ADVANCES IN DATA-DRIVEN MODELING AND GLOBAL OPTIMIZATION OF CONSTRAINED GREY-BOX COMPUTATIONAL SYSTEMS

A Dissertation

by

BURCU BEYKAL

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Efstratios N. Pistikopoulos
Committee Members,	Mahmoud M. El-Halwagi
	Eduardo Gildin
	M. M. Faruque Hasan
Head of Department,	Arul Jayaraman

May 2020

Major Subject: Chemical Engineering

ABSTRACT

The effort to mimic a chemical plant's operations or to design and operate a completely new technology *in silico* is a highly studied research field under process systems engineering. As the rising computation power allows us to simulate and model systems in greater detail through careful consideration of the underlying phenomena, the increasing use of complex simulation software and generation of multi-scale models that spans over multiple length and time scales calls for computationally efficient solution strategies that can handle problems with different complexities and characteristics. This work presents theoretical and algorithmic advancements for a range of challenging classes of mathematical programming problems through introducing new data-driven hybrid modeling and optimization strategies.

First, theoretical and algorithmic advances for bi-level programming, multi-objective optimization, problems containing stiff differential algebraic equations, and nonlinear programming problems are presented. Each advancement is accompanied with an application from the grand challenges faced in the engineering domain including, food-energy-water nexus considerations, energy systems design with economic and environmental considerations, thermal cracking of natural gas liquids, and oil production optimization.

Second, key modeling challenges in environmental and biomedical systems are addressed through employing advanced data analysis techniques. Chemical contaminants created during environmental emergencies, such as hurricanes, pose environmental and health related risks for exposure. The goal of this work is to alleviate challenges associated with understanding contaminant characteristics, their redistribution, and their biological potential through the use of data analytics.

DEDICATION

To my mother Sıdıka, and my father Ömer.

ACKNOWLEDGMENTS

This dissertation was made possible with the tremendous amount of support and guidance provided by my Ph.D. advisor, Professor Efstratios N. Pistikopoulos. Throughout the past 5 years, Professor Pistikopoulos has been my strongest advocate academically and personally, encouraging me to seek excellence in all aspects of my life, and always pushing me to live up to my fullest potential. Furthermore, I would like to thank my late advisor, Professor Christodoulos A. Floudas, whom I had the pleasure of working with in my first year of Ph.D. studies before his sudden passing away. Professor Floudas was my inspiration to join process systems engineering and I have always considered myself lucky to be a part of his research group. Of many academic advisors that I have worked with, Professor Floudas was one of the key people alongside Professor Pistikopoulos, who shaped my academic career and whom I owe my greatest appreciation. I was truly honored to work with these two amazing scientists whom I learned a lot professionally and academically. I am forever grateful for the things that they have done for me and I will continue to cherish their teachings throughout my professional career and my personal life.

Within the past 5 years, I was extremely blessed to work with a number of talented and hard-working colleagues and mentors. I would like to first acknowledge the senior members of Floudas and Pistikopoulos research groups; Dr. Fani Boukouvala, Dr. Chris Kieslich, Dr. Yannis Guzman, Dr. Alexander M. Niziolek, Dr. Onur Onel, Dr. Logan Matthews, Dr. Nikolaos A. Diangelakis, Dr. Maria M. Papathanasiou, Dr. Richard Oberdieck, Dr. Ioana Nascu, Dr. Styliani Avraamidou, and Dr. Hari Ganesh. I would like to give special thanks to my friends and my former colleagues Dr. Onur Onel and Dr. Melis Onel who were pivotal in my transition to process systems engineering. They have always welcomed me to their home when I needed a couch to crash on. I always enjoyed seeking advice from Dr. Onur Onel and was happy to collaborate with him in one of his final research projects at Texas A&M. I also feel very fortunate for collaborating with Dr. Melis Onel whom I shared many fun memories at conferences. I very much enjoyed working on numerous projects with Dr. Melis Onel and honored to co-author many publications together.

I would like to express my sincerest gratitude to the current members of the Pistikopoulos research group. I started my Ph.D. studies together with Dr. Justin Katz, Coşar Doğa Demirhan, Barış Burnak, and William W. Tso. I was glad to be in the same boat with you all when we were struggling through assignments in our first year. I would also like to recognize Dr. Gerald S. Ogumerem and my colleagues Yuhe Tian, Iosif Pappas, Stefanos Baratsas, Denis Su-Feher, Christopher Gordon, and Cory Allen for many fruitful discussions at the office. Beyond the two research groups, I would also like to thank Dr. Aurora Vargas for always inviting me to game night and sharing fun stories at the office. I would also like to thank Salih Emre Demirel for making the best tea in town and for being a great host and a nice conversationalist.

