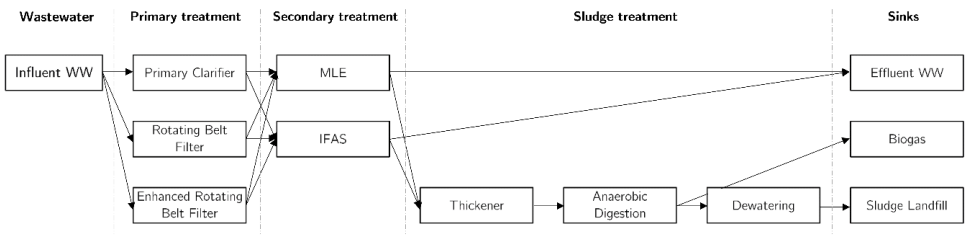

Figure 4.1: The superstructure of different treatment units that are considered for the BSM2 case study. The overall process is divided into primary, secondary, and sludge treatment sub-processes, each of which contains a number of alternative processing technologies [80].

4.1.2.1 Process models

The models of the processes considered in the superstructure are selected as follows. The primary clarifier is modeled based on gravity settling, as presented in Gernaey et al. [11], whereas the rotating belt filters (both RBF and ERBF) are modeled based on cake filtration and sieving as presented in Behera et al. [88]. The MLE process is modeled with the extended activated sludge model (ASMG1), as presented in [89] and further implemented in [88]. The IFAS system is a carrier-based mainstream Anammox process and modeled using a multi-scale modeling approach as developed in [84]. The anaerobic digestion process is modeled with the well-established ADM1 model of Batstone et al. [90].

The sludge thickening and dewatering units are modeled using the same implementations found in the original BSM2 plant [11].

4.1.2.2 Key performance indicators

The operating cost model of the plant consists of a number of cost elements, which are added together to represent a plant-wide operational cost index (OCI). The formula used for calculating this index is taken from [11] with the exclusion of pumping energy cost and reads as follows:

$$\mathrm{OCI} = AE + 3 \cdot SP + 3 \cdot EC + ME - 6 \cdot MP + HE^{net} \qquad (1)$$

where AE denotes aeration energy, SP is sludge production for disposal, EC is external carbon addition, ME is mixing energy, MP represents methane production, and HE^{net} is the net heating energy needed to heat the sludge in the anaerobic digester unit. Containing nonmonetary terms, OCI is often used to compare different design alternatives and control strategies in WWTPs on a dimensionless scale, providing a location-independent metric for comparing operational costs. Following the simulation procedure for the BSM2 plant, as described in [11], these operational costs are calculated separately for each specific sub-unit, and a plant-wide sum averaged per day is returned.

Similarly to the OCI, another widely-used performance metric for comparing the effluent quality in different WWTPs is the effluent quality index (EQI), which is an aggregated weighted sum of all pollution loads that have a major influence on the effluent water quality, such as total suspended solids (TSS), COD, biochemical oxygen demand (BOD), total Kjeldahl nitrogen (S_{TKN}) and the oxidized forms of nitrogen (S_{NOX}), leaving the plant. The formula for calculating this index is taken from [11] and reads as follows:

$$EQI = \frac{1}{1000} . (W_{TSS} . TSS_e + W_{COD} . COD_e + W_{TKNe} . S_{TKNe} + W_{NO} . S_{NO,e} + W_{BOD} . BOD_e) . Q_e \qquad (2)$$

with the weighting factors used as tabulated in Table 4.3.

Table 4.3: Weighting factors used for calculating the effluent quality index (adopted from [11])

Factors	W_{TSS}	W_{COD}	W_{TKN}	W_{NO}	W_{BOD}
Value (g Pollution unit/g)	2	1	30	10	2

4.1.3 Design space exploration

For each different plant configuration, the design and operational decisions that offer extra degrees of freedom to the designer and affect the plant performance indicators (e.g., the hydraulic residence time in aerobic tanks of an MLE system) are gathered into a global (plant-wide) parameter space (hereafter referred to as the design space), which is then explored using the Monte Carlo sampling-based design space exploration algorithm. This step requires a database of design and operational degrees of freedom offered by all of the technologies considered in the superstructure, as well as their typical ranges of variation. Table 4.4 shows the list of all the design and operational parameters of the superstructure, which are acquired from several design guidelines for wastewater treatment plants. The framework automatically generates the design space for any given plant layout, depending on the units contained in that configuration. An exploratory space-filling sampling plan, Latin hypercube sampling (LHS), is employed to uniformly sample the design spaces of each different plant layout. Figure 4.2 shows an example of the design space covered by 500 LHS samples for the plant configuration 1. The diagonal elements in the figure show the uniformity of the sampling, which equally covers regions between the lower and upper bounds of the decisions.

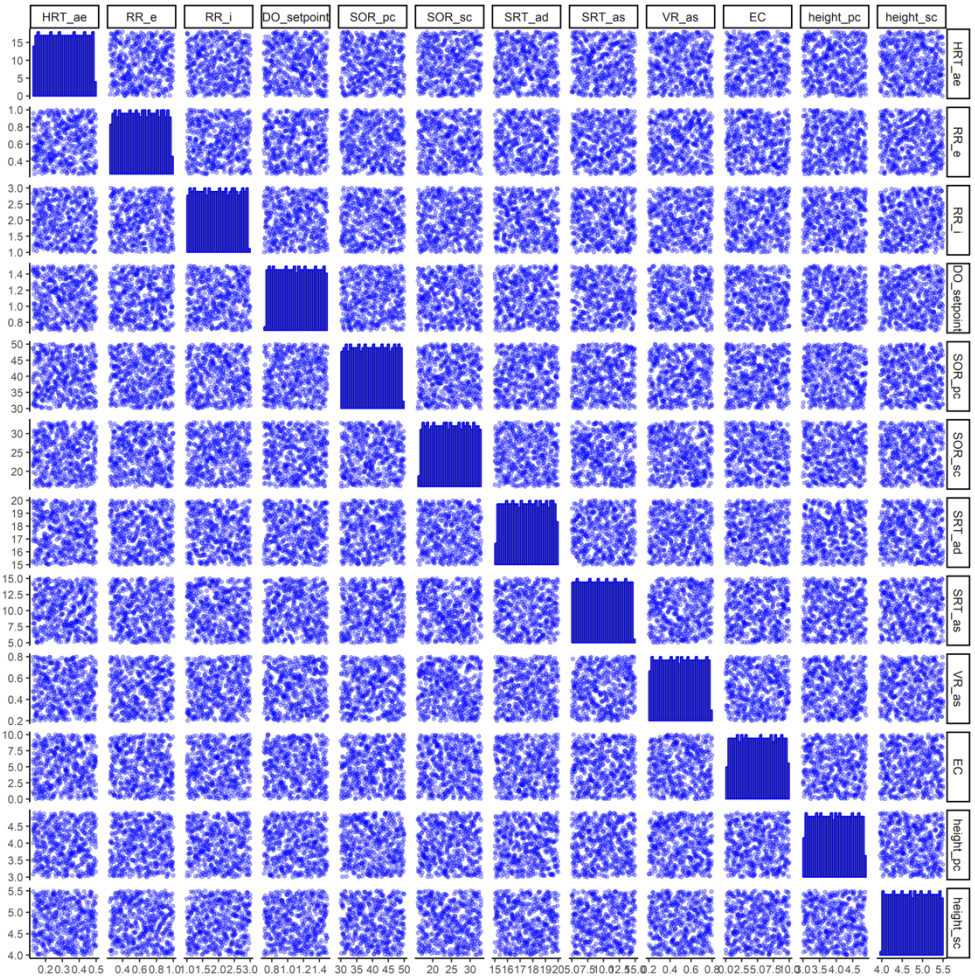

Figure 4.2: Illustration of samples generated by uniform space-filling sampling design (Latin hypercube samples) for the design space of plant configuration 1. The diagonal elements of the matrix show the uniform distribution.

Table 4.4: Design space parameters of all the treatment units with their typical ranges of variation [80].

Parameter	Description	Unit	Min	Max	Ref.
SOR_{PC}	Surface overflow rate of PC	$m^3m^{-2}day^{-1}$	30	50	[91]
h_{PC}	Height of the PC unit	m	3	4.9	[91]
HRT_{ae}	Hydraulic retention time in aerobic tanks	day	0.125	0.5	[92]
SRT_{AS}	Solid retention time in AS	day	5	15	[92]
DO_{AS}	Dissolved oxygen set point in AS	$g.m^{-3}$	0.10	0.50	[93]
VR_{AS}	Volume ratio of anoxic tank to aerobic tank	-	0.2	0.8	[91]
RR_i	Ratio of internal recycle flowrate to influent flowrate	-	1	3	[92]
RR_e	Ratio of external recycle flowrate to influent flowrate	-	0.25	1	[92]
Q_{EC}	External carbon flowrate in AS	$m^3.d^{-1}$	0	10	[80]
SOR_{SC}	Surface overflow rate of SC	$m^3m^{-2}day^{-1}$	16	33	[92]
h_{SC}	Height of the SC unit	m	4	5.5	[92]
SRT_{AD}	Solids retention time in AD	day	15	20	[92]
NLR_{IFAS}	Nitrogen loading rate		0.05	0.5	[94]
$MLSS_{IFAS}$	Mixed liquor suspended solids in IFAS	mg TSS/L	1000	5000	[92]
FF_{IFAS}	Fill fraction in IFAS	%	30	60	[92]
RR_{IFAS}	Ratio of recycle flowrate in IFAS	-	1	3	[92]
DO_{IFAS}	Dissolved oxygen set point in IFAS	$g.m^{-3}$	0.1	0.5	[94]

4.1.4 Identifying promising layouts

For each of the six configurations in the superstructure, 500 Monte Carlo simulations are performed using the samples generated in the design spaces. This results in 3000 plant-wide simulations in total, which were performed using the first principles models developed in MATLAB/Simulink environment. In order to speed-up the computational time, the parallel computing architecture available at the DTU HPC cluster was exploited to divide the total simulation workload among 100 CPU cores. The results from each layout are gathered together to allow for a comparative data visualization.

Six key plant performance indicators (KPI) are considered in the output space. Effluent concentrations of organic matter (COD), total nitrogen (TN), and ammonia (S_{NH}) are calculated to identify designs that satisfy effluent quality limits. Moreover, the amount of methane produced

in the anaerobic digester (CH_4) and the previously discussed plant-wide indices of EQI and OCI are also calculated from the simulation results. Figure 4.3 visualizes the feasible regions of the output space by comparing only the feasible designs found from all of the six configurations according to their plant KPIs. The diagonal elements in the figure show the ranges of variation in the outputs. For example, configurations employing the IFAS technology, C2, C4, and C6, are expectedly found to perform better in terms of their OCI metric. This is mainly because of the technology's lower aeration energy demand in comparison to the conventional activated sludge systems.

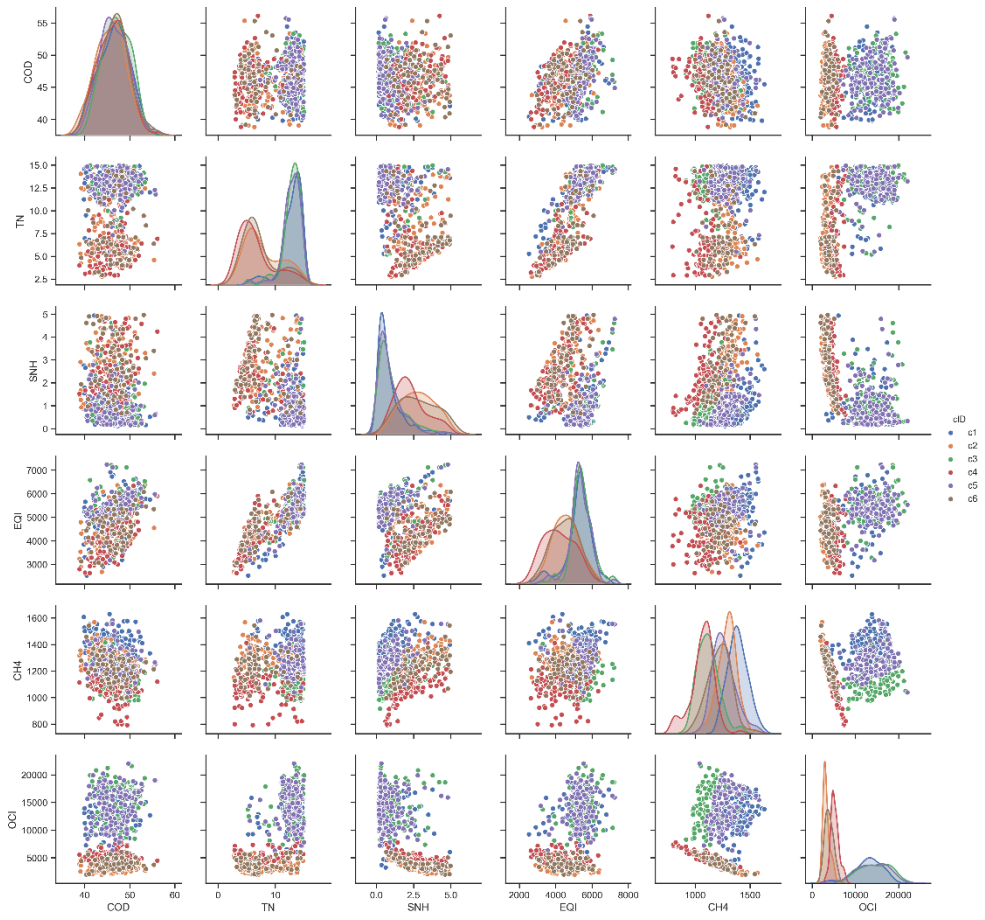

Figure 4.3: Visualization of the feasible regions of the output space of the plant performance indicators generated using the results of Monte Carlo simulations. The colors represent each different plant configuration in the superstructure. Certain configurations show a narrower variation range in the outputs, as shown in the diagonal plots [80].

Similarly, bar plots can also be used to allow for a layout comparison and selection. Figure 4.4 visualizes the results of each layout in box plots and compares them for the operational cost index and the amount of methane produced. As can be seen from the figure, IFAS configurations not only have a narrower range of variation for OCI, but also their designs have an OCI that lies below the OCI of MLE configurations (C1, C3, and C5). Therefore, design engineers interested in minimizing operational costs can proceed with these layouts, whereas for maximizing methane production, the configuration C1 can be investigated further as it outperforms the other layouts.

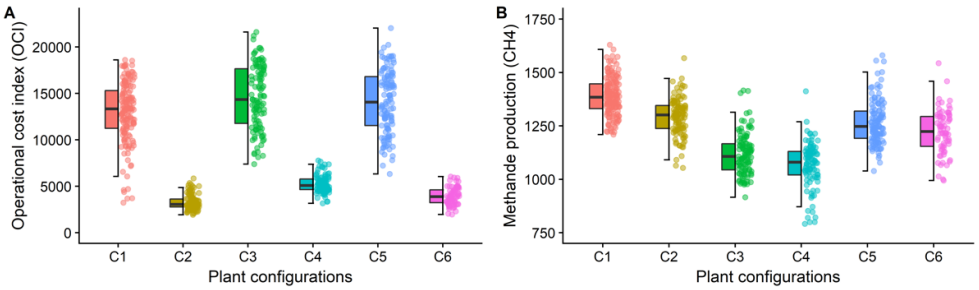

Figure 4.4: Comparison of plant configurations for results of operational cost index (OCI) shown in A and methane production shown in B. Plant configurations C2, C4, and C6 outperform other configurations for OCI whereas C1 delivers comparatively higher methane production [80].

However, box plots can also be normalized between 0 and 1 to allow for comparison among multi-KPIs. For instance, Figure 4.5 shows the comparison of six plant KPIs, all normalized between 0 and 1, in each of the six configurations. This visualization is intended for decision-making based on multi-KPI metrics. Prompt identification of promising layouts can be made possible by comparing layouts based on the design objective KPIs and constraint KPIs. For example, it can be seen from Figure 4.5 that the layout selection does not make a significant difference for the COD in the effluent, whereas it significantly matters for the TN and OCI metrics.

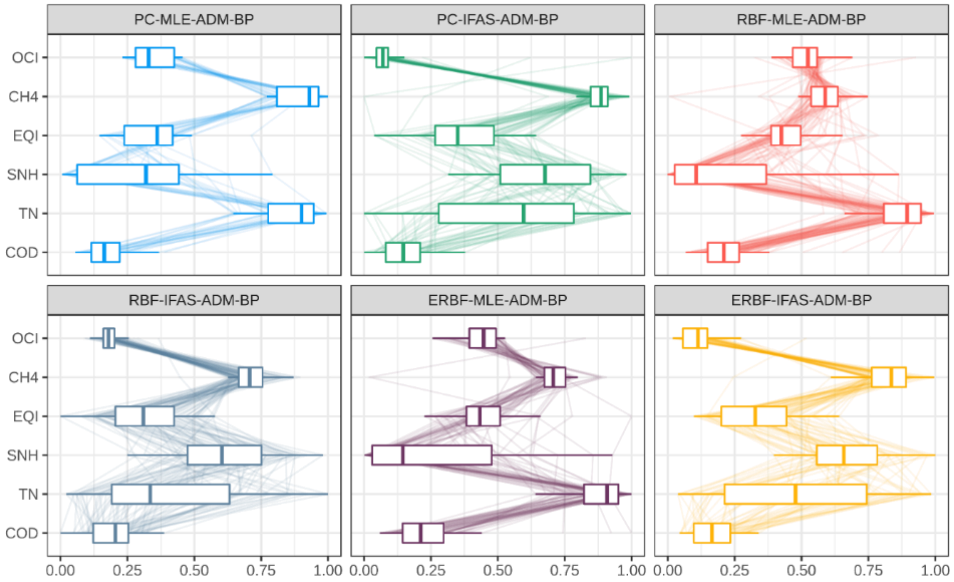

Figure 4.5: Multi-KPI comparison of different plant configurations. Results are normalized to a range of 0 to 1 to allow for visual comparison. Layouts containing the IFAS unit perform considerably better in terms of OCI than those employing the MLE unit [80].

4.2 Retrofitting municipal WWTPs

The second case study is formulated in order to investigate opportunities to improve existing wastewater treatment facilities as well as to further demonstrate the applicability of the proposed framework to industrially relevant problem scales. To this end, the proposed methodology of the framework is applied to explore the design spaces of two full-scale municipal WWTPs, namely Avedøre WWTP and Valladolid WWTP.

Avedøre WWTP is located west of Copenhagen, Denmark, and serves a population of approximately 265,000 person equivalent (PE), treating around 25-30 million m^3 of wastewater annually [81]. Valladolid WWTP, on the other hand, is located in the northwest of Spain and serves a population of 1,067,000 PE approximately [84]. Both treatment plants utilize similar physical, biological and chemical processes to treat the incoming wastewater with the main water line treatment process of activated sludge along with an anaerobic digestion process for the sludge treatment. The final sludge for disposal is incinerated in Avedøre WWTP, whereas in Valladolid WWTP, it is further used for agricultural applications [84]. In both plants, combined heat and power engines (CHP) are employed to convert the produced biogas into heat and electricity. A simplified plant layout representing the main water and sludge line treatment processes is shown in Figure 4.6.

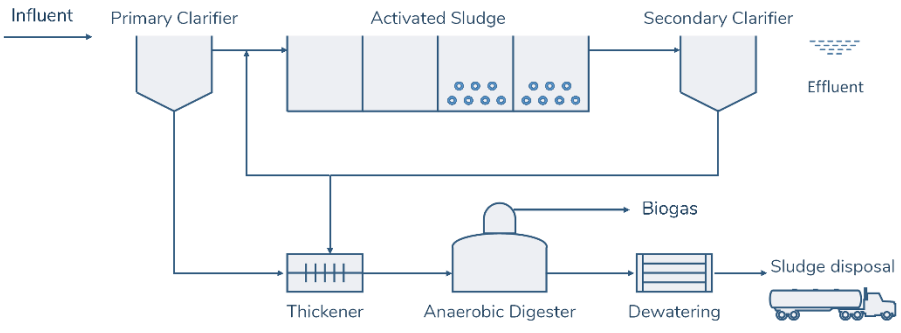

Figure 4.6: Representative flow diagram for the full-scale municipal WWTPs of Avedøre and Valladolid.

The characterization of wastewater for the influents of Avedøre WWTP and Valladolid WWTP are presented in [81] and [84], respectively. The fractionation of the measured yearly average organic matter (COD) and total nitrogen (TN) concentrations into more detailed pollutant components, such as soluble or particulate COD, is needed in order to allow for detailed modelling of the underlying biological processes. The procedure for calculating detailed fractionations can be found elsewhere [84]. Table 4.5 presents only the major pollutant characteristics of the influents of Avedøre and Valladolid WWTPs. Compared to the BSM2 plant, the flowrates of incoming wastewater are significantly larger in both plants, with Valladolid having almost 10 times the flowrate of the BSM2 plant, albeit with significantly lower pollutant concentrations.

Table 4.5: Influent wastewater characterization for Avedøre and Valladolid WWTPs.

Characterization	**Unit**	**Avedøre WWTP** [81]	**Valladolid WWTP** [84]
Chemical Oxygen Demand (COD)	g COD/m^3	476	363
Biochemical Oxygen Demand (BOD)	g COD/m^3	180	203
Ammonia (NH_4-N)	g N/m^3	33	19
Total Kjeldahl Nitrogen (TKN)	g N/m^3	43	31
Flowrate (Q) (m^3/d)	m^3/d	72,037	213,408
Capacity	PE	265,000	1,067,000

4.2.1 Design problem definition

A design problem definition is made as follows: Given the influent compositions of Avedøre and Valladolid WWTPs and a number of alternative primary, secondary and tertiary treatment technologies, identify the most promising plant layouts among all possible WWTP configurations generated using the superstructure approach to process synthesis. To this end, similar to the previous case study, the design objective is set to minimize the operational cost index of the final plant layout, whereas the imposed effluent quality limits are tightened, compared to the previous case study, as follows: maximum COD concentration of 100 mg/L, the total nitrogen concentration of 10 mg/L,

and ammonia (S_{NH}) concentration of 1 mg/L. In this case study, the design objective is intentionally kept relatively simple as the main emphasis is placed on the applicability of the developed methodology to the challenging design problems. Using the collaboratively developed decision support tool (SPDLab), more comprehensive design objectives involving considerations regarding capital costs of treatment technologies as well as greenhouse gas emissions were studied by Behera et al. [78].

4.2.2 WWTP alternative layout synthesis with superstructures

As shown in Figure 4.7, an extensive superstructure is developed for the second case study by extending the layout of the BSM2 plant. The overall process is divided into four main sub-processes, each of which includes a number of alternatives. Firstly, the primary treatment of the incoming wastewater is achieved with one of the four alternative technologies, namely primary clarifier (PC), rotating belt filter (RBF), enhanced rotating belt filter (ERBF), and high rate activated sludge (HRAS) process. Secondly, three main water line treatment options are considered, MLE, IFAS, and a granular Anammox technology (ELANM). Sludge treatment using anaerobic digesters is kept the same as before. An option is considered for the treatment of sludge reject water using granular Anammox technology (ELANS) for the side stream. Using the combinatorial synthesis and expert knowledge-based synthesis approach, 24 different potential plant layout configurations, as tabulated in Table 4.6, are generated from the resulting superstructure. From an energy recovery point of view, all candidate layouts are enforced to have anaerobic digesters since the design objective is to minimize the operational costs.

In their extensive study, Behera et al. [78] presented the model library development, which included the models of RBF, HRAS, IFAS, ELAN processes, as well as the validation of these models against the influent data of Avedøre and Valladolid WWTPs. These models were employed

in this case study in order to investigate the operational cost performance of alternative plant layouts.

Table 4.6: Alternative plant layout configurations contained in the superstructure developed for the second case study [80].

Configuration ID	Alternative units connected within the configuration
1 (base)	PC – MLE – TH – AD – DW – BP
2	PC – MLE – TH – AD – DW – ELANS
3	PC – IFAS – TH – AD – DW – BP
4	PC – IFAS – TH – AD – DW – ELANS
5	PC – ELANM – TH – AD – DW – BP
6	PC – ELANM – TH – AD – DW – ELANS
7	RBF – MLE – TH – AD – DW – BP
8	RBF – MLE – TH – AD – DW – ELANS
9	RBF – IFAS – TH – AD – DW – BP
10	RBF – IFAS – TH – AD – DW – ELANS
11	RBF – ELANM – TH – AD – DW – BP
12	RBF – ELANM – TH – AD – DW – ELANS
13	ERBF – MLE – TH – AD – DW – BP
14	ERBF – MLE – TH – AD – DW – ELANS
15	ERBF – IFAS – TH – AD – DW – BP
16	ERBF – IFAS – TH – AD – DW – ELANS
17	ERBF – ELANM – TH – AD – DW – BP
18	ERBF – ELANM – TH – AD – DW – ELANS
19	HRAS – MLE – TH – AD – DW – BP
20	HRAS – MLE – TH – AD – DW – ELANS
21	HRAS – IFAS – TH – AD – DW – BP
22	HRAS – IFAS – TH – AD – DW – ELANS
23	HRAS – ELANM – TH – AD – DW – BP
24	HRAS – ELANM – TH – AD – DW – ELANS

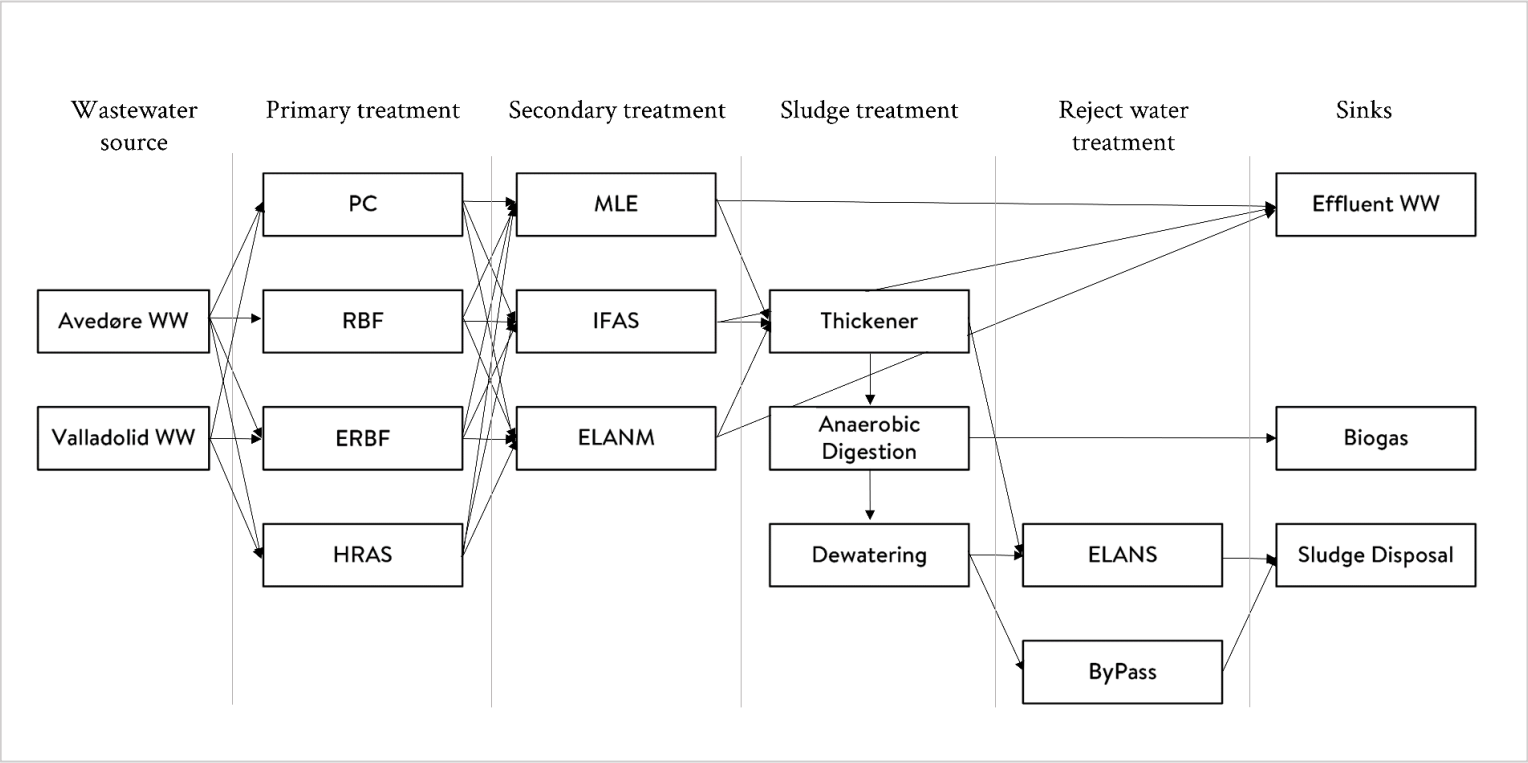

Figure 4.7: Superstructure developed for the retrofitting case study of WWTPs.

4.2.3 Design space exploration

Similarly to the BSM2 case study, the design and operational degrees of freedom in each of the candidate plant layouts are explored with extensive Monte Carlo simulations. Table 4.7 tabulates the additional design and operational degrees of freedom that are introduced due to the inclusion of HRAS, ELANM, and ELANS units in the superstructure of the second case study. The design spaces of the other units are kept the same as the BSM2 case study, which was tabulated in Table 4.4. For each layout, uniformly space-filling 500 LHS samples representing different sets of design and operational decisions were randomly drawn and simulated using the parallel computing architecture. For each candidate layout, a step-by-step initialization strategy, which simulates units one at a time in order to provide reliable initial conditions for the plant-wide simulation, is also developed. Successful full-scale plant-wide simulations of alternative plant layouts were only made possible by the use of this sequential initialization strategy. In total, 24,000 Monte Carlo simulations are performed for both the Avedøre and Valladolid WWTP influents. The results obtained from these simulations are gathered into a large dataset, which is then exposed to the effluent quality limits of the case study.

Table 4.7: Additional design and operational degrees of freedom explored in the superstructure due to the addition of HRAS, ELANM, and ELANS units.

Parameter	Description	Unit	Min	Max	Ref.
HRT_{HRAS}	Hydraulic retention time in HRAS	day	0.01	0.04	[92]
SRT_{HRAS}	Solid retention time in HRAS	day	0.10	1	[92]
DO_{HRAS}	Dissolved oxygen set point in HRAS	$g.m^{-3}$	0.10	0.50	[93]
RR_{HRAS}	Recycle rate in HRAS	-	0.25	1	[91]
NLR_{ELANM}	Nitrogen loading rate in ELANM	kg N/m^3d	0.05	0.30	[92]
SRT_{ELANM}	Solid retention time in ELANM	day	20	80	[94]
DO_{ELANM}	Dissolved oxygen set point in ELANM	g/m^3	0.10	0.50	[92]
NLR_{ELANS}	Nitrogen loading rate in ELANS	kg N/m^3d	0.05	0.65	[92]
SRT_{ELANS}	Solid retention time in ELANS	day	20	80	[92]
DO_{ELANS}	Dissolved oxygen set point in ELANS	g/m^3	0.10	0.50	[94]

Figure 4.8 and Figure 4.9 show the results of the design space exploration performed in 24 layouts using the Avedøre and the Valladolid WWTP influents, respectively. As expected, in most of the layouts, the majority of the designs violate the stringent effluent ammonia concentration limit of 1 mg/L, whereas the COD limit is rarely violated, especially for the Valladolid influent, which is due to the more diluted influent COD concentration.

The box plots in Figure 4.8 and Figure 4.9 represent the ranges of variation in the plant KPIs using the interquartile ranges. The middle 50 % of the data, also referred to as interquartile range (IQR), are contained within the boxes whose boundaries represent the first (Q1) and the third (Q3) quartiles. The lower and the upper whiskers represent Q1 - 1.5*IQR and Q3 + 1.5*IQR, respectively. The width of the whiskers represents the skewness of the data. Points beyond the whiskers are often referred to as outliers, i.e., designs with significantly different values than the rest of the dataset. As can be seen from Figure 4.8 and Figure 4.9, most of these points are falling into the infeasible regions of the output space, shaded in the red area, mostly at the higher ends of the plant KPIs.

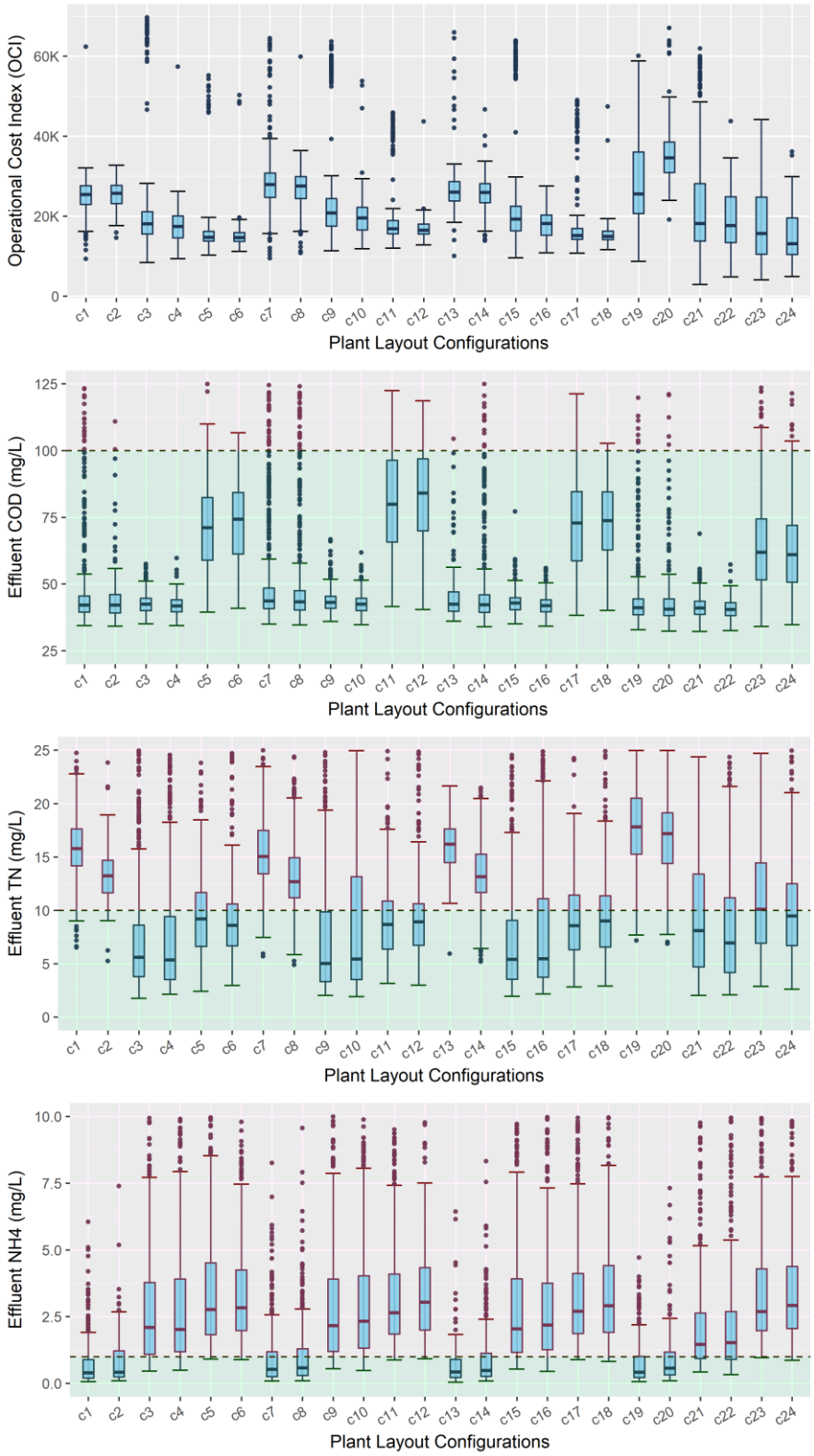

Figure 4.8: Design space exploration results of all the configurations simulated with the Avedøre WWTP influent. Most solutions violate the stringent maximum effluent ammonia concentration limit of 1 mg/L.

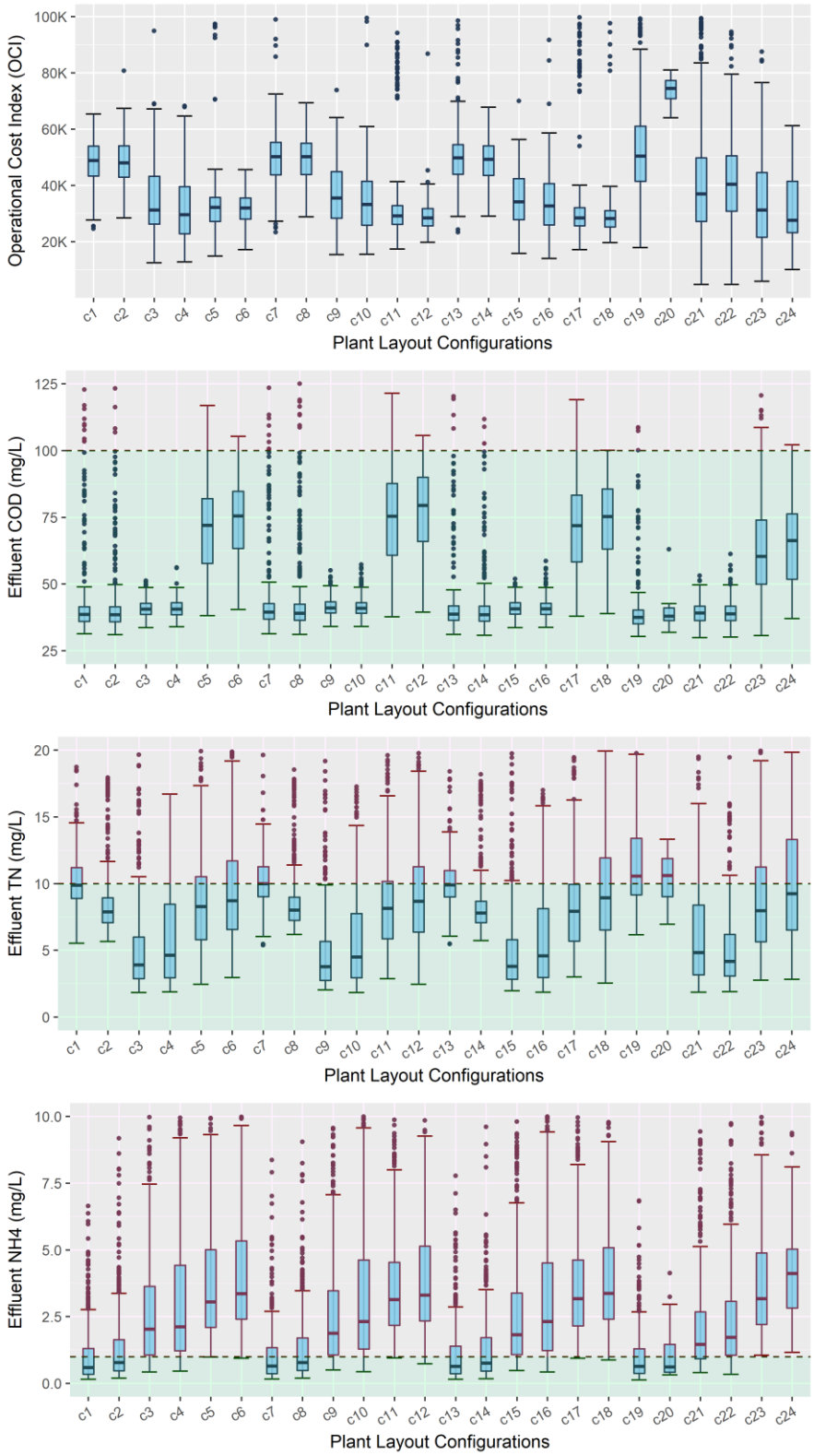

Figure 4.9: Design space exploration results of all the configurations simulated with the Valladolid WWTP influent. Most solutions violate the stringent maximum effluent ammonia concentration limit of 1 mg/L.

4.2.4 Identifying promising layouts

In order to identify the most promising configurations, a ranking-based approach is employed. Previously identified feasible designs in each layout are gathered together, and their ranges of variation in terms of the design objective (OCI) are plotted in opened boxplots of Figure 4.10. As can be seen, the layouts differ significantly in terms of the number of feasible designs they contain. In general, satisfying the stringent quality limits for the Avedøre influent is harder than for the Valladolid influent as significantly fewer feasible designs were found in layouts simulated with the Avedøre influent, compared to those simulated with the Valladolid influent. This can be attributed to the higher influent pollutant loads entering the Avedøre WWTP. The most promising layouts for satisfying the quality limits with minimal operational costs for the influent of Avedøre are configurations 21 and 22, both of which employ the HRAS and IFAS processes as the primary and secondary treatment technology, respectively. Configurations employing PC and IFAS combination (configuration 3 and 4) as well as ERBF and IFAS combination (configuration 15 and 16) were also found to hold the highest number of designs satisfying the quality limits.

For the Valladolid influent, similar to the Avedøre results, the configurations employing the IFAS process (configurations 21 and 22) are found to be the most effective at achieving the desired pollutant removal rates. However, unlike the Avedøre results, the configurations employing the MLE process, such as configurations 1, 2, 19, were also found to contain the most number of feasible designs. Moreover, no feasible designs were found in configurations 23 and 24, whereas configurations 5, 6, 11, 12, 17, 18 contain less than 5 feasible designs out of 500 Monte Carlo simulations. As the common unit in these configurations is the ELANM, the results suggest that highly stringent ammonia limits might be violated in WWTPs employing the ELANM process.

Concerning the design objective of minimum OCI, the top 5 layouts for achieving desired water quality in both WWTPs are ranked in Table 4.8. The best design performance is found in configuration 21 in both cases, with most of the top configurations employing HRAS primary treatment, which serves a higher carbon recovery purpose by diverting more of the incoming COD to the anaerobic digester unit, thereby lowering the OCI by increased methane production. Anammox-based secondary treatment processes like IFAS and ELANM are also found in top layouts as these processes are known to demand significantly less aeration energy. However, most designs in ELANM configurations are suffering from the very low effluent ammonia limit even though they contain designs with competitive performances. After exploring their design spaces and feasibility, it can be concluded that the top layouts identified in this case study are worthy of further optimization efforts, for which Chapter 6 will present a simulation-based methodology.

Table 4.8: Top 5 layouts for treating the influents of Avedøre and Valladolid WWTPs.

Rank	cID	Units contained in the configuration	# of feasible designs	Best design objective (OCI)
		Avedøre WWTP		
1	c21	HRAS – IFAS – TH – AD – DW – BP	66	10929
2	c22	HRAS – IFAS – TH – AD – DW – ELANS	76	11497
3	c24	HRAS – ELANM – TH – AD – DW – ELANS	2	14879
4	c19	HRAS – MLE – TH – AD – DW – BP	3	17634
5	c5	PC – ELANM – TH – AD – DW – BP	3	17795
		Valladolid WWTP		
1	c21	HRAS – IFAS – TH – AD – DW – BP	110	25471
2	c19	HRAS – MLE – TH – AD – DW – BP	88	25914
3	c22	HRAS – IFAS – TH – AD – DW – ELANS	90	28326
4	c1	PC – MLE – TH – AD – DW – BP	180	33177
5	c18	ERBF – ELANM – TH – AD – DW – ELANS	2	35201

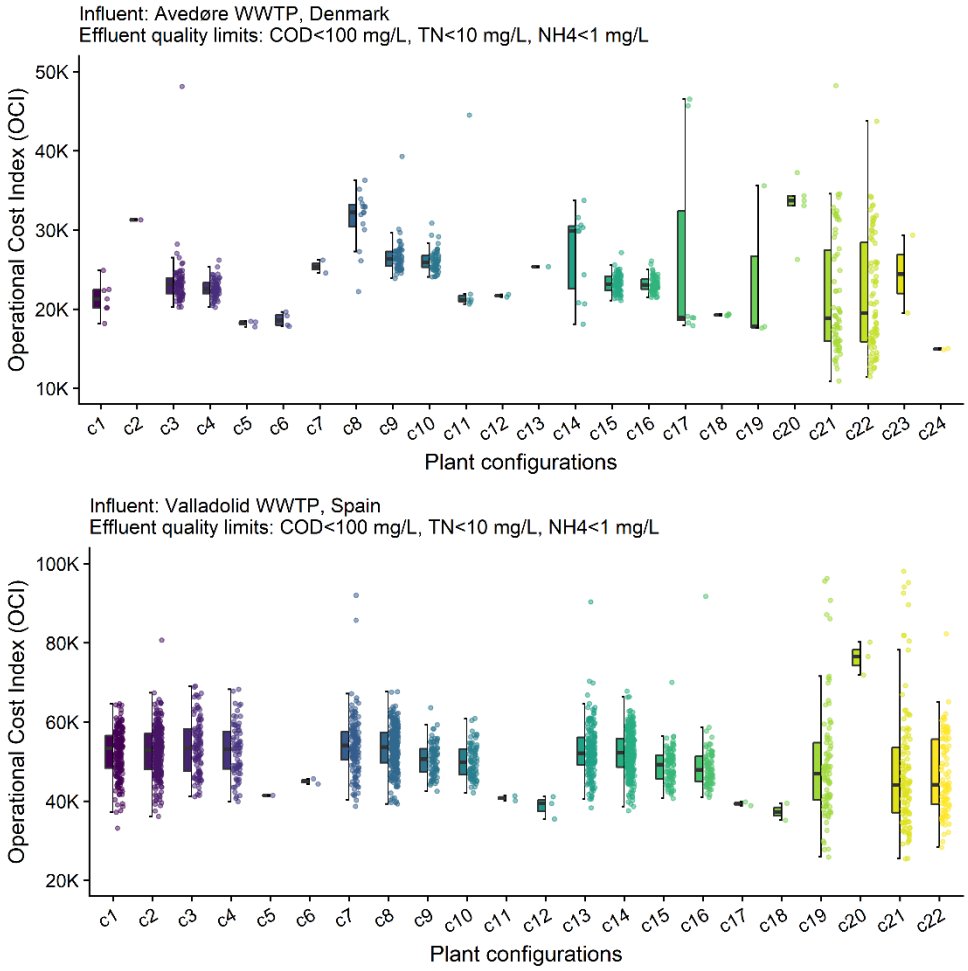

Figure 4.10: Feasible designs found from the Monte Carlo simulation-based design space exploration performed with Avedøre (top) and Valladolid (bottom) WWTP influents. The imposed effluent quality limits were COD<100 mg/L, TN<10 mg/L, and NH4<1 mg/L.

4.3 Tool implementation: SPDLab

The methodology presented in this chapter was implemented in a decision support tool for technology selection during the early stage of WWTP design. The tool, named SPDLab, encompasses a model library of first principles wastewater treatment process models, including both conventional and emerging treatment technologies. As stated earlier, the development and the validation of the emerging models in the model library were conducted by Behera et al. [84], whereas the author of this thesis made methodological contributions in the form of algorithms designed for effectively addressing the resulting complexities.

With modularity in mind, SPDLab was designed in an attempt to automate most of the tasks necessary to complete the process design activity using complex process models. Therefore, algorithms were scripted using the functional programming paradigm, which does not require further code modification once the data is changed. Table 4.9 presents a non-exhaustive list of these algorithms, which are maintained in a publicly available GitHub repository [95]. Most of these algorithms are developed in the MATLAB programming language, along with interfaces to Simulink, which provides a graphical editor, customizable block libraries, and numerical solvers for modeling and simulating complex engineering systems.

Table 4.9: A list of algorithms developed for the SPDLab tool.

#	Algorithm name	What the algorithm does
1	superstructure	Creates a superstructure from the models in the model library.
2	synthesize	Generates alternative, executable plant layout configurations from a given superstructure.
3	designSpace	Creates a design space of design and operational decisions based on technological units contained in a given configuration.
4	initialize	Simulates a given flowsheet using the sequential initialization strategy to store initialization data.
5	run	Simulates a complete flowsheet to its steady state.
6	getKPIs	Calculates key plant performance indicators from simulation results.
7	parallelize	Parallelizes the total number of simulations into separate job files.
8	gather	Gathers output data from parallelized simulations into one dataset.
9	pplot	Creates a parallel coordinates plot from Monte Carlo simulations.

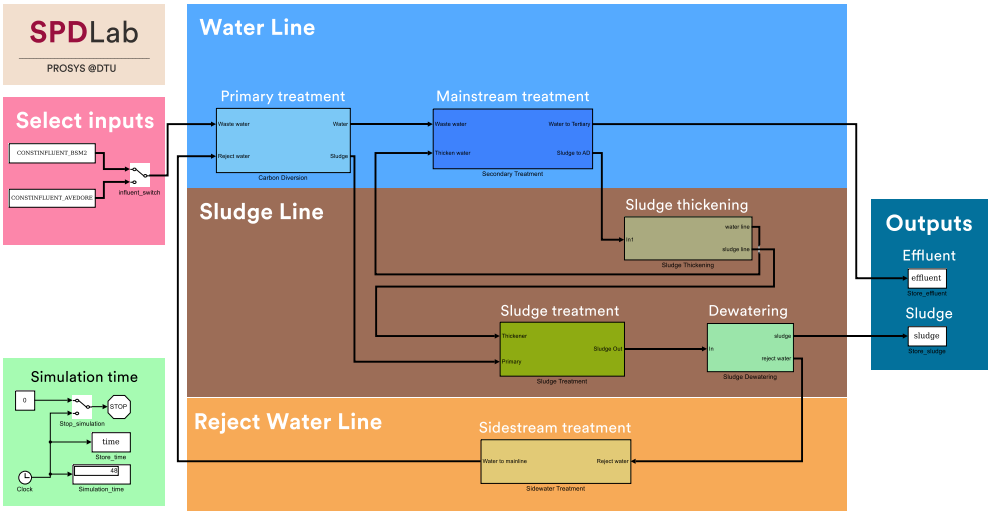

Figure 4.11: The main user interface of the decision support tool, SPDLab [16].

Figure 4.11 shows the Simulink-based main user interface of the SPDLab tool. The tool also comes with a number of advanced data visualization workflows written in both R and JavaScript programming languages. For instance, the results of the design space exploration step are visualized in an interactive parallel coordinates plot, as shown in Figure 4.12, where the users can define their own effluent quality limits to explore feasible designs from different layouts.

The user inputs to the tool are intended from the graphical user interface, the model library, and the inventory database. For example, the model library can readily be extended with the Simulink building blocks, each of which can seamlessly call models written in MATLAB, Simulink, and C environments without the need for an external interface. However, to be able to interconnect models written in other environments, such as gPROMS models, external interfaces, such as gO:MATLAB of gPROMS, are needed. Although the tool encourages the integration of first principles models into its library, care must be exercised in interconnecting models with a different number of state variables. Therefore, the current version of the tool is only intended for users familiar with mechanistic process modelling practices.

As the rankings of the top layouts and the designs are made based on the design objectives and constraints, it might be the case that the obtained top layout results are highly sensitive to the inventory data that is used to calculate the key plant performance indicators. Similarly, if the ranges of the design and operational parameters are changed, the number of feasible designs found in a layout can also change, which might also consequently affect the final rankings of the top layouts. Therefore, it is intended that SPDLab could be used in an iterative fashion with ongoing feedback and user updates to the database until a satisfactory level of confidence in the obtained results is achieved.

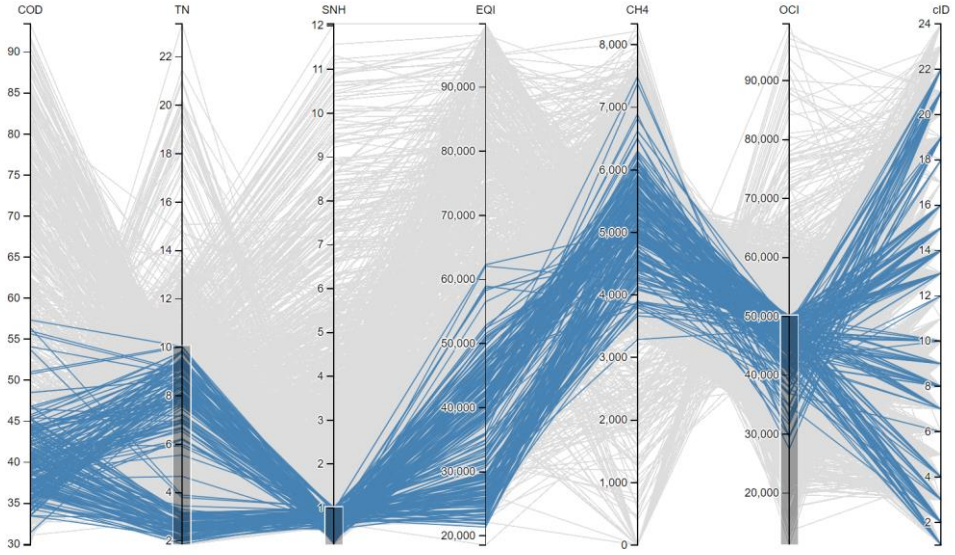

Figure 4.12: Interactive visualization of Monte Carlo simulation results allows for setting different effluent quality limits to explore feasible designs found in different plant layout configurations.

4.4 Summary and conclusions

This chapter has illustrated the application of the proposed framework to process synthesis and design space exploration problems in two case studies, involving both benchmark and full-scale municipal WWTPs. The design objective was set to design a WWTP layout with minimal operational cost while also satisfying the regulated effluent quality metrics with various levels of stringency. The following main conclusions are drawn.

- The proposed framework can effectively synthesize alternative WWTP configurations and explore their design and operational degrees of freedom using the first principles process models, and identify the most promising plant layouts for a given design objective and a set of constraints.
- The simulation-based approach can deliver a reliable estimation of design performances by incorporating the rigorous plant-wide simulations into the early-stage design.
- Anammox-based secondary treatment process IFAS presents desirable operational cost savings in comparison to the conventional MLE process with design performances satisfying even the most stringent effluent quality limits.
- WWTP configurations employing the ELANM process are suffering from the violation of highly stringent ammonia limits with design spaces containing very few feasible designs.
- Advanced data visualizations can allow for informed decision making for the identification of the promising WWTP layout configurations as well as the trade-offs between design objectives and constraints.

Overall, the proposed Monte Carlo and simulation-based framework allows for model-based design space exploration and can be used to simultaneously adjust the critical operating variables as well as the plant layout configuration to improve the WWTP's performance. Further

optimization efforts can be invested in those layouts identified as the most promising, including considerations for managing uncertainty.

Chapter 5

Plant-wide global sensitivity analysis

This chapter investigates the use of complex plant-wide models as well as advanced data-driven models for understanding the sensitivity of key performance indicators of wastewater treatment plants. The chapter also introduces a generic global sensitivity analysis tool, easyGSA, which provides workflows for performing mechanistic or machine learning model-based global sensitivity analysis. This chapter is based on the following published article:

R. Al, C.R. Behera, A. Zubov, K. V. Gernaey, G. Sin, Meta-modeling based efficient global sensitivity analysis for wastewater treatment plants – An application to the BSM2 model, Comput. Chem. Eng. 127 (2019) 233–246. doi:10.1016/j.compchemeng.2019.05.015.

5.1 Introduction

Computational models describing technical and natural systems have become increasingly complex, requiring the incorporation of many model parameters, which add up to model output uncertainty. Sensitivity analysis is broadly recognized as a good modeling practice to better understand model behavior, providing valuable insights into how much of the model output uncertainty can be attributed to the uncertainty in a specific model parameter. While different methods exist for sensitivity analysis, global sensitivity analysis (GSA) constitutes a class of methods which look at the model behavior in a global fashion, allowing more than one factor (i.e., an investigated model parameter) to vary at the same time as opposed to one-factor-at-a-time (OAT) methods. Therefore, unlike OAT methods, GSA methods can account for and quantify the effects of interactions among the investigated input factors on the model outputs. Although the use of OAT methods in sensitivity analysis is fiercely criticized [8], OAT is still the most widely used technique in sensitivity analysis [96]. Among two distinct classes of GSA methods are the variance decomposition-based methods, such as the Sobol' indices method [97], and the regression-based methods, such as standardized regression coefficients (SRC) [98]. Sobol sensitivity indices are considered as the reference sensitivity measures among practitioners, setting the benchmark for all other sensitivity analysis methods. These indices are traditionally evaluated by employing a Monte Carlo simulations-based approach on randomly generated input matrices. Although this approach provides robust results, one major drawback is its high computational demand, requiring a large number of model evaluations, typically in the order of 2,000 to 10,000 simulations per input parameter to ensure convergence of the Sobol indices with a satisfactory precision level [99]. Therefore, to reduce the computational cost associated with the Sobol method, alternative ways of calculating Sobol indices, such as meta-modeling-based approaches, have gained significant interest in recent years. Among the most widely employed types of

surrogate models for global sensitivity analysis are polynomial chaos expansions [100], Gaussian processes regression [101], and artificial neural networks [102].

In the field of wastewater treatment plant modeling, the importance of performing global sensitivity analysis has been widely recognized with applications on both individual process models and plant-wide modelling frameworks. In some of the earliest applications of the Monte Carlo method, Flores-Alsina et al. [103] investigated the effect of activated sludge input uncertainties on the control strategies of the BSM2 plant, whereas Sin et al. [12] identified and analyzed different sources of uncertainties in the BSM1 plant. Using the SRC method, Sin et al. [104] also conducted a global sensitivity analysis on the BSM1 plant to identify the most critical parameters influencing the plant performance criteria. Flores-Alsina et al. [105] applied the SRC method to identify and rank the most important design variables of the activated sludge plants. Rojas and Zhelev [106] used GSA to identify the operating conditions with the strongest impact on the energy requirements of a thermophilic aerobic digestion system, on which they perform energy efficiency optimization using the optimization variables identified by the GSA. Cosenza et al. [107] comparatively studied SRC, Morris screening, and Extended-FAST methods to a membrane bioreactor (MBR) model and reported that Morris screening provides inconsistent results in comparison to the other two methods. The same authors also studied variance decomposition-based methods in WWTP modeling and concluded that for model outputs like ammonia, nitrate, and phosphorus, interactions among input factors provided significantly different sensitivity results compared to regression-based methods, such as SRC, which do not consider interactions [108]. Ramin et al. [109] studied first order and second order secondary settling tank models within the BSM2 plant and conducted a global sensitivity analysis using both the SRC method and the Morris screening method. Dragan et al. [110] used the Morris screening method to identify significant design degrees of freedom of the BSM2 plant, which

later are used to perform design space optimization. Mannina et al. [111] used the SRC method, albeit with a poor linearization of the MC simulation results (R^2<0.7), to identify the calibration parameters of an integrated activated sludge and membrane bioreactor model. Ochoa et al. [112] studied process synthesis of wastewater treatment network design and applied GSA using the Sobol method on the kinetic dynamic model parameters of the optimal network configuration to identify opportunities for further design optimization. More recently, Fortela et al. [113] utilized a Morris screening-based GSA method along with principal component analysis to identify sensitive biochemical mechanism parameters of anaerobic digestion kinetic models.

Although the SRC method is broadly practiced by the researchers, benchmarking of the reported sensitivity measures with more reliable variance decomposition-based methods, such as the Sobol indices method, is needed to strengthen the conclusions drawn from the global sensitivity studies, especially in the context of full-scale WWTP modeling case studies. However, to the best of the author's knowledge, the Sobol indices method has thus far not seen an application in plant-wide models of WWTPs, such as the BSM2 plant model. This is mainly due to the high computational cost associated with the calculation of these indices. Since the Sobol indices method delivers the researchers with more reliable sensitivity measures compared to the popular SRC method, there is a need for a systematic framework that will extend the knowledge on how to build computationally efficient meta-models to be used as surrogates for global sensitivity analysis purposes. The core purpose of this chapter is, therefore, to establish a systematic methodology to perform efficient Sobol global sensitivity analysis of complex biological systems, such as WWTPs, with the help of advanced data-driven surrogate models, such as polynomial chaos expansions, Gaussian processes regression, and artificial neural networks. For that purpose, four distinct scenarios involving various sources of epistemic uncertainties in WWTPs are framed as follows: (1) uncertainty in the influent fractionation, (2)

uncertainty in stoichiometric and kinetic model parameters, (3) uncertainty about the hydraulics and design-related parameters of the plant, and (4) simultaneous realization of uncertainties in all of previous the scenarios (1), (2), and (3). The computational efficiency and the accuracy of the different approaches for global sensitivity analysis are compared and discussed.

This chapter is organized as follows. Firstly, the theory behind the traditional global sensitivity analysis methods, as well as the more recent meta-modeling based methods, is presented. Secondly, a framework is proposed for the construction of efficient surrogate models with the different steps being detailed and the implemented techniques briefly described. Thirdly, four practical scenarios, in which a global sensitivity analysis is performed to address the questions design engineers encounter, are detailed. Then, the obtained results are presented with discussions regarding the validation of the PCE, GPR and ANN models, comparison of the SRC and the Sobol indices methods, as well as the effectiveness of the meta-modeling-based approach for GSA. Lastly, an implementation of the proposed framework in a generic global sensitivity analysis tool, easyGSA, is also presented. Finally, the chapter is concluded with a summary of the main contributions and conclusions.

5.2 Methods

5.2.1 Global sensitivity analysis methodologies

Global sensitivity analysis is a well-established area of study, providing methods quantifying how much of the uncertainty in the output of interest is due to the uncertainty in the input parameters of an underlying mathematical model. There exists a wide spectrum of techniques and sensitivity measures available in the literature, e.g., the Morris screening method [114], linear regression-based methods, variance-based methods [98], to name a few. For a recent review of these methods, readers are referred to the studies of [115,116]. Among the most widely employed methods in GSA are standardized regression coefficients (SRCs) and Sobol sensitivity indices. The SRC method presents a measure of sensitivity using a linear regression approximation of the model response, whereas the Sobol method provides sensitivity indices by decomposing the total model output variance into each single input variance and the combinations thereof. The latter proves especially useful for the case of highly nonlinear computational models. Both methods rely on Monte Carlo simulations, a technique which is discussed next before explaining the two GSA methods.

5.2.2 Monte Carlo simulations approach

Monte Carlo simulation (MCS) is a universal mathematical technique relying on repeated random sampling to numerically attain results that are otherwise intractable or unavailable by analytical methods. As a technique, it has found widespread applications in fields as disparate as finance, energy, uncertainty, and sensitivity analyses. Calculation of the Sobol sensitivity indices using the MCS technique is a fairly well-established procedure which can be divided into 4 steps: (1) specifying the uncertainty ranges for the uncertain input parameters; (2) sampling of the parameters within their ranges using a sampling scheme such as Sobol sampling, Latin hypercube sampling, etc.; (3) propagation of uncertainty

by repeated evaluations of the model for each combination of sampled parameters in the input space; (4) post-processing of the obtained dataset of quantities of interests. Irrespective of the complexity associated with the model evaluation, the MCS procedure remains the same, and as the repeated runs are independent of one another, it intrinsically suitable for parallelization. However, the technique comes with an obvious downside of low computational efficiency, since the size of the sampling matrices needed to converge to accurate results for sensitivity indices is no less than 10^3, typically in the order of 10^4 [99]. In practice, this often translates to an unaffordable number of required model evaluations. Also in practice, slightly varying Monte Carlo estimators are used for the numerical computation of Sobol sensitivity indices with the Monte Carlo simulations approach. Saltelli et al. [117] presented a comparative study of such estimators, and in their conclusions, they have found the Jansen's estimator as being more accurate. Hence, it was selected as the Monte Carlo estimator of the Sobol sensitivity indices calculated from the MCS approach in this study.

5.2.3 Standardized regression coefficients (SRC)

The SRC method requires attaining model output observations by performing Monte Carlo simulations and fitting a linear regression model to the input-output dataset of those simulations using the input variables $X_{1,..,k}$ as in the following functional form.

$$y = b_0 + \sum_{k=1}^{n} b_k X_k \tag{3}$$

where the coefficients $b_0,..,b_k$ are determined by a least-squares minimization of the residuals (differences between the y-values obtained from the regressed linear model and the actual model outputs obtained by the Monte Carlo simulations). The SRCs are obtained by scaling the raw coefficients $b_0,..,b_k$ with the standard deviations of the input variables and the output as follows:

$$SRC_k = b_k \frac{\sigma_{X_k}}{\sigma_y} \tag{4}$$

The absolute value of SRC_k provides a direct measure of the individual impact of an input variable X_k on the output variance; the higher the value, the stronger the influence. However, one major drawback of this method is that it does not provide interactional effects, i.e., the amount of variance caused on the output due to the interactions of input variables X_k. The sum of squares of the SRCs should add up to 1 ($\Sigma_k SRC^2 = 1$) for a perfectly linear model, whereas for non-linear models, this sum amounts to the coefficient of determination known as the R^2. The R^2 metric in this method also represents the fraction of the MCS data variance that can be explained by the regressed model. According to Saltelli et al. [98], the SRCs are valid measures of sensitivity if the R^2 of the underlying linear model is higher than 0.7. When the R^2 is below 0.7, the use of SRC as a direct sensitivity measure comes at the risk of being ignorant of 100×(1- R^2) % of the original output variance. In practice, the calculation of these SRCs is relatively straightforward using any regression software. The developed toolbox for this study, easyGSA, provides a Monte Carlo-based workflow for generating SRCs using the `fitlm` regressor in the MATLAB computing environment. More details of this toolbox will be discussed in section 5.6.

5.2.4 Sobol sensitivity indices (Si and STi)

Sobol's sensitivity indices contain valuable information about the contributions of individual model inputs, or sub-groups of inputs, to the variations in the output. Among the most widely encountered indices in the literature are the first order (S_i) and the total order (S_{Ti}) Sobol indices.

The first order index quantifies the individual contribution of an input variable x_i to the total output variance. The formula for calculating this index reads as follows:

$$S_i = \frac{V[E(y \mid x_i)]}{V(y)} \tag{5}$$

where $V[E(y \mid x_i)]$ is the conditional variance, and $V(y)$ is the total unconditional output variance. One practical interpretation of this index could be made as the expected amount of variance, which can be subtracted from the total output variance if the value of that input variable was to be fixed within its uncertainty range. The S_i indices are only indicative of the individual contribution of each input, excluding any contribution that may arise from interactions with other inputs. They are also called the main effects. The sum of all the main effects should add up to 1 for perfectly additive (linear) models, whereas that sum is always less than 1 for non-additive (nonlinear) models. In essence, the difference ($1-\Sigma_i S_i$) is widely regarded as an indicator of the presence of interactions between the input variables [98].

The total order indices, on the other hand, quantify the total impact of an input variable on the model output, taking into account also the impacts made due to its interactions with the other input variables. The expression for the Sobol's total sensitivity indices reads as follows:

$$S_{Ti} = 1 - \frac{V[E(y \mid x_{-i})]}{V(y)} \tag{6}$$

where $V[E(y \mid x_{-i})]$ represents the conditional variance due to all variables but x_i. One practical interpretation of S_{Ti} can be made as the expected amount of output variance that would remain unexplained if only that input variable were allowed to vary over its uncertain range. Similarly to the first order indices, the sum of all total order indices should be equal to 1 for perfectly additive (linear) models whereas, unlike the first order indices, that sum is always higher than 1 for non-additive models [98]. Another interesting interpretation of the Sobol indices is that the difference ($S_{Ti} - S_i$) can be seen as a direct indicator of the strength of interactions, if any, among the input variables. However, Sobol indices rely on the fundamental assumption of independent inputs. Therefore,

the definitions of the first and the total indices may vary when there are dependent inputs, for which the studies of Mara et al. [118] and Kucherenko et al. [119] can be consulted. In this study, the definitions given above are used for models with independent inputs only. Traditionally, nested Monte Carlo simulations are used to numerically calculate these indices. However, to ease the computational burden of such simulations, more recent approaches increasingly employ computationally efficient meta-models, which are described in the next subsection.

5.2.5 *Meta-modeling-based approaches*

As computational models became much more sophisticated and computationally demanding, meta-modeling (also referred to as surrogate modeling) received enormous research attention in recent years. Such models offer a computationally efficient alternative for usually complex simulation models in numerous engineering applications. In the context of GSA, they promise to help overcome the aforementioned high computational cost issue of the MCS approach. Extensive discussions on how to build these models can be found elsewhere [54]. Although they come in many types, three distinct classes of meta-models are commonly employed for global sensitivity analysis applications, namely, polynomial chaos expansions (PCE), Gaussian processes regression (GPR), and artificial neural networks (ANN). This selection of surrogate models is mainly motivated by the fact that they were found to be particularly effective for GSA applications in an earlier study [120], which also investigated radial basis function (RBF) interpolation, multivariate adaptive regression splines (MARS) comparatively. Readers are referred to the original work for more discussions regarding the performance comparison of the other models. In addition to these meta-models, relatively newer studies also suggested the use of orthogonal augment radial basis functions [121] and support vector regression [122] type models for performing efficient global sensitivity analysis.

5.2.6 Polynomial chaos expansions

Polynomial chaos expansions offer polynomial approximations of model responses generated by complex simulators. To approximate a random model response $y = M(x)$ using N independent random variables $x = \{x_1, x_2, ..., x_N\}$, PCE assumes the following functional shape:

$$y^{PC} = M^{PC}(x) = \sum_{\alpha \in A} S_\alpha \Psi_\alpha(x) \tag{7}$$

using the following chaos representation,

$$\begin{aligned} y^{PC} = s_0 \psi_0 + \sum_{i=1}^{N} s_i \psi_1(x_i) + \sum_{j \geq i}^{N} s_{ij} \psi_2(x_i, x_j) + \\ \sum_{k \geq j}^{N} s_{ijk} \psi_3(x_i, x_j, x_k) + ... \end{aligned} \tag{8}$$

where ψ_α is an α-th order multivariate orthogonal polynomial and S_α denotes the polynomial coefficients. By definition, the orthogonality property of the polynomials translates to an ANOVA decomposition [123]. Each multivariate polynomial ψ_α in this representation is a tensor product of univariate polynomials that are selected based on the distribution of the input variables,—e.g., Legendre polynomials for uniform distributions and Hermite polynomials for standard Gaussian distributions. More details about these univariate polynomials can be found elsewhere [124]. Due to practical reasons, this chaos representation is usually truncated to a simpler form in such a way that one only keeps those basis polynomials with degrees not higher than a predetermined degree p. Hence the number of unknown coefficients to be calculated in a PCE model, denoted by N_p, leads to the following:

$$N_p = \binom{N+p}{p} = \frac{(N+p)!}{N!\,p!} \tag{9}$$

To estimate these unknown coefficients, least-squares minimization, among other techniques, is often used by minimizing the mean square residual. However, in order to circumvent encountering an ill-

conditioned regression problem, one needs to create an initial experimental design of a size greater than N_p, which increases factorially with respect to N and p. To tackle this issue, a degree adaptive least angle regression algorithm, which tries to choose the best polynomial degree given the experimental design, is recently proposed [125]. This algorithm creates a sparse PC expansion consisting of only the significant coefficients of the full PC expansion. The authors suggest using the PC coefficients of that sparse expansion model in order to analytically compute the Sobol sensitivity indices. The generalization error of the resulting sparse PCE model can then be estimated by calculating the leave-one-out cross-validation error ϵ_{LOO} as follows:

$$\epsilon_{LOO} = \frac{\sum_{i=1}^{N}(M(x^i) - M^{PC\setminus i}(x^i))^2}{\sum_{i=1}^{N}\left(M(x^i) - \mu_y\right)^2} \tag{10}$$

where $M^{PC\setminus i}(x^i)$ is the prediction of the model constructed from an experimental design excluding x^i and $M(x^i)$ is the actual model response at the design point x^i. The upper part of this expression is also called the predictive error sum of squares (PRESS), whereas the denominator is a total sum of squares. The corresponding coefficient of determination, with an analogy to R^2, is often called the leave-one-out coefficient Q^2 and defined as follows:

$$Q^2 = 1 - \epsilon_{LOO} \tag{11}$$

Unlike R^2, Q^2 does not approach 1 as the number of model parameters increases. Hence, it provides a more comprehensive measure for the purpose of model selection. The UQLab software framework [124] developed for the MATLAB computing environment provides an implementation of the above-mentioned algorithm for the sparse PCE models, which was used in this study.

5.2.7 Gaussian process regression

Gaussian process regression (GPR) is a Bayesian probabilistic nonlinear regression algorithm, which uses kernel functions to explain a given model response as a realization of a random function of the following shape,

$$y^{GP} = M^{GP}(x) = \beta f(x) + \sigma^2 Z(x,\omega) \tag{12}$$

where the first term $\beta f(x)$ is the mean value of the Gaussian process, σ^2 is the variance, and $Z(x,\omega)$ is a zero-mean, unit-variance stochastic process. In the expression above, β denotes the regressed coefficients and $f(x)$ a set of basis functions (e.g., constants, polynomials, etc.). The algorithm uses the experimental design (i.e., the dataset of simulation inputs and outputs used to construct the GPR model) to estimate the error variance σ^2 and the coefficients β in the model. However, the model owes its expressiveness in large part to its stochastic component $Z(x,\omega)$, which contains crucial information of correlations among the data observations. To put it concisely, this component introduces latent variables ω to correlate observations x and new points x' in the input space using correlation functions (also referred to as kernel or covariance functions in the literature). Among the most frequently used kernel function types are squared exponential kernel, exponential kernel, matern 3/2, matern 5/2, and rational quadratic kernel. For more details about Gaussian process regression and the kernel functions, the readers are referred to the seminal book of Rasmussen and Williams [126].

With a wide variety of available off-the-shelf correlation functions that can be employed to define the similarity between two samples in input space, the GPR provides researchers and engineers with a powerful surrogate model that is capable of approximating highly nonlinear system responses. Once a GPR model is built from an experimental design, one can then use this model as a surrogate for the original model of the response. Regarding the GSA application, the GRP model replaces the

computationally costly original model in the Monte Carlo procedure described above to calculate Sobol sensitivity indices. The Statistics and Machine Learning toolbox of the MATLAB software (The MathWorks) offers implementations of different types of kernel functions for GPR models. This implementation was used for developing GPR models in this study.

5.2.8 *Artificial neural networks*

Inspired initially by biological neurons, artificial neural networks have attracted enormous research attention in recent years, finding successful applications in highly challenging machine learning tasks, such as image classification, speech recognition, computer vision, etc. [127]. Although today they come in many types and topologies, one of the earliest and the simplest type of neural networks is called feedforward neural networks (FNN), where connections between the nodes do not form a cycle as in more recent recurrent neural networks (RNN). One example of FNNs is the multilayer perceptron (MLP), where there are at least three layers (input, hidden, and output layers) with interconnecting transfer functions (also referred to as activation functions) such as sigmoids, logistic function, hyperbolic tangent, etc. Figure 5.1 illustrates a single hidden layer architecture of a feedforward MLP, where weights and bias terms are used as the model parameters. When there is more than one hidden layer in the network architecture, it is then often called a deep neural network (DNN). Due to their highly versatile and scalable architectures, neural networks are proven to be very effective at capturing vastly nonlinear relationships in datasets, making them an ideal candidate also for the generation of surrogate models. One of the most widely used training algorithms for neural networks is the backpropagation algorithm [128], in which a prediction is made for each training observation in a forward pass, and the error contribution from each connection in the network is calculated in a reverse pass to adjust the corresponding weights so that the error in the final prediction is reduced. The Deep Learning

Toolbox of the MATLAB software provides implementations of the backpropagation algorithm variants, which are comparatively used to develop the ANN models in this study.

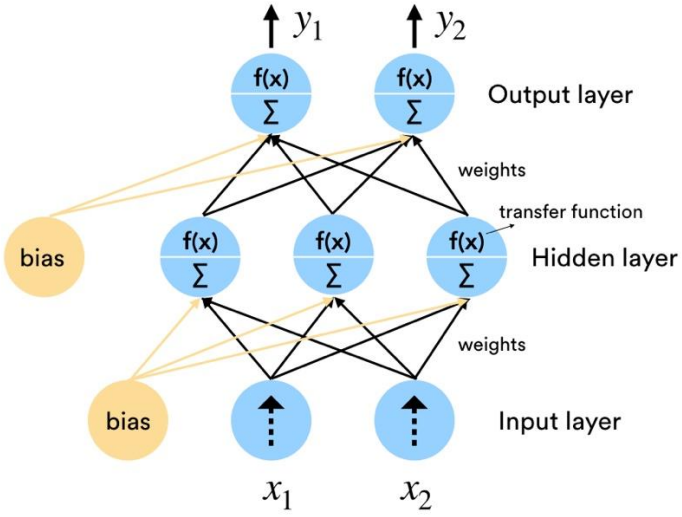

Figure 5.1: A network architecture of a feed-forward MLP is shown with a single hidden layer and an output layer. Input variables (x1 and x2) are passed through the input, the hidden, and the output layers (also being affected by the bias and the weight terms) to produce the network outputs (y1 and y2) [129].

5.3 A generic framework for meta-model development

A generic framework for constructing different types of surrogate models to be used for global sensitivity analysis is presented in Figure 5.2. The framework starts with the definition of input space, i.e., a set of parameters that are subject to the sensitivity analysis. The step requires all the available information regarding bounds and distributions of the input parameters. Secondly, an experimental design is created using a sampling algorithm, such as space-filling designs like Latin Hypercube sampling (LHS) [86], or quasi-random sequences with uniformity properties, such as Sobol and Halton sequences. These sampling algorithms aim at addressing the problem of how to effectively and efficiently span the input spaces of usually higher dimensions with as few samples as necessary. A recent comparative study found the best results when using samples generated with quasi-random Sobol sequences compared to the LHS and Halton designs [130]. In recent studies, more efficient sampling designs are also sought with the use of adaptive algorithms, such as those defined in [99,131]. After placing the random samples in the input space, Monte Carlo simulations are consecutively performed to generate the experimental design, i.e., a paired dataset of responses corresponding to each input sample. This experimental design is then fed into the surrogate model development algorithms to train the surrogate models. After the training, one often needs to estimate the generalization capability of the trained surrogate models before their use. Among the techniques that can be used for this purpose are the holdout validation method, the k-fold cross-validation method, the leave-one-out cross-validation method, and the Monte Carlo cross-validation method. For a comparative discussion of these cross-validation methods, readers are referred to the work of Molinaro et al. [132]. The size of the experimental design directly dictates the total computational cost of surrogate model development. Hence, an appropriate size of this training sample is decided by the framework, which monitors the evolution of a

user-defined target accuracy on an error metric while incrementally increasing the size of this sample.