Finally, I would like to thank my family; my father Dr. Ömer Beykal, my mother Dr. Sıdıka Beykal, my sister Duygu Beykal İz, my brother-in-law Mustafa Kemal İz, and my nephew Bulut İz, whom without their existence and their endless support, I wouldn't be where I am today. I am grateful for having such a hard-working family who taught me discipline and self-motivation at a very young age. I would also like to thank and acknowledge my closest friends; Cemre Özmenci, Miray Gizem Yılmaz and Dr. İrem Altan, whom I survived undergrad with and shared my graduate school journey despite the time differences and being on different continents. My family and my closest friends have always been on my side and I am very blessed to share their love and support, which I know without them, I would not have made this far.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professors Efstratios N. Pistikopoulos, Mahmoud M. El-Halwagi, and M. M. Faruque Hasan of the Artie McFerrin Department of Chemical Engineering, and Professor Eduardo Gildin of the Harold Vance Department of Petroleum Engineering.

The UNISIM reservoir model simulation presented in Chapter 5 is developed by Nadav Sorek and Hardikkumar Zalavadia under the guidance of Professor Eduardo Gildin. The experimental data in Section 6.1 is provided by Gopal Bera, Krisa Camargo, José L. Sericano, Yina Liu, Stephen T. Sweet, and Terry L. Wade under the guidance of Professor Anthony H. Knap of the Geochemical and Environmental Research Group at Texas A&M University. The experimental data in Section 6.2 is provided by Adam T. Szafran, Fabio Stossi, and Maureen G. Mancini under the guidance of Professor Michael A. Mancini of the Department of Molecular and Cellular Biology at Baylor College of Medicine. All other work conducted for the dissertation was completed by the student independently.

Funding Sources

Graduate study was supported by U.S. National Institutes of Health Superfund Research Program (NIH P42-ES027704), National Science Foundation (NSF CBET-1548540), the Texas A&M University Superfund Research Center and the Texas A&M Energy Institute. Portions of this work were conducted with the advanced computing resources provided by Texas A&M High Performance Research Computing. The dissertation contents are solely the responsibility of the grantee and do not necessarily represent the official views of the NIH. Further, NIH does not endorse the purchase of any commercial products or services mentioned in the publication.

NOMENCLATURE

AIC	Akaike Information Criterion
Al	Aluminum
ANN	Artificial Neural Network
ANTIGONE	Algorithms for coNTinuous/Integer Global Optimization of Nonlinear Equations
AR	Androgen Receptor
ARGONAUT	AlgoRithms for Global Optimization of coNstrAined grey-box compUTational problems
p-ARGONAUT	Parallel AlgoRithms for Global Optimization of coNstrAined grey-box compUTational problems
BARON	Branch-And-Reduce Optimization Navigator
BgP	Benzo(ghi)perylene
BHP	Bottom Hole Pressure
B-LP	Bi-level Linear Program
B-MILP	Bi-level Mixed-Integer Linear Program
B-MINLP	Bi-level Mixed-Integer Nonlinear Program
B-POP	Bi-level Parametric Optimization toolbox
CA	Crustal Abundance
CERAPP	Collaborative Estrogen Receptor Activity Prediction Project
CHP	Combined Heat and Power
CO_2	Carbon Dioxide
COBYLA	Constrained Optimization BY Linear Approximations
CV	Cross-Validation

CVMSE	Cross-Validation Mean Squared Error
DAE	Differential Algebraic Equation
DFO	Derivative-Free Optimization
DoE	Design of Experiments
DOMINO	Data-driven Optimization of bi-level Mixed-Integer NOnlinear Problems
EDC	Endocrine Disrupting Chemical
EPA	United States Environmental Protection Agency
ER	Estrogen Receptor
FEW-N	Food-Energy-Water Nexus
FM	Fowlkes-Mallows
FN	False Negative
FLA	Fluoranthene
FP	False Positive
GC-ECD	Gas Chromatography Electron Capture Detection
GC-MS-MS	Gas Chromatography Mass Spectrometry
GFP	Green Fluorescent Protein
HD	Hellinger Distance
ICP-MS	Inductively Coupled Plasma Mass Spectrometer
INLP	Integer Nonlinear Program
InP	Indeno(1,2,3-cd)pyrene
ISRES	Improved Stochastic Ranking Evolution Strategy
kNN	k-Nearest Neighbors
LHD	Latin Hypercube Design
LLP	Lower Level Problem
LP	Linear Program