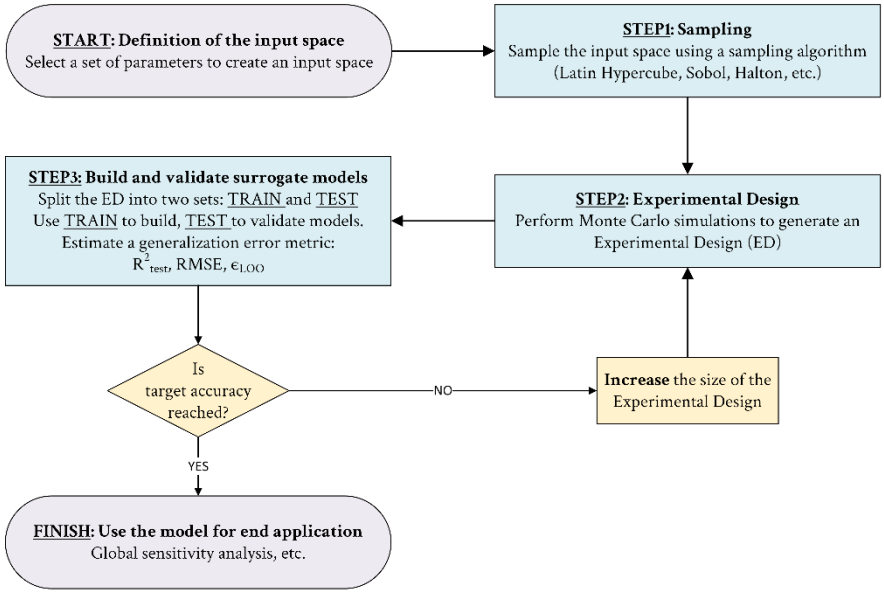

Figure 5.2: The flowchart of the generic framework for building surrogate models. An iterative loop is added to ensure a user-set target model accuracy by incrementally increasing the size of the experimental design [129].

It should also worth noting that the generic feature of the framework allows for the construction of all different types of meta-models considered in this study from the same dataset without a further need for tuning the dataset. On the other hand, each meta-model training, which is coded as functions returning a model given a dataset, has its own corresponding model development algorithm to identify an optimal meta-model structure and its parameters. For instance, hyperparameters of a GPR model, such as kernel function-specific parameters, are optimized using a Bayesian solver to improve its performance of cross-validation statistics [126] while the number of neurons in an ANN model is found using a grid search algorithm.

5.4 Case study

In this section, a case study is formulated to elucidate the application of meta-models for plant-wide global sensitivity analysis of WWTPs. For that purpose, the widely-studied Benchmark Simulation Model No. 2 framework, as initially developed by Jeppsson et al. [133], was taken as the base WWTP layout and is further modified as shown in Figure 5.3. The modified layout contains a primary clarifier (PC), an activated sludge (AS) process, and a secondary clarifier (SC) as the mainstream technologies, as well as a thickener, an anaerobic digester (AD), and a dewatering unit for the treatment of the sludge coming out of the AS and PC units. The AS process contains five tanks that are modeled using the Activated Sludge Model No. 1 (ASM1) [134]. The first two tanks are anoxic (aiming at denitrification), whereas the remaining three tanks are aerobic (aiming at nitrification and carbon removal).

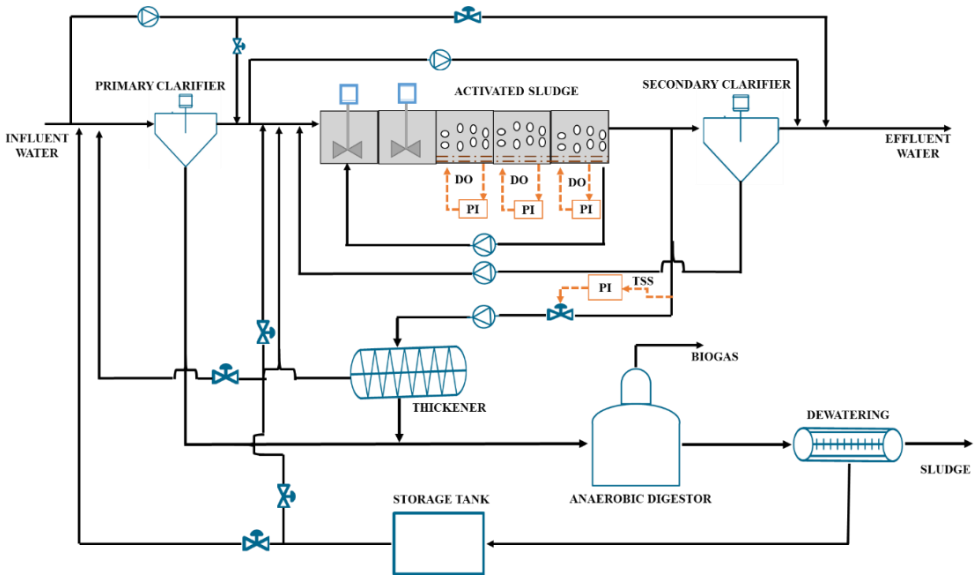

Figure 5.3: Schematic overview of the modified Benchmark Simulation Model No. 2 (BSM2) framework used for global sensitivity analysis [129].

In the modified layout, the waste activated sludge is discharged from the last aerobic tank of the AS process (adopted from other studies [12,88,135]) instead of the SC unit as in the original BSM2 configuration. The SC unit separates active biomass from the effluent and is modeled as a non-reactive system using an exponential settling velocity function

proposed by Takács et al. [136]. The AD unit is modeled using the Anaerobic Digestion Model No. 1 (ADM1) [90]. Further details about the employed modeling and simulation strategy can be found in Appendix B.

5.4.1 Scenarios for global sensitivity analysis

As shown in Table 5.1, four realistic scenarios are developed for this case study. In scenario 1, the objective is to quantify the sensitivity of the plant's key performance indicators (KPIs); such as methane gas production, aeration energy demand, etc., to the uncertainty in the data regarding the incoming influent wastewater characterization, which is represented with a set of influent fractionation parameters and their uncertainty ranges as shown in Table 8.1. In scenario 2, the emphasis is placed on the model parametric sources of uncertainties of the plant, namely the stoichiometric and biokinetic parameters of the ASM1 model. Table 8.2 tabulates the list of those uncertain parameters and their uncertain bounds considered in this scenario. In scenario 3, the goal is to study the impacts of the plant's design and hydraulics-related parameters (as listed in Table 8.3) on the KPIs. Finally, in scenario 4, all different sources of uncertainty considered in the previous scenarios are combined in order to identify the most crucial parameters and quantify their effects on the plant KPIs. More detailed descriptions of all scenarios are also included in Appendix B.

Table 5.1: Different sources of uncertainties considered in the four scenarios to be studied [129].

Scenarios	Sources of uncertainty		
	Influent data	Kinetics and stoichiometry	Hydraulics and design-related parameters
1	Uncertain	Certain	Certain
2	Certain	Uncertain	Certain
3	Certain	Certain	Uncertain
4	Uncertain	Uncertain	Uncertain

The sensitivity analysis results of each scenario are evaluated based on 6 different plant performance indicators; namely effluent quality index

(EQI), methane gas production in the AD tank, aeration energy demand by the AS tank, net sludge discharge from the plant, effluent ammonium, and effluent nitrate concentrations. The EQI definition is adopted from [11], and it represents a sum of pollutant loads of the effluent. It is important to note that the EQI is calculated for one day (average load used for steady-state simulation), and not for a longer period (i.e., for the last 364 days) as used for dynamic simulation of the BSM2 plant. The aeration energy demand in the AS tank is calculated using the correlation suggested in [11]. The equations used for calculating the above-mentioned performance indicators (such as EQI, AE demand, etc.) are provided in Appendix B.

5.5 Results and discussions

5.5.1 Validation of PCE, GPR, and ANN models

PCE models were developed following the proposed generic framework, which for PCE modes sequentially increased the experimental design size and the maximum allowable polynomial order until a target accuracy ($0.95< Q^2$) is reached. Table 5.2 shows the performance statistics of the final PCE models attained for scenario 1, in which the input space consisted of 7 influent parameters. Despite imposing a maximum order limit of 20, the final returned PCE models had moderately low orders with a maximum of 6. The size of the experimental design was initially set to 100, with an incremental size increase of 50, to a maximum of 250 samples, which was hit by the framework only for developing models of effluent ammonium and effluent quality index for the scenario 1 (see Table 5.2) as the target accuracy was not reached. The leave-one-out error metric (ϵ_{LOO}) was used to select the best PCE models in the adaptive sparse PC expansion algorithm since it gives a better estimate of the model's accuracy compared to R^2 [125]. On the whole, considering the low number of coefficients (see the number of nonzero coefficients in Table 5.2) and the polynomial orders, it is possible to build highly accurate PCE models that require as few samples as a few hundred by following the proposed framework along with the underlying degree adaptive sparse expansion algorithm. Performance statistics of the PCE models developed for scenarios 2, 3, and 4 are also included in Appendix B.

Table 5.2: Performance statistics of PCE models developed for scenario 1 [129].

PCE model output	Size of experimental design	PCE order p	Training R^2	Leave-1-out coefficient Q^2	# of nonzero coeffs
Effluent nitrate	150	6	0.988	0.950	37
Effluent ammonium	250	5	0.955	0.900	26
Effluent quality index	250	5	0.976	0.910	48
Sludge disposal	100	2	0.999	0.999	21
Aeration energy	100	4	0.993	0.968	28
Methane production	100	5	0.999	0.999	24

For developing the GPR models, the proposed framework employed a training algorithm, which incrementally increased the size of the experimental design, and for each different size, it sought the best kernel function among a family of available alternatives (from MATLAB implementation) by also performing a hyper-parameter optimization. The target accuracy was set to obtain models with a cross-validation score ($0.95 < Q^2$). Similar to the PCE model development, the chosen increment was 50. An increase in the size of the experimental design is only made if the desired accuracy was not reached. Table 5.3 shows the statistics of the final selected scenario 1 GPR models, which had a maximum of 150 samples. The results also show the superior performance of the GPR models over the PCE models. Appendix B also contains the statistics of the models developed for the other scenarios.

Table 5.3: Performance statistics of GPR models developed for scenario 1 [129].

GPR model output	Size of experimental design	Training R^2	Leave-1-out coefficient Q^2	Kernel function
Effluent nitrate	100	1	0.977	ARD Matern 3/2
Effluent ammonium	150	0.999	0.976	Sq Exponential
Effluent quality index	150	1	0.984	ARD Matern 5/2
Sludge disposal	100	0.999	0.999	ARD Matern 3/2
Aeration energy	100	1	0.994	ARD Matern 3/2
Methane production	100	0.999	0.999	ARD Matern 3/2

For developing the ANN models, the proposed framework employed a grid search approach that tried to find the best combination of user-configurable network parameters, which included the number of neurons in each layer, training algorithms, and the transfer functions. The Deep Learning Toolbox of the MATLAB software is used to develop the ANNs with backpropagation training algorithm variants, e.g., Levenberg-Marquardt (LM), Bayesian regularization (BR), scaled conjugate gradient (SCG), and conjugate gradient with Powell-Beale restarts (CGB) were all tested for each output. To decide on the right size of the training sample, 5-fold cross-validation is used. Further increase in the sample size is avoided if the target accuracy (cross-validation R^2 above 0.95) is reached.

One-fifth of the training sample is allocated as a validation dataset while the rest is used for training. Bayesian regularization is most often selected as the training algorithm by the framework, whereas the size of the hidden layer (i.e., number of hidden nodes) is allowed to vary from 10 to 15 in order to keep the number of network connections at a reasonable level. Table 5.4 shows the obtained performance scores and configuration details (number of nodes and the training algorithm) of the final returned ANN models for scenario 1. Results show the highest cross-validation scores obtained with the ANN models. Appendix B also contains the statistics of the ANN models developed for the other scenarios.

Table 5.4: Performance statistics of ANN models developed for scenario 1 [129].

ANN model output	Size of experimental design	Training R^2	Cross-validation R^2 (5-fold)	Selected configuration
Effluent nitrate	150	0.999	0.978	7×12×1 with BR
Effluent ammonium	150	1	0.992	7×10×1 with BR
Effluent quality index	150	1	0.994	7×10×1 with BR
Sludge disposal	100	0.999	0.998	7×15×1 with BR
Aeration energy	150	0.999	0.993	7×11×1 with BR
Methane production	100	0.999	0.998	7×14×1 with BR

It is important to emphasize here that these meta-models are developed for global sensitivity analysis purpose, and hence the accuracy of these models are valid within the entire domain of their defined input space. Said differently, if used beyond the limits of their input space, these models do not guarantee the same accuracy. Moreover, for different application purposes, the proposed framework might be aligned accordingly. For instance, if one is interested in understanding the use of meta-models in a local sensitivity analysis context (i.e., how much a small variation of a single input parameter affects the outputs), then the scope of design of experiments during the sampling stage needs to be aligned with this purpose. In that case, one should define a narrower range of input space parameters around their nominal values (where local sensitivity analysis is to be performed), and construct a meta-model for that range.

5.5.2 Comparison of SRCs and first order Sobol indices

As described earlier, as a sensitivity measure, the standardized regression coefficients are considered valid only when the linear regression model (fitted to the Monte Carlo simulation results) has a coefficient of determination (R^2) higher than 0.7, and the sum of squared SRCs should add up to 1, for a perfectly linear model [137], similar to the Sobol's first order indices. Hence, these two sensitivity measures are comparable to one another, despite relying upon distinct theoretical grounds (regression for SRCs and variance decomposition for S_is). To show such a comparison, the Monte Carlo simulation results of scenario 1 were used to calculate the SRCs of the input parameters.

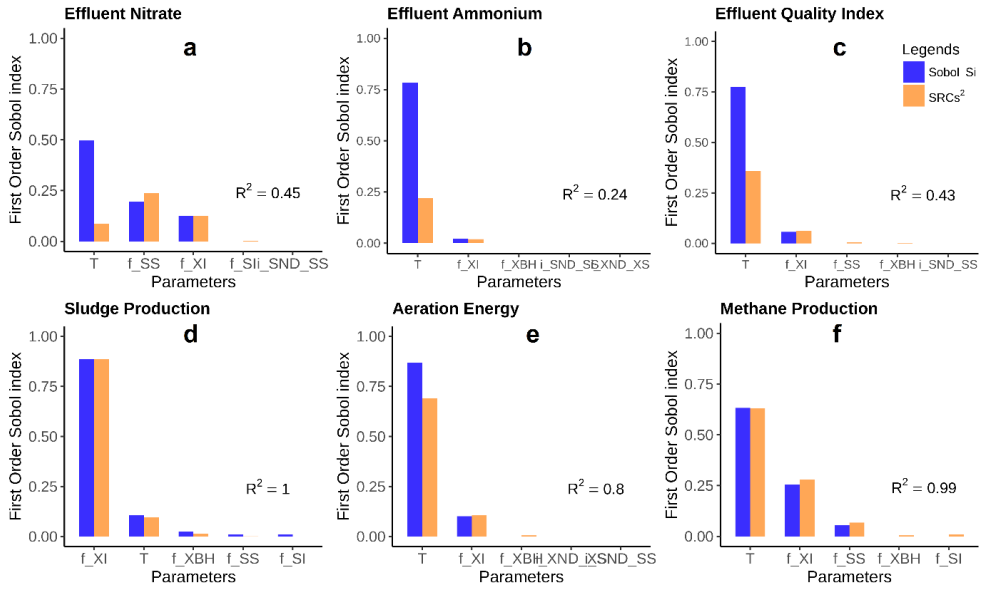

Figure 5.4: Comparison of standardized regression coefficients (squared) with first order Sobol indices. As the linear model R2 decreases, SRCs provide increasingly diverging indices compared to the corresponding Sobol indices [129].

Figure 5.4 shows the results of the comparison of squared SRCs with the first order Sobol indices obtained from the same Monte Carlo simulations. Also shown in Figure 5.4 are the R^2 values attained from the regressed linear models. As can be seen, for KPIs whose linear models had low values of R^2, such as effluent ammonium, the two measures diverge significantly, whereas, for KPIs with higher linear model R^2, such as sludge production, they converge to the same values and rankings. For

the effluent ammonium, however, the SRC method and the Sobol method yield different rankings for the most significant parameters. As was also pointed out by [137], the Sobol indices are more reliable than regression-based methods whose accuracy is upper bounded by the percent of the output variance explained by the fitted linear model, i.e., R^2. Thus, these results simultaneously demonstrate why and when the Sobol method should be preferred over the SRC method for global sensitivity analysis.

Aside from the sensitivity aspects, the varying performance results of the linear models also imply that certain KPIs of the WWTP system, such as effluent ammonium and nitrate concentrations, exhibit a nonlinear behavior with the chosen set of input parameters. For instance, the input-output relationships between the influent temperature and effluent ammonium, as well as the influent temperature and effluent nitrate, are visualized in Figure 8.1 and Figure 8.2, respectively. Both figures show a strong nonlinear type correlation; therefore, the predictions from the linear model diverge from the true simulation results. Moreover, Figure 5.5 shows the poor predictions of the linear model for effluent nitrate in comparison to those obtained from PCE and GPR models. Overall, these results further motivate the importance of systematically exploring different functional forms (meta-model structure) to describe input-output data relations, as presented in this framework.

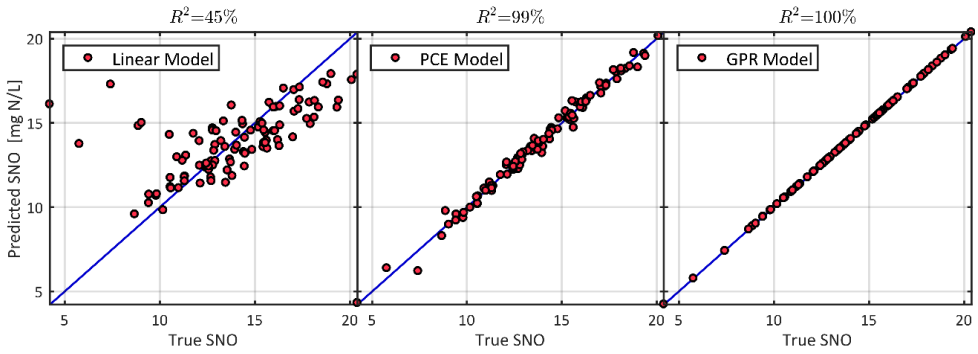

Figure 5.5: Predictive performance comparison of the linear model used by the SRC method to more advanced meta-models of type PCE and GPR for scenario 1 results of effluent nitrate concentration [129].

5.5.3 Comparison of Monte Carlo and meta-model-based approaches for calculation of Sobol indices

5.5.3.1 Results of scenario 1

Figure 5.6 shows the results of scenario 1 for the total order Sobol indices, which are obtained from using four different approaches: the Monte Carlo indices (using the BSM2 simulation model with the Monte Carlo procedure), the PCE indices (generated from the PCE model itself), the GPR indices (using the GPR models with the Monte Carlo procedure), and the ANN indices (using the ANN models with the Monte Carlo procedure). By comparing the different approaches, it is found that the benchmark Monte Carlo indices were in very good agreement with the indices obtained from PCE, GPR, and ANN meta-models, with the GPR and ANN models giving slightly better results compared to the PCE models. This could be explained by the fact that the GPR and ANN models have better cross-validation performance scores compared to the PCE models (see Table 5.2, Table 5.3, Table 5.4), especially for the models of effluent nitrate, effluent ammonium, and effluent quality index. However, for the effluent quality index, the ANN model performs even better than the GPR model by giving closer results to the benchmark Monte Carlo indices. For methane and sludge production, all the models give highly agreeing results, as they all have cross-validation scores close to 1.

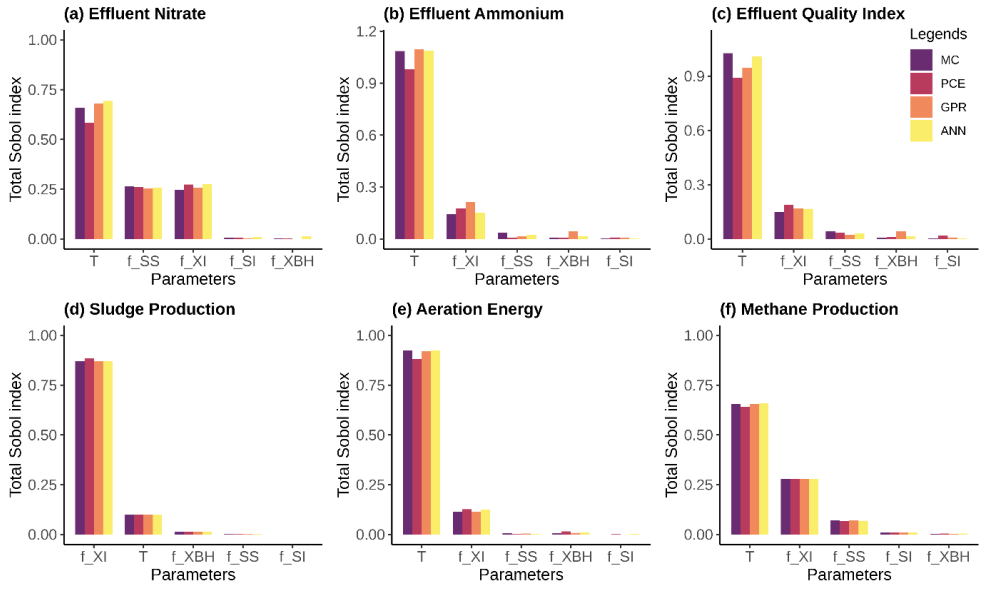

Figure 5.6: Comparison of Sobol indices obtained from the original plant model (BSM2) using the MC procedure and meta-models (PCE, GPR, and ANN) for scenario 1. The most influential influent fractionation parameters are shown for the key plant performance indicators: (a) Effluent nitrate (b) Effluent ammonium (c) Effluent quality index (d) Sludge production (e) Aeration Energy (f) Methane production [129].

From a process understanding point of view, the results show that the most significant parameters in this scenario are the wastewater temperature (T), the influent fractions of f_{SS}, and f_{XI}, respectively. For the effluent ammonium, the temperature comes as the most influential input, capturing more than 95% of the output variation. This is expected and consists with the current process understanding, as the nitrification rate is described by an Arrhenius equation, which increases rapidly with the increasing temperature T. Similarly, the effluent nitrate concentration depends on the rate of the underlying denitrification process, which inherently also relies on the availability of biodegradable COD (both soluble and particulates), and the heterotrophic growth rate. In this sense, by ranking T (due to its influences on the growth of heterotrophs and autotrophs), f_{SS} (representing the soluble biodegradable COD), and f_{XI} (inversely related to biodegradable particulate COD) as influential parameters, the sensitivity results support the process understanding.

Concerning the results for aeration energy demand and EQI, T and f_{XI} are the most influential parameters as the growth of heterotrophs and

autotrophs depends on temperature. Higher growth rate and availability of biodegradable particulate COD (which correlates with lower fractions of inerts f_{XI}) translate to a higher demand for dissolved oxygen (hence, higher aeration energy demand). The EQI calculation depends on the effluent COD, BOD_5, TSS, TKN, and nitrate concentration (see Eq (2)), and the temperature influences the effluent concentration by altering the growth rate.

The methane production in the AD unit is directly related to the COD load, which comes from the sludge produced in the AS and PC units. The availability of biodegradable COD influences the sludge production in the AS unit, which among other factors, is also affected by the biomass decay rate, which is itself also a function of the temperature. Similarly, the influent COD fractions also directly influence the composition of the underflow solids of the PC, which are then sent to the AD unit for methane production. Hence, the sensitivity analysis results identify T, f_{XI}, and f_{SS} as the most influential parameters for methane gas production as well. On the other hand, the sludge production from the plant is mainly influenced by the influent f_{XI} (which directly affects the amount of sludge produced in the plant). The temperature has a lower impact here, mainly because of its effect on the decay rate of the biomass in the system. All in all, these results are all in good agreement with the general process understanding of the activated sludge-based wastewater systems.

The results of scenario 1, where the objective was to study the importance of influent compositions on the plant KPIs, also imply that uncertainty in the influent wastewater (such as in COD fractionations) and temperature, which could be affected by climate change, exert crucial impacts on all of plant KPIs. Therefore during the design of WWTPs, to ensure the robustness of plant performance against future changes in influent quality and temperature, appropriate engineering measures need to be proposed (e.g., by increasing design safety margins).

5.5.3.2 Results of scenario 2

In scenario 2, the purpose was to apply and compare the meta-models for GSA and also to assess the importance of model parametric uncertainties of the plant. In BSM2, ASM1 is employed to describe microbial conversion dynamics in activated sludge tanks. Therefore, as the source of uncertainty, the uncertain stoichiometric and kinetic parameters of the ASM1 model are investigated in this scenario, which had 20 uncertain parameters in total. The sensitivity analysis results of this scenario are shown in Figure 5.7. Regarding the comparison of the Sobol indices returned by the Monte Carlo and the meta-modeling approaches, there is no distinguishable difference in the results, with the exception of PCE models giving slightly diverging results in comparison to the others, which can be attributed to their slightly lower cross-validation scores. Moreover, GSA results from meta-models also show a good agreement with those obtained from the MCS approach in terms of correctly ranking the most important parameters.

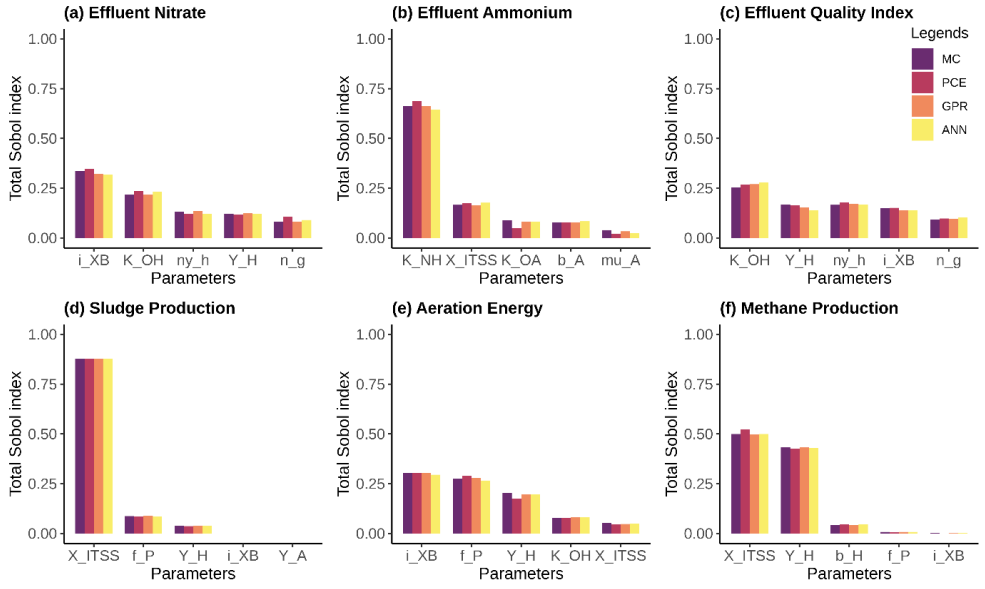

Figure 5.7: Comparison of Sobol indices obtained from the original plant model (BSM2) using the MC procedure and meta-models (PCE, GPR, and ANN) for scenario 2. The most influential stoichiometric and kinetic parameters are shown for the key plant performance indicators: (a) Effluent nitrate (b) Effluent ammonium (c) Effluent quality index (d) Sludge production (e) Aeration energy (f) Methane production [129].

From the wastewater treatment perspective, Figure 5.7 reveals the parameters i_{XB}, K_{OH}, η_{yh}, Y_H, and η_g (in descending order) as the most important for the effluent nitrate. This is because when the nitrogen fraction in biomass (i_{XB}) increases, the ammonium concentration inside the reactor increases too since this nitrogen fraction is hydrolyzed after biomass decay. Therefore, the increased availability of ammonium for nitrification/oxidation leads to higher effluent nitrate concentration, which is also dependent on the anoxic growth rate of heterotrophs (K_{OH}, η_g), which convert nitrate to gaseous N_2. As expected, the stoichiometric parameter heterotrophic yield (Y_H) is also important for effluent nitrate. The same set of parameters also comes as the most important for the EQI, because the effluent nitrate concentration has a considerable weight in the EQI calculation (see Eq (2)). On the other hand, Figure 5.7 reveals the nitrification parameters (K_{NH}, K_{OA}, b_A, u_A) as the most significant for the effluent ammonium as well as X_{ITSS} (i.e., TSS to COD ratio), which is found to be the dominating factor for the sludge production in the plant. This is because this parameter affects the sludge wastage rate, which determines the amount of nitrifying biomass in the system. It indicates that the sludge production in the AS system increases (recycling more active biomass from SC to AS) with an increase in X_{ITSS} value, and the SRT decreases (as constant TSS is maintained by the controller) by manipulating the flowrate Q_w (waste sludge goes to the AD via the thickener) and reduces the nitrification rate. This explanation agrees well with the findings also reported by Sin et al. [12].

5.5.3.3 Results of scenario 3

Scenario 3 looked at the influence of hydraulic and design-related parameters on the plant KPIs and the Sobol sensitivity analysis results of this scenario are shown in Figure 5.8. Similar to scenario 2, the good agreement between the results of the different GSA approaches is still preserved with the target Monte Carlo indices being indistinguishably followed by the indices generated by the meta-models. As for the process understanding, Figure 5.8 pinpoints the slow settling parameter (r_P) as

the most important in all plant KPIs expect the effluent ammonium. The same parameter was also found to be playing the most significant role in plant KPIs in an earlier study of Ramin et al. [138], which looked at sensitivities of the settling model parameters. In addition to r_P, the volumes of anoxic tanks (*VOL1*, *VOL2*) and the internal recycle flowrate (Q_{intr}) also come as the important parameters for the effluent nitrate concentration in this scenario. The *VOL1* and *VOL2* affect the hydraulic residence time (HRT) spent in the anoxic tank, whereas the internal recycle (Q_{intr}) in the AS system transports nitrate from the aerobic tank to the anoxic tank [139]. Therefore, altering these parameters can lead to inefficient denitrification, which can then influence the effluent nitrate concentration. On the other hand, the effluent ammonia is influenced by aerobic tank volumes (i.e., *VOL3*, *VOL4*, and *VOL5*) as the longer aerated HRT will allow more nitrification to take place in the system [140]. As expected, the aeration energy is mainly influenced by the aerated tank volumes. Based on scenario 3 results, it can be argued that hydraulics and effective volumes of reactors (which may change due to construction or mixing issues in aeration and settling tanks) have the potential to affect the design performance metrics. Therefore good engineering practices and design standards (such as Metcalf & Eddy Inc. et al. [92]) should be followed in plant design and commissioning.

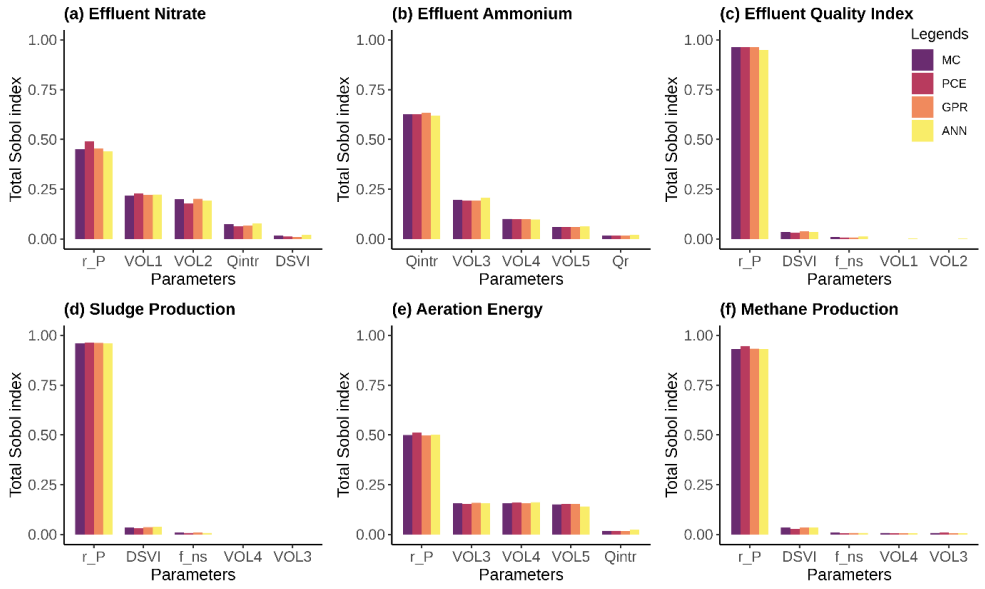

Figure 5.8: Comparison of Sobol indices obtained from the original plant model (BSM2) using the MC procedure and meta-models (PCE, GPR, and ANN) for scenario 3. The most influential hydraulics and design parameters are shown for the key plant performance indicators: (a) Effluent nitrate (b) Effluent ammonium (c) Effluent quality index (d) Sludge production (e) Aeration energy (f) Methane production [129].

5.5.3.4 Results of scenario 4

The objective of the scenario 4 is twofold: to analyze the comparative effects of different uncertainty scenarios, and also to test the robustness of the meta-model based GSA approaches in high dimensional settings. Therefore, all the parameters considered in previously studied scenarios are gathered together to attain a more extensive parameter set. Figure 5.9 shows the Sobol sensitivity results obtained for this scenario.

As was also found in scenario 1, all the plant KPIs show high sensitivity towards the variations in the incoming influent temperature, except total sludge production, which is primarily dictated by the particulate organic matter fraction (f_{XI}) of the influent, as expected. For effluent ammonium, effluent quality index, and aeration energy demand, the only predominant factor is the influent temperature whereas, for methane production, the influent fractions of f_{XI} and f_{SS} were also found among the significant parameters, as in scenario 1.

In comparison, the top-listed parameters of this scenario differ from those of scenario 1 only with the inclusion of some scenario 2 parameters (such as i_{XB}, f_P, X_{ITSS}). In essence, this comparison of different sets of uncertain parameters on the plant KPIs implies that influent uncertainty is the most critical source of uncertainty in WTTP design, as it causes the most substantial deviations from the means of the performance metrics (see Table 8.4). On the other hand, hydraulics and design-related parameters of scenario 3 play a negligible role when compared with the other sets. Concerning the parameter ranking results obtained from the meta-models, a good agreement with the costly MCS approach is still preserved (see Figure 5.9), even when the dimension of the parameter space was relatively high (d=37). PCE models give slightly less accurate results compared to the ANN and GPR models (for example, effluent nitrate in Figure 5.9), though they all correctly rank the most sensitive parameters. This further strengthens the reliability of the use of the meta-modeling-based approach for global sensitivity analysis of complex plant-wide models.

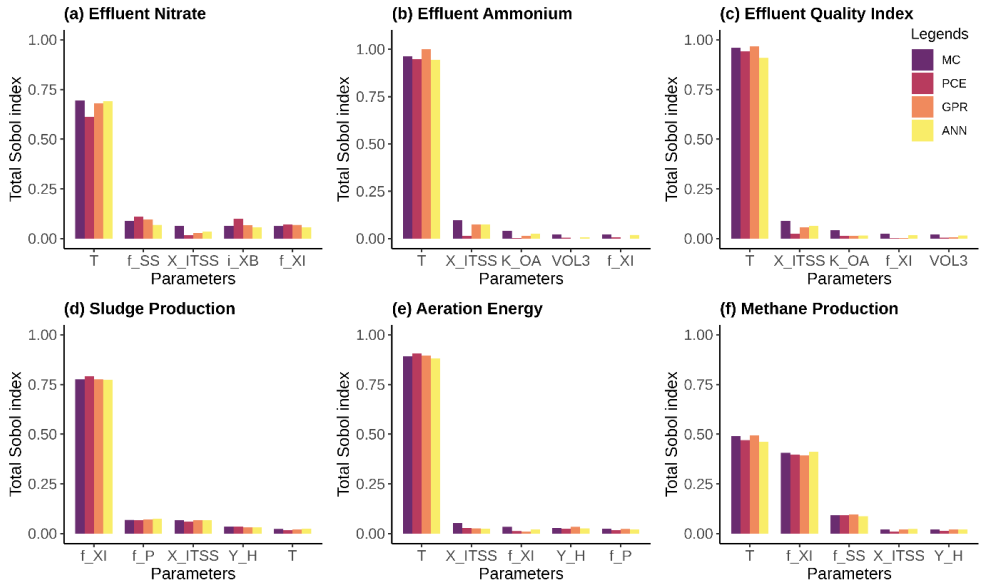

Figure 5.9: Comparison of Sobol indices obtained from the original plant model (BSM2) using the MC procedure and meta-models (PCE, GPR, and ANN) for scenario 4. The most influential parameters considered in all scenarios are shown for the key plant performance indicators: (a) Effluent nitrate (b) Effluent ammonium (c) Effluent quality index (d) Sludge production (e) Aeration energy (f) Methane production [129].

5.5.4 Computational cost comparison of different GSA approaches

In terms of their computational costs, the GSA approaches discussed above differ considerably. As the actual elapsed clock time for performing the required simulations will vary depending on the simulation and the solver settings, the type of processing units, the computing architecture (clusters, clouds, etc.), and the computing strategy (level of parallelization), a comparison is given in terms of the number of costly model evaluations required, which is listed in Table 5.5 for each approach. To obtain Sobol indices from the Monte Carlo procedure, sampling matrices of size 2000 were used, which resulted in 78000 plant model simulations for scenario 4 for which a total of 37 parameters were sampled. For investigating all the scenarios, a total of 164000 plant-wide simulations were performed in parallel using a high-performance cluster connected to 100 processing units. The relatively large size of the input matrices is necessary to ensure accurate convergence of each Sobol index of all the investigated parameters. As expected, the meta-modeling based approaches needed a significantly lower number of original model simulations, which is the size of the experimental design from which they were built. The incremental algorithm of the framework used as few as 100 simulations to build the GPR and ANN models for most of the outputs of scenarios 1, 2, and 3; however, a larger dataset is needed to attain the target cross-validation scores in scenario 4. A total of 600 original model simulations were used to produce the GPR based indices, plus the time spent on performing the Monte Carlo simulations with meta-models (around 10 minutes in the cluster), which is negligible considering a single plant simulation time of 6 minutes (t_{BSM2}). On the other hand, the PCE models required a total of 750 simulations without the need for Monte Carlo simulations, as explained above. The standardized regression coefficients were generated using the Monte Carlo simulation results (of the BSM2), for which a sample size of 1000 was used for each scenario, resulting in 4000 simulations in total. Unlike

the Sobol method, the MC procedure for SRC does not depend on the number of parameters sampled. In overall, using the meta-models for conducting the scenario analysis presented in this work results in an order of magnitude computational gain for the SRC method, and a two orders of magnitude gain for the Sobol method.