MILP	Mixed-Integer Linear Program
MINLP	Mixed-Integer Nonlinear Program
MIQCP	Mixed-Integer Quadratically Constrained Program
MOO	Multi-Objective Optimization
MRST	MATLAB Reservoir Simulation Toolbox
NIEHS	National Institute of Environmental Health Sciences
NOMAD	Nonlinear Optimization by Mesh Adaptive Direct Search
NPV	Net Present Value
NSGA-II	Non-dominated Sorting Genetic Algorithm-II
OBBT	Optimality Based Bound Tightening
OC	Organochlorine Pesticides
ODE	Ordinary Differential Equation
OSCAR	Optimal Scenario Reduction lgorithm
PAH	Polycyclic Aromatic Hydrocarbon
PBDE	Polybrominated Diphenyl Ethers
PCB	Polychlorinated Biphenyl
PI	Pixel Intensity
PYR	Pyrene
QCP	Quadratically Constrained Program
QSAR	Quantitative Structure-Activity Relationship
RBF	Radial Basis Function
RF	Random Forest
RMSE	Root Mean Square Error
SOR	Secondary Oil Recovery
SPV	Solar Photovoltaic

SQP	Sequential Quadratic Programming
SVM	Support Vector Machines
TN	True Negative
TP	True Positive
ULP	Upper Level Problem
WC	Water-Cut

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION*

Many engineering design and optimization problems in the fields of mechanical, aerospace, civil, petroleum, chemical and biomedical engineering, and geosciences, are characterized by complex first principle models, in the form of large systems of nonlinear partial differential equations [21]. The aim of these rigorous and highly detailed models is to simulate industrial processes in such a way that any mechanical, chemical and/or physical phenomena, which spans over multiple length and time scales, is captured with highest accuracy. In such complex systems, locating the globally optimal solution poses a formidable challenge due to the lack of analytical mathematical forms (i.e., simulation or proprietary model dependence) or due to the noise and/or computational expense associated with the calculation or approximation of the derivatives. These are commonly referred to as "grey-box" or "black-box" problems, where the entirety or a portion of the system characteristics are provided in the form of input-output data. In this dissertation, a special attention is given to; (1) constrained optimization of grey-box/black-box problems, (2) using surrogate modeling and high-performance computing for the explicit handling of constraints in these systems, (3) employing algorithmic features of grey-box optimization strategies to postulate theoretical advances in solving challenging classes of mathematical programming formulations, and (4) developing data-driven predictive models for environmental and biological systems.

The goal of this chapter is to provide an introduction to grey-box optimization (Section 1.1), discuss previous algorithmic advances for the constrained optimization of these problems (Section 1.2), state the challenges and open research questions existing in this field (Section 1.3), and deliver the objectives and the structure of this dissertation (Section 1.4).

*Part of this chapter is reprinted with permissions from "Global optimization of grey-box computational systems using surrogate functions and application to highly constrained oil-field operations" by B. Beykal, F. Boukouvala, C.A. Floudas, N. Sorek, H. Zalavadia, E. Gildin, 2018. *Computers & Chemical Engineering*, vol. 114, pp. 99-110, Copyright [2018] by Elsevier and Copyright Clearance Center, and "DOMINO: Data-driven Optimization of bi-level Mixed-Integer NOnlinear Problems" by B. Beykal, S. Avraamidou, I.P.E. Pistikopoulos, M. Onel, E.N. Pistikopoulos, 2020. *Journal of Global Optimization*, DOI: https://doi.org/10.1007/s10898-020-00890-3, Copyright [2020] by Springer Nature and Copyright Clearance Center.

1.1 Constrained Grey-Box Optimization

General constrained grey-box problems have the mathematical form described in Equation 1.1,

$$\begin{aligned} &\min_{\boldsymbol{x}} f(\boldsymbol{x}) \\ &s.t.\ g_m \leq 0 \ \ \forall m \in \{1, \ldots, M\} \\ &\quad g_k(\boldsymbol{x}) \leq 0 \ \ \forall k \in \{1, \ldots, K\} \\ &\quad x_i \in [x_i^L, x_i^U] \ i = 1, \ldots, n \\ &\quad \boldsymbol{x} \in \mathbb{R}^n \end{aligned} \tag{1.1}$$

where set $k \in \{1, \ldots, K\}$ represents the constraints with known closed-form (i.e., known constraints), and n represents the dimensionality of the problem or else, the number of decision variables, with known lower and upper bounds $[\boldsymbol{x}^L, \boldsymbol{x}^U]$. The mechanistic expressions defining the objective, $f(\boldsymbol{x})$, and the constraints, represented by set $m \in \{1, \ldots, M\}$, are not explicitly available as a function of the continuous decision variables. However, the values of these unknown formulations can be retrieved as outputs to the problem simulator, which is typically computationally expensive.

This class of problems is tackled using data-driven or derivative-free optimization (DFO) techniques where the derivative information of the original formulation is not utilized to get the optimal solution [22]. A typical DFO procedure starts with an initial design of experiments on the decision variables $\boldsymbol{x}$, which provides a set of pre-determined locations for evaluating the system and collecting the corresponding outputs (objective function value and constraint violations) from the simulated high-fidelity model. This input-output data will be further used by the data-driven optimizer to find the true optimum of the original model either through (a) a purely sample-based methodology, which only uses function-call data guided by pattern-based rules; or (b) a hybrid methodology (model-based methods), which uses samples in order to fit parametric functions that are subsequently used as surrogates of the original optimization formulation. Many algorithmic advances have been made in the last decade for data-driven grey-box optimization of both box-constrained

problems [23, 24] and general constrained problems [25–27] including, the ARGONAUT framework [28–30], the ALAMO framework [31, 32] and the SO-MI algorithm [33]. Further details on DFO and other algorithmic advances in this field can be found in the textbook by Conn et al. [22], which introduces the rich theory of sample-based DFO, and in several recent and valuable review articles and surveys, including a review by Kolda et al. [34] on sample-based methods, by Rios and Sahinidis [35] on box-constrained DFO and comparison of software implementations, by Boukouvala et al. [21] on constrained DFO, and by Bhosekar and Ierapetritou [36] and Vu et al. [37] on surrogate-based DFO.

1.2 Literature Review on Constraint Handling Strategies in Grey-Box Optimization

1.2.1 Constraint Handling in Search-based Methods

Traditionally, constraint handling in search-based methods has been done through augmented Lagrangian formulations [38–40], penalty methods [41–44] and restoration steps [45, 46]. Recently, Di Pillo et al. [47] introduced a DIRECT-type approach for the global optimization of general constrained optimization problems without using the derivatives. The authors make use of the well-known DIRECT algorithm and further combine it with a constrained derivative-free local minimization algorithm for improved solutions, where the nonlinear constraints are handled via an exact penalty function. In another study, Liuzzi et al. [48] employ an exact merit function to penalize the nonlinear constraints while converting the original constrained problem to a box-constrained problem in a multi-objective optimization framework.

1.2.2 Constraint Handling in Model-based Methods

In model-based approaches, the unknown constraints can be handled through surrogate models. Regis and Shoemaker [49] and Müller et al. [33], have proposed to optimize costly black-box systems using radial basis functions to create inexpensive approximations for the objective and the constraints. In another study by Bajaj et al. [26], the authors have introduced a trust-region based two-phase algorithm for the constrained optimization of grey/black-box problems. The two-phase algorithm starts with a feasible point identification, and proceeds with the optimization step, where

cubic radial basis function (RBF) with linear polynomial tail is used as a surrogate to approximate any unknown equation. Furthermore, several noteworthy studies have used local kriging approximations [50], local linear approximations [51, 52] and quadratic models [53] for handling constraints in grey/black-box optimization problems. In a more recent study, Müller and Shoemaker [54] showed that the selection of the surrogate function type affects the accuracy of obtaining the optimal solution, by taking into consideration the combination of different surrogate functions. A recent review describes advances in constrained DFO theory, applications, literature, algorithms and software along with advances in Mixed-Integer Nonlinear Optimization (MINLP) and their potential interactions [21].

1.3 Challenges in Grey-Box Optimization

Despite the many advances in the past and recent literature, there still exist several challenges towards the development of efficient DFO methods and algorithms. First, guarantee of convergence to ϵ-global optimality has not been achieved by any method within a finite number of steps. Second, DFO methods suffer from the curse-of-dimensionality, since sampling requirements and the number of parameters of surrogate models increase at high rates with the number of dimensions. Third, the presence of a large number of grey/black-box constraints within the optimization formulation cannot be handled efficiently by most existing DFO frameworks. Last, efficient methods are needed which can optimize hybrid problems comprised of both unknown information and mathematically known functions, in a way that maximizes the communication between the known and unknown components.