Table 5.5: Comparison of computational costs of different approaches for global sensitivity analysis [129].

Approach	**# of plant-wide simulations used**				**Total computational cost**
	Scenario 1 ($d = 7$)	Scenario 2 ($d = 20$)	Scenario 3 ($d = 10$)	Scenario 4 ($d = 37$)	
SRC with MCS	1000	1000	1000	1000	$4000 \times t_{BSM2}$
Sobol with MCS using BSM2	18000	44000	24000	78000	$164000 \times t_{BSM2}$
Sobol with MCS using GPR	150	100	100	250	$600 \times t_{BSM2}$
Sobol with MCS using ANN	150	100	100	450	$800 \times t_{BSM2}$
Sobol with PCE	250	150	100	250	$750 \times t_{BSM2}$

5.6 Tool implementation: easyGSA

In an effort to make global sensitivity analysis methods accessible to non-specialist users (i.e., people with basic experience in sensitivity analysis), a generic toolbox named easyGSA was developed. The easyGSA toolbox contains functionality for both Monte Carlo and meta-modeling-based calculation of global sensitivity indices, as discussed in this chapter. The current release of the easyGSA toolbox incorporates the Sobol indices method, standardized regression coefficient method, Gaussian process regression-based, and artificial neural network-based sensitivity indices.

As shown in Figure 5.10, a public GitHub repository [141] (freely accessible from https://github.com/resulal/easyGSA) was created to maintain the latest version (version 1.4) of the toolbox. The toolbox is MIT-licensed, which grants free-of-charge permission to use given a reference to the original published version [129]. Upon download of the toolbox, the users need no further installation and can directly work with the toolbox by including its folder path to their current MATLAB session. The folder organization of the toolbox is shown in Figure 5.11. The subfolder *demos* contains a set of benchmark test problems whose sensitivities are calculated using the toolbox. The subfolder *docs* includes documentation which comprises of a set of lecture slides and commented scripts, some of which are also shown in Appendix B..

Different GSA methods and surrogate models in the easyGSA toolbox can be activated using the input argument structure of its primary function easyGSA, whose basic syntax is shown in Figure 5.12. The complete set of input arguments that can be selected by the users is shown in Figure 5.13. Of all the input arguments, only 3 of them are set as compulsory, with the rest being set to default values. The compulsory arguments are the model subject to the GSA, size of the sampling matrices desired for the GSA, and the input space of parameters subject to sampling. The input space needs to be defined in a cell array data structure with parameter names and distribution bounds, as shown in Appendix B

examples. The current implementation only allows for two of the most commonly used distributions (i.e., uniform and Gaussian input distributions).

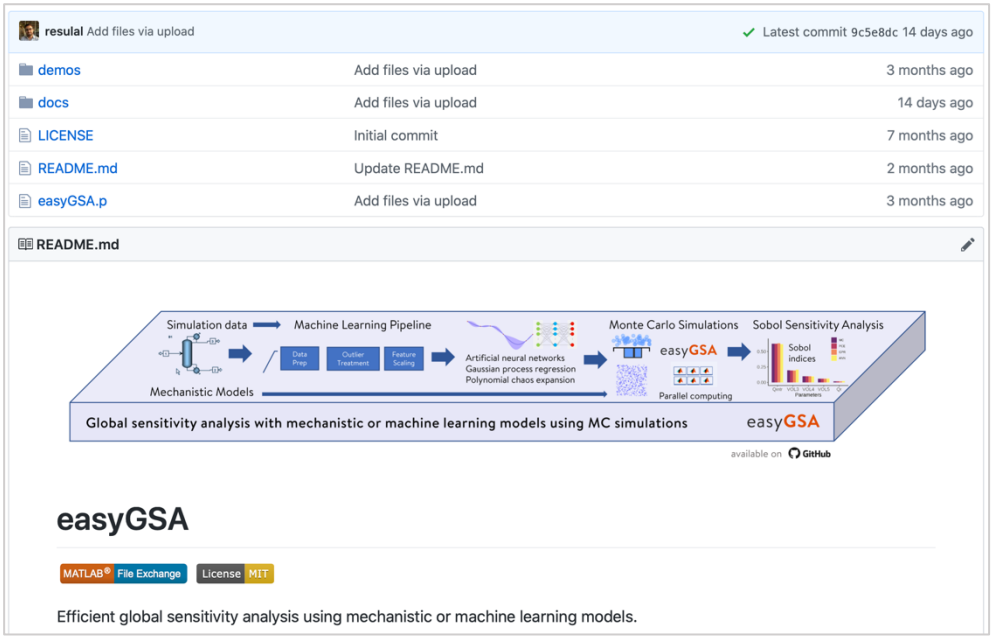

Figure 5.10: GitHub repository containing the toolbox easyGSA.

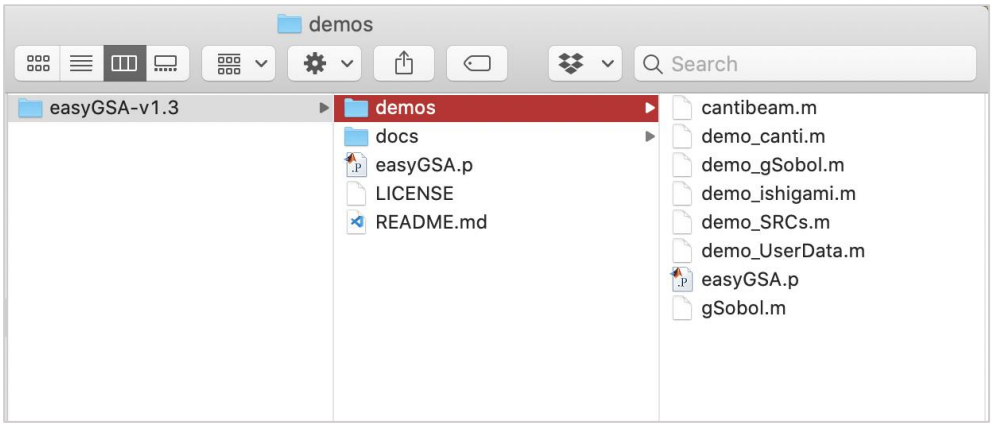

Figure 5.11: Folder organization of the easyGSA toolbox.

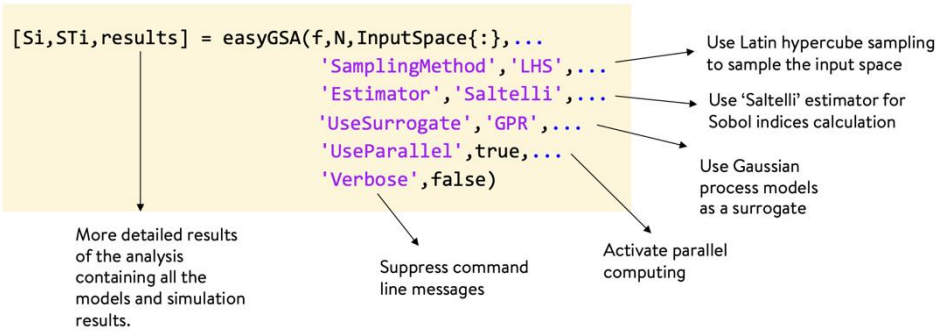

Figure 5.12: Basic syntax of the easyGSA function written in MATLAB.

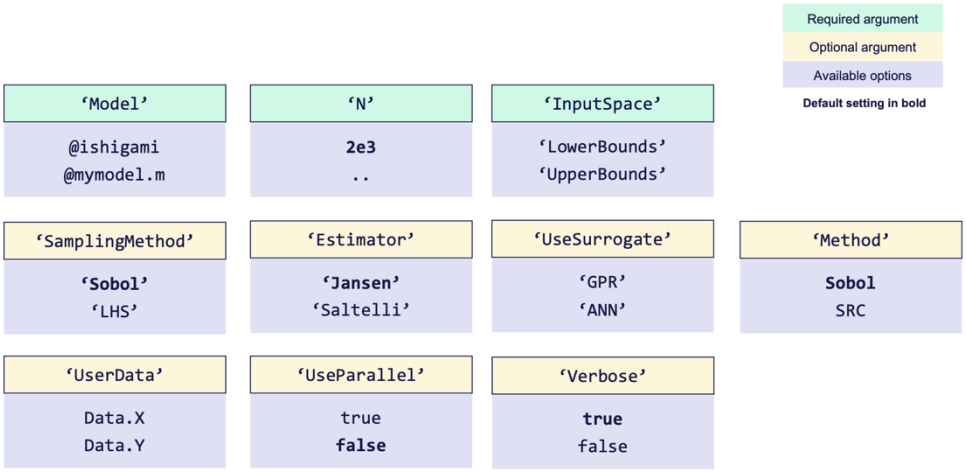

Figure 5.13: Input arguments overview of the easyGSA toolbox. Required arguments in green, optional arguments in yellow, and default settings in bold.

The optional arguments allow for switching between the GSA methods, the sampling methods, Monte Carlo estimators for indices, surrogate model types, as well as options for whether to use parallel computing, whether to print out detailed progress messages, among others. Further improvements to the easyGSA toolbox are planned as follows: the capability to sample different types of distributions among input space parameters, new sensitivity indices for the case of dependent inputs, and inclusion of deep learning models (such as deep neural networks) to surrogate model options.

5.7 Summary and conclusions

In summary, this chapter presented a new methodological framework for constructing advanced surrogate models in order to perform an efficient global sensitivity analysis of complex plant-wide models. Applying the framework, it has also demonstrated four different scenarios covering various kinds of epistemic system uncertainties related to wastewater treatment plants. The key conclusions of the presented work can be summarized as follows:

- The proposed framework enables gaining valuable insights into the performance of complex process systems like WWTPs by combining advanced GSA methods (such as variance decomposition-based Sobol sensitivity analysis) with powerful machine learning algorithms.
- The Sobol sensitivity method provides more reliable sensitivity measures compared to the broadly-used SRC method, especially when highly nonlinear relations exist between the model outputs of interest and the input parameters. When the degree of linearization is low, the SRC method fails to provide correct rankings of significant parameters, and hence should be discouraged in plant-wide applications that feature strong nonlinearities between inputs and outputs.
- Overall, in comparison to MC-based techniques, using surrogate models to carry out plant-wide GSA of WWTP models yields computational gains as large as two orders of magnitude.
- The easyGSA toolbox provided with the framework expedites the process of building highly accurate surrogate models and enables performing computationally heavy model-based analysis such as global sensitivity analysis and beyond.
- Polynomial chaos expansion is a particularly efficient technique that can be used to calculate Sobol sensitivity indices of complex biological systems at a very low computational cost. To further improve the accuracy of the PCE-based indices and to cross-

validate them, the GPR and ANN type meta-models can also be constructed to replace the computationally demanding models in Monte Carlo simulations.

- Influent fractionations, stoichiometric and biokinetic model parameters, as well as hydraulics and design-related parameters, are all found to be influential in estimating performance metrics of WWTP systems when analyzed separately. However, the comparative analysis of all the parameters in scenario 4 indicates that the influent parameters play more important roles for most of the plant KPIs.

Finally, despite not seeing extensive use in plant-wide modeling of WWTPs, the Sobol-based global sensitivity analysis method is a powerful tool for revealing parameter dependence of the underlying complex system behaviors. The method's high computational cost can be further brought down by exploring advanced surrogate modeling techniques, for which the sequential framework provides a useful workflow. As the proposed framework is flexible, the cross-validated surrogate models generated from this framework can also be used for other application purposes than global sensitivity analysis, such as process synthesis and model-based design optimization, for which Chapter 6 will present an example.

Chapter 6

Optimization of WWTP networks under uncertainty

This chapter introduces a novel simulation-based optimization under uncertainty workflow. The previously-identified promising WWTP configurations in Chapter 4 are further optimized for minimizing operational costs subject to influent uncertainty. The chapter also introduces a new solver (MCSKopt) that generically implements the workflow developed within this work. Portions of this chapter are based on the following article.

R. Al, C.R. Behera, K. V. Gernaey, G. Sin, Stochastic simulation-based superstructure optimization framework for process synthesis and design under uncertainty, Comput. Chem. Eng. (under review).

6.1 Introduction

WWTP design is generally associated with significant uncertainties, be it due to natural variability in the influent, measurement errors, uncertain parameters in plant-wide models, etc. As was also found in Chapter 5, recent research shows that the uncertainty in the incoming influent compositions constitutes the most critical source of uncertainty for WWTPs [12,104,129]. As a result, WWTP designs that are feasible under nominal influent conditions might prove infeasible under different realizations of uncertain parameters. Therefore, explicit considerations of uncertainties in the optimal design and operation problems are needed in order to obtain realistic results, which would provide enhanced insights into the robustness of the design solutions. In WWTPs, robustness is defined as the ability of a proposed process design to satisfy the regulated effluent quality limits, even when the values of the uncertain parameters (such as influent compositions) are changed from their nominal values.

The simulation-based optimization under uncertainty approach adopted in this design framework allows for quantification of uncertainties using Monte Carlo simulations during the search for an optimal set of design and operational decisions. By incorporating various techniques from simulation optimization, such as stochastic Kriging-based infill algorithms, as well as techniques from uncertainty and sensitivity analyses literature, a computationally tractable workflow for design optimization of WWTPs under uncertainty was developed. This chapter will introduce the complete workflow and apply it to three case studies of increasing complexity. The remainder of this chapter is organized as follows: The following subsection introduces specific methods employed in the workflow, followed by a description of the adopted optimization under uncertainty workflow. Thereafter, case studies, as well as results and discussions, will be presented. Finally, a generic simulation-based design optimization tool developed in this study will also be introduced before presenting the summary and conclusions.

6.2 Methods

The simulation-based optimization under uncertainty approach developed in this thesis makes uses of several methods from different fields of literature. It is the purpose of this section to introduce the reader to these methods before presenting the workflow. It will be beyond the scope of the thesis to give a complete review or survey of these methods; however, references will be provided to the pertinent literature.

6.2.1 Benchmark method: Exhaustive sampling

The exhaustive sampling method represents the standard Monte Carlo sampling-based approach to uncertain design space exploration. The pseudocode for this method is presented in Table 6.1, as taken from Kusumo et al. [142]. In this method, both the design space of the process parameters and the uncertainty space of uncertain input parameters are discretized by the use of sampling. Depending on the dimensionality of the spaces, different sampling techniques can be employed, — grid sampling for low dimensions (<4), space-filling Latin hypercube sampling or quasi-Monte Carlo sampling for high dimensions are recommended practices [143]. At each design sample, $x_i \in S_d$, the probability map of satisfying active constraints under uncertainty is computed via Monte Carlo simulations performed within the uncertainty space.

Table 6.1: Pseudocode for the exhaustive sampling-based method for design space exploration.

Algorithm 1:	**Design space exploration via exhaustive sampling**
1:	**input**
	$S_d = \{x_i \in X_d : i = 1,...,N_d\}$, and $S_u = \{u_j \in X_u : j = 1,...,N_u\}$
2:	$DS \leftarrow U$
3:	**for all** $x_i \in S_d$ **do**
4:	$DS \leftarrow DS \cup \left\{x_i;\ \mathrm{P}[G(x_i,\cdot) \le 0 \mid S_u]\right\}$
5:	**end for**
6:	**return** DS

Similar to the probability map of Kucherenko et al. [143], other statistical metrics for hedging against uncertainties, such as mean, upper confidence interval of the mean, etc., can also be computed to assist with the selection of optimal designs found in the design space. Table 6.2 tabulates these hedging strategies, which will be used in this work.

Despite being effective, the exhaustive sampling method requires $N_d \times N_u$ times evaluating the underlying simulation model, making it computationally extremely intensive, if not prohibitive, especially for real practical applications, where the computational cost of running the simulation model can be very high. To reduce the required number of simulations and also to increase the efficacy of this method, simulation-based methods described next employ surrogate models that assist with performing a more informed exploration within the design space.

Table 6.2: Uncertainty hedging strategies for satisfying stochastic constraints.

Name	Explanation
Mean	The mean of constraint observations acquired from the Monte Carlo simulations is less than the constraint limit.
UCI95	The upper confidence interval for the mean of constraint observations acquired from the Monte Carlo simulations is less than the constraint limit.
PF80	The probability of the feasibility calculated from the Monte Carlo simulations is higher than 80 %.
MeanPlusSigma	The mean plus one standard deviation of constraint observations acquired from the Monte Carlo simulations is less than the constraint limit.

6.2.2 Stochastic Kriging modeling

Originally introduced by Ankenman et al. [144] as a meta-modeling methodology for stochastic simulations, stochastic Kriging (SK) has later found promising uses in simulation-based optimization frameworks [63,145]. It was developed as an extension to Kriging, which is a commonly-used interpolation-based meta-modeling technique. Since the outputs from a stochastic simulation may contain noise or uncertainty, interpolating outputs from a simulation model as in Kriging becomes

undesirable for stochastic systems. Therefore, SK employs an approximation-based approach that includes a noise model to provide a statistical framework that can handle both intrinsic and extrinsic types of uncertainty in meta-modeling of the stochastic responses [146]. A brief explanation of the SK meta-modeling approach is given in the following, whereas a detailed explanation can be found elsewhere [63,144,147]. The SK model assumes the following functional form:

$$y^{SK}(x) = \mu + M(x) + \varepsilon_r(x) \quad \text{with } x \in \mathbb{R}^k \text{and } r = 1, \dots, m_i. \tag{13}$$

where μ is a constant term denoting the mean, x is the input vector, and k is the number of data points contained within the model. In Ankenman et al. [144], the term $M(x)$ is defined as the *extrinsic* uncertainty, which is the lack of certainty of the meta-model about the simulation model's behavior at regions where no observations are obtained whereas the term $\varepsilon_r(x)$ denotes the *intrinsic* uncertainty which is due to the lack of certainty in the original simulation model's responses. According to the assumptions of Ankenman et al. [144], the former uncertainty is a stationary multivariate Gaussian random field with zero mean and a k-by-k covariance matrix $\sum_M$ while the latter follows a Gaussian distribution with zero mean and variance $Var[\varepsilon(x_i)]$, which is estimated from m_i independent and identically distributed (IID) simulation replications at a sample point x_i. The formula used in SK to estimate this variance reads as follows:

$$\hat{V}(x_i) = \frac{\sum_{r=1}^{m_i} (y_r(x_i) - \bar{y}(x_i))^2}{m_i - 1} \qquad \bar{y}(x_i) = \sum_{r=1}^{m_i} y_r(x_i)/m_i \tag{14}$$

The unbiased SK predictor based on the assumptions above is derived as follows:

$$\hat{y}(x) = \mu + \sum_M(x)'[\sum_M + \widehat{\sum_\varepsilon}]^{-1}(\bar{y} - \mu 1_k) \tag{15}$$

where $\sum_M(x)$ is a k-by-1 covariance vector, $\bar{y}$ is the k-by-1 mean vector, and $\widehat{\sum_\varepsilon}$ denotes the estimated diagonal covariance matrix as follows:

$$\widehat{\sum_\varepsilon} = \begin{bmatrix} \hat{V}(x_1)/m_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \hat{V}(x_k)/m_k \end{bmatrix} \tag{16}$$

The prediction variance of a SK model can also be derived as follows:

$$\hat{s}^2(x) = \tau^2 - \sum_M(x)'[\sum_M + \sum_\varepsilon]^{-1}\sum_M(x) + \delta^T\delta(1_k^T[\sum_M + \sum_\varepsilon]^{-1}1_k)^{-1} \tag{17}$$

where $\delta = 1 - 1_k^T[\sum_M + \sum_\varepsilon]^{-1}\sum_M(x)$ and τ^2 denote the variance of the noise inherent in stochastic simulation. Moreover, the model parameters μ and τ^2 are estimated by maximizing the log-likelihood as derived in Ankenman et al. [144].

6.2.3 Expected improvement

The expected improvement (EI) is a popular infill criterion used in Kriging-assisted simulation-based optimization algorithms. Jones et al. [148] popularized the EI criterion in their well-known Efficient Global Optimization (EGO) algorithm, which was developed for unconstrained deterministic global optimization of computationally expensive black-box problems. The EI provides a metric to quantify the merit of a new sample point where an improvement in the objective function value relative to the currently found minimum objective $f_{\min}$ can be expected. Given n observations $f^n = \{f(x_1), f(x_2), ..., f(x_N)\}$ of an objective function $F(x)$, an improvement at an unvisited point x is defined as follows:

$$I(x) = \max\{f_{\min} - F(x), 0\} \tag{18}$$

The formula for the expected improvement criterion reads as follows:

$$\begin{aligned} E\left[I(x)\,|\,f^n\right] &= \mathrm{E}\left[\max\{f_{\min} - F(x), 0\}\,|\,f^n\right] \\ &= \left(f_{\min} - \mu_f(x)\right)\Phi\left(\frac{f_{\min} - \mu_f(x)}{\hat{s}_f(x)}\right) + s(x)\phi\left(\frac{f_{\min} - \mu_f(x)}{\hat{s}_f(x)}\right) \end{aligned} \tag{19}$$

where Φ denotes the standard normal cumulative distribution function, ϕ denotes the standard normal probability density function, $\mu_f(x)$ and $\hat{s}_f(x)$ respectively refer to the prediction mean and standard deviation. The value of the EI criterion will inflate in those regions of the input space where the prediction mean is lower than the observed minimum or the prediction variance gets higher. Consequently, as in EGO, maximizing the EI criterion at each iteration of an infill optimization algorithm will provide a balance between local exploitation and global exploration of the initial search space where an optimum to the expensive simulation model is sought. One of the main drawbacks of the EI criterion, however, is its inability to accommodate constraints. Further detailed analysis of the pros and cons of this infill criterion can be found in Forrester et al. [149]. Figure 6.1 shows a visual example of the EI criterion calculated on a one-dimensional problem.

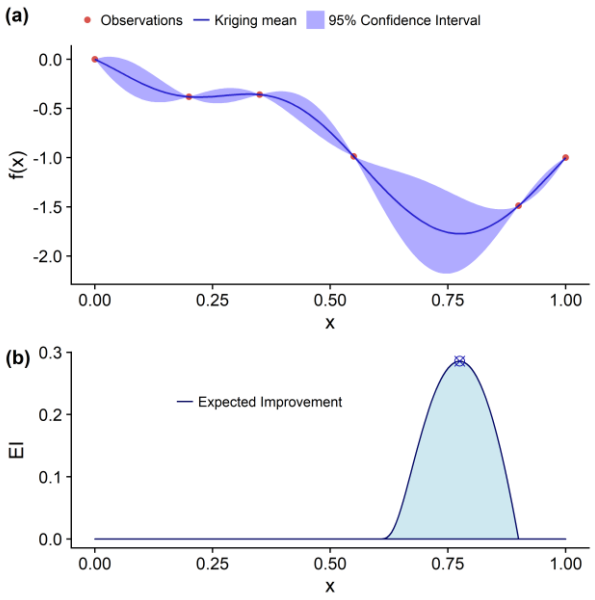

Figure 6.1: Uniformly sampled observations of $f(x) = -x \cdot \left(sin(3\pi x)^2 + 1\right)$ *are used to fit a Kriging model whose mean and 95% confidence intervals are shown in (a). The corresponding expected improvement criterion is computed and plotted in (b) [80].*

6.2.4 Expected quantile improvement (EQI)

As an extension of the original EI criterion for stochastic systems, the concept of quantile improvement was proposed by Picheny et al. [145].

The EI relies on the currently found minimum objective ($f_{\min}$), which is not so intuitive while comparing stochastic responses obtained at different sample points. Therefore, the expected quantile improvement (EQI) compares the β-quantiles of the predictions returned by the Kriging model, and declares a design point among a set of candidates X^n as the currently found best if it has the lowest β-quantile as follows:

$$\begin{aligned} x^* &= \arg\min_{x \in X_n} [q_n(x)] \\ &= \arg\min_{x \in X_n} [m_n(x) + \Phi^{-1}(\beta) s_n(x)] \end{aligned} \tag{20}$$

Then, an improvement is redefined to be the decrease in the lowest β-quantile, between iterations n and $n+1$ as follows:

$$I_n(x) = (\min(q_n(x)) - q_{n+1}(x^{n+1}))^+ \tag{21}$$

where q_{n+1} is the quantile of the kriging updated with x^{n+1}. The corresponding expected quantile improvement criterion for the optimization replaces the current best objective $f_{\min}$ for EI with the current best quantile $q_{\min}$ and reads as follows:

$$EQI_n(x) = (q_{\min} - m_Q(x))\Phi\left(\frac{q_{\min} - m_Q(x)}{s_Q(x)}\right) + s_Q(x)\phi\left(\frac{q_{\min} - m_Q(x)}{s_Q(x)}\right) \tag{22}$$

$$\begin{aligned} q_{\min} &= \min(q_n(x^i)) \\ m_Q(x) &= m_n(x) + \Phi^{-1}(\beta)\sqrt{\frac{\tau^2 s_n^2(x)}{\tau^2 + s_n^2(x)}} \\ s_Q(x) &= \sqrt{\frac{s_n^2(x)^2}{\tau^2 + s_n^2(x)}} \end{aligned}$$

The parameter β takes values $\beta \in [0.5,1)$ and allows for tuning of the level of conservativeness desired on the returned optimum result. For example, a more conservative approach against uncertainty can be selected by setting β close to 1 as doing so will penalize designs with higher prediction uncertainty. Correspondingly, setting β = 0.5 will only compare the means of the Kriging predictions, ignoring their prediction variances. The noise variance term τ^2 in the EQI formulation is the crucial term making it possible to accommodate various kinds of stochasticity in the

observations of an output (especially of constraints). Figure 6.2 shows a visual example of an SK model fitted to noisy observations of a simple one-dimensional function and its corresponding EQI criterion. For further extensions made to the EI criterion for optimizing stochastic systems and their comparison, we refer to the work of Jalali et al. [150].

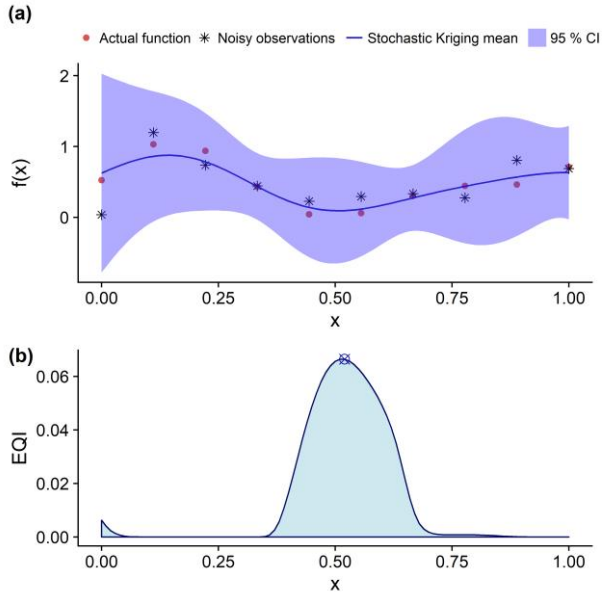

Figure 6.2: Uniformly sampled observations of $f(x) = -x \cdot \left(sin(3\pi x)^2 + 1\right)$ *and its noisy observations are used to fit a stochastic Kriging model, whose mean and 95% confidence intervals are shown in (a). The corresponding expected quantile improvement criterion is computed and plotted in (b).*

6.2.5 Constrained expected improvement

When constraints are present in a simulation optimization problem, the EI criterion is extended using the probability of feasibility (PF), which quantifies the probability of being feasible at an unvisited design point x using the following formulation as defined in [151]:

$$P(G_i(x) \leq 0) = \Phi\left(\frac{0 - \mu_{g_i}}{\hat{s}_{g_i}}\right) \tag{23}$$

where μ_{g_i} and $\hat{s}_{g_i}$ denote prediction mean and standard deviations of the constraint models. Maximizing the EI criterion in conjunction with the PF directs the search towards feasible areas of the input space. To also

accommodate variability in the objective observations, the following multiplicative term is introduced to the EI criterion to obtain the augmented expected improvement (AEI), which aims at a more balanced search for the stochastic systems where uncertainty or noise variances in the objective function observations are taken into account as follows:

$$AEI(x) = EI_f(x) \cdot \left(1 - \frac{\tau_f}{\sqrt{\hat{s}_f^2 + \tau_f^2}}\right) \tag{24}$$

where τ_f^2 represents the noise variance in the objective function at an unvisited point x, which can be estimated through replicated simulations. By combining the probability of feasibility with the augmented expected improvement, a constrained augmented expected improvement (cAEI) criterion is proposed in [63] as follows:

$$cAEI_f(x) = E\left[I_f(x)\right] \cdot P\left(G_i(x) \le 0\right) \cdot \left(1 - \frac{\tau_f}{\sqrt{\hat{s}_f^2 + \tau_f^2}}\right) \tag{25}$$

Maximizing the cAEI infill criterion directs the search for the next sample point to those areas where the probability of feasibility is higher while also maintaining a balance between global and local searches under stochastic observations. As the introduced noise term τ_f only accounts for variance in the objective observations, the cAEI criterion ignores the possible variances of the constraint observations.

6.2.6 Feasibility enhanced constrained expected improvement (FEI)

To incorporate further enhanced knowledge of feasibility into the optimization search, Wang and Ierapetritou [63] proposed a new infill criterion, named Feasibility-enhanced Expected Improvement (FEI), which, compared to the cAEI criterion, included an additive feasibility enhancement term in order to better account for constraint variance. The additive term consists of two multiplied terms: the Expected Quantile

Improvement for Feasibility (denoted as $EQIF_g(x)$) criterion, which promotes feasible points of the constraint model by also taking into account its ordinary Kriging-based variance model, and a penalty factor $P(Y_f(x) \le f^{***})$ for directing the feasibility search towards the promising areas of the objective. The final expression for FEI reads as follows:

$$
\begin{aligned}
P(Y_f(x) \le f^{***}) &= \Phi\left(\frac{f^{***} - \hat{y}_f}{\hat{s}_f}\right) \\
EQIF_g &= \hat{s}_Q \cdot \phi\left(\frac{q^t - \hat{y}_Q}{\hat{s}_Q}\right) \\
&\quad where \\
\hat{y}_Q &= \hat{y}_g + \Phi^{-1}(\alpha) \cdot \sqrt{\frac{\tau_g^2 \hat{s}_g^2}{\tau_g^2 + \hat{s}_g^2}} \\
\hat{s}_Q &= \sqrt{\frac{(\hat{s}_g^2)^2}{\tau_g^2 + \hat{s}_g^2}} \\
q^t &= g^{**} + \Phi^{-1}(\alpha) \cdot \hat{s}_g \\
FEI(x) &= cAEI_f(x) + pEQI_g(x) \\
&= cAEI_f(x) + \hat{s}_Q \cdot \phi\left(\frac{(g^{**} + \Phi^{-1}(\alpha) \cdot \hat{s}_g) - \hat{y}_Q}{\hat{s}_Q}\right) \cdot \Phi\left(\frac{f^{***} - \hat{y}_f}{\hat{s}_f}\right)
\end{aligned}
\tag{26}
$$

For further details of the FEI criterion, readers are referred to the original article [63]. Although it is the only infill criterion that takes into account the constraint model and its uncertainty, the FEI criterion is designed to work with a single constraint model and a constraint noise model. When there is more than one constraint, the authors suggested a lumped approach:

$$g(x) = \max([g_1(x), g_2(x), ..., g_c(x)]) \tag{27}$$

which reduces the feasibility information of multiple constraints into a single value. However, this approach falls short of handling multiple stochastic constraints, each of which might have a different amount of uncertainty associated with it. Besides, the lumped constraint function, $g(x)$, is of highly discontinuous shape, making it very difficult to converge to solutions that lie at the constraint boundaries. To overcome

these issues, we propose the multiple constrained FEI criterion, which is explained in the next section.

6.2.7 Multiple constrained feasibility enhanced expected improvement (mcFEI)

To further account for multiple stochastic constraints and their variances in the optimization of stochastic black-box systems, we proposed to model each constraint and its uncertainty separately. The infill criterion should then account for each constraint model to calculate its probability of feasibility. To that end, we propose to calculate the expected quantile improvement for feasibility $EQIF_{g_i}$ criterion for each constraint separately, using their models for the mean and the uncertainty, and multiply them to direct the infill search towards the feasible points. The final proposed criterion reads as follows:

$$\begin{aligned} mcFEI(x) &= mcAEI(x) + P(Y_f(x) \le f^{***}) \cdot \prod_{i=1}^{c} EQI_{g_i}(x) \\ mcAEI(x) &= E[I_f(x)] \cdot \left(1 - \tau_f / \sqrt{\tau_f^2 + \hat{s}_f^2}\right) \cdot \prod_{i=1}^{c} P[G_i(x) \le 0] \end{aligned} \tag{28}$$

where the term $P[G_i(x) \le 0]$ calculates the probability that a new design point x is feasible according to the constraint model $\hat{g}_i(x)$. In summary, $mcFEI$ provides a means to explicitly consider the knowledge of feasibility of multiple stochastic constraints while also searching for improvements in the objective value.

6.2.8 Monte Carlo-based uncertainty quantification

Monte Carlo methods constitute a broad class of computational algorithms that rely on random sampling and simulation in order to estimate the statistical properties of a response of interest. They found extensive uses in a variety of science and engineering applications, including propagation of input uncertainty, numerical integration, benchmarking of robust optimization results, etc. [152]. Uncertainty quantification is a multi-disciplinary field that has been on the rise in recent decades. For an extensive discussion of methods developed in this

field, the recent book of Ghanem et al. can be consulted. [153]. Among other methods, Monte Carlo methods can be used to quantify uncertainties arising in different domains. The basic principle behind these methods is that provided a sufficient number of random samples, the expected value of a random variable can be approximated with an arbitrary level of accuracy by taking the empirical mean. The procedure for Monte Carlo-based uncertainty propagation is as follows. The uncertainties that are present in the input parameters of a simulation model are characterized by statistically representative distributions, which are then sampled with, among others, a space-filling sampling algorithm such as Latin hypercube sampling (LHS). To obtain a complete distribution of the model output uncertainty, Monte Carlo simulations are performed, and the results are represented using statistical mean, standard deviation, percentiles, and cumulative distribution functions. Irrespective of the type of the model used, the Monte Carlo procedure for uncertainty quantification remains the same, making the Monte Carlo method extremely suitable for large scale applications, without the problem of combinatorial growth in computational demand, as is the case in other uncertainty quantification methods, such as stochastic programming [79].

Regarding applications to wastewater treatment plants, the Monte Carlo-based uncertainty analysis methodology adopted in the case studies of this work follows from the work of Sin et al. [12], which involves three major steps: (1) specification of the input uncertainty (e.g., influent fractionations); (2) sampling of the uncertain input space; (3) propagation of the input uncertainty to the design objective (e.g., operating cost index, etc.). In searching for a robust design for WWTPs, integrating the uncertainty analysis into the design optimization workflow provides a hedge against natural fluctuations that might occur in, for example, influent compositions. Hence, a more robust evaluation of design performances under such disturbances can be made possible with the use of these Monte Carlo methods.

6.2.9 Surrogate-based global sensitivity analysis

As was extensively discussed in Chapter 5, a global sensitivity analysis using surrogate models provides a computationally efficient alternative for calculating variance decomposition-based Sobol sensitivity indices. In the third case study of this chapter, we apply three different types of surrogate models, namely polynomial chaos expansions (PCE), Gaussian process regression (GPR), and artificial neural networks (ANN). Detailed explanations of these surrogate models and how they can be used for global sensitivity analysis can be found in Chapter 5. The variance decomposition-based Sobol sensitivity indices quantify both the individual (S_i) and the total effects (S_{Ti}) of input parameters as well as the effects due to the interactions among input parameters. The PCE-based Sobol indices are calculated from the PCE model coefficients, whereas the GPR and ANN-based calculation of these indices uses the surrogate models instead of complex plant-wide WWTP simulation models and follows the Monte Carlo procedure for global sensitivity analysis as discussed in Chapter 5. In addition to the PCE, GPR and ANN-based Sobol sensitivity indices, an ensemble approach is also used to return indices that are calculated by taking the mean of the indices returned by the three surrogate models. In doing so, it is expected that the specific surrogate model-based deviations in the results of Sobol indices are minimized.

6.3 Simulation-based optimization under uncertainty workflow

The overall workflow for simulation-based optimization under uncertainty is depicted in Figure 6.3. It builds on the generic workflow for surrogate model-based simulation-based optimization, which usually consists of three steps: the creation of surrogate models based on an initial set of design samples, adaptive sampling with infill optimization, and termination. However, in addition to this generic workflow, the workflow developed in this study uses extra steps, such as uncertainty and sensitivity analyses, which are explained in the following subsections.

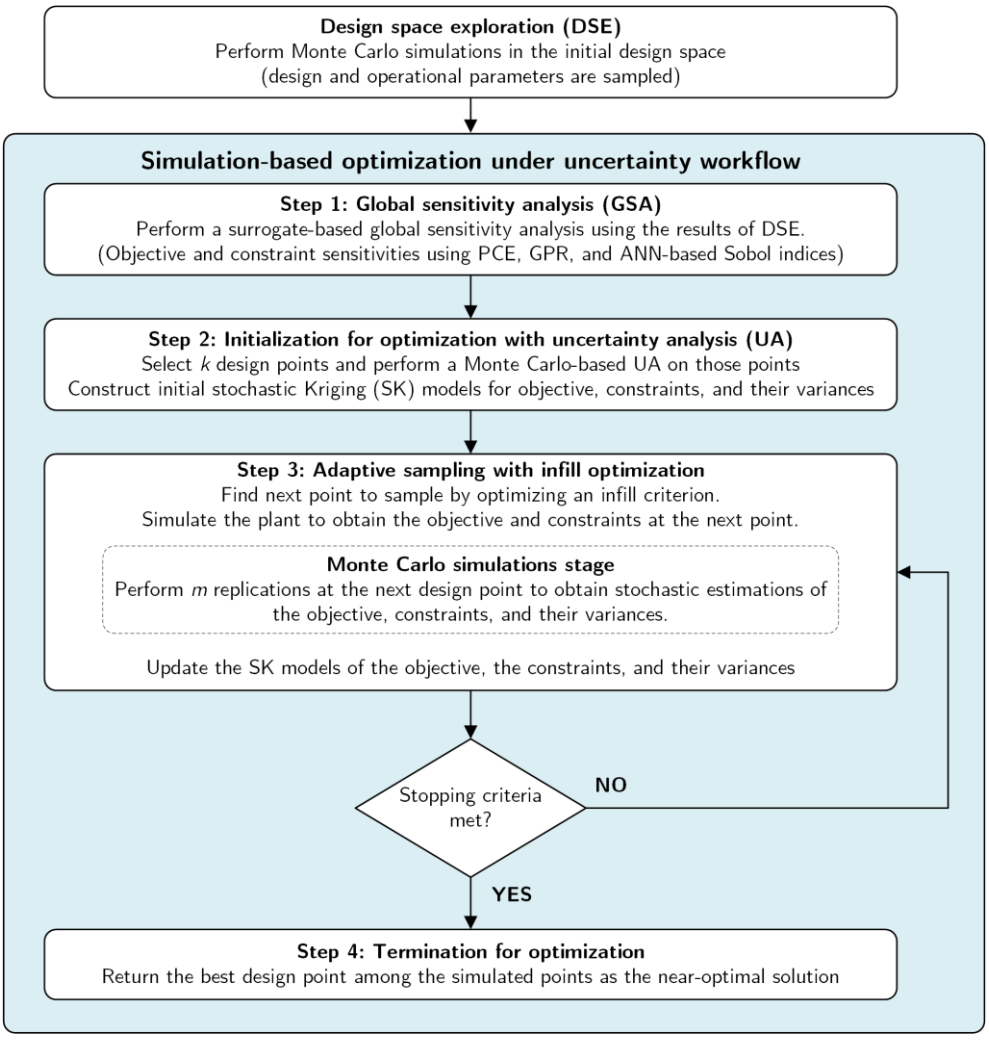

Figure 6.3: General overview of the workflow developed for simulation-based optimization under uncertainty.

6.3.1 Step 1: Global sensitivity analysis

The Monte Carlo-based design space exploration performed in Chapter 4 used the entire set of design and operational parameters within a flowsheet to explore its design space. However, as the design optimization under uncertainty comes at the cost of increased computational effort, before investing such efforts, it needs to be ensured that the objective of the optimization problem as well as its constraints are sensitive to the decision variables, i.e., such variables significantly contribute to the optimization objective and constraints. A prior global sensitivity analysis step is, therefore, added to the workflow to ensure strong relationships among the objective and decision variables of the optimization problem. Moreover, the results of the design space exploration can readily be used to perform a surrogate-based global sensitivity analysis. In doing so, only the key decision variables identified by global sensitivity analysis are made subject to optimization under uncertainty, reducing the size of the optimization problem.

6.3.2 Step 2: Initialization for optimization with uncertainty analysis

The initialization step for optimization requires the selection of initial design points, to which initial stochastic Kriging models will be fitted. This initial dataset usually consists of a relatively low number of initial design points, proportional to the number of decision variables ($10d$ for d decision variables). In the literature, space-filling LHS designs effectively spanning the entire input space are widely employed to select these initial points [63,154]. However, in constrained simulation optimization, an issue arises as how to find a feasible design point in the initial dataset, which is not necessarily guaranteed by the use of pure LHS. This is due to the use of expected improvement-based infill criteria, which necessitates the inclusion of at least one feasible design point within the initial set of samples so that a relative improvement can be calculated. With regard to this issue, Boukouvala and Ierapetritou [155] suggested a feasibility

analysis method using adaptive sampling with Kriging model, whereas Boukouvala and Floudas [156] employed a two-step strategy, which included obtaining a large LHS sample in the first step and filtering out infeasible points in the next. In our workflow, we combine both approaches as follows. Feasible design samples found from the previously performed Monte Carlo based design space exploration step are selected for inclusion in the initial set of samples. If they are less than the required initial sample size ($10d$), the remaining samples are drawn using a space-filling LHS design. If there is still no feasible design sample found in the initial dataset, then the workflow proceeds with the next adaptive sampling step, which switches the infill optimization objective to find a feasible point by the use of the expected quantile improvement for feasibility criterion (EQI_f), as was also suggested in [63].

Before proceeding with the adaptive sampling step for optimization, in order to provide a framework for taking into account the possible uncertainty in the objective and constraint responses of a black-box system, Monte Carlo simulations are integrated into this workflow. To that end, a Monte Carlo-based uncertainty analysis as described above is performed for each design sample of the initial dataset. In doing so, instead of deterministically evaluating design performances at the nominal values of the uncertain parameters, one obtains the reliable estimates of the magnitude of the output uncertainties at different design points, which may all have different magnitudes throughout the entire domain of the design space. These uncertainty estimates of the objective and constraints are then fed into the stochastic Kriging models, which are later used to find infill design points with a higher likelihood of improving the objective under uncertainty. Therefore, the choice of using stochastic Kriging models as the surrogate in this workflow is due to their ability to take into account uncertainty information during the model building process.

6.3.3 Step 3: Adaptive sampling with infill optimization

The third step in the workflow involves adaptive sampling with internal infill optimization. As engineering design is usually associated with a variety of challenging constraints (be it technical, economical, safety, or environmental), the infill strategy employed in adaptive sampling plays the key role for the convergence toward a feasible near-optimal design in constrained simulation optimization problems. The challenge gets exacerbated when multiple constraints are present. There are several different ways of handling such constraints. One of the most common approaches in the literature is to use penalty functions, which inflate the objective function value whenever a design violates one or more constraints. However, this approach is found to result in severe discontinuities in the objective function surface, which makes it very difficult to converge to any final design that lies on the constraint boundaries and it does not take into consideration which and how many constraints are violated at any infeasible design point [149]. Another approach is to lump multiple constraints into one and model the resulting value with a surrogate in an infill optimization [63]. However, this approach has the drawback that the resulting constraint model is of no real system response and has discontinuities caused by the violation of various constraints. Besides, this approach, as well as the former one, do not consider the effect of uncertainties that might accompany those multiple constraints and affect their violations. To address this issue in this work, modeling of each constraint and its uncertainty separately with stochastic Kriging and ordinary Kriging models, respectively, is proposed using the multiple constrained FEI (mcFEI) infill criterion. In the current workflow, therefore, this infill criterion is employed as the primary infill objective, and its performance is compared to that of the FEI infill criterion.

The adaptive sampling with infill optimization continues for a pre-defined number of iterations. In each iteration, a new design point is suggested as the next point (also referred to as infill point) with the

greatest expected improvement of the objective under constraints. In each iteration, the infill design point is then simulated using the expensive black-box simulation model, and responses of the design objectives and constraints are appended to the dataset. By using this updated dataset, the surrogate models are retrained at each iteration, and these updated models are then used for a new infill optimization toward the more promising areas of the search space. Also integrated into each iteration of this step is the uncertainty propagation of the uncertain parameters, which is accomplished via Monte Carlo simulations. This increases the computational load of the overall workflow proportionally to the sample size of the integrated Monte Carlo simulations. To overcome this issue, we set the size of Monte Carlo sampling (equal to 100) the same as the number of available CPU cores, which effectively reduces the computational time.

6.3.4 Step 4: Termination

The last step in the simulation optimization workflow is to terminate the infill optimization search. As the simulation optimization problems often include computationally expensive simulation models, a termination criterion is usually set based on the available computational budget. In doing so, the number of calls made to the expensive simulation model is effectively restricted to a priori selected budget. This step also identifies and returns the best-found feasible solution as the near-optimal solution, by comparing among the simulated design points.

6.4 Case study 1: An illustrative test problem

To illustrate the benefits of the simulation-based approach, we first consider an illustrative test problem named Sasena, which is a widely-studied benchmark problem in simulation optimization literature [157]. With its two design variables and three active design constraints, Sasena readily allows for design space visualization and also provides a case that is well representative of actual engineering design optimization problems subject to multiple stochastic constraints.

6.4.1 Mathematical formulation of Sasena problem

In its deterministic form, the mathematical formulation of this problem reads as follows:

$$\begin{aligned} \min_{x} \quad & f(x) = -(x_1-1)^2-(x_2-0.5)^2 \\ s.t. \quad & g_1(x) = ((x_1-3)^2+(x_2+2)^2)e^{-x_2^7}-12 \\ & g_2(x) = 10x_1+x_2-7 \\ & g_3(x) = (x_1-0.5)^2+(x_2-0.5)^2-0.2 \\ & 0 \le x_i \le 1 \text{ for } i=1,\ 2 \end{aligned} \tag{29}$$

To make this problem an optimization under uncertainty problem, we arbitrarily introduce four uncertain parameters entering the expressions of both the objective and the constraints, as shown below.

$$\begin{aligned} \min_{x} \quad & f(x) = -u_1(x_1-1)^2-(x_2-0.5)^2 \\ s.t. \quad & g_1(x) = (u_2(x_1-3)^2+(x_2+2)^2)e^{-x_2^7}-12 \\ & g_2(x) = 10u_3x_1+x_2-7 \\ & g_3(x) = (x_1-0.5)^2+u_4(x_2-0.5)^2-0.2 \\ & 0 \le x_i \le 1 \text{ for } i=1,\ 2 \\ & \ln(u_j) \approx \mathcal{N}(\mu,\sigma) \text{ for j}=1,..,4 \end{aligned} \tag{30}$$

We solve this problem using both the benchmark method based on exhaustive sampling and also the simulation-based method developed in this work. In the following, the obtained results are compared and analyzed.

6.4.2 Solution using the Exhaustive Sampling method

Firstly, the exhaustive sampling-based benchmark method is employed to obtain the best feasible solution under uncertainty. To that end, the two-dimensional design space of the problem is sampled using Latin hypercube sampling with a varying number of samples from a very coarse (10) to a very fine (10^5) design. Each sample in the sampling design is a candidate design. Hence, an uncertainty analysis is performed for each of these samples using vectorized Monte Carlo simulations, thanks to MATLAB's powerful vector operations, with an additional 1000 Latin hypercube samples. Figure 6.4 shows the LHS sampling of the problem's design space, which is uniformly sampled between the lower and upper bounds of the decision variables.

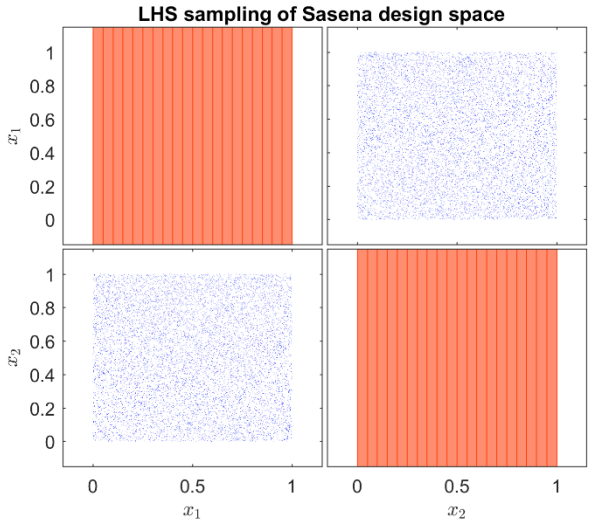

Figure 6.4: LHS sampling of Sasena design space. The number of samples is 10000. Diagonal elements show the uniform distribution of the sampling.

On the other hand, the input uncertainty space of the problem is arbitrarily defined with two different uncertainty scenarios, each of which has its distribution and scaling factor, as tabulated in Table 6.3. The resulting uncertainty space is sampled with 1000 LHS samples, as shown in Figure 6.5.

Table 6.3: Input uncertainty scenarios. Uncertain parameters follow different distributions scaled with a scaling factor.

Scenario	Distribution	Mean (μ)	Coefficient of variation (σ/μ)	Scaling factor
1	Normal	1	0.50	1/2
2	Lognormal	1	0.50	1/7

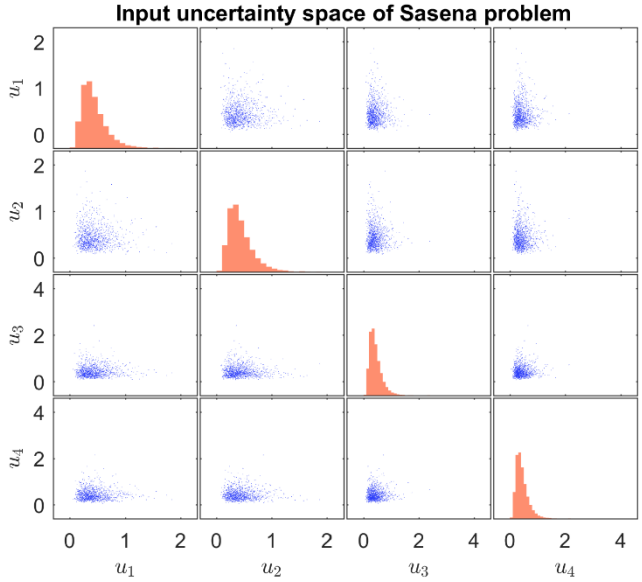

Figure 6.5: The input uncertainty space of the Sasena problem. The space is sampled with 1000 Latin hypercube samples, following a lognormal distribution.

Table 6.4: The average results of the optimum value of the objective and the location of the optimum design under uncertainty obtained using the exhaustive sampling method. Uncertainty hedge is set to MeanPlusSigma.

Number of design samples	Normal uncertainty		Lognormal uncertainty	
	Optimum value of the objective	Location of the optimum (x)	Optimum value of the objective	Location of the optimum (x)
10	-0.395	[0.310, 0.662]	-0.399	[0.280, 0.806]
100	-0.452	[0.204, 0.376]	-0.431	[0.238, 0.482]
1000	-0.464	[0.197, 0.465]	-0.449	[0.268, 0.445]
10000	-0.467	[0.202, 0.447]	-0.453	[0.287, 0.444]
100000	-0.468	[0.200, 0.453]	-0.454	[0.288, 0.479]

For each different sampling design, the results are repeated 50 different times so as to alleviate the effects of the randomness in the

sample locations. Table 6.4 shows the averaged results of the optimum and its location, which are obtained from these 50 different repetitions. As expected, increasing the number of design samples results in improved objective as the finer sampling designs cover the design space more rigorously. However, the results of two different uncertainty scenarios do not differ considerably with slightly higher optimum results being obtained for the lognormal uncertainty scenario. This is mainly because of the scaling factor applied to the uncertain parameters, which ensures the existence of a feasible region within the design spaces of both uncertain scenarios.

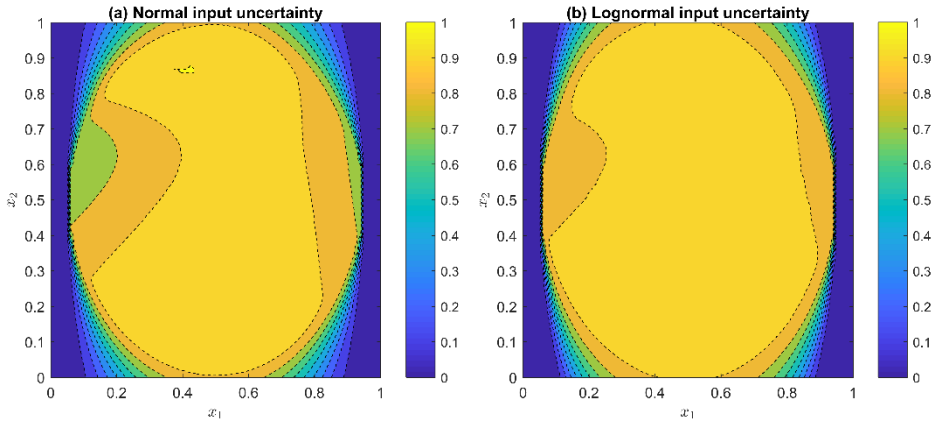

Figure 6.6: The probabilistic design spaces calculated for the Sasena problem. The uncertain input parameters follow a normal distribution in (a), and a lognormal distribution in (b).

At each design sample, the feasibility is decided based on the results of the uncertainty analysis by adopting a conservative hedging strategy. Essentially, a design is declared feasible if the mean plus the standard deviations of each constraint satisfies the corresponding constraint limits. For example, Figure 6.6 calculates the probability of feasibility at each design sample and draws the boundaries between different regions of feasibility. As can be seen from the figure, the shape of the feasible regions in both scenarios appears similar, with the edges of the design spaces being infeasible areas. Similarly, the other hedging strategies, as was tabulated in Table 6.2, are also computed, and the obtained results are shown in Figure 6.7 and Figure 6.8. Using only the mean of the

uncertainty simulations to decide feasibility yields optimum values that are lower (- 0.549 for 10^5 design samples), whereas more conservative approaches uplift the optimum towards the positive axis.

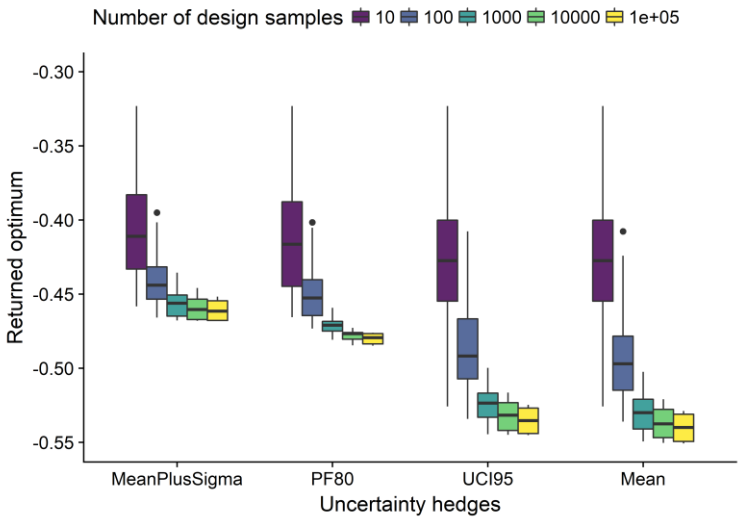

Figure 6.7: Uncertainty hedging strategies are compared along with the increasing number of design samples. The returned optimum shifts upwards for more conservative hedging strategies.

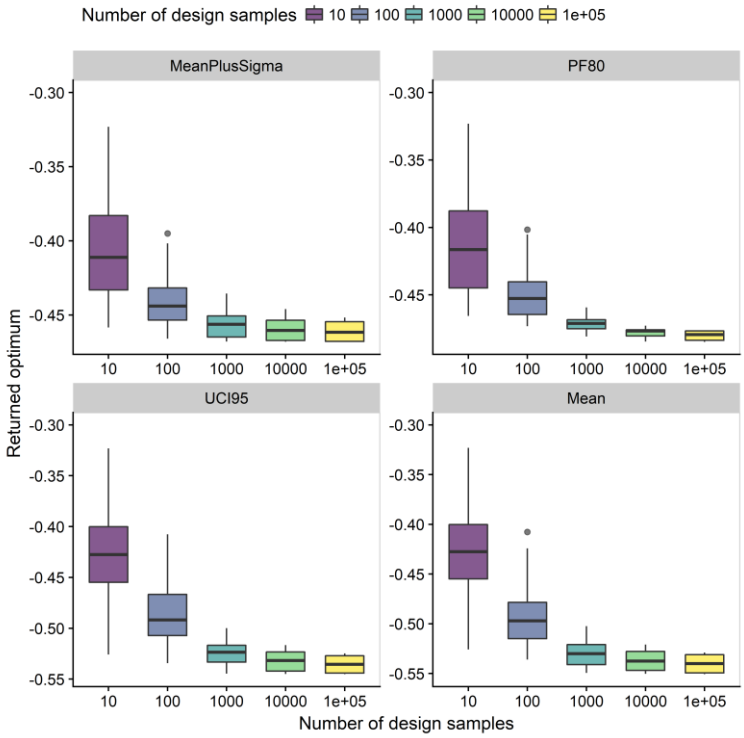

Figure 6.8: The results of the exhaustive sampling method for different sizes of design samples and also for different types of hedging strategies. Each boxplot represents the results obtained from 50 different independent runs. The narrower the boxplot, the better the convergence.

Concerning the total computational cost, finding optimum under uncertainty using the exhaustive sampling-based approach described in this work required 100 million calls to the underlying simulation model for the finest sampling design. This exemplifies the limiting problem behind the exhaustive sampling-based methods, which are known to suffer from the rapid growth of the computational costs (known as the curse of dimensionality) as the size of the design space increases [143]. This massive computational demand makes these methods often impractical for real engineering design applications, where the time it takes to perform a single simulation can be several minutes, if not hours. In order to circumvent this issue, we resort to the simulation-based approaches using the previously described methods.

6.4.3 Solution via simulation-based optimization

In this subsection, the same Sasena optimization under uncertainty problem is solved with the in-house developed simulation-based optimization solver (MCSKopt). Two infill criteria, namely FEI and mcFEI, are comparatively employed to solve the resulting optimization under uncertainty problem along with four different hedging strategies against the uncertainties. Each test is repeated 50 times with different random initial design points to start with, and the optimization is terminated when a total of 100 samples are taken from the design space (10 initial samples, 90 adaptive iteration samples). The obtained results are visualized with box plots, as shown in Figure 6.9.

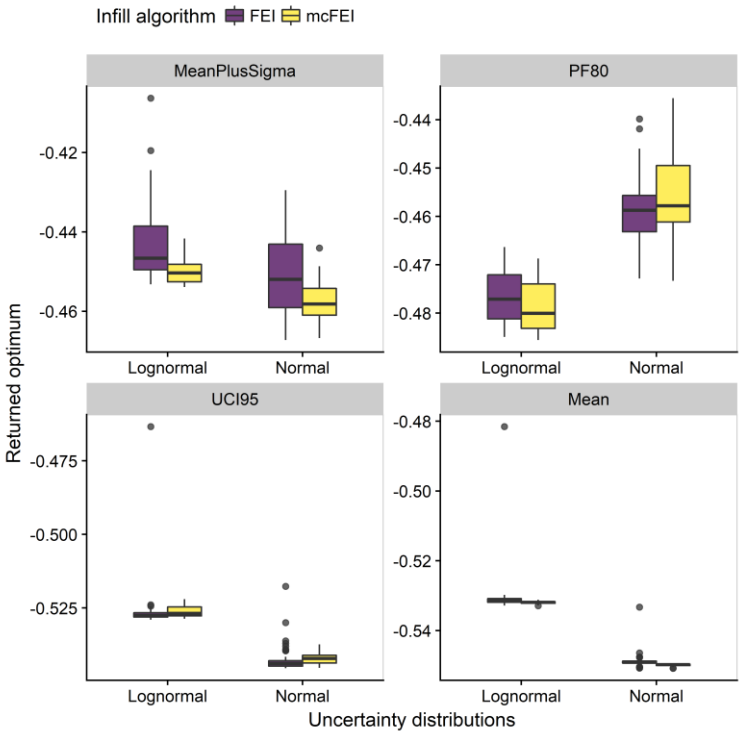

Figure 6.9: Results of simulation-based optimization for the uncertain scenarios of Sasena problem. Each boxplot represents the results obtained from 50 different independent runs. The results are in line with the exhaustive sampling-based method.

In general, both infill algorithms performed comparably well and returned optimum values that are in agreement with what is achievable by the benchmark method of exhaustive sampling approach. The results of both infill algorithms for mean hedging are indistinguishable, whereas

hedging using mean and the standard deviation results in slightly better optimum values returned by the newly proposed mcFEI criterion compared to the values returned by the FEI criterion. When comparing different uncertainty distributions, the returned optimum values are slightly higher for lognormal uncertainty except for the PF80 hedging strategy, which requires 80% of uncertainty simulations to satisfy the constraint limits. This could be explained by the fact that PF80 is the only hedging strategy that does not use the mean or the standard deviations to decide the feasibility of a design sample. Instead, it uses direct observations of the Monte Carlo uncertainty simulations. Table 6.5 shows the averaged returned optimum values and their locations for both infill algorithms. The optimum values are very close to each other despite at different locations, especially for the x_2 variable. This could be because of the response characteristics of the problem at hand, especially of its constraints, which are more sensitive to the variations in x_1 than in x_2 dimension. Indeed, the response surface plots of the objective and the lumped constraints (as shown in Figure 8.3) of the nominal Sasena problem concurs with this reasoning.

Table 6.5: Results of simulation-based optimization for Sasena problem, averaged over 50 independent repetitions, reported for the MeanPlusSigma hedging strategy.

	Normal uncertainty		**Lognormal uncertainty**	
Infill algorithm	**Optimum value of the objective**	**Location of the optimum (x)**	**Optimum value of the objective**	**Location of the optimum (x)**
FEI	-0.4507	[0.2749, 0.9316]	-0.4424	[0.3037, 0.9776]
mcFEI	-0.4577	[0.2469, 0.4445]	-0.4500	[0.3002, 0.6745]

Although this case study highlights the effectiveness of the simulation-based solver in returning design solutions that are in agreement with the computationally expensive benchmark method, it does not clearly show the added value created by the multiple constraint modeling implemented in the mcFEI criterion. To that end, we apply the two algorithms for a higher dimensional case study in the following section.

6.5 Case study 2: A high dimensional test problem

In order to further investigate both the reliability and the performance comparison of both infill criteria in higher-dimensional problems, we solve the Rosen Suzuki test problem, taken from Carpio et al. [158].

6.5.1 Mathematical formulation of Rosen Suzuki problem

In its deterministic form, the Rosen Suzuki problem has four dimensions and a mathematical formulation as follows:

$$\begin{aligned}
\min_{x} \quad & f(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4 \\
s.t. \quad & g(x) = [g_1(x),\, g_2(x),\, g_3(x)] \leq 0 \\
& g_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 - 8 \\
& g_2(x) = x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 - 10 \\
& g_3(x) = 2x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 - 5 \\
& -3 \leq x_i \leq 3 \text{ for } i = 1,..,4
\end{aligned} \tag{31}$$

Similar to the previous case study, we introduce uncertain parameters in the expressions of the objective and the constraints along with scaling applied to the constraints as follows:

$$\begin{aligned}
\min_{x} \quad & f(x) = x_1^2 + x_2^2 + 2u_1 x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4 \\
s.t. \quad & g(x) = [g_1(x),\, 100g_2(x),\, 1000g_3(x)] \leq 0 \\
& g_1(x) = x_1^2 + u_2 x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 - 8 \\
& g_2(x) = x_1^2 + 2x_2^2 + x_3^2 + 2u_3 x_4^2 - x_1 - x_4 - 10 \\
& g_3(x) = 2x_1^2 + u_4 x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 - 5 \\
& -3 \leq x_i \leq 3 \text{ for } i = 1,..,4 \\
& u_j \approx \mathcal{N}(\mu, \sigma) \text{ for j} = 1,..,4 \text{ and } \mu = 1,\ \sigma = 0.25
\end{aligned} \tag{32}$$

Carpio et al. [158] give the nominal solution for this problem as -44 obtained at $x^* = [0\,,\,1\,,\,2\,,\,-1]$. However, the optimum solution for the uncertain version of the problem may or may not lie close to the nominal solution. In order to obtain a benchmark solution, we apply the exhaustive sampling method to the uncertain problem in the following.

6.5.2 Solution using the Exhaustive Sampling method

Similar to the previous case study, we sample the design space of the problem with the incrementally increased number of LHS samples. The obtained results from 50 different independent replications are visualized in Figure 6.10. As can be seen from the figure, the best convergence is achieved with 10^5 design samples, whereas the results of coarser sampling designs vary over a wide range. The benchmark solution is, therefore, taken from the finest design and tabulated in Table 6.6. As each design sample is evaluated with 10^3 Monte Carlo simulations for uncertainty analysis, the total number of calls made to the underlying model to achieve this solution is 100 million, making it an extremely costly solution from a computational standpoint.

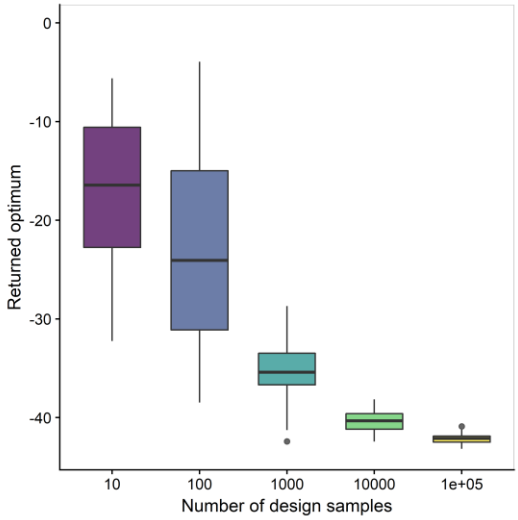

Figure 6.10: The results of the exhaustive sampling method for different sizes of design samples and also for different types of hedging strategies. Each boxplot represents the results obtained from 50 different independent runs. The narrower the boxplot, the better the convergence.

Table 6.6: The average results of the optimum value of the objective and the location of the optimum design under uncertainty obtained using the exhaustive sampling method. Uncertainty hedge is set to MeanPlusSigma.

Number of LHS design samples	Total number of calls to the model	Optimum value of the objective	Location of the optimum (x)
10^5	10^8	-42.152	[-0.026, 0.841, 1.951, -0.914]

6.5.3 Solution via simulation-based optimization

In this subsection, the uncertain Rosen Suzuki problem is solved using the simulation-based solver with 20 initial LHS design samples taken from the four-dimensional design space. These samples are then uncertainty analyzed with the Monte Carlo simulations, and the obtained responses of the objective and the constraints are fed to the solver. To alleviate the effects of randomness in the locations of the initial design samples, 50 independent repetitions are made using the solver with two infill criteria, FEI and mcFEI. Figure 6.11 shows the progress of the returned optimum solution in 50 iterations in the design space, compared between the two infill criteria. As can be seen, the mcFEI criterion consistently outperforms the FEI criterion in returning better optimum solutions that satisfy the three stochastic constraints of the problem. This can be explained with the separate modeling of each constraint response and their uncertainty, as implemented in the mcFEI criterion, instead of the lumped approach of the FEI criterion. As each constraint has different scales, separate modeling helps to better identify the uncertain characteristics of the problem's output space and hence leads to the better performance of the mcFEI criterion.

Besides the improved performance over the FEI criterion, solving the uncertain Rosen Suzuki problem using the mcFEI criterion also unequivocally demonstrates the advantages of the simulation-based approach over the exhaustive sampling-based approach. In most of the 50 repetitions made with the solver, the simulation-based approach with the mcFEI criterion using only 50 design samples yields better optimum values of the objective, as tabulated in Table 6.7, than the exhaustive sampling approach using 10^5 design samples. In fact, the total of number calls made to the model is significantly decreased from 100 million to 50 thousand. This is mainly because of the increased dimension of the design space, which requires an exponentially increasing number of design samples to converge for the exhaustive sampling method.

Table 6.7: Results of simulation-based optimization for the uncertain Rosen Suzuki problem. Results are averaged over 50 independent repetitions.

Infill algorithm	Number of LHS design samples	Total # of calls to the model	Optimum value of the objective	Location of the optimum (x)
FEI	50	50×10^3	-24.915	[-0.368, 0.607, 1.468, 0.128]
mcFEI	50	50×10^3	-42.701	[0.028, 0.608, 1.996, -0.996]

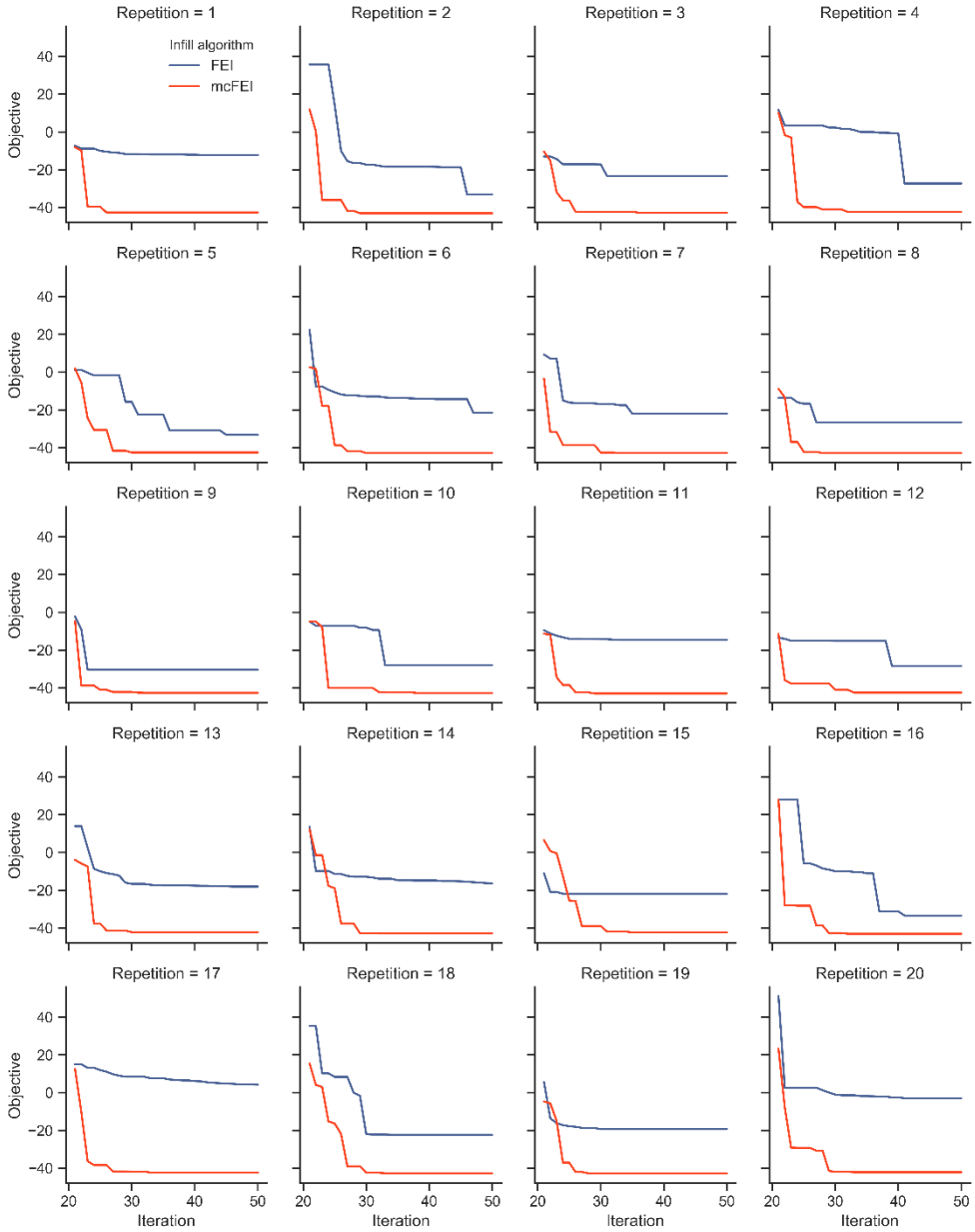

Figure 6.11: Comparison of FEI and mcFEI infill criteria for the uncertain Rosen Suzuki problem. The multiple constraint modeling of the mcFEI criterion leads to better performance over the FEI criterion.

6.6 Case study 3: the BSM2 plant optimization under uncertainty

After demonstrating the motivations behind the simulation-based approach with the simple test problems, we now turn to the engineering design problems subject to uncertainties. To that end, this case study builds on the results of the first case study presented in Chapter 4 and illustrates the application of the simulated-based optimization under uncertainty workflow. In Chapter 4, a design space optimization was performed to identify the most promising plant layout configurations, which were configuration 2, 4, and 6. The further design optimization of those layouts, as schematically shown in Figure 6.12, under influent uncertainty considerations will be presented in the following subsections.

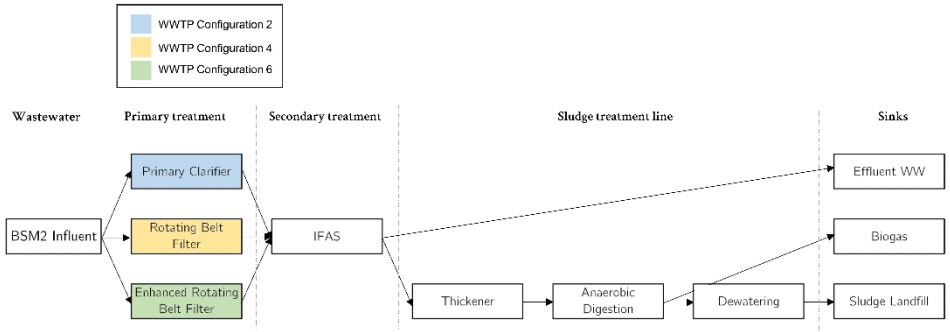

Figure 6.12: Schematic layouts of three promising WWTP configurations subject to design optimization under influent uncertainty.

6.6.1 Problem statement for design optimization under uncertainty

As discussed earlier, reducing the operating costs of a WWTP is of increasing importance for the water sector. The plant-wide WWTP simulation models provide a means of credibly estimating those costs as well as the effluent water quality. Such plant-wide models can also be used to improve the performance of the WWTP through a systematic optimization workflow. However, as was also found in Chapter 5, the uncertainties present in the incoming wastewater compositions significantly affect the effluent water quality, and such effects need to be

taken into consideration when searching for an optimal set of design and operational decisions in WWTPs. From this perspective, an informal statement of the design optimization problem in this case study is given as follows: Given a plant-wide model of a WWTP, find the optimal design and operational decision variables minimizing the plant's operational cost index (OCI), while meeting the effluent regulations, under the influent pollutant loadings uncertainties. A more formal optimization problem definition can then be formulated as follows:

$$\begin{aligned} \min_{x} \quad & OCI(x) = AE + 3\cdot SP + 3\cdot EC + ME - 6\cdot MP + HE^{net} \\ s.t. \quad & COD_{eff}(x) \le 100 \text{ g COD m}^{-3} \\ & TN_{eff}(x) \le 15 \text{ g N m}^{-3} \\ & SNH_{eff}(x) \le 5 \text{ g N m}^{-3} \\ under \quad & \left.\begin{matrix} SI_{inf} \\ SS_{inf} \\ XI_{inf} \\ XBH_{inf} \end{matrix}\right\} \approx \mathcal{N}(\mu,\sigma) \end{aligned} \tag{33}$$

where x represents the optimization decision variables, AE is aeration energy consumption, SP is sludge production, EC is external carbon addition, ME is mixing energy, MP is methane production, HE^{net} is net heating energy needed to heat the sludge in the anaerobic digester. Following the simulation procedure for the BSM2 plant, as described in [11], these operational costs are calculated separately for each specific sub-unit, and a plant-wide sum averaged per day is returned. The effluent water quality limits are introduced, as shown above. Four influent fractionations, namely soluble inert organic matter (SI_{inf}), readily biodegradable substrate (SS_{inf}), particulate inert organic matter (XI_{inf}), and active heterotrophic biomass (XBH_{inf}), are allowed to be uncertain around their nominal values (μ) as reported in Gernaey et al. [11] with a coefficient of variation (σ / μ) of 25 %.

6.6.2 Identifying key decision variables with global sensitivity analysis

The optimization decision variables are decided by the use of the surrogate-based global sensitivity analysis methodology. Firstly, PCE, GPR, and ANN type surrogate models were developed using the Monte Carlo simulations datasets, which were generated from the design space exploration performed for 6 configurations in case study 1 of Chapter 4. These datasets were outlier treated to prevent developing surrogate models that are overfitting to the unlikely observations. This ensures a better surrogate model quality before proceeding with global sensitivity analysis. Then, variance decomposition-based Sobol sensitivity indices were calculated using each of these surrogate models, following the methodology describer earlier. The PCE-based Sobol sensitivity indices were generated using the UQLab software [124], whereas the GPR and the ANN-based indices were generated using the easyGSA tool [129].