1.4 Dissertation Objectives and Structure

The goal of this dissertation is to present theoretical and algorithmic advances towards alleviating a subset of the aforementioned challenges in grey-box optimization, postulating novel strategies for solving challenging classes of mathematical programming problems, and extending these data-driven modeling capabilities to applications in the environmental and biomedical sciences domain. The challenging class of mathematical programming problems that are investigated in this

dissertation include bi-level optimization, multi-objective optimization, stiff differential algebraic equations (DAEs), and nonlinear nonconvex optimization. Specific objectives of this dissertation are further listed below.

1. Develop a data-driven optimization framework for solving bi-level mixed-integer nonlinear programming problems with guaranteed feasibility.

2. Establish a hybrid framework for solving multi-objective programming problems through reformulation and grey-box optimization strategies.

3. Introduce a Support Vector Machine-based constraint handling scheme for handling the stiffness in multi-dimensional DAE systems.

4. Enable distributed computing for the parallel execution of a grey-box optimization solver such that a realistic high-dimensional highly constrained black-box problem is solved to optimality.

5. Employ exploratory data analysis techniques for an effective interpretation of environmental contaminants and for the diagnosis of their potential pathways for redistribution.

6. Create predictive data-driven models for understanding the biological responses of environmental contaminants.

The following sections summarize the theoretical and algorithmic advances presented in this dissertation alongside the application areas explored in each chapter.

1.4.1 Theoretical Advances in Data-Driven Modeling and Optimization

- Feasibility guarantee for special classes of bi-level mixed-integer nonlinear programming problems (Chapter 2).

- Feasibility guarantee for general constrained continuous multi-objective optimization problems (Chapter 3).

- Derivation of the stability constraint for ill-conditioned (i.e., stiff) DAE systems that do not have an analytical solution (Chapter 4).

- Feasibility guarantee for high-dimensional highly constrained grey/black-box optimization problems with expensive simulators (Chapter 5).

1.4.2 Algorithmic Advances in Mathematical Programming

- DOMINO algorithm for bi-level mixed-integer nonlinear optimization (Chapter 2).

- Data-driven multi-objective optimization using ϵ-constraint reformulation and grey-box optimization algorithms (Chapter 3).

- Support Vector Machine-based constraint handling scheme for the data-driven optimization of stiff DAE systems without the full discretization of the underlying first-principles model (Chapter 4).

- Parallelization of a grey-box optimization solver, namely the ARGONAUT algorithm, for solving high-dimensional highly constrained nonlinear programming problems (Chapter 5).

1.4.3 Application Areas

- Land allocation problem in Food-Energy-Water Nexus considerations (Chapter 2).

- Energy systems design under economic and environmental considerations (Chapter 3).

- Reactor design and operation for thermal cracking of natural gas liquids (Chapter 4).

- Water-flooding control operations for secondary oil recovery (Chapter 5).

- Visualization of environmental contaminants for facilitating the interpretation and diagnosis of potential pathways for redistribution in a post-hurricane event (Chapter 6).

- Characterization of the estrogenic potential of chemical compounds (Chapter 6).

2. DATA-DRIVEN BI-LEVEL MIXED-INTEGER NONLINEAR OPTIMIZATION WITH APPLICATIONS TO FOOD-ENERGY-WATER NEXUS*

The Data-driven Optimization of bi-level Mixed-Integer NOnlinear problems (DOMINO) framework is presented for addressing the optimization of bi-level mixed-integer nonlinear programming problems. In this framework, bi-level optimization problems are approximated as single-level optimization problems by collecting samples of the upper-level objective and solving the lower-level problem to global optimality at those sampling points. This process is done through the integration of the DOMINO framework with a grey-box optimization solver to perform design of experiments on the upper-level objective, and to consecutively approximate and optimize bi-level mixed-integer nonlinear programming problems that are challenging to solve using exact methods. The performance of DOMINO is assessed through solving numerous bi-level benchmark problems, a land allocation problem in Food-Energy-Water Nexus, and through employing different data-driven optimization methodologies, including both local and global methods. Although this data-driven approach cannot provide a theoretical guarantee to global optimality, we present an algorithmic advancement that can guarantee feasibility to large-scale bi-level optimization problems when the lower-level problem is solved to global optimality at convergence.

This chapter is organized as follows. In Section 2.1, a background on bi-level programming is provided along with a literature review on data-driven bi-level optimization. The DOMINO framework is introduced in Section 2.2. Furthermore, in Section 2.3, the results for an extensive set of benchmark studies are presented alongside the results of a large-scale case study of land allocation in Food-Energy-Water Nexus problem. Finally, the concluding remarks are provided in Section 2.4.

*Part of this chapter is reprinted with permission from "DOMINO: Data-driven Optimization of bi-level Mixed-Integer NOnlinear Problems" by B. Beykal, S. Avraamidou, I.P.E. Pistikopoulos, M. Onel, E.N. Pistikopoulos, 2020. *Journal of Global Optimization*, DOI: https://doi.org/10.1007/s10898-020-00890-3, Copyright [2020] by Springer Nature and Copyright Clearance Center.

2.1 Multi-level Programming

Multi-level programming is a class of mathematical optimization with hierarchical structures, where one optimization problem is constrained by other optimization problems. It arises in the presence of multiple decision makers, where each of them is concerned with optimizing its own objective function. As a result, multi-level programming problems are encountered in many different application areas, including supply chain planning [55, 56], scheduling [57–59], government policy decision [60], price setting problems [61, 62], economics [63], and other multi-stage decision making problems [64, 65].

This chapter presents a data-driven framework for the solution of bi-level mixed-integer nonlinear problems with the general mathematical form shown in Equation 2.1. The considered class of problems contain two optimization levels with $F(\boldsymbol{x}, \boldsymbol{y})$ and $f(\boldsymbol{x}, \boldsymbol{y})$ representing the objective functions of the upper and lower-level problems, respectively. The upper-level problem (ULP) is constrained by the inequality $\boldsymbol{G}(\boldsymbol{x}, \boldsymbol{y})$, whereas the lower-level problem (LLP) is constrained both by the inequality $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y})$ and the equality constraint $\boldsymbol{h}(\boldsymbol{y})$, where $\boldsymbol{y}$ is a vector of continuous and/or integer variables strictly controlled by the LLP, and $\boldsymbol{x}$ is a vector of continuous variables strictly controlled by the ULP. It is worth noting here that the developed framework cannot address bi-level problems with upper-level integer variables, although lower-level integer variables can appear in the ULP.

$$
\begin{aligned}
\min_{\boldsymbol{x}} \quad & F(\boldsymbol{x}, \boldsymbol{y}) \\
\text{s.t.} \quad & \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{y}) \leq \boldsymbol{0} \\
& \boldsymbol{y} \in \underset{\boldsymbol{y}}{\operatorname{argmin}} \{ f(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}) \leq \boldsymbol{0}, \boldsymbol{h}(\boldsymbol{y}) = \boldsymbol{0} \} \\
& [x_1, ..., x_n] \in \mathbb{R}^n \\
& [y_1, ..., y_p] \in \mathbb{R}^p, [y_{p+1}, ..., y_r] \in \mathbb{Z}^{r-p}
\end{aligned}
\tag{2.1}
$$

This hierarchical structure can be viewed as a Stackelberg game [66, 67] where the upper-level objective will lead and decide on the decision variables $\boldsymbol{x}$, and the lower-level decision maker will then follow the leader by reacting accordingly, choosing the optimal values for $\boldsymbol{y}$ to opti-

mize its own objective function. Previously, the solutions of bi-level and multi-level programming problems have been studied extensively using branch and bound algorithms [68–72] and multi-parametric optimization techniques [73–80]. Although the aforementioned studies represent important theoretical advances for retrieving either ϵ-optimal or exact solutions of bi-level and multi-level optimization problems, the primary goal of this work is to tackle problems where the deterministic solution strategies cannot be applied due to the highly nonlinear nonconvex nature of many two-level large-scale optimization problems (i.e., problems that contain high number of variables and/or constraints).

To this end, many studies have focused on implementing evolutionary algorithms (i.e., genetic and meta-heuristic algorithms) and trust-region approaches to solve problems with multiple nested layers as presented in the detailed review by Sinha et al. [81]. Although evolutionary algorithms are very-well established and can be applicable to bi-level optimization problems, these methodologies typically require a large number of function evaluations for convergence, which come with a significant computational burden. Furthermore, evolutionary algorithms are generally implemented to unconstrained or box-constrained problems which limit their applicability to many real-life, constrained optimization problems. Extensions of evolutionary algorithms are proposed in the literature for handling constraints using aggregated approaches, through penalty functions [82, 83] or Augmented Lagrangian techniques [84].