As seen in Figure 6.13, the sensitivity indices results obtained from different surrogates are in good agreement with one another, with the GPR indices giving slightly lower sensitivities for the less sensitive parameters, compared to the PCE and ANN-based indices. Among the investigated parameters, the OCI is found to be the most sensitive to the operational parameter external carbon (EC) in MLE configurations, as the addition of external carbon directly increases the operational costs of a plant, whereas in IFAS configurations, the nitrogen loading rate (NLR) of the IFAS unit affects the OCI the most.

Before carrying out design optimization, sensitivity indices calculated from the three surrogates are used to return the ensemble indices, and a threshold for sensitivity was set to 1% to eliminate design and operational parameters that are not influential in terms of their effect on the design objective. This approach ensures that there is a strong relationship between the decision variables and the optimization objective, and reduces the unnecessary dimensionality in the optimization search space.

Moreover, dimensionality reduction is also a good practice in developing machine learning-based models, especially when the size of the set of training samples is limited. The key optimization decision variables obtained after this elimination are shown in Table 6.8 for the three promising layouts (as identified in Chapter 4), for which further computational efforts will be invested in performing a simulation-based optimization.

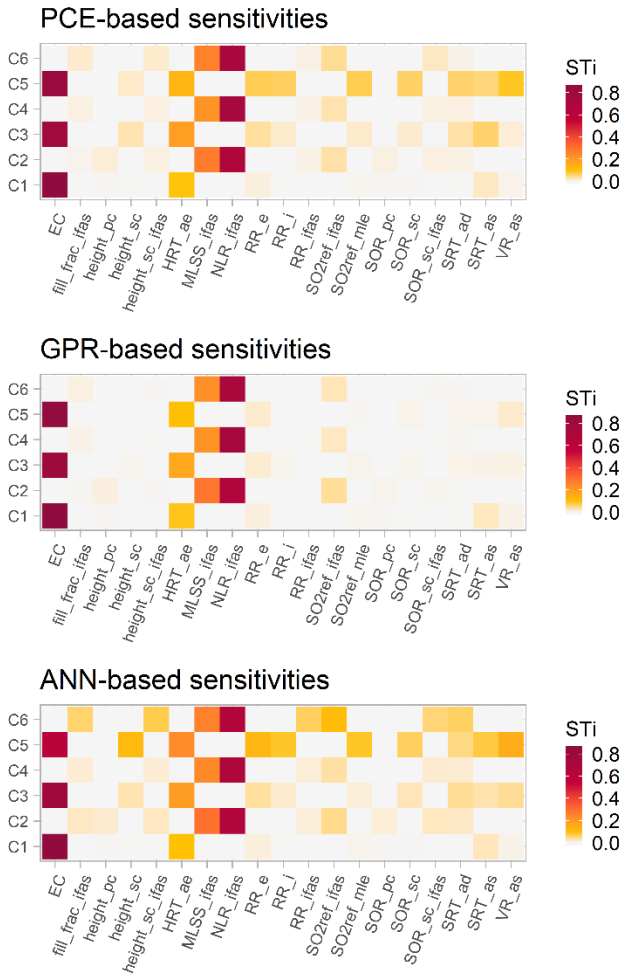

Figure 6.13: Global sensitivity analysis of the design objective OCI to the design and operational parameters (x-axis), shown for each plant configuration (y-axis). Similar results were obtained from using surrogate models of type PCE (top), GPR (middle), and ANN (bottom) [80].

Table 6.8: Key decision variables identified in plant configurations 2, 4, and 6, by the use of surrogate-based global sensitivity analysis [80].

Configuration ID	Decision variable	Lower bound	Upper bound	Unit
2	$MLSS_{IFAS}$	1000	5000	g TSS/m^3
	NLR_{IFAS}	0.05	0.5	kg N/m^{3}d
	DO_{IFAS}	0.1	0.5	g/m^3
4	$MLSS_{IFAS}$	1000	5000	g TSS/m^3
	NLR_{IFAS}	0.05	0.5	kg N/m^{3}d
	DO_{IFAS}	0.1	0.5	g/m^3
	FF_{IFAS}	30	60	%
6	$MLSS_{IFAS}$	1000	5000	g TSS/m^3
	NLR_{IFAS}	0.05	0.5	kg N/m^{3}d
	DO_{IFAS}	0.1	0.5	g/m^3

6.6.3 Uncertainty management via embedded Monte Carlo simulations

The previously described Monte Carlo-based uncertainty analysis was performed for each of the optimization initialization samples in order to quantify and thereby model the uncertainty accompanying optimization objective and constraints. To that end, as described in the problem statement above, four influent fractionation parameters were allowed to vary from their nominal values with a coefficient of variation of 25 %. The probability model for this uncertainty was expected to follow a normal distribution, as these parameters are more likely to have values around their means than around their extremes. The resulting probability space was sampled with 100 LHS samples, which are illustrated in Figure 6.14 (A), whose diagonal elements show the underlying assumption of a Gaussian distribution. The propagation of these influent uncertainties to the plant performance indicators was achieved by performing simulations for each of the 100 LHS influent samples. Then, the results obtained from these simulations were gathered into a dataset, which is then analyzed to

statistically estimate distributions of the key plant outputs, e.g., mean, standard deviation, variance, etc.

Figure 6.14 (B) shows Monte Carlo-estimated properties for the optimization objective (OCI) and constraints (effluent COD, S_{NH}, and TN). The standard deviation provides a quantitative measure of how much deviation from the mean should be expected in a random variable, and as such, it constitutes one of the standard measures of uncertainty, among other measures. Consequently, the wider the width of the green shaded area in Figure 6.14 (B), which represents the standard deviation, the greater the uncertainty in the output. The kernel density estimation, on the other hand, gives an insight into the shape of the probability distribution of the output, which follows a normal-like distribution in this case, using a kernel density function.

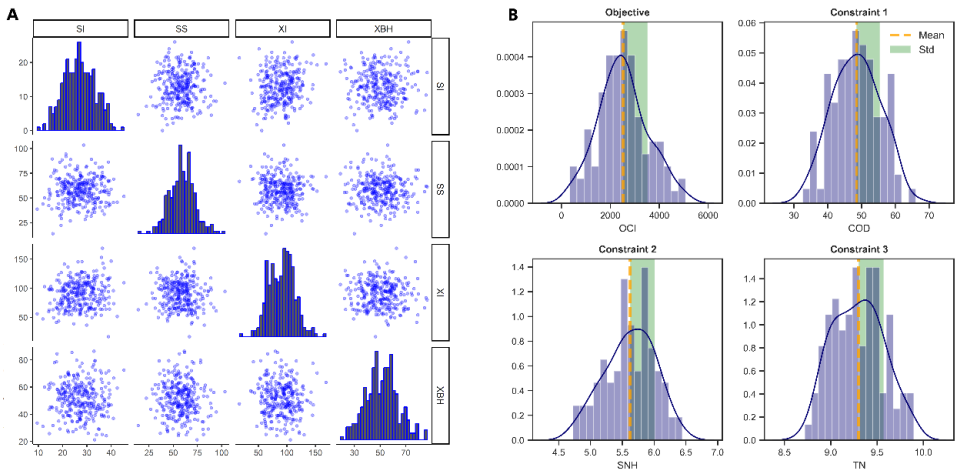

Figure 6.14: Uncertain influent fractionation parameters uniformly sampled around their nominal values are shown in A. Output distributions (bins) of the objective and effluent quality constraints obtained from Monte Carlo simulations are shown in B along with their mean (dashed orange line), standard deviation (green shaded area), and kernel density estimations (blue lines) [80].

In the workflow, it is important to note that this uncertainty analysis is performed for each design point during both the initialization and the adaptive sampling steps. In doing so, uncertainty at each different set of design decisions (as tried out by the solver during the optimization) is quantified and taken into account when deciding on feasibility as the

output uncertainty might vary for each candidate design in the design space. These Monte Carlo estimates of uncertainty are then fed into the black-box solver, which internally fits Kriging type surrogate models to the output variance. These Kriging models are then employed by the solver in order to estimate uncertainty at unvisited design points in the design space, and by using also the stochastic Kriging models, the solver then finds the next best design point which has the highest likelihood of improving the objective while satisfying the constraints and taking into account their uncertainties propagated from the inputs.

6.6.4 Optimization of the BSM2 plant under influent uncertainty

The refined design spaces of the promising plant configurations, as identified in Chapter 4 (configuration 2, 4, and 6), were selected for optimization under uncertainty. The two infill criteria (the FEI and the mcFEI) were selected for comparison and separately employed to perform the optimization. The termination criterion in the workflow was set to 150 total iterations. With these settings, the software implementation of the workflow (MCSKopt) was called to optimize the selected configurations. A new candidate design, as suggested by the solver during the adaptive sampling iterations, was declared feasible only if the sum of the mean and the standard deviation of the integrated Monte Carlo simulations for uncertainty at that design point is less than the regulatory limits for each of the three effluent quality constraints. Moreover, at each iteration, all surrogate models were updated with the newly generated simulation data from the plant-wide model, and the convergence of the best observed objective value is shown in Figure 6.15 for each of the select configurations.

Regarding the comparison of the two infill criteria, the results show that mcFEI is able to return better candidate design points, whereas the FEI criterion significantly suffers from the existence of three active stochastic constraints. This is especially evident for the case of configuration 2, where the FEI criterion achieves no improvement in the

best observed objective value, whereas the proposed mcFEI criterion significantly improves the objective for each of the studied configurations. Also shown in Figure 6.15 by the shaded areas are the 95 % confidence intervals of the best objective value, which are calculated using the results of the Monte Carlo simulations. Table 6.9 shows the solver-returned values of the key decision variables that yield the best mean objective value under the influent uncertainty. Among the three configurations, the results of the configuration 2 achieve the minimum objective value though the amount of improvement in the objective value is similar to the other two configurations. It should be noted that the interpretation of the returned objective results, as well as the decision variables, need to take into account the amount of the propagated uncertainties, especially in the constraints. If the uncertainties are large, the addition of such uncertainties to the mean values of the constraints while evaluating feasibility might prove as a conservative approach, making those constraints violated, hence a feasible design might be harder to find. Therefore, the framing and the magnitude of the input uncertainty, as well as the definition of what should be considered as a feasible design during the iterations, significantly affect the results of simulation-based optimization.

Table 6.9: Simulation-based optimization under uncertainty results acquired for the case study 3 [80].

Configuration ID	Decision variable	Lower bound	Upper bound	Values at optimal	Best feasible (OCI)
2	$MLSS_{IFAS}$	1000	5000	1105.3	1921
	NLR_{IFAS}	0.05	0.5	0.139	
	DO_{IFAS}	0.1	0.5	0.305	
4	$MLSS_{IFAS}$	1000	5000	2481.3	3833
	NLR_{IFAS}	0.05	0.5	0.216	
	DO_{IFAS}	0.1	0.5	0.339	
	FF_{IFAS}	30	60	40.1	
6	$MLSS_{IFAS}$	1000	5000	1587.9	3034
	NLR_{IFAS}	0.05	0.5	0.183	
	DO_{IFAS}	0.1	0.5	0.5	

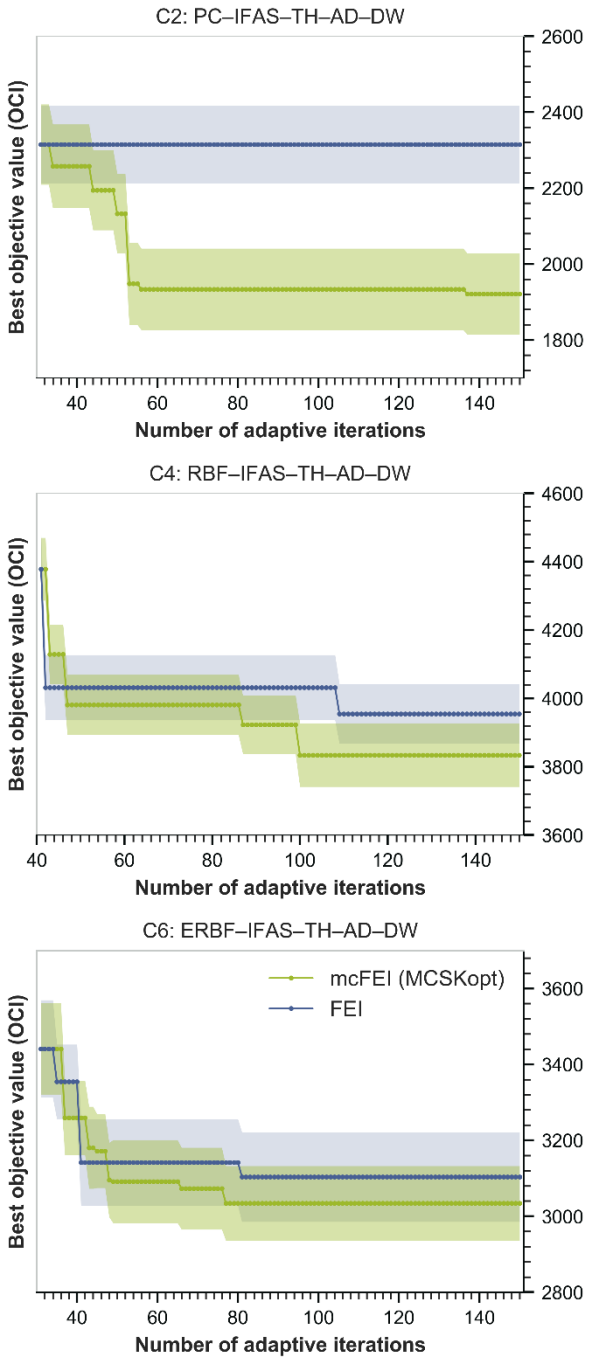

Figure 6.15: The progress of the best feasible objective by each iteration of the optimization solver, for each of the three promising layouts, shown for mcFEI (green) and FEI (blue) infill criteria. Lines show the mean, and the shaded areas show the 95% confidence intervals of the mean of the best feasible objectives. mcFEI consistently outperforms the FEI criterion in returning better candidate design points and converges to a lower value for the objective [80].

6.7 Tool implementation: MCSKopt

The simulation-based optimization under uncertainty workflow developed in this work has been implemented in a generic stochastic black-box optimization solver, named MCSKopt. The MCSKopt solver implements the stochastic Kriging-based algorithm described above with the integrated Monte Carlo simulations for uncertainty quantification. To manage the software complexity, the solver is written in the MATLAB programming language in an object-oriented fashion, i.e., the core of the algorithm is implemented in several different class files along with function files intended for user-definable infill criteria. All the previously-defined infill criteria, including the multiple-constrained FEI (mcFEI) criterion proposed in this work, are already implemented and being called as part of the overall workflow.

As shown in Figure 6.16, a public GitHub repository [159] (freely accessible from https://github.com/resulal/MCSKopt) was created to maintain the latest version (version 1.0) of the solver. The solver is MIT-licensed, which grants free-of-charge permission to use given a reference to the original published version. Upon download of the solver, the users need no further installation and can work with the solver by including its folder path to their current MATLAB session. The overall folder organization of the solver is shown in Figure 6.17. The subfolder *Examples* contains a number of simple example scripts to get started with using the solver and its functionalities. These scripts are conceived as the starting point for anyone who wants to apply the workflow for their own black-box process, and as such, they show how to connect user's data and models into the solver. Further detailed documentation of the solver will be prepared for public use. Appendix C of this thesis also contains some example scripts for both the test problems and the problem of WWTP optimization under influent uncertainty.

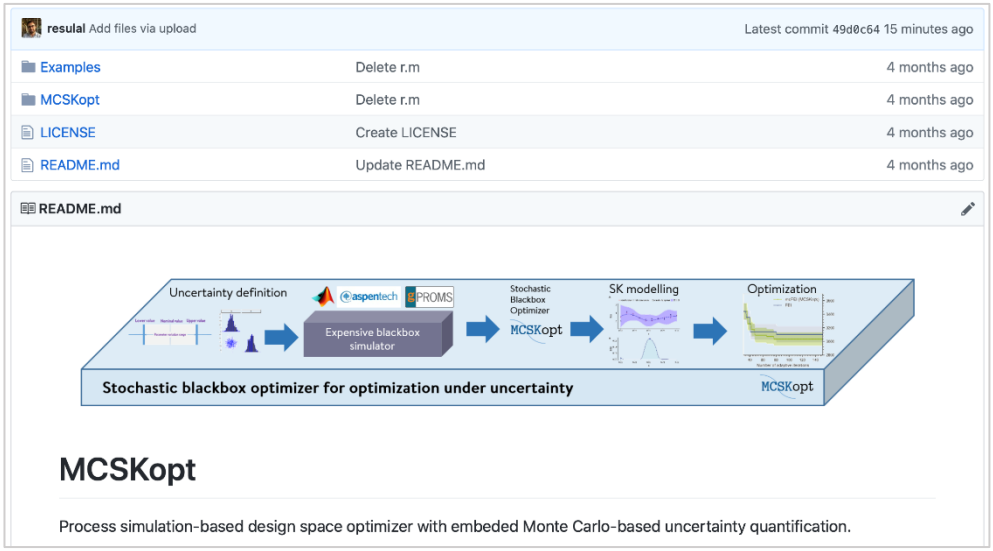

Figure 6.16: GitHub repository containing the solver MCSKopt.

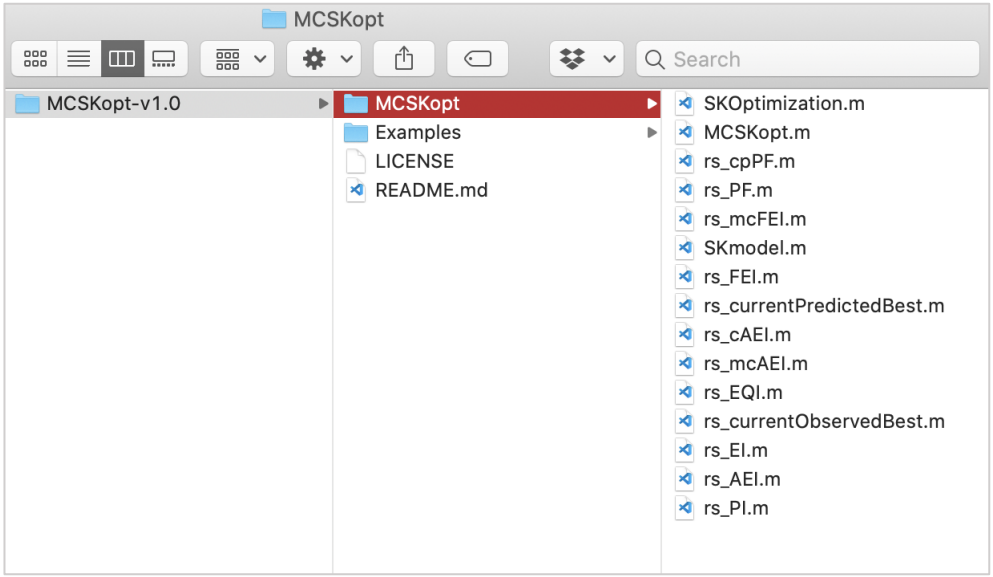

Figure 6.17: Folder organization of the MCSKopt solver.

6.8 Summary and conclusions

This chapter has presented a novel, comprehensive, Monte Carlo and simulation-based optimization under uncertainty workflow for the design of stochastic engineering systems. The objective was to incorporate diverse techniques from the fields of simulation-based optimization, sensitivity analysis, and uncertainty quantification in order to derive a complete systematic workflow for the design optimization of complex engineering systems under uncertainty. The workflow was successfully applied to the wastewater treatment plant design optimization problem, which took into account the effects of uncertain influent compositions. By integrating Monte Carlo-based uncertainty quantification into the optimization, it is also made possible to robustify design and operational decisions against the effects of possible variations in the influent wastewater compositions. All in all, the following main conclusions are drawn in this chapter.

- Plant-wide WWTP simulation models can be employed to formulate rigorous design optimization problems using the developed workflow based on a stochastic Kriging-based simulation optimization algorithm.
- As opposed to deterministic mathematical programming, simulation-based optimization not only allows for using arbitrarily complex process models for optimization purposes without the need for further model simplifications but also provides a means of quantifying the surrounding uncertainties.
- Global sensitivity analysis using surrogate models can provide fast yet valuable insights into identifying the key design and operational parameters with respect to their influence on the plant performance indicators. In doing so, the search space of the optimization problems can be effectively reduced to the only significant decisions.

- Monte Carlo based uncertainty management procedures can be effectively integrated into simulation-based optimization workflows in order to converge to robust design solutions that maintain feasibility at different realizations of the uncertain parameters.
- Unlike other surrogate models, a stochastic Kriging type surrogate model provides a means to model output uncertainties in optimization objectives and constraints, hence making it suitable for simulation-based optimization under uncertainty problems.
- In simulation-based optimization, maintaining feasibility while improving objective poses a significant challenge, especially for systems subject to multiple stochastic constraints. The proposed mcFEI infill criterion outperforms the FEI criterion in returning near-optimal solutions for such systems.
- The solver implementation (MCSKopt) constitutes an initial step towards making the developed workflow applicable to design under uncertainty problems arising in other domains. Thanks to its object-oriented structure, further improvement of the code, as well as its infill algorithms, can readily be achieved.

Finally, by integrating plant-wide simulation models, the developed generic workflow alleviates the shortcomings of simple algebraic models previously used in the optimization of complex engineering systems, paving the way for a more accurate simulation-based evaluation of design objectives and constraints. Although the results obtained are only specific for this case study, the workflow of simulation-based optimization using plant-wide models is generic in nature and can be applied to other engineering systems subject to severe uncertainties to improve their design and operational strategies.

Chapter 7

Conclusions and future perspectives

This chapter provides final conclusions drawn from this Ph.D. project and presents the author's future perspectives for further research.

7.1 Achievements

The design of WWTPs is a truly multifaceted problem, requiring holistic thinking across key stakeholders involved in the decision-making process. From solely an engineering standpoint, it constitutes a complex decision-making problem involving multiple and sometimes conflicting performance objectives and constraints under various uncertain situations. In this project, we targeted to manage this complexity by offering another way of approaching this problem within a new systematic framework that strengthens the decision-making by factoring in mechanistic model-based process understanding. Underpinned by a systematic process systems engineering approach, this new simulation-based conceptual design framework effectively divides the overall design problem into three sub-problems: process synthesis, process design, and process optimization, and for each sub-problem, it provides computationally efficient solution strategies that exploit both the first principles simulation models and also the powerful machine learning and surrogate models, including stochastic Kriging, Gaussian process regression, and artificial neural network models.

In designing the framework, a special emphasis was given to the integration of uncertainty considerations into the framework's solution algorithms, such as the simulation-based optimization under uncertainty algorithm, to enable informed decision-making by the use of well-established Monte Carlo methods for uncertainty quantification. Also emphasized and accomplished by the framework was providing the capability to harness the parallel computing architectures, such as high-performance computing clusters, clouds, etc., which are shown to substantially ease the computational burden of the exhaustive Monte Carlo simulations. Among other most significant achievements of this project are the developments of generic software tools that are aimed at better equipping the non-specialist practitioners of process design and optimization with the state-of-the-art surrogate modeling, machine learning, global sensitivity analysis, and simulation optimization

algorithms. These tools include SPDLab, easyGSA, and MCSKopt, which are all made publicly available on GitHub repositories.

On the whole, the simulation-based approaches to conceptual design have been an underinvestigated area. There has not been as extensive research conducted on conceptual design approaches integrating simulation-based optimization as there has been for approaches using mathematical programming-based superstructure optimization. In this context, the proposed framework in this thesis represents a significant step forward in simulation-based process design, which itself is an achievement in terms of its contribution to the state-of-the-art.

7.2 Remaining challenges

Despite the significant number of achievements attained within this Ph.D. project, there are still several areas that remain open for further improvement.

Firstly, the model library of the simulation-based synthesis tool (SPDLab) needs to be extended with newly arising wastewater treatment technologies, and the database containing technology-specific information needs to be updated accordingly. To further assist decision-makers and ensure wider adoption of the proposed method, a graphical user interface (GUI) simplifying the user-inputs needs to be developed. The framework SPDLab relies on may prove computational demanding if the number of considered alternative flowsheets is too many. In such cases, more efficient simulation-based algorithms for identifying the promising configurations among many alternatives need to be investigated.

Secondly, the Sobol sensitivity indices provided by the easyGSA toolbox assumes no dependency among the input parameters subject to the GSA. To be able to accommodate such a priori knowledge of input dependencies, the easyGSA tool can be extended to include the state-of-the-art GSA methods for correlated inputs, such as the Kucherenko indices method [119]. In doing so, a more realistic representation of the input space by the sampling, hence a more reliable sensitivity measure of the inputs, can be achieved.

Lastly, the black-box solver MCSKopt relies on a stochastic variant of Kriging models, which are, in general, known to suffer from scalability issues. Training Kriging models with datasets larger than a few thousand observations remains a significant challenge since the computational time required for the inversion of the covariance matrix increases cubically with the number of observations in the dataset. To overcome this issue, a sparse variant of a Gaussian process model [160] can be trained. Moreover, when the observations in the dataset are too close to each

other, the covariance matrix becomes ill-conditioned for inversion, further exacerbating the model construction. To address this issue, regularization techniques [161] for Gaussian process models can be investigated to be applied to the stochastic Kriging models.

7.3 Future perspectives

Although simulation models prove powerful for describing intricate WWTP processes like the activated sludge process, there are still some complex process phenomena that are beyond the reach of current mechanistic modeling tools, such as the N_2O emissions from the WWTPs, which pose a threat to the climate. To better understand the characteristics of N_2O emissions, tools incorporating global sensitivity analysis with purely data-driven (plant) modeling approaches leveraging the recent breakthroughs, especially in deep learning, are needed. As part of future work, it would be interesting to extend the capabilities of the easyGSA tool to return plant-data driven sensitivity indices. Indeed, there is already an ongoing effort made by the author for such an enabling tool named deepGSA.

Multiple objectives are frequently encountered in engineering design, and how to handle them in simulation-based optimization remains an open research topic. Although the black-box solver MCSKopt can be set to address such problems by converting objectives into constraints, the solver can also be extended with more explorative multi-objective infill criteria such as expected hypervolume improvement [162], which would allow for the estimation of the Pareto front of multiple objectives in stochastic simulations. As part of future work, it would be interesting to investigate the applicability of such newly proposed infill criteria [163] to the WWTP design optimization problems.

Finally, it is my hope that the conceptual design framework presented in this thesis will lay the foundations of a new and exciting area of simulation-based process synthesis and design in the PSE field, by combining fundamental mechanistic modeling and modern machine learning techniques in a principled and practical way and also by leveraging exponential opportunities offered by the state-of-the-art cloud computing environments.

Chapter 8
Appendices

This chapter provides supplementary material for the previous chapters, as well as code scripts for the software tools developed within this project. For the latest version of the tools and their example scripts, please refer to the author's GitHub page at https://github.com/resulal.

8.1 Appendix A

This appendix provides supplementary material for Chapter 4.

8.1.1 Scripts for SPDLab

The following scripts present a condensed version of the algorithms of SPDLab. The more extensive codes with the most up-to-date versions are stored in the GitHub repository [95].

```
function SS = rs_superstructure()
    % Creates a superstructure from models

    % Put all the alternative, callable model names in differents process steps.
    SS = struct;
    SS.primary_units   = {'PC','RBF','ERBF','HRAS'};
    SS.secondary_units = {'MLE','IFAS','ELANM'};
    SS.sludge_units    = {'ADM'};
    SS.side_units      = {'BP_side','ELANS'};
end
```

```
function [T, designIDs] = rs_synthesize(SS)
    % Returns a table of configurations contained in a superstructure

    designIDs = {};
    for p=1:length(SS.primary_units)
        for m=1:length(SS.secondary_units)
            for s=1:length(SS.sludge_units)
                for r=1:length(SS.side_units)
                    did ={SS.primary_units{p}, SS.secondary_units{m}, ...
                        SS.sludge_units{s}, SS.side_units{r}};
                    designIDs{end+1,1} = did; % to get all dids, designIDs{:}
                end
            end
        end
    end

    didc = {};
    for d=1:length(designIDs), did=designIDs{d}; didc{d,1}=strjoin(did,'_'); end
    T = table;
    T.cID=[1:length(didc)]';
    T.designID = didc;
end
```

```
function dSpace = rs_designSpace(dID)
    % Creates a design space object of InputSpace class from a given designID
    common = {'TH','DW'};
    uMap   = rs_unitMap(); % load units.
    dSpace = InputSpace;
    dID    = [dID common];
    for i=1:length(dID) % for KEY in dID
        KEY = dID{i};
        dSpace = merge(dSpace, uMap(KEY).DesignSpace);
    end
end
```

```
classdef InputSpace  % Class for the objects to define design spaces.
    properties
        ParNames
        LowerBounds
        UpperBounds
        %         Distributions
    end
    methods
        function obj = InputSpace(varargin) % Constructor
            if nargin==1 % a structure entered.
                obj.ParNames = varargin{1}.ParNames;
                obj.LowerBounds = varargin{1}.LowerBounds;
                obj.UpperBounds = varargin{1}.UpperBounds;
            elseif nargin==3 % data entered.
                obj.ParNames    = varargin{1};
                obj.LowerBounds = varargin{2};
                obj.UpperBounds = varargin{3};
            end
        end
        function X = sample_space(obj,N,sampling)
            if nargin<3, sampling='LHS'; end
            d = numel(obj.ParNames);
            Xp=lhsdesign(N,d); % sampling in unit probability space
            % convert probability space to real value space via inverse distribution
            L=repmat(obj.LowerBounds,N,1);
            U=repmat(obj.UpperBounds,N,1);
            X = unifinv(Xp,L,U);
        end
        % merge variable number of spaces
        function obj = merge(varargin)
            a              = horzcat(varargin{:}); % array of objects
            [ParNames,ind] = unique([a.ParNames]);
            LowerBounds    = [a.LowerBounds];
            UpperBounds    = [a.UpperBounds];
            obj = InputSpace(ParNames,LowerBounds(ind),UpperBounds(ind));
        end
        % substract space2 from space1
        function obj = substract(space1,space2) %
            [pars,ind] = setdiff(space1.ParNames,space2.ParNames);
            obj = InputSpace(pars,space1.LowerBounds(ind),space1.UpperBounds(ind));
        end
        function obj = remove(obj,pars) % remove a specified parameter from a space.
            if isstring(pars)|ischar(pars), pars=cellstr(pars), end
            props = properties(obj);
            for p=1:length(pars)
                ind=find(contains(obj.ParNames,pars{p}));
                for pr=1:length(props)
                    prop=props{pr};
                    obj.(prop)(ind)=[];
                end
            end
        end
        function bounds(obj,pars) % prints lbs and ubs of given pars
            if isstring(pars)|ischar(pars), pars=cellstr(pars), end
            for p=1:length(pars)
                ind=find(contains(obj.ParNames,pars{p}));
                disp(['Lower bound for ', obj.ParNames{ind}, ' is ', num2str(obj.LowerBounds(ind))]);
                disp(['Upper bound for ', obj.ParNames{ind}, ' is ', num2str(obj.UpperBounds(ind))]);
            end
        end
        function show(obj)
            fprintf(['Your InputSpace object named ',inputname(1),' has the following variables.\n\n'])
            T = table(obj.ParNames',obj.LowerBounds',obj.UpperBounds',...
                      'VariableNames',{'ParNames','LowerBounds','UpperBounds'});
            disp(T)
        end
    end
end
```

```
classdef UnitClass % Class for storing technology specific design and uncertainty spaces.
    properties
        KEY
        FullName
        DesignSpace
        UncertainSpace
    end
    methods
        function obj = UnitClass(S) % Constructor
            if nargin>0
                obj.KEY            = S.KEY;
                obj.FullName       = S.FullName;
                obj.DesignSpace    = S.DesignSpace;
                obj.UncertainSpace = S.UncertainSpace;
            end
        end
    end
end
```

```
function [I6,G] = rs_initialize(rowXd,c)
    % Initializes a given set of design and operational variables to find
    % initial conditions by applying the sequential simulation methodology

    influent = 'BSM2'; % influent for the initialization.
    [INFLUENT,CONSTINFLUENT_BSM2,AVEDORE] = rs_get_influent(influent);

    % get the key design space and the designID
    load(c.GSAdata,'kspace');    % GSA data of the configuration
    load(c.MCSdata,'designID');  % MCS data of the configuration


    % update the design variables that will be executed in vecs.
    designrow = table2struct(array2table(rowXd, 'VariableNames', kspace.ParNames));
    space    = rs_designSpace(designID); sspace = substract(space,kspace); % unimportant space
    for up=1:numel(sspace.ParNames), designrow.(sspace.ParNames{up})=mean([sspace.Lower-
Bounds(up) sspace.UpperBounds(up)]); end
    struct2vars(designrow);

    T_sim    = 2000;

    % Run step1
    step1_pc
    I1 = rs_extractInit1(designrow,primaryin,primaryout,PC_load1,PC_load2);

    % Run step2
    step2_pc_mle_th
    I2 = rs_extractInit2(settler_mle,XINITDE-
LAY_mle,reac1,reac2,reac3,reac4,reac5,TH_load1,TH_load2);

    % Run step3 intermediate
    step3_pc_mle_th_AD_initial
    I3 = rs_extractInit3(DIGESTERINIT);

    % Run step3
    step3_pc_mle_th_AD_DE
    I4 = rs_extractInit4(digesterout,DE_load1,DE_load2);

    % Run step4
    step4_pc_mle_th_AD_DE_rec1
    I5 = rs_extractInit5(primaryout,primaryin,settler_mle,digesterout,digesterinpreinter-
face,reac1,reac2,reac3,reac4,reac5,TH_load1,XINITDELAY_mle);

    % Run step5
    step5_pc_mle_th_AD_DE_rec1_rec2
    I6 = rs_extractInit6(primaryout,primaryin,ASinput1,reac1,reac2,reac3,reac4,reac5,set-
tler_mle,digesterout,rec);
    G  = rs_ex-
tractGeom(VOL_P,VOL1,VOL2,VOL3,VOL4,VOL5,V_liq,V_gas,area_sc,height_sc,Qintr,Qr,Qw); % Store
geometry
    bdclose('all');
end
```

```
function [KPI,ME] = rs_getKPI(runS,c)
    % Simulates a single set of design decisions for a single realization of
    % influent uncertainty and returns plant KPIs.

    rs_addpaths;
    rowXd=runS.rowXd; rowXi=runS.rowXi; I6=runS.I6; G=runS.G;

    % get the key design space and the designID
    load(c.GSAdata,'kspace');
    load(c.MCSdata,'designID');

    try
        KPI  = rs_run(I6,G,rowXd,rowXi,designID,kspace,c);
        ME = '';
    catch ME
        KPI  = NaN;
        %rethrow(ME)
    end
end
```

```
function KPI = rs_run(I6,G,rowXd,rowXi,designID,kspace,c)
    % Simulates a single set of design decisions for a single realization of
    % influent uncertainty and returns plant KPIs.

    influent=c.influent;
    [~,AVEDORE] = rs_print_influent('avedore');

    struct2vars(I6);
    struct2vars(G);
    rs_init_bsm2Na_pars % pass all that is needed.

    % Update the design variables that will be executed in vecs.
    designrow = table2struct(array2table(rowXd, 'VariableNames', kspace.ParNames));
    space    = rs_designSpace(designID); sspace = substract(space,kspace); % unimportant space
    for up=1:numel(sspace.ParNames)
        designrow.(sspace.ParNames{up})=mean([sspace.LowerBounds(up) sspace.UpperBounds(up)]);
    end
    struct2vars(designrow);

    % Update the influent file in the workspace (CONSTINFLUENT_BSM2)
    [pars,mu,unc] = rs_getSet(c); ispace = InputSpace(pars,mu,unc);
    [INFLUENT,CONSTINFLUENT_BSM2,AVEDORE] = rs_newINFLUENT(rowXi,ispace,influent);

    T_sim = 1000;

    % Simulate
    rs_init_bsm2Na_vecs
    sys=c.sys;  %Open the simulink file    load_system(sys);
    load_system(sys);
    rs_select_influent; % needs influent
    myOptions = simset('SrcWorkspace','current','DstWorkspace','current');
    sim(sys,[], myOptions) %Simulate the simulink file

    try
        cr_bsm2N_plant_perf
    catch
        disp('Performandce file didnt work.')
    end
end
```

```
% Perform design space exploration using Monte Carlo simulations
% within the design space of a given configuration

tic; clc; clear; rng(42); rs_addpaths;

% Choose the configuration
cID = 6;
[~, designIDs] = rs_synthesize(rs_superstructure);
designID = designIDs{cID};  % designID = {'ERBF','IFAS','ADM','BP_side'};

% Generate and sample the design space
space    = rs_designSpace(designID); show(space);
N        = 50*numel(space.ParNames);
X        = sample_space(space,N,'LHS');
influent = 'avedore'; % 'bsm2', 'lynetten', 'valladolid'
filename = ['results/MCS_' char(join(designID,'_')) '_' influent];

% Perform Parallel Monte Carlo Simulations
pp = gcp('nocreate'); if isempty(pp); pp = parpool('local',20); end;
parfor_progress(N);
parfor i=1:N
    try
        if exist(sprintf('mcsims/row%d',i),'file')==2, continue; end
        designrow = table2struct(array2table(X(i,:), 'VariableNames', space.ParNames));
        [I6,G]    = cr_initdesign(designrow,influent,designID)
        KPI       = cr_run(designrow,designID,I6,G,influent);
        %
        D(i).KPI  = KPI;
        D(i).I6   = I6;
        D(i).G    = G;
        D(i).S    = '';
        m=matfile(sprintf('mcsims/row%d',i),'writable',true); m.KPI=KPI; m.I6=I6; m.G=G;
        fprintf('Finished a successful run.\n')
    catch ME
        D(i).KPI  = NaN;
        D(i).I6   = NaN;
        D(i).G    = NaN;
        D(i).S    = ME;
        %rethrow(ME)
    end
    parfor_progress;
end
parfor_progress(0)
ttoc = toc;
save(filename)
```

```
%% Gather the results of MCS
parfor_progress(N);
parfor i=1:N
    try
        m=matfile(sprintf('mcsims/row%d',i),'writable',true);
        D(i).KPI  = m.KPI;
        D(i).I6   = m.I6;
        D(i).G    = m.G;
        D(i).S    = '';
    catch ME
        D(i).KPI  = NaN;
        D(i).I6   = NaN;
        D(i).G    = NaN;
        D(i).S    = ME;
        %rethrow(ME)
    end
    parfor_progress;
end
parfor_progress(0)
```

```
function T = rs_MCstats(M)
    % Returns the summary of MCS results including MC error.
    varnames = {'OCI','COD','SNH','TN'};
    R = [nanmean(M); nanstd(M); nanvar(M); nanstd(M)./sqrt(size(M,1)); nanstd(M)./nanmean(M)];
    T = array2table(R,'VariableNames',varnames,...
              'RowNames',{'Mean','Std','Var','MCerr','CoV'})
end
```

8.2 Appendix B

This appendix provides supplementary material for Chapter 5.

8.2.1 Scenarios considered for global sensitivity analysis

8.2.1.1 Scenario 1: Uncertainty in the influent wastewater characterization

In this scenario, a plant design engineer is in charge of building a WWTP and he/she has some experience with activated sludge system design. Although the engineer is confident about handling the uncertainties associated with kinetic and stoichiometry parameters of the mathematical model (such as ASM1), he/she is more concerned about the influent wastewater characterization due to lack of reliable plant measurement data. Therefore, he/she is interested in quantifying the influence of influent wastewater load on the key plant performance indicators (such as methane gas production, aeration energy demand, etc.).

Table 8.1: Uncertain influent fractionation parameters considered in scenario 1.

Parameter	Description	Unit	Min	Max	Ref
f_{SI}	Soluble inert organic matter	g COD (g COD_{total})$^{-1}$	0.034425	0.057375	[12]
f_{SS}	Readily biodegradable substrate	g COD (g COD_{total})$^{-1}$	0.0491	0.1473	[12]
f_{XI}	Particulate inert organic matter	g COD (g COD_{total})$^{-1}$	0.07805	0.23415	[12]
f_{XBH}	Active heterotrophic biomass	g COD (g COD_{total})$^{-1}$	0.04275	0.12825	[12]
i_{SND_SS}	Soluble nitrogen fraction of readily biodegradable substrate	g N (g COD)$^{-1}$	0.072825	0.12138	[12]
i_{XND_XS}	Particulate nitrogen fraction of slowly biodegradable substrate	g N (g COD)$^{-1}$	0.033	0.055	[12]
$T_{influent}$	Influent temperature	°C	5	25	[129]

8.2.1.2 Scenario 2: Uncertainty in the kinetic and stoichiometry parameters

In this scenario, a plant design engineer is certain about the influent load characterization of a WWTP and he/she wants to use a mathematical model (ASM1) to help design the activated sludge system. However, he/she is concerned about the parametric uncertainties (due to the kinetic and stoichiometric parameters) of that model and wants to understand which of those uncertain parameters affect the plant KPIs the most.

Table 8.2: Uncertain stoichiometric and kinetic model parameters of ASM1 considered in scenario 2.

Parameter	Description	Unit	Min	Max	Ref
μ_H	Heterotrophic max. specific growth rate	day^{-1}	3.00	5.00	[12]
K_S	Substrate saturation constant	g COD m^{-3}	5.00	15.00	[12]
K_{OH}	Oxygen saturation constant	g O_2 m^{-3}	0.10	0.30	[12]
K_{NO}	Nitrate saturation constant	g N m^{-3}	0.25	0.75	[12]
b_H	Specific decay rate	day^{-1}	0.29	0.32	[12]
μ_A	Autotrophic max. specific growth rate	day^{-1}	0.48	0.53	[12]
K_{NH}	Ammonium saturation constant	g N m^{-3}	0.50	1.50	[12]
K_{OA}	Oxygen saturation constant	g COD m^{-3}	0.30	0.50	[12]
b_A	Specific decay rate	day^{-1}	0.04	0.06	[12]
n_g	Anoxic growth correction factor	Dimensionless	0.60	1.00	[12]
k_a	Ammonium saturation constant	m^3 (g COD $d)^{-1}$	0.03	0.08	[12]
k_h	Max. specific hydrolysis rate	g X_S (g X_{BH} COD $day)^{-1}$	2.25	3.75	[12]
K_X	Hydrolysis half-saturation coeff.	g X_S (g X_{BH} $COD)^{-1}$	0.075	0.125	[12]
n_{hyh}	Anoxic hydrolysis correction factor	Dimensionless	0.60	1.00	[12]
Y_H	Heterotrophic yield	g COD (g $COD)^{-1}$	0.64	0.70	[12]
Y_A	Autotrophic yield	g COD (g $N)^{-1}$	0.23	0.25	[12]
f_{Pobs}	Observed inert fractions	Dimensionless	0.15	0.25	[12]
i_{XB}	Nitrogen fraction in biomass	g N (g $COD)^{-1}$	0.04	0.12	[12]
i_{XP}	Nitrogen fraction in endogenous mass	g N (g $COD)^{-1}$	0.06	0.065	[12]
X_{ITSS}	TSS to COD ratio	g TSS (g $COD)^{-1}$	0.7	0.95	[12]

8.2.1.3 Scenario 3: Uncertainty in the hydraulic and design parameters

In this scenario, a plant design engineer wants to expand an existing WWTP by redesigning the activated sludge process and secondary settler to address new effluent quality limits. He/she gathers a set of hydraulic and design parameters, among which he/she wants to identify the most influential parameters affecting the KPIs of the WWTP.

Table 8.3: Uncertain hydraulics and design parameters considered in scenario 3.

Parameter	Description	Unit	Min	Max	Ref
VOL1	Volume of tank 1	m^3	1350	1500	[12]
VOL2	Volume of tank 2	m^3	1350	1500	[12]
VOL3	Volume of tank 3	m^3	2700	3000	[12]
VOL4	Volume of tank 4	m^3	2700	3000	[12]
VOL5	Volume of tank 5	m^3	2700	3000	[12]
Q_r	External recycle flow	m^3/d	15486	20648	[12]
Q_{intr}	Internal recycle flow	m^3/d	46459	77431	[12]
DSVI	Settling characteristic (r_H, v_0 correlation)	ml/g	100	160	[138]
r_P	Low conc./slow settling parameter	m^3/g	0.0027	0.01	[138]
f_{ns}	Non-settleable fraction	-	0.00123	0.00259	[138]

8.2.1.4 Scenario 4: Uncertainty in the influent wastewater characterization, kinetic and stoichiometry parameters, hydraulic and design parameters

In this scenario, a plant design engineer is interested in understanding and comparing the impact of the most important uncertain parameters affecting the plant KPIs. To achieve that, he/she gathers all the parameters considered in the previous three scenarios and performs an uncertainty quantification and a global sensitivity analysis, respectively.

8.2.2 The BSM2 plant simulation strategy and performance evaluation

The wastage sludge from the activated sludge tank is thickened using a thickener that is modeled as an ideal separator with no biological activity [11] before being sent to an anaerobic digester unit, which is modeled using Anaerobic Digestion Model No. 1 (ADM1) [90] for biogas production. The sludge produced from the anaerobic digester is further dewatered using a dewatering unit (modeled as an ideal separator with no biological activity [11] before being discharged from the plant. A storage tank is used to store the reject water from the dewatering unit, and the reject water is then recycled back to the primary clarifier. The three aerobic tanks in the activated sludge unit are maintained at a dissolved oxygen concentration level of 1 mg/l (as also used in other publications [88,135]) using proportional-integral (PI) controllers. Furthermore, the oxygen mass transfer coefficient ($K_L a$) is used as a manipulated variable (i.e. an ideal DO controller is assumed). Similarly, the total suspended solids (TSS) concentration in the last aerobic tank is maintained at a concentration of 3500 mg/l (aiming to maintain a constant sludge retention time (SRT) in the AS tank) using a PI controller and the wastage flowrate is considered as the manipulated variable. The external carbon addition to the first anoxic tank (for ensuring efficient denitrification) is fixed to 400000 mg COD/l with a flowrate of 2 m^3/day. In this way, it is ensured that the influence of DO, carbon addition and SRT on the global sensitivity analysis results is minimal.

The aeration energy consumption in the activated sludge system is calculated by the following equation.

$$AE = \frac{S_O^{sat}}{1.8 \times 1000} . (\sum_{k=1}^{5} V_k . K_L a_k) \tag{34}$$

The effluent quality index is calculated using equation (2) introduced in Chapter 4.

8.2.3 Analysis of the BSM2 model outputs

Table 8.4: Mean (μ) and standard deviation (σ) of Monte Carlo simulation results of BSM2 model outputs are tabulated for scenario 1, 2, 3, and 4.

BSM2 model output	Scenario 1		Scenario 2		Scenario 3		Scenario 4	
	μ	**σ**	**μ**	**σ**	**μ**	**σ**	**μ**	**σ**
Effluent nitrate	13.90	3.11	11.42	2.32	9.93	0.26	13.48	5.45
Effluent ammonium	1.31	3.99	0.28	0.16	0.18	0.01	4.8	11.6
Effluent quality index	6525.5	2252.1	5209	445.26	6644.4	1027.1	12545	6371.3
Sludge disposal	2715.3	380.17	3542.5	356.25	2545.2	113.65	2800	345.6
Aeration energy	4158.2	388.34	3523	146.8	3917.1	18.22	3555	569.85
Methane production	1080.9	106.57	1088.2	21.82	1060.2	15.97	1046	94.87

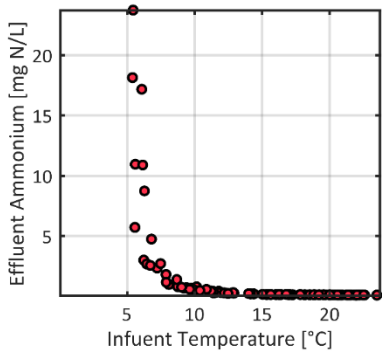

Figure 8.1: The input-output data relationship between influent temperature and effluent ammonium.

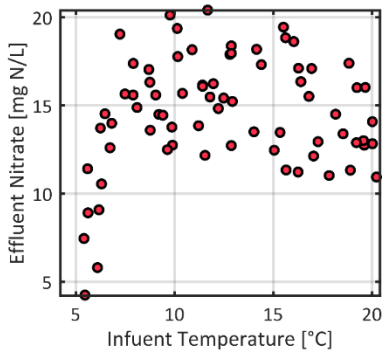

Figure 8.2: The input-output data relationship between influent temperature and effluent nitrate.

8.2.4 Model development statistics of the meta-models

8.2.4.1 ANN models

Table 8.5: Statistics of ANN models developed for scenario 2.

ANN model output	Size of experimental design	Training R^2	Cross-validation R^2 (5-fold)	Selected configuration
Effluent nitrate	100	0.998	0.966	20×11×1 with BR
Effluent ammonium	100	0.999	0.988	20×14×1 with BR
Effluent quality index	100	0.999	0.961	20×12×1 with BR
Sludge disposal	100	0.999	0.998	20×14×1 with BR
Aeration energy	100	0.999	0.983	20×10×1 with BR
Methane production	100	0.999	0.993	20×13×1 with BR

Table 8.6: Statistics of ANN models developed for scenario 3.

ANN model output	Size of experimental design	Training R^2	Cross-validation R^2 (5-fold)	Selected configuration
Effluent nitrate	100	0.999	0.981	10×11×1 with BR
Effluent ammonium	100	0.998	0.992	10×12×1with LM
Effluent quality index	100	0.999	0.990	10×12×1 with BR
Sludge disposal	100	0.999	0.995	10×15×1 with BR
Aeration energy	100	0.999	0.989	10×14×1 with BR
Methane production	100	0.999	0.990	10×12×1 with BR

Table 8.7: Statistics of ANN models developed for scenario 4.

ANN model output	Size of experimental design	Training R^2	Cross-validation R^2 (5-fold)	Selected configuration
Effluent nitrate	450	0.951	0.694	37×13×1 with LM
Effluent ammonium	400	0.976	0.440	37×13×1 with LM
Effluent quality index	400	0.945	0.448	37×13×1 with LM
Sludge disposal	100	0.999	0.978	37×11×1 with BR
Aeration energy	450	0.979	0.611	37×14×1 with LM
Methane production	100	0.998	0.960	37×14×1 with BR

8.2.4.2 GPR models

Table 8.8: Statistics of GPR models developed for scenario 2.

GPR model output	Size of experimental design	Training R^2	Leave-1-out coefficient Q^2	Kernel function
Effluent nitrate	100	1	0.990	ARD Matern 3/2
Effluent ammonium	100	1	0.996	ARD Matern 3/2
Effluent quality index	100	1	0.976	ARD Matern 3/2
Sludge disposal	100	0.999	0.999	ARD Matern 3/2
Aeration energy	100	1	0.995	ARD Matern 3/2
Methane production	100	0.999	0.999	ARD Matern 3/2

Table 8.9: Statistics of GPR models developed for scenario 3.

GPR model output	Size of experimental design	Training R^2	Leave-1-out coefficient Q^2	Kernel function
Effluent nitrate	100	0.986	0.967	Sq Exponential
Effluent ammonium	100	0.998	0.998	ARD Matern 3/2
Effluent quality index	100	0.999	0.999	ARD Matern 3/2
Sludge disposal	100	0.999	0.999	ARD Quadratic
Aeration energy	100	0.999	0.997	ARD Sq Quadratic
Methane production	100	0.999	0.999	ARD Sq Quadratic

Table 8.10: Statistics of GPR models developed for scenario 4.

GPR model output	Size of experimental design	Training R^2	Leave-1-out coefficient Q^2	Kernel function
Effluent nitrate	250	0.927	0.852	ARD Matern 3/2
Effluent ammonium	250	1	0.950	ARD Matern 3/2
Effluent quality index	250	0.996	0.926	ARD Matern 3/2
Sludge disposal	100	0.999	0.997	ARD Matern 3/2
Aeration energy	200	0.987	0.957	ARD Matern 3/2
Methane production	100	1	0.994	ARD Matern 3/2

8.2.4.3 PCE models

Table 8.11: Statistics of PCE models developed for scenario 2.

PCE model output	Size of experimental design	PCE order p	Training R^2	Leave-1-out coefficient Q^2	# of nonzero coeffs
Effluent nitrate	100	2	0.991	0.961	31
Effluent ammonium	100	2	0.993	0.968	31
Effluent quality index	150	2	0.995	0.979	45
Sludge disposal	100	3	0.999	0.999	18
Aeration energy	100	2	0.995	0.979	28
Methane production	100	4	0.993	0.988	10

Table 8.12: Statistics of PCE models developed for scenario 3.

PCE model output	Size of experimental design	PCE order p	Training R^2	Leave-1-out coefficient Q^2	# of nonzero coeffs
Effluent nitrate	100	3	0.996	0.981	29
Effluent ammonium	100	2	0.999	0.999	38
Effluent quality index	100	4	0.999	0.997	23
Sludge disposal	100	3	0.997	0.995	13
Aeration energy	100	5	1	0.995	99
Methane production	100	3	0.996	0.989	19

Table 8.13: Statistics of PCE models developed for scenario 4.

PCE model output	Size of experimental design	PCE order p	Training R^2	Leave-1-out coefficient Q^2	# of nonzero coeffs
Effluent nitrate	250	3	0.947	0.907	55
Effluent ammonium	150	3	0.997	0.986	87
Effluent quality index	150	3	0.991	0.972	60
Sludge disposal	100	2	0.999	0.997	36
Aeration energy	250	3	0.973	0.955	49
Methane production	100	2	0.996	0.992	26

8.2.5 Scripts for easyGSA

8.2.5.1 Example 1: Ishigami function

Analytical expression for the Ishigami function reads as follows [101]:

$$\begin{aligned} f(x) &= \sin(x_1) + a\sin^2(x_2) + bx_3^4 \sin(x_1) \\ x_i &\sim U(-\pi,\pi) \text{ for } i = 1,..,3 \text{ and } a = 7,\ b = 0.1 \end{aligned} \tag{35}$$

The following script computes Monte Carlo, GRP, and ANN-based Sobol sensitivity indices using easyGSA.

```
% Tutorial: Performing Sobol GSA on Ishigami function using easGSA
% By Resul Al @DTU

% Model: Ishigami function [https://www.sfu.ca/~ssurjano/ishigami.html]
f = @(x) sin(x(:,1)) + 7.*sin(x(:,2)).^2 + 0.1.*x(:,3).^4.*sin(x(:,1));
N = 1e4; % Number of MC samples. Minimum recommended: 1e3

% Uniform Input Space
pars = strseq('x',1:3); % input parameter names
lbs  = -pi.*ones(1,3);  % lower bounds of input parameters
ubs  =  pi.*ones(1,3);  % upper bounds of input parameters
InputSpace = {'ParNames',pars,'LowerBounds',lbs,'UpperBounds',ubs};

% call easyGSA tool to perform Sobol sensitivity analysis with MC approach
[Si,STi] = easyGSA(f,N,InputSpace{:},'UseParallel',true)

% Suppress command line messages
[Si,STi] = easyGSA(f,N,InputSpace{:},'UseParallel',true,'Verbose',false)

% Change sampling method and MC estimator
[mcSi,mcSTi] = easyGSA(f,N,InputSpace{:}, ...
                   'SamplingMethod','LHS',... % default: 'Sobol'
                   'Estimator','Saltelli')    % default: 'Jansen'

% use a GPR model instead
[gprSi, gprSTi] = easyGSA(f,N,InputSpace{:},'UseSurrogate','GPR')

% use an ANN model instead
[annSi, annSTi] = easyGSA(f,N,InputSpace{:},'UseSurrogate','ANN')


% Analytical first order indices from doi:10.1016/j.ress.2008.07.008
Si_analytical = [0.3139 0.4424 0]';

% Plot comparative results
T = table(Si_analytical,mcSi,gprSi,annSi,...
    'VariableNames', {'Analytical','MonteCarlo','GPR','ANN'}, ...
    'RowNames', strseq('x',1:3));
fprintf("\n\nFirst Order Sensitivity Indices of Ishigami function\n\n")
disp(T)

H = [Si_analytical,mcSi,gprSi,annSi]; c = categorical(pars);
bar(c,H); legend({'Analytical','MonteCarlo','GPR','ANN'});
ylabel('First Order Sobol indices'); xlabel('Input Parameters');
print('Si_ishigami','-dpng','-r1200')

H = [mcSTi,gprSTi,annSTi]; c = categorical(strseq('x',1:3));
bar(c,H); legend({'MonteCarlo','GPR','ANN'});
ylabel('Total Order Sobol indices'); xlabel('Input Parameters');
print('STi_ishigami','-dpng','-r1200')
```

8.2.5.2 Example 2: g-function of Sobol

Analytical expression for the g-function of Sobol reads as follows [101]:

$$f(x)=\prod_{i=1}^{d}\frac{|4x_i-2|+a_i}{1+a_i},$$
$$a_i=\frac{i-2}{2},\ \text{and}\ x_i\sim U(0,1)\ \text{for}\ i=1,..,d \tag{36}$$

The following script computes Monte Carlo, GRP, and ANN-based Sobol sensitivity indices using easyGSA.

```
% Tutorial: Perform GSA on g-function of Sobol using easGSA
% By Resul Al @DTU

f = @(x) gSobol(x);
N = 1e6; % Number of MC samples

pars = strseq('x',1:5); % input parameter names
lbs  = zeros(1,5);      % lower bounds of input parameters
ubs  = ones(1,5);       % upper bounds of input parameters
InputSpace = {'ParNames',pars,'LowerBounds',lbs,'UpperBounds',ubs};

% Monte Carlo indices from the original model, use parallel computing.
[mcSi,mcSTi] = easyGSA(f,N,InputSpace{:},'UseParallel',true);

% GPR indices
[gprSi, gprSTi] = easyGSA(f,N,InputSpace{:},'UseSurrogate','GPR',...
                                            'UseParallel',true)

% ANN indices
[annSi, annSTi] = easyGSA(f,N,InputSpace{:},'UseSurrogate','ANN',...)
                                            'UseParallel',true)

% Analytical first order indices from doi:10.1016/j.ress.2008.07.008
Si_analytical = [0.48 0.21 0.12 0.08 0.05]';

T = table(Si_analytical,mcSi,gprSi,annSi,...
    'VariableNames', {'Analytical','MonteCarlo','GPR','ANN'}, ...
    'RowNames', strseq('S',1:5));
fprintf("\n\nFirst Order Sensitivity Indices of Sobol' g-function\n\n")
disp(T)

% put all indices in a bar plot
H = [Si_analytical,mcSi,gprSi,annSi]; c = categorical(strseq('x',1:5));
bar(c,H); legend({'Analytical','MonteCarlo','GPR','ANN'});
ylabel('First Order Sobol indices'); xlabel('Input Parameters');
print('gSobol','-dpng','-r1200')

function y = gSobol(x)
    % 5 dimensional implementation of g-function of Sobol.
    a=[1:5];
    prod = 1;
    for j = 1:5
        xi = x(:,j);
        ai = a(j);
        term1 = abs(4.*xi-2) + ai;
        term2 = 1 + ai;
        prod = prod .* term1./term2;
    end
    y = prod;
end
```

8.2.5.3 Example 3: Sobol sensitivity indices from a user dataset

The following script computes and compares GRP and ANN-based Sobol sensitivity indices calculated from a standard MATLAB dataset using easyGSA.

```
% Tutorial: Inputting your own dataset to perform GPR and ANN-based GSA
% By Resul Al @DTU

% Step 1: Load your own dataset, eg. simulation results, etc.
[X,Y]=chemical_dataset; X=X'; Y=Y'; % a standard MATLAB dataset

% Step 2: Put your data into a struct. Only X and Y fields are expected.
Data.X = X; % inputs
Data.Y = Y; % outputs

% Step 3: pass your data into easyGSA
[Si,STi,results] = easyGSA('UserData',Data) % uses GPR models by default.

% Step 4: Fit ANN models and perform a Sobol GSA
[Si,STi,results] = easyGSA('UserData',Data,...
                           'UseSurrogate','ANN')

% Step 5: Change size of sampling matrices, N, used for Sobol analysis
[Si,STi,results] = easyGSA('UserData',Data,...
                           'UseSurrogate','ANN','N',1e4)


% Step 5: Change size of sampling matrices used for Sobol analysis
[Si,STi,results] = easyGSA('UserData',Data,...
                           'UseSurrogate','ANN',...
                           'N',4e4,'SamplingMethod','LHS')

% Change the input space to a Gaussian.
InputSpace={'Means',mean(Data.X),'Sigmas',std(Data.Y)};
[Si,STi,results] = easyGSA('UserData',Data,...
                           'UseSurrogate','ANN',...
                           InputSpace{:})
```

8.2.5.4 Example 4: Standardized regression coefficients

The following script returns standardized regression coefficients (SRCs) for the same dataset as above.

```
% Tutorial: Standardized regression coefficients using easyGSA
% By Resul Al @DTU

% Load a built-in dataset for the analysis
[X,Y] = chemical_dataset;
Data.X = X'; % rows are observations.
Data.Y = Y'; % columns are outputs.

% call the easyGSA tool with the following arguments for SRCs.
[SRCs,results] = easyGSA('UserData',Data,...
                         'Method','SRC')

% Visualize the outputs in a barplot
H = [SRCs]; c = categorical(strseq('x',1:8));
bar(c,H);
ylabel('Standardized regression coefficients'); xlabel('Input Parameters');
print('ChemData-SRC','-dpng','-r600');
```

8.3 Appendix C

This appendix provides supplementary material for Chapter 6.

8.3.1 Multiple constrained FEI infill criterion implementation

```
function FEI = rs_mcFEI(x,skf,skg,fstar,xstar,noisef,noiseg,m)
    % Matlab implementation of mcFEI (Al et al., 2020)
    % Models multiple stochastic constraints/variances
    % By Resul Al @DTU, April 15, 2020

    % Multiple constrained AEI ctiterion
    mcAEI = rs_mcAEI(x,skf,skg,fstar,noisef,noiseg,m);

    % EQIFg term for multiple constraints (see Wang et al. 2018)
    kg=noiseg.predictor;
    for c=1:numel(skg)
        EQIFg(:,c) = rs_EQIFg(x,skg{c},kg{c},xstar,m);
    end
    mEQIFg = prod(EQIFg,2);

    % P term
    [yf,sf] = skf.predict(x);
    [ystar,sfstar] = skf.predict(xstar);
    f3star = fstar + 3.*sfstar;
    P = normcdf((f3star-yf)./sf);

    pEQIF = mEQIFg.*P;
    FEI = mcAEI + pEQIF;

    % Expected Quantile Improvement for Feasibility
    function EQIFg = rs_EQIFg(x,skg,kg,xstar,m)
        [yg,sg] = skg.predict(x); sg2=sg.^2;
        gstar = skg.predict(xstar); alpha = 0.1;
        qt = gstar + norminv(alpha).*sg;

        Vg = kg.predict(x);
        epsilong  =  10.^(Vg); % noise variance predictor
        taug2  = epsilong.^2/m;
        yq = yg + norminv(alpha).*sqrt(taug2.*sg2./(taug2 + sg2));
        sq2 = (sg2.^2./(taug2 + sg2)); sq=sqrt(sq2);
        EQIFg = sq.*normpdf((qt-yq)./sq);
    end
end

function mcAEI = rs_mcAEI(x,skf,skg,fCurrentBest,noisef,noiseg,m)
    % Multiple constrained augmented expected improvement infill criterion.
    % By Resul Al @DTU, April 15, 2020

    % Augmented Expected Improvement ( AEI)
    AEI = rs_AEI(x,skf,fCurrentBest,noisef,m);

    % Probability of Feasibility (PF): All constraints satisfied
    PF = rs_PF(x,skg,noiseg,m);

    % mcAEI
    mcAEI = AEI.*PF;
end
```

8.3.2 Feasibility enhanced EI (FEI) criterion implementation

```
function FEI = rs_FEI(x,skf,skg,fstar,xstar,noisef,noiseg,m)
    % Matlab implementation of FEI criterion of Wang et al. (2018)
    % By Resul Al @DTU, April 15, 2020

    if iscell(skg), skg=skg{1}; end % only one constraint model

    % Constrained augmented expected improvement infill criterion
    cAEI = rs_cAEI(x,skf,skg,fstar,noisef,m);

    % Predictions from SK models
    [yf,sf] = skf.predict(x);
    [yg,sg] = skg.predict(x); sg2=sg.^2;

    % Expected Quantile Improvement for Feasibility (EQIFg)
    gstar = skg.predict(xstar); alpha = 0.1;
    qt = gstar + norminv(alpha).*sg; % target feasibility contour
    kg = noiseg.predictor; if iscell(kg), kg=kg{1}; end % only one constraint noise model
    Vg = kg.predict(x);
    epsilong  =  10.^(Vg); % noise variance predictor
    taug2  = epsilong.^2/m;
    yq = yg + norminv(alpha).*sqrt(taug2.*sg2./(taug2 + sg2));
    sq2 = (sg2.^2./(taug2 + sg2)); sq=sqrt(sq2);
    EQIFg = sq.*normpdf((qt-yq)./sq);

    % P term: search feasible around the promising areas of the objective
    [ystar,sfstar] = skf.predict(xstar);
    f3star = fstar + 3.*sfstar;
    P = normcdf((f3star-yf)./sf);

    pEQIF = EQIFg.*P;
    FEI = cAEI + pEQIF;
end
```

8.3.3 Constrained augmented EI criterion implementation

```
function cAEI = rs_cAEI(x,skf,skg,fCurrentBest,noisef,m)
    % Matlab implementation of constrained AEI criterion (see Wang et al., 2018)
    % By Resul Al @DTU, April 15, 2020

    if iscell(skg), skg=skg{1}; end % only one constraint model

    % Prediction
    [mug,sg] = skg.predict(x); % only one constraint model

    % Augmented expected improvement infill criterion
    AEI = rs_AEI(x,skf,fCurrentBest,noisef,m);

    % Constrained AEI criterion
    cAEI = AEI.*normcdf((0-mug)./sg);
end

function AEI = rs_AEI(x,skf,fCurrentBest,noisef,m)
    % Augmented expected improvement infill criterion
    % By Resul Al @DTU, April 15, 2020

    % Prediction
    [muf,sf] = skf.predict(x);

    % Expected improvement
    EI = rs_EI(x,skf,fCurrentBest);

    % Get the noise variance at the new x
    kf = noisef.predictor; % was trained on log10(fStd)
    Vf = kf.predict(x);
    epsilonf  =  10.^(kf.predict(x)); % noise std predictor
    tauf2  = epsilonf.^2/m;

    % Augmented EI
    AEI = EI.*(1 - sqrt(tauf2)./sqrt(tauf2+sf.^2));
end

function EI = rs_EI(x,skf,fCurrentBest)
    % Returns expected improvement score of any design point x given
    % a current best (fCurrentBest), ignores noise and constraints.
    % By Resul Al @DTU, April 15, 2020

    % Prediction
    [muf,sf] = skf.predict(x);

    % EI
    xcr = (fCurrentBest-muf)./sf;
    EI  = (fCurrentBest-muf).*normcdf(xcr) + sf.*normpdf(xcr);
End
```

8.3.4 Scripts for MCSKopt

8.3.4.1 Example 1: Sasena test problem

The following script solves the Sasena test problem presented in section 6.4 using the MCSKopt solver.

```
% Simulation-based optimization of Sasena test problem using MCSKopt
% By Resul Al @DTU

% Define an optimization problem structure p
p      = struct;
p.lbs  = zeros(1,2);      % Lower bounds of the design space
p.ubs  = ones(1,2);       % Upper bounds of the design space
p.x0   = (p.lbs+p.ubs)/2;
p.dim  = numel(p.x0);     % Dimensinality of the design space
p.m    = 1e3;             % MC sample size for integrated uncertainty analysis
p.cv   = 0.50;            % Coefficient of variation in uncertain parameters
k      = 5*p.dim;         % Size of initial (coarse) design
Nmax   = 150;              % MaxFunEval (SK iterations)

% Create an initial (coarse) design space (InitialX)
Xp = lhsdesign(k,p.dim);  % LHS in probability space
for i=1:p.dim, InitialX(:,i) = unifinv(Xp(:,i),p.lbs(i),p.ubs(i)); end
[InitialObjectiveObservations, InitialConstraintObservations] = rs_simulate(InitialX,p);


% Prepare variables and the objective function for SK optimization
vars=[];
for i=1:p.dim
    eval(sprintf("x%d = optimizableVariable('x%d',[%d,%d]);",i,i,p.lbs(i),p.ubs(i)));
    eval(sprintf("vars = [vars x%d];",i));
end

% Call the interface to the optimizer
fun = @(xx) myObj(xx,p); % objective function
[x,fval,results] = MCSKopt(fun,vars,'Verbose',1,...
                           'SaveEachNiters',100,...
                           'MaxObjectiveEvaluations',Nmax,...
                           'NumSeedPoints',k,...
                           'NumRepetitions',p.m,...
                           'InitialX',array2table(InitialX),...
                           'InitialObjectiveObservations',InitialObjectiveObservations,...
                           'InitialConstraintObservations',InitialConstraintObservations,...
                           'NumCoupledConstraints',3,...
                           'CoupledConstraintTolerances',1e-3*ones(1,3),...
                           'InfillCriterion','mcFEI',... % 'FEI', 'cAEI'
                           'InfillSolver','particleswarm',... % 'GlobalSearch', 'MultiStart'
                           'UncertaintyHedge','Mean') % 'MeanPlusSigma', 'UCI95', 'PF80'
```

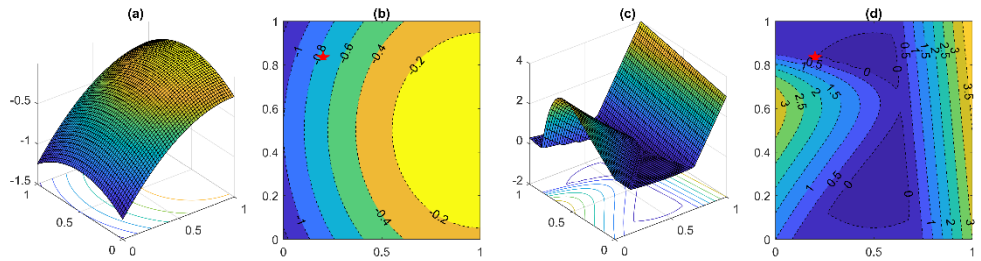

Figure 8.3: Response surface plots of Sasena objective in (a), constraints in (c). Contour plots of Sasena objective in (b), constraints in (d) where the star shows the location of the best-known solution for the nominal problem.

8.3.4.2 Example 2: Rosen Suzuki test problem

The following script solves the Rosen Suzuki test problem presented in section 6.5 using the MCSKopt solver.

```
% Simulation-based optimization of Rosen Suzuki test problem using MCSKopt
% By Resul Al @DTU

% Define an optimization problem structure p
p      = struct;
p.lbs  = -3*ones(1,4);    % Lower bounds of the design space
p.ubs  =  3*ones(1,4);    % Upper bounds of the design space
p.x0   = (p.lbs+p.ubs)/2;
p.dim  = numel(p.x0);     % Dimensinality of the design space
p.m    = 1e3;             % MC sample size for integrated uncertainty analysis
p.cv   = 0.50;            % Coefficient of variation in uncertain parameters
p.seed = 0;               % Seed for rng
p.dist = 'norm';          % Distribution for the uncertain parameters
k      = 5*p.dim;         % Size of initial (coarse) design
Nmax   = 50;              % MaxFunEval (SK iterations)

% Create an initial (coarse) design space (InitialX)
rng(p.seed,'Twister'); Xp = lhsdesign(k,p.dim);
for i=1:p.dim, InitialX(:,i) = unifinv(Xp(:,i),p.lbs(i),p.ubs(i)); end
[InitialObjectiveObservations, InitialConstraintObservations] = rs_simulate(InitialX,p);

% Prepare variables and the objective function for SK optimization
vars=[];
for i=1:p.dim
    eval(sprintf("x%d = optimizableVariable('x%d',[%d,%d]);",i,i,p.lbs(i),p.ubs(i)));
    eval(sprintf("vars = [vars x%d];",i));
end

% Call the interface to the optimizer
fun = @(xx) myObj(xx,p); % objective function
[x,fval,results] = MCSKopt(fun,vars,'Verbose',1,...
                           'MaxObjectiveEvaluations',Nmax,...
                           'NumSeedPoints',k,...
                           'NumRepetitions',p.m,...
                           'InitialX',array2table(InitialX),...
                           'InitialObjectiveObservations',InitialObjectiveObservations,...
                           'InitialConstraintObservations',InitialConstraintObservations,...
                           'NumCoupledConstraints',3,...
                           'CoupledConstraintTolerances',1e-3*ones(1,3),...
                           'InfillCriterion','mcFEI',... % 'FEI', 'cAEI'
                           'InfillSolver','particleswarm',...  % 'GlobalSearch', 'MultiStart'
                           'UncertaintyHedge','MeanPlusSigma') % 'MeanPlusSigma', 'UCI95',
'PF80'

function [f,g,UserData] = myObj(xx,p)
    % Handle of the objective function that returns the objective and the constraint
    % observations (each with m MC simulations).
    x=[];
    for i=1:p.dim
        eval(sprintf("x = [x ; xx.x%d];",i))
    end
    [f_observations, g_observations] = rs_simulate(x',p); % Calls the simulator

    % Outputs: Mean of the objective, constraints as well as entire dataset from MC simula-
tions.
    f = nanmean(f_observations,2);
    g = cellfun(@(X) nanmean(X,2), g_observations ,'UniformOutput',false);
    UserData.ObjectiveObservations    = f_observations;
    UserData.ConstraintObservations   = g_observations;
end
```

8.3.4.3 Example 3: WWTP optimization under uncertainty

The following script solves the WWTP design optimization under influent uncertainty problem presented in section 6.6 using the MCSKopt solver.

```
% Simulation-based optimization of a WWTP design optimization under uncertainty using MCSKopt
% By Resul Al @DTU

clc; clear; rs_addpaths; % clear and add paths of SPDLab and MCSKopt
pp = gcp('nocreate'); if isempty(pp); pp = parpool('local',20); end;

% Get the key decision variables space from ensemble GSA before optimization
cID=4; threshold=0.01; kspace=rs_kspace(cID,threshold); show(kspace);

% Optimization: Define an optimization problem structure p, and call the interface to the op-
timizer.
p     = struct;
p.lbs = kspace.LowerBounds';  % Lower bounds of the design space
p.ubs = kspace.UpperBounds';  % Upper bounds of the design space
p.x0  = (p.lbs+p.ubs)/2;      % x0 for optimization
p.dim  = numel(p.x0);     % Dimensinality of the design space
p.m    = 1e3;             % MC sample size for integrated uncertainty analysis
p.cv   = 0.50;            % Coefficient of variation in the uncertain space
p.dist = 'norm';          % Distribution for the uncertain parameters
k      = 5*p.dim;         % Size of initial (coarse) design
Nmax   = 150;             % Max number of iterations (SK iterations)

% Decide on initial set of samples using both MCS results and LHS samples.
FeasibleX  = rs_FeasibleX('MCS_c4'); % needs a completed MC simulation dataset
n_feasible = size(FeasibleX,1); % Number of feasible samples found from design space explora-
tion with MCS.
if n_feasible<k/2
    FeasibleX   = FeasibleX; % use all feasible samples
    AdditionalX = rs_sampleLHS(k/2-n_feasible,p.lbs,p.ubs); % additional LHS samples
else
    FeasibleX   = FeasibleX(randsample(k,k/2),:); % use only k/2 feasible samples
    AdditionalX = rs_sampleLHS(k/2,p.lbs,p.ubs);  % additional k/2 LHS samples
end
InitialX = [FeasibleX; AdditionalX];

% Simulate the initial samples with MCS for uncertainty.
[InitialObjectiveObservations, InitialConstraintObservations] = rs_simulate(InitialX,p);


% Prepare variables and the objective function for SK optimization
vars=[];
for i=1:p.dim
    eval(sprintf("x%d = optimizableVariable('x%d',[%d,%d]);",i,i,p.lbs(i),p.ubs(i)));
    eval(sprintf("vars = [vars x%d];",i));
end

% Call the interface to the optimizer
fun = @(xx) myObj(xx,p); % objective function
[x,fval,results] = MCSKopt(fun,vars,'Verbose',1,...
                           'MaxObjectiveEvaluations',Nmax,...
                           'NumSeedPoints',k,...
                           'NumRepetitions',p.m,...
                           'InitialX',array2table(InitialX),...
                           'InitialObjectiveObservations',InitialObjectiveObservations,...
                           'InitialConstraintObservations',InitialConstraintObservations,...
                           'NumCoupledConstraints',3,...
                           'CoupledConstraintTolerances',1e-3*ones(1,3),...
                           'InfillCriterion','mcFEI',... % 'FEI', 'cAEI'
                           'InfillSolver','particleswarm',...  % 'GlobalSearch', 'MultiStart'
```

References

[1] D. Griggs, M. Stafford-Smith, O. Gaffney, J. Rockström, M.C. Öhman, P. Shyamsundar, W. Steffen, G. Glaser, N. Kanie, I. Noble, Sustainable development goals for people and planet, Nature. 495 (2013) 305–307. doi:10.1038/495305a.

[2] I.E. Grossmann, I. Harjunkoski, Process systems Engineering: Academic and industrial perspectives, Comput. Chem. Eng. 126 (2019) 474–484. doi:10.1016/j.compchemeng.2019.04.028.

[3] Q. Chen, I.E. Grossmann, Recent Developments and Challenges in Optimization-Based Process Synthesis, 2017. doi:10.1146/annurev-chembioeng-080615-033546.

[4] J.M. Douglas, Conceptual design of chemical processes, McGraw-Hill, 1988. https://books.google.dk/books?id=M6JTAAAAMAAJ.

[5] H. Yeomans, I.E. Grossmann, A systematic modeling framework of superstructure optimization in process synthesis, Comput. Chem. Eng. 23 (1999) 709–731. doi:10.1016/S0098-1354(99)00003-4.

[6] X. Chen, R. Al, C.R. Behera, G. Sin, Process Synthesis, Design, and Control of Wastewater Treatment Plants, in: Ref. Modul. Chem. Mol. Sci. Chem. Eng., Elsevier, 2018: pp. 1–14. doi:10.1016/B978-0-12-409547-2.14345-8.