In fact, several novel genetic and evolutionary algorithms have been presented for the solution of integer linear bi-level problems [85, 86] but both of these studies cannot guarantee global optimality or feasibility. Further advances to genetic algorithms have also been presented for the solution of mixed-integer nonlinear bi-level problems in the last decade [87, 88]. However, the study by Hecheng and Yuping [87] is not applicable to bi-level programming problems with general nonlinear lower-level problems. In addition, similar to the integer linear algorithms, these nonlinear genetic algorithms [87, 88] cannot also guarantee global optimality or feasibility. As an alternative approach to evolutionary algorithms, Sinha et al. [81] suggested building a local single-level approximation of the bi-level problem using Artificial Neural Networks (ANNs). The

authors briefly discuss how local surrogate modeling efforts can be a useful tool for solving bi-level optimization problems. However, the challenges associated with training an ANN, such as the hyperparameter optimization, decisions on the architecture of the network, and the number of samples required for training are not addressed. Therefore, new algorithmic approaches are necessary for solving nonlinear nonconvex bi-level mixed-integer optimization problems with improved constraint handling capabilities and maximum computational efficiency.

Hence, in this work, a new data-driven optimization framework is proposed to alleviate the aforementioned challenges as well as to bridge the gaps in solving a special class of bi-level programming problems, as shown in Equation 2.1. To this end, the Data-driven Optimization of bi-level Mixed-Integer NOnlinear problems (DOMINO) algorithm is presented where this approach reformulates bi-level optimization problems into single-level approximations through collecting samples on the upper-level objective, while the lower-level is solved to global optimality at these sampling points. This data-driven approach enables the collected input-output information to be utilized by a grey-box optimization solver, where the upper-level objective is solved to optimality via a derivative-free optimization methodology. Through this work, the aim is to:

- Establish a powerful computational algorithm for solving large-scale bi-level mixed-integer nonlinear programming (B-MINLP) problems of the form provided in Equation 2.1, which are difficult to solve using deterministic algorithms,
- Test the framework on an extensive list of bi-level optimization benchmark problems,
- Assess the performance of different grey-box solvers on the benchmark problems,
- Utilize the framework for the optimization of a large scale bi-level engineering problem.

2.2 DOMINO Framework

The Data-driven Optimization of bi-level Mixed-Integer NOnlinear problems (DOMINO) framework solves the constrained bi-level mixed-integer nonlinear nonconvex optimization problems following a similar procedure as a generic grey-box optimization algorithm, where the novelty

of the work underlies in approximating the bi-level problem into a single-level grey-box optimization problem. A general overview of the algorithm is provided in Figure 2.1. Given a bi-level programming problem, the first step to DOMINO framework is to pass the dimensionality information of the ULP (i.e., number of upper-level decision variables, n, and their respective bounds) along with any known constraints (i.e., constraints that are explicitly and solely imposed on the upper-level decision variables) to the design of experiments, if the data-driven optimizer can explicitly handle this information. In the absence of such a capability, the known constraints are directly handled as grey-box constraints.

The dimensionality information of the ULP is further processed by the data-driven optimizer to identify an initial starting point or an initial design of experiments at random. The choice of starting with a random initial point or a random design of experiments strictly depends on the type of grey-box solver that is incorporated in the framework. Typically, local black/grey-box solvers, such as a direct search algorithm [41], start with random single initial point whereas global approaches like ARGONAUT [28, 29] create a random space-filling maximin Latin Hypercube Design within the provided bounds. Then, at each of these pre-determined candidate locations of $\boldsymbol{x}$, the corresponding optimal value of the LLP, $\boldsymbol{y}^*$, is determined using either a local solver such as CPLEX [89], or global MINLP solvers such as ANTIGONE [90–92] and BARON [93], depending on the problem type. CPLEX is implemented for linear (LP), mixed-integer linear (MILP), quadratically constrained (QCP), and mixed-integer quadratically constrained (MIQCP) programming problems, whereas BARON and ANTIGONE are implemented to general nonlinear (NLP) and mixed-integer nonlinear (MINLP) programming problems at the lower-level. Thus, the LLP is solved deterministically to global optimality at each iteration at the given upper-level sampling points. Later, the optimal solution of the LLP, $\boldsymbol{y}^*$, and the pre-determined sampling points will be used to enumerate the upper-level objective, $F(\boldsymbol{x}, \boldsymbol{y}^*)$, and the constraint violations of both levels, $\boldsymbol{G}(\boldsymbol{x}, \boldsymbol{y}^*)$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}^*)$. This input-output data will be further passed onto the derivative-free optimization stage to retrieve a candidate solution of the original bi-level programming problem once the DFO convergence criteria are met. If this returned solution violates any of the grey-box constraints, the