[7] J. Xu, E. Huang, L. Hsieh, L.H. Lee, Q.-S. Jia, C.-H. Chen, Simulation optimization in the era of Industrial 4.0 and the Industrial Internet, J. Simul. 10 (2016) 310–320. doi:10.1057/s41273-016-0037-6.

[8] A. Saltelli, K. Aleksankina, W. Becker, P. Fennell, F. Ferretti, N. Holst, S. Li, Q. Wu, Why so many published sensitivity analyses are false: A systematic review of sensitivity analysis practices,

Environ. Model. Softw. 114 (2019) 29–39. doi:10.1016/j.envsoft.2019.01.012.

[9] M. Henze, W. Gujer, T. Mino, M.C.M. van Loosdrecht, Activated Sludge Models ASM1, ASM2, ASM2d and ASM3, IWA Publ. (2000) 121. doi:10.1007/s13398-014-0173-7.2.

[10] U. Jeppsson, M.-N. Pons, I. Nopens, J. Alex, J.B. Copp, K.V. Gernaey, C. Rosen, J.-P. Steyer, P.A. Vanrolleghem, Benchmark simulation model no 2: general protocol and exploratory case studies, Water Sci. Technol. 56 (2007) 67–78. doi:10.2166/wst.2007.604.

[11] K. V. Gernaey, U. Jeppsson, P.A. Vanrolleghem, J.B. Copp, Benchmarking of Control Strategies for Wastewater Treatment Plants, IWA Publishing, 2014. doi:10.2166/9781780401171.

[12] G. Sin, K. V. Gernaey, M.B. Neumann, M.C.M. van Loosdrecht, W. Gujer, Uncertainty analysis in WWTP model applications: A critical discussion using an example from design, Water Res. 43 (2009) 2894–2906. doi:10.1016/j.watres.2009.03.048.

[13] D. Puyol, D.J. Batstone, T. Hülsen, S. Astals, M. Peces, J.O. Krömer, Resource Recovery from Wastewater by Biological Technologies: Opportunities, Challenges, and Prospects, Front. Microbiol. 7 (2017) 1–23. doi:10.3389/fmicb.2016.02106.

[14] N. Diaz-Elsayed, N. Rezaei, T. Guo, S. Mohebbi, Q. Zhang, Wastewater-based resource recovery technologies across scale: A review, Resour. Conserv. Recycl. 145 (2019) 94–112. doi:10.1016/j.resconrec.2018.12.035.

[15] Y. Shen, J.L. Linville, M. Urgun-Demirtas, M.M. Mintz, S.W. Snyder, An overview of biogas production and utilization at full-scale wastewater treatment plants (WWTPs) in the United States: Challenges and opportunities towards energy-neutral WWTPs, Renew. Sustain. Energy Rev. 50 (2015) 346–362. doi:10.1016/j.rser.2015.04.129.

[16] R. Al, C.R. Behera, K. V. Gernaey, G. Sin, Towards development of a decision support tool for conceptual design of wastewater treatment plants using stochastic simulation optimization, in: 29th

Eur. Symp. Comput. Aided Process Eng., Elsevier Masson SAS, 2019: pp. 325–330. doi:10.1016/B978-0-12-818634-3.50055-2.

[17] S.J. Qin, L.H. Chiang, Advances and opportunities in machine learning for process data analytics, Comput. Chem. Eng. 126 (2019) 465–473. doi:10.1016/j.compchemeng.2019.04.003.

[18] D. Torregrossa, U. Leopold, F. Hernández-Sancho, J. Hansen, Machine learning for energy cost modelling in wastewater treatment plants, J. Environ. Manage. 223 (2018) 1061–1067. doi:10.1016/j.jenvman.2018.06.092.

[19] L. Chiang, B. Lu, I. Castillo, Big Data Analytics in Chemical Engineering, Annu. Rev. Chem. Biomol. Eng. 8 (2017) annurev-chembioeng-060816-101555. doi:10.1146/annurev-chembioeng-060816-101555.

[20] J.H. Lee, J. Shin, M.J. Realff, Machine Learning: Overview of the Recent Progresses and Implications for the Process Systems Engineering Field, Comput. Chem. Eng. (2017). doi:10.1016/j.compchemeng.2017.10.008.

[21] V. Venkatasubramanian, The promise of artificial intelligence in chemical engineering: Is it here, finally?, AIChE J. 65 (2019) 466–478. doi:10.1002/aic.16489.

[22] S. Cremaschi, A perspective on process synthesis: Challenges and prospects, Comput. Chem. Eng. 81 (2015) 130–137. doi:10.1016/j.compchemeng.2015.05.007.

[23] J.J. Siirola, D.F. Rudd, Computer-Aided Synthesis of Chemical Process Designs. From Reaction Path Data to the Process Task Network, Ind. Eng. Chem. Fundam. 10 (1971) 353–362. doi:10.1021/i160039a003.

[24] J.M. Douglas, A hierarchical decision procedure for process synthesis, AIChE J. 31 (1985) 353–362. doi:10.1002/aic.690310302.

[25] L.T. Biegler, I.E. Grossmann, A.W. Westerberg, Systematic methods for chemical process design, Prentice Hall, Old Tappan, NJ (United States), United States, 1997. http://www.osti.gov/scitech/servlets/purl/293030.

[26] J. Ryu, L. Kong, A.E. Pastore de Lima, C.T. Maravelias, A generalized superstructure-based framework for process synthesis, Comput. Chem. Eng. 133 (2020) 106653. doi:10.1016/j.compchemeng.2019.106653.

[27] E. Ahmetović, I.E. Grossmann, Global superstructure optimization for the design of integrated process water networks, AIChE J. 57 (2011) 434–457. doi:10.1002/aic.12276.

[28] C.L. Gargalo, P. Cheali, J.A. Posada, K. V. Gernaey, G. Sin, Economic Risk Assessment of Early Stage Designs for Glycerol Valorization in Biorefinery Concepts, Ind. Eng. Chem. Res. 55 (2016) 6801–6814. doi:10.1021/acs.iecr.5b04593.

[29] M.J. Andrecovich, A.W. Westerberg, An MILP formulation for heat-integrated distillation sequence synthesis, AIChE J. 31 (1985) 1461–1474. doi:10.1002/aic.690310908.

[30] R. Lotfi, R.B. Boozarjomehry, Superstructure Optimization in Heat Exchanger Network (HEN) Synthesis Using Modular Simulators and a Genetic Algorithm Framework, Ind. Eng. Chem. Res. 49 (2010) 4731–4737. doi:10.1021/ie901215w.

[31] M.-O.O. Bertran, R. Frauzem, A.-S.S. Sanchez-Arcilla, L. Zhang, J.M. Woodley, R. Gani, A generic methodology for processing route synthesis and design based on superstructure optimization, Comput. Chem. Eng. 106 (2017) 892–910. doi:10.1016/j.compchemeng.2017.01.030.

[32] A. Quaglia, C.L. Gargalo, S. Chairakwongsa, G. Sin, R. Gani, Systematic network synthesis and design: Problem formulation, superstructure generation, data management and solution, Comput. Chem. Eng. 72 (2015) 68–86. doi:10.1016/j.compchemeng.2014.03.007.

[33] C.A. Henao, C.T. Maravelias, Surrogate-based process synthesis, Elsevier B.V., 2010. doi:10.1016/S1570-7946(10)28189-0.

[34] C.A. Henao, C.T. Maravelias, Surrogate-based superstructure optimization framework, AIChE J. 57 (2011) 1216–1232. doi:10.1002/aic.12341.

[35] M.M. Daichendt, I.E. Grossmann, Integration of hierarchical decomposition and mathematical programming for the synthesis of process flowsheets, Comput. Chem. Eng. 22 (1998) 147–175. doi:10.1016/S0098-1354(97)88451-7.

[36] W. Marquardt, S. Kossack, K. Kraemer, A Framework for the Systematic Design of Hybrid Separation Processes, Chinese J. Chem. Eng. 16 (2008) 333–342. doi:10.1016/S1004-9541(08)60084-1.

[37] J.R. Birge, F. Louveaux, Introduction to Stochastic Programming, Springer New York, New York, NY, 2011. doi:10.1007/978-1-4614-0237-4.

[38] G.E. Paules IV, C.A. Floudas, Stochastic programming in process synthesis: A two-stage model with MINLP recourse for multiperiod heat-integrated distillation sequences, Comput. Chem. Eng. 16 (1992) 189–210. doi:10.1016/0098-1354(92)85006-T.

[39] P. Liu, E.N. Pistikopoulos, Z. Li, Decomposition Based Stochastic Programming Approach for Polygeneration Energy Systems Design under Uncertainty, Ind. Eng. Chem. Res. 49 (2010) 3295–3305. doi:10.1021/ie901490g.

[40] Z. Zhou, J. Zhang, P. Liu, Z. Li, M.C. Georgiadis, E.N. Pistikopoulos, A two-stage stochastic programming model for the optimal design of distributed energy systems, Appl. Energy. 103 (2013) 135–144. doi:10.1016/j.apenergy.2012.09.019.

[41] J. Steimel, S. Engell, Conceptual design and optimization of chemical processes under uncertainty by two-stage programming, Comput. Chem. Eng. 81 (2015) 200–217. doi:10.1016/j.compchemeng.2015.05.016.

[42] A. Ben-Tal, L. El Ghaoui, A. Nemirovski, Robust optimization. Princeton series in applied mathematics, 2009.

[43] C. Shang, X. Huang, F. You, Data-driven robust optimization based on kernel learning, Comput. Chem. Eng. 106 (2017) 464–479. doi:10.1016/j.compchemeng.2017.07.004.

[44] Y. Yuan, Z. Li, B. Huang, Nonlinear robust optimization for

process design, AIChE J. 64 (2018) 481–494. doi:10.1002/aic.15950.

[45] N.H. Lappas, C.E. Gounaris, Robust optimization for decision-making under endogenous uncertainty, Comput. Chem. Eng. 111 (2018) 252–266. doi:10.1016/j.compchemeng.2018.01.006.

[46] C. Ning, F. You, Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming, Comput. Chem. Eng. 125 (2019) 434–448. doi:10.1016/j.compchemeng.2019.03.034.

[47] I.E. Grossmann, R.M. Apap, B.A. Calfa, P. García-Herreros, Q. Zhang, Recent advances in mathematical programming techniques for the optimization of process systems under uncertainty, Comput. Chem. Eng. 91 (2016) 3–14. doi:10.1016/j.compchemeng.2016.03.002.

[48] I. Fahmi, S. Cremaschi, A prototype simulation-based optimization approach to model feedstock development for chemical process industry, Chem. Eng. Res. Des. 91 (2013) 1499–1507. doi:10.1016/j.cherd.2013.05.021.

[49] L.T. Biegler, New directions for nonlinear process optimization, Curr. Opin. Chem. Eng. 21 (2018) 32–40. doi:10.1016/j.coche.2018.02.008.

[50] G. Hüllen, J. Zhai, S.H. Kim, A. Sinha, M.J. Realff, F. Boukouvala, Managing uncertainty in data-driven simulation-based optimization, Comput. Chem. Eng. 136 (2020) 106519. doi:10.1016/j.compchemeng.2019.106519.

[51] M.C. Fu, C.C. Price, J. Zhu, F.S. Hillier, Handbook of Simulation Optimization, Springer New York, New York, NY, 2015. doi:10.1007/978-1-4939-1384-8.

[52] S. Amaran, N. V. Sahinidis, B. Sharda, S.J. Bury, Simulation optimization: a review of algorithms and applications, 4OR. 12 (2014) 301–333. doi:10.1007/s10288-014-0275-2.

[53] C. Audet, W. Hare, Derivative-Free and Blackbox Optimization, Springer International Publishing, Cham, 2017. doi:10.1007/978-

3-319-68913-5.

[54] A.I.J. Forrester, A. Sbester, A.J. Keane, Engineering Design via Surrogate Modelling, John Wiley & Sons, Ltd, Chichester, UK, 2008. doi:10.1002/9780470770801.

[55] A. Bhosekar, M. Ierapetritou, Advances in surrogate based modeling, feasibility analysis, and optimization: A review, Comput. Chem. Eng. 108 (2018) 250–267. doi:10.1016/j.compchemeng.2017.09.017.

[56] J.C. Eslick, B. Ng, Q. Gao, C.H. Tong, N. V. Sahinidis, D.C. Miller, A Framework for Optimization and Quantification of Uncertainty and Sensitivity for Developing Carbon Capture Systems, Energy Procedia. 63 (2014) 1055–1063. doi:10.1016/j.egypro.2014.11.113.

[57] Z.T. Wilson, N. V. Sahinidis, The ALAMO approach to machine learning, Comput. Chem. Eng. 106 (2017) 785–795. doi:10.1016/j.compchemeng.2017.02.010.

[58] M.M.F. Hasan, R.C. Baliban, J.A. Elia, C.A. Floudas, Modeling, Simulation, and Optimization of Postcombustion CO 2 Capture for Variable Feed Concentration and Flow Rate. 2. Pressure Swing Adsorption and Vacuum Swing Adsorption Processes, Ind. Eng. Chem. Res. 51 (2012) 15665–15682. doi:10.1021/ie301572n.

[59] F. Boukouvala, M.M.F. Hasan, C.A. Floudas, Global optimization of general constrained grey-box models: new method and its application to constrained PDEs for pressure swing adsorption, J. Glob. Optim. 67 (2017) 3–42. doi:10.1007/s10898-015-0376-2.

[60] I. Bajaj, S.S. Iyer, M.M. Faruque Hasan, A trust region-based two phase algorithm for constrained black-box and grey-box optimization with infeasible initial point, Comput. Chem. Eng. 116 (2018) 306–321. doi:10.1016/j.compchemeng.2017.12.011.

[61] A. Arora, I. Bajaj, S.S. Iyer, M.M.F. Hasan, Optimal synthesis of periodic sorption enhanced reaction processes with application to hydrogen production, Comput. Chem. Eng. 115 (2018) 89–111. doi:10.1016/j.compchemeng.2018.04.004.

[62] A. Arora, S.S. Iyer, I. Bajaj, M.M.F. Hasan, Optimal Methanol

Production via Sorption-Enhanced Reaction Process, Ind. Eng. Chem. Res. 57 (2018) 14143–14161. doi:10.1021/acs.iecr.8b02543.

[63] Z. Wang, M. Ierapetritou, Constrained optimization of black-box stochastic systems using a novel feasibility enhanced Kriging-based method, Comput. Chem. Eng. 118 (2018) 210–223. doi:10.1016/j.compchemeng.2018.07.016.

[64] C. Puchongkawarin, Y. Vaupel, M. Guo, N. Shah, D.C. Stuckey, B. Chachuat, Wastewater treatment: Toward the synthesis of wastewater recovery facilities using enviroeconomic optimization, in: Water-Food-Energy Nexus Process. Technol. Challenges, 2017: pp. 129–144.

[65] M. Poch, J. Comas, I. Rodríguez-Roda, M. Sànchez-Marrè, U. Cortés, Designing and building real environmental decision support systems, Environ. Model. Softw. 19 (2004) 857–873. doi:10.1016/j.envsoft.2003.03.007.

[66] R. Karuppiah, I.E. Grossmann, Global optimization of multiscenario mixed integer nonlinear programming models arising in the synthesis of integrated water networks under uncertainty, Comput. Chem. Eng. 32 (2008) 145–160. doi:10.1016/j.compchemeng.2007.03.007.

[67] M.J. Bagajewicz, D.C. Faria, On the appropriate architecture of the water/wastewater allocation problem in process plants, in: 2009: pp. 1–20. doi:10.1016/S1570-7946(09)70006-9.

[68] X. Flores, I. Rodríguez-Roda, M. Poch, L. Jiménez, R. Bañares-Alcántara, Systematic Procedure to Handle Critical Decisions during the Conceptual Design of Activated Sludge Plants, Ind. Eng. Chem. Res. 46 (2007) 5600–5613. doi:10.1021/ie061426a.

[69] M. Garrido-Baserba, R. Reif, F. Hernández, M. Poch, Implementation of a knowledge-based methodology in a decision support system for the design of suitable wastewater treatment process flow diagrams, J. Environ. Manage. 112 (2012) 384–391. doi:10.1016/j.jenvman.2012.08.013.

[70] M. Garrido-Baserba, R. Reif, M. Molinos-Senante, L. Larrea, A. Castillo, M. Verdaguer, M. Poch, Application of a multi-criteria

decision model to select of design choices for WWTPs, Clean Technol. Environ. Policy. 18 (2016) 1097–1109. doi:10.1007/s10098-016-1099-x.

[71] N. Alasino, M.C. Mussati, N. Scenna, Wastewater treatment plant synthesis and design, Ind. Chem. Eng. Res. 46 (2007) 7497–7512. doi:10.1021/ie0704905.

[72] N. Alasino, M.C. Mussati, N.J. Scenna, P. Aguirre, Wastewater Treatment Plant Synthesis and Design: Combined Biological Nitrogen and Phosphorus Removal, Ind. Eng. Chem. Res. 49 (2010) 8601–8612. doi:10.1021/ie1000482.

[73] J. Hakanen, K. Sahlstedt, K. Miettinen, Wastewater treatment plant design and operation under multiple conflicting objective functions, Environ. Model. Softw. 46 (2013) 240–249. doi:10.1016/j.envsoft.2013.03.016.

[74] H. Bozkurt, A. Quaglia, K. V. Gernaey, G. Sin, A mathematical programming framework for early stage design of wastewater treatment plants, Environ. Model. Softw. 64 (2015) 164–176. doi:10.1016/j.envsoft.2014.11.023.

[75] H. Bozkurt, M.C.M. van Loosdrecht, K. V. Gernaey, G. Sin, Optimal WWTP process selection for treatment of domestic wastewater – A realistic full-scale retrofitting study, Chem. Eng. J. 286 (2016) 447–458. doi:10.1016/j.cej.2015.10.088.

[76] A. Castillo, P. Cheali, V. Gómez, J. Comas, M. Poch, G. Sin, An integrated knowledge-based and optimization tool for the sustainable selection of wastewater treatment process concepts, Environ. Model. Softw. 84 (2016) 177–192. doi:10.1016/j.envsoft.2016.06.019.

[77] C. Puchongkawarin, C. Gomez-Mont, D.C. Stuckey, B. Chachuat, Optimization-based methodology for the development of wastewater facilities for energy and nutrient recovery, Chemosphere. 140 (2015) 150–158. doi:10.1016/j.chemosphere.2014.08.061.

[78] C.R. Behera, R. Al, K. V. Gernaey, G. Sin, A process synthesis tool for WWTP – An application to design sustainable energy recovery

facilities, Chem. Eng. Res. Des. 156 (2020) 353–370. doi:10.1016/j.cherd.2020.02.014.

[79] C.M. Marques, S. Moniz, J.P. de Sousa, A.P. Barbosa-Póvoa, A simulation-optimization approach to integrate process design and planning decisions under technical and market uncertainties: A case from the chemical-pharmaceutical industry, Comput. Chem. Eng. 106 (2017) 796–813. doi:10.1016/j.compchemeng.2017.04.008.

[80] R. Al, C.R. Behera, K. V. Gernaey, G. Sin, Stochastic simulation-based superstructure optimization framework for process synthesis and design under uncertainty, Comput. Chem. Eng. (under Rev. (2020).

[81] H. Bozkurt, Computer-aided Framework for Synthesis, Design and Retrofit of Wastewater Treatment Plants, PhD Thesis, Department of Chemical and Biochemical Engineering, Technical University of Denmark, 2015.

[82] A. Quaglia, An Integrated Business and Engineering Framework for Synthesis and Design of Processing Networks, PhD Thesis, Department of Chemical and Biochemical Engineering, Technical University of Denmark, 2013.

[83] M.-O. Bertran, Modelling , synthesis and analysis of biorefinery networks, PhD Thesis, Department of Chemical and Biochemical Engineering, Technical University of Denmark, 2017.

[84] C.R. Behera, A Decision Support Tool for Screening Novel WWT Processes, PhD Thesis, Department of Chemical and Biochemical Engineering, Technical University of Denmark, 2019.

[85] M. Gries, Methods for evaluating and covering the design space during early design development, Integr. VLSI J. 38 (2004) 131–183. doi:10.1016/j.vlsi.2004.06.001.

[86] M.D. McKay, R.J. Beckman, W.J. Conover, Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code, Technometrics. 21 (1979) 239–245. doi:10.1080/00401706.1979.10489755.

[87] T. Luukkonen, K. Věžníková, E.-T. Tolonen, H. Runtti, J. Yliniemi, T. Hu, K. Kemppainen, U. Lassi, Removal of ammonium from municipal wastewater with powdered and granulated metakaolin geopolymer, Environ. Technol. 39 (2018) 414–423. doi:10.1080/09593330.2017.1301572.

[88] C.R. Behera, D. Santoro, K. V. Gernaey, G. Sin, Organic carbon recovery modeling for a rotating belt filter and its impact assessment on a plant-wide scale, Chem. Eng. J. 334 (2018) 1965–1976. doi:10.1016/j.cej.2017.11.091.

[89] L. Guo, P.A. Vanrolleghem, Calibration and validation of an activated sludge model for greenhouse gases no. 1 (ASMG1): prediction of temperature-dependent N2O emission dynamics, Bioprocess Biosyst. Eng. 37 (2014) 151–163. doi:10.1007/s00449-013-0978-3.

[90] D.J. Batstone, J. Keller, I. Angelidaki, S.V. Kalyuzhnyi, S.G. Pavlostathis, A. Rozzi, W.T.M. Sanders, H. Siegrist, V.A. Vavilin, The IWA Anaerobic Digestion Model No 1 (ADM1), Water Sci. Technol. 45 (2002) 65–73. doi:10.2166/wst.2002.0292.

[91] WEF, Design of Water Resource Recovery Facilities, Sixth Edition, 6th editio, McGraw-Hill Education, New York, 2018. https://www.accessengineeringlibrary.com/content/book/9781260031188.

[92] Metcalf & Eddy Inc., G. Tchobanoglous, F.L. Burton, R. Tsuchihashi, H.D. Stensel, Wastewater Engineering: Treatment and Resource Recovery, 5th ed., McGraw-Hill Education, 2014.

[93] J. Jimenez, M. Miller, C. Bott, S. Murthy, H. De Clippeleir, B. Wett, High-rate activated sludge system for carbon management – Evaluation of crucial process mechanisms and design parameters, Water Res. 87 (2015) 476–482. doi:10.1016/j.watres.2015.07.032.

[94] A. Malovanyy, J. Yang, J. Trela, E. Plaza, Combination of upflow anaerobic sludge blanket (UASB) reactor and partial nitritation/anammox moving bed biofilm reactor (MBBR) for municipal wastewater treatment, Bioresour. Technol. 180 (2015) 144–153. doi:10.1016/j.biortech.2014.12.101.

[95] R. Al, SPDLab - A decision support tool for simulation-based plant design of WWTPs, (2020). https://github.com/resulal/SPDLab.

[96] F. Ferretti, A. Saltelli, S. Tarantola, Trends in sensitivity analysis practice in the last decade, Sci. Total Environ. 568 (2016) 666–670. doi:10.1016/j.scitotenv.2016.02.133.

[97] I.M. Sobol, S.S. Kucherenko, Global sensitivity indices for nonlinear mathematical models. Review, Wilmott. 2005 (2005) 56–61. doi:10.1002/wilm.42820050114.

[98] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, S. Tarantola, Global Sensitivity Analysis. The Primer, in: Glob. Sensit. Anal. Prim., John Wiley & Sons, Ltd, Chichester, UK, 2008: pp. 237–275.

[99] E. Burnaev, I. Panin, B. Sudret, Efficient design of experiments for sensitivity analysis based on polynomial chaos expansions, (2017). http://arxiv.org/abs/1705.03944.

[100] B. Sudret, Global sensitivity analysis using polynomial chaos expansions, Reliab. Eng. Syst. Saf. 93 (2008) 964–979. doi:10.1016/j.ress.2007.04.002.

[101] A. Marrel, B. Iooss, B. Laurent, O. Roustant, Calculations of Sobol indices for the Gaussian process metamodel, Reliab. Eng. Syst. Saf. 94 (2009) 742–751. doi:10.1016/j.ress.2008.07.008.

[102] S. Li, B. Yang, F. Qi, Accelerate global sensitivity analysis using artificial neural network algorithm: Case studies for combustion kinetic model, Combust. Flame. 168 (2016) 53–64. doi:10.1016/j.combustflame.2016.03.028.

[103] X. Flores-Alsina, I. Rodríguez-Roda, G. Sin, K. V. Gernaey, Multi-criteria evaluation of wastewater treatment plant control strategies under uncertainty, Water Res. 42 (2008) 4485–4497. doi:10.1016/j.watres.2008.05.029.

[104] G. Sin, K. V. Gernaey, M.B. Neumann, M.C.M. van Loosdrecht, W. Gujer, Global sensitivity analysis in wastewater treatment plant model applications: Prioritizing sources of uncertainty, Water Res. 45 (2011) 639–651. doi:10.1016/J.WATRES.2010.08.025.

[105] X. Flores-Alsina, L. Corominas, M.B. Neumann, P.A. Vanrolleghem, Assessing the use of activated sludge process design guidelines in wastewater treatment plant projects: A methodology based on global sensitivity analysis, Environ. Model. Softw. 38 (2012) 50–58. doi:10.1016/j.envsoft.2012.04.005.

[106] J. Rojas, T. Zhelev, Energy efficiency optimisation of wastewater treatment: Study of ATAD, Comput. Chem. Eng. 38 (2012) 52–63. doi:10.1016/j.compchemeng.2011.11.016.

[107] A. Cosenza, G. Mannina, P.A. Vanrolleghem, M.B. Neumann, Global sensitivity analysis in wastewater applications: A comprehensive comparison of different methods, Environ. Model. Softw. 49 (2013) 40–52. doi:10.1016/j.envsoft.2013.07.009.

[108] A. Cosenza, G. Mannina, P.A. Vanrolleghem, M.B. Neumann, Variance-based sensitivity analysis for wastewater treatment plant modelling, Sci. Total Environ. 470–471 (2014) 1068–1077. doi:10.1016/j.scitotenv.2013.10.069.

[109] E. Ramin, X. Flores-Alsina, G. Sin, K. V. Gernaey, U. Jeppsson, P.S. Mikkelsen, B.G. Plósz, Influence of selecting secondary settling tank sub-models on the calibration of WWTP models – A global sensitivity analysis using BSM2, Chem. Eng. J. 241 (2014) 28–34. doi:10.1016/j.cej.2013.12.015.

[110] J.M. Dragan, A. Zubov, G. Sin, Methodology for Plantwide Design and Optimization of Wastewater Treatment Plants, Elsevier Masson SAS, 2017. doi:10.1016/B978-0-444-63965-3.50145-8.

[111] G. Mannina, A. Cosenza, G. Viviani, G.A. Ekama, Sensitivity and uncertainty analysis of an integrated ASM2d MBR model for wastewater treatment, Chem. Eng. J. 351 (2018) 579–588. doi:10.1016/j.cej.2018.06.126.

[112] M.P. Ochoa, V. Estrada, P.M. Hoch, Wastewater stabilisation ponds system: Global sensitivity analysis on network design, Chem. Eng. Trans. 50 (2016) 187–192. doi:10.3303/CET1650032.

[113] D.L.B. Fortela, W.W. Sharp, E.D. Revellame, R. Hernandez, D. Gang, M.E. Zappi, Computational evaluation for effects of feedstock variations on the sensitivities of biochemical mechanism

parameters in anaerobic digestion kinetic models, Biochem. Eng. J. 143 (2019) 212–223. doi:10.1016/j.bej.2019.01.001.

[114] M.D. Morris, Factorial Sampling Plans for Preliminary Computational Experiments, Technometrics. 33 (1991) 161–174. doi:10.2307/1269043.

[115] B. Iooss, P. Lemaître, A Review on Global Sensitivity Analysis Methods BT - Uncertainty Management in Simulation-Optimization of Complex Systems: Algorithms and Applications, in: G. Dellino, C. Meloni (Eds.), Springer US, Boston, MA, 2015: pp. 101–122. doi:10.1007/978-1-4899-7547-8_5.

[116] F. Pianosi, K. Beven, J. Freer, J.W. Hall, J. Rougier, D.B. Stephenson, T. Wagener, Sensitivity analysis of environmental models: A systematic review with practical workflow, Environ. Model. Softw. 79 (2016) 214–232. doi:10.1016/j.envsoft.2016.02.008.

[117] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, S. Tarantola, Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index, Comput. Phys. Commun. 181 (2010) 259–270. doi:10.1016/j.cpc.2009.09.018.

[118] T.A. Mara, S. Tarantola, P. Annoni, Non-parametric methods for global sensitivity analysis of model output with dependent inputs, Environ. Model. Softw. 72 (2015) 173–183. doi:10.1016/j.envsoft.2015.07.010.

[119] S. Kucherenko, S. Tarantola, P. Annoni, Estimation of global sensitivity indices for models with dependent variables, Comput. Phys. Commun. 183 (2012) 937–946. doi:10.1016/j.cpc.2011.12.020.

[120] R. Al, C.R. Behera, A. Zubov, G. Sin, Systematic framework development for the construction of surrogate models for wastewater treatment plants, in: Comput. Aided Chem. Eng., 2018: pp. 1909–1914. doi:10.1016/B978-0-444-64241-7.50313-X.

[121] Z. Wu, W. Wang, D. Wang, K. Zhao, W. Zhang, Global sensitivity analysis using orthogonal augmented radial basis function, Reliab. Eng. Syst. Saf. 185 (2019) 291–302. doi:10.1016/j.ress.2018.12.028.

[122] K. Cheng, Z. Lu, K. Zhang, Multivariate output global sensitivity analysis using multi-output support vector regression, Struct. Multidiscip. Optim. (2019). doi:10.1007/s00158-018-2184-z.

[123] N. Fajraoui, F. Ramasomanana, A. Younes, T.A. Mara, P. Ackerer, A. Guadagnini, Use of global sensitivity analysis and polynomial chaos expansion for interpretation of nonreactive transport experiments in laboratory-scale porous media, Water Resour. Res. 47 (2011) 1–14. doi:10.1029/2010WR009639.

[124] S. Marelli, B. Sudret, UQLab The Framework for Uncertainty Quantification, (2016) 2016.

[125] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression, J. Comput. Phys. 230 (2011) 2345–2367. doi:10.1016/j.jcp.2010.12.021.

[126] C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, MIT Press, Cambridge, Massachusetts, 2006.

[127] A. Géron, Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems, " O'Reilly Media, Inc.," 2017.

[128] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, Nature. 323 (1986) 533–536. doi:10.1038/323533a0.

[129] R. Al, C.R. Behera, A. Zubov, K. V. Gernaey, G. Sin, Meta-modeling based efficient global sensitivity analysis for wastewater treatment plants – An application to the BSM2 model, Comput. Chem. Eng. 127 (2019) 233–246. doi:10.1016/j.compchemeng.2019.05.015.

[130] S.E. Davis, S. Cremaschi, M.R. Eden, Efficient Surrogate Model Development: Impact of Sample Size and Underlying Model Dimensions, in: Comput. Aided Chem. Eng., Elsevier Masson SAS, 2018: pp. 979–984. doi:10.1016/B978-0-444-64241-7.50158-0.

[131] M. Steiner, J.M. Bourinet, T. Lahmer, An adaptive sampling method for global sensitivity analysis based on least-squares support vector regression, Reliab. Eng. Syst. Saf. 183 (2019) 323–

340. doi:10.1016/j.ress.2018.11.015.

[132] A.M. Molinaro, R. Simon, R.M. Pfeiffer, Prediction error estimation: A comparison of resampling methods, Bioinformatics. 21 (2005) 3301–3307. doi:10.1093/bioinformatics/bti499.

[133] U. Jeppsson, C. Rosen, J. Alex, J. Copp, K. V. Gernaey, M.N. Pons, P.A. Vanrolleghem, Towards a benchmark simulation model for plant-wide control strategy performance evaluation of WWTPs, Water Sci. Technol. 53 (2006) 287–295. doi:10.2166/wst.2006.031.

[134] M. Henze, C.P.L. Grady, W. Gujer, G.V.R. Maris, T. Matsuo, Activated sludge process model No. 1, 1987.

[135] R. Boiocchi, K. V. Gernaey, G. Sin, Understanding N2O formation mechanisms through sensitivity analyses using a plant-wide benchmark simulation model, Chem. Eng. J. 317 (2017) 935–951. doi:10.1016/j.cej.2017.02.091.

[136] I. Takács, G.G. Patry, D. Nolasco, A dynamic model of the clarification-thickening process, Water Res. 25 (1991) 1263–1271. doi:10.1016/0043-1354(91)90066-Y.

[137] A. Saltelli, S. Tarantola, F. Campolongo, M. Ratto, Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models, Chichester, John Wiley & Sons, Inc., 2004.

[138] E. Ramin, G. Sin, P.S. Mikkelsen, B.G. Plósz, Significance of settling model structures and parameter subsets in modelling WWTPs under wet-weather flow and filamentous bulking conditions, Water Res. 63 (2014) 209–221. doi:10.1016/j.watres.2014.05.054.

[139] J.A. Baeza, D. Gabriel, J. Lafuente, Effect of internal recycle on the nitrogen removal efficiency of an anaerobic/anoxic/oxic (A2/O) wastewater treatment plant (WWTP), Process Biochem. 39 (2004) 1615–1624. doi:10.1016/S0032-9592(03)00300-5.

[140] H. Li, Y. Zhang, M. Yang, Y. Kamagata, Effects of hydraulic retention time on nitrification activities and population dynamics of a conventional activated sludge system, Front. Environ. Sci. Eng. China. 7 (2013) 43–48. doi:10.1007/s11783-012-0397-8.

[141] R. Al, easyGSA - Efficient global sensitivity analysis using mechanistic or machine learning models, (2019). doi:10.17632/MCJ4J3RDX9.1.

[142] K.P. Kusumo, L. Gomoescu, R. Paulen, S. García Muñoz, C.C. Pantelides, N. Shah, B. Chachuat, Bayesian Approach to Probabilistic Design Space Characterization: A Nested Sampling Strategy, Ind. Eng. Chem. Res. 59 (2020) 2396–2408. doi:10.1021/acs.iecr.9b05006.

[143] S. Kucherenko, D. Giamalakis, N. Shah, S. García-Muñoz, Computationally efficient identification of probabilistic design spaces through application of metamodeling and adaptive sampling, Comput. Chem. Eng. 132 (2019) 106608. doi:10.1016/j.compchemeng.2019.106608.

[144] B. Ankenman, B.L. Nelson, J. Staum, Stochastic Kriging for Simulation Metamodeling, Oper. Res. 58 (2010) 371–382. doi:10.1287/opre.1090.0754.

[145] V. Picheny, D. Ginsbourger, Y. Richet, G. Caplin, Quantile-Based Optimization of Noisy Computer Experiments With Tunable Precision, Technometrics. 55 (2013) 2–13. doi:10.1080/00401706.2012.707580.

[146] J. Staum, Better simulation metamodeling: The why, what, and how of stochastic kriging, in: Proc. 2009 Winter Simul. Conf., IEEE, 2009: pp. 119–133. doi:10.1109/WSC.2009.5429320.

[147] J.P.C. Kleijnen, E. Mehdad, Estimating the variance of the predictor in stochastic Kriging, Simul. Model. Pract. Theory. 66 (2016) 166–173. doi:10.1016/j.simpat.2016.03.008.

[148] D.R. Jones, M. Schonlau, W.J. Welch, Efficient Global Optimization of Expensive Black-Box Functions, J. Glob. Optim. 13 (1998) 455–492. doi:10.1023/A:1008306431147.

[149] A.I.J. Forrester, A. Sbester, A.J. Keane, A. Sóbester, A.J. Keane, Engineering Design via Surrogate Modelling, Wiley, Chichester, UK, 2008. doi:10.1002/9780470770801.

[150] H. Jalali, I. Van Nieuwenhuyse, V. Picheny, Comparison of

Kriging-based algorithms for simulation optimization with heterogeneous noise, Eur. J. Oper. Res. 261 (2017) 279–301. doi:10.1016/j.ejor.2017.01.035.

[151] A.I.J. Forrester, A.J. Keane, Recent advances in surrogate-based optimization, Prog. Aerosp. Sci. 45 (2009) 50–79. doi:10.1016/j.paerosci.2008.11.001.

[152] M.H. Kalos, P.A. Whitlock, Monte Carlo Methods, Wiley, Dordrecht, 2009. doi:10.1002/9783527626212.

[153] R. Ghanem, H. Owhadi, D. Higdon, Handbook of Uncertainty Quantification, Springer International Publishing, Cham, 2017. doi:10.1007/978-3-319-12385-1.

[154] R.G. Regis, C.A. Shoemaker, Constrained global optimization of expensive black box functions using radial basis functions, J. Glob. Optim. 31 (2005) 153–171. doi:10.1007/s10898-004-0570-0.

[155] F. Boukouvala, M.G. Ierapetritou, Feasibility analysis of black-box processes using an adaptive sampling Kriging-based method, Comput. Chem. Eng. 36 (2012) 358–368. doi:10.1016/j.compchemeng.2011.06.005.

[156] F. Boukouvala, C.A. Floudas, ARGONAUT: AlgoRithms for Global Optimization of coNstrAined grey-box compUTational problems, Optim. Lett. 11 (2017) 895–913. doi:10.1007/s11590-016-1028-2.

[157] A. Habib, H.K. Singh, T. Ray, A study on the effectiveness of constraint handling schemes within Efficient Global Optimization framework, in: 2016 IEEE Symp. Ser. Comput. Intell., IEEE, 2016: pp. 1–8. doi:10.1109/SSCI.2016.7850205.

[158] R.R. Carpio, R.C. Giordano, A.R. Secchi, Enhanced surrogate assisted framework for constrained global optimization of expensive black-box functions, Comput. Chem. Eng. 118 (2018) 91–102. doi:10.1016/j.compchemeng.2018.06.027.

[159] R. Al, MCSKopt - Process simulation-based design space optimizer with Monte Carlo-based embeded uncertainty quantification, (2020). https://github.com/resulal/MCSKopt.

[160] M. McIntire, D. Ratner, S. Ermon, Sparse Gaussian processes for Bayesian optimization, 32nd Conf. Uncertain. Artif. Intell. 2016, UAI 2016. (2016) 517–526.

[161] H. Mohammadi, R. Le Riche, N. Durrande, E. Touboul, X. Bay, An analytic comparison of regularization methods for Gaussian Processes, (2016). http://arxiv.org/abs/1602.00853.

[162] K. Shimoyama, K. Sato, S. Jeong, S. Obayashi, Updating Kriging Surrogate Models Based on the Hypervolume Indicator in Multi-Objective Optimization, J. Mech. Des. 135 (2013) 1–7. doi:10.1115/1.4024849.

[163] J. Zhang, Y. Ma, T. Yang, L. Liu, Estimation of the Pareto front in stochastic simulation through stochastic Kriging, Simul. Model. Pract. Theory. 79 (2017) 69–86. doi:10.1016/j.simpat.2017.09.006.