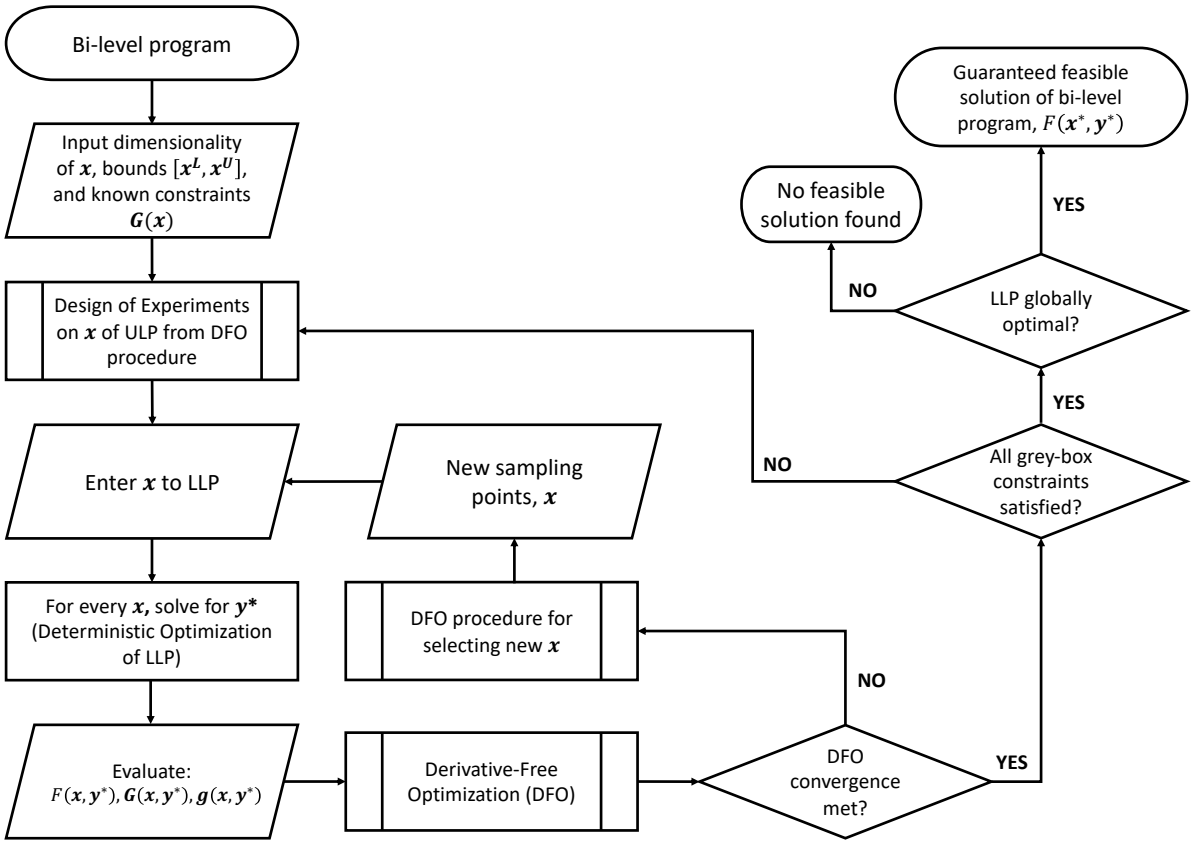

Figure 2.1: Algorithmic flowchart of the DOMINO framework. DOMINO is integrated with a DFO algorithm and a deterministic global optimizer for solving bi-level programming problems. The LLP is solved to global optimality at each iteration for a given vector of upper-level decision variables, $\boldsymbol{x}$ (input data). The objective function and the constraint violations (output data) that contain at least one upper-level variable are enumerated using the optimal solution $\boldsymbol{y}^*$ and the corresponding input upper-level decision variables $\boldsymbol{x}$. This input-output data is later passed to a DFO subroutine to retrieve a candidate solution of the bi-level programming problem.

algorithm is restarted to explore a feasible solution, starting with a new initial point/design. If all constraints are satisfied but the LLP is only locally optimal or feasible, then the algorithm will terminate without identifying a feasible solution to the bi-level programming problem. If the solution satisfies all grey-box constraints, and the LLP is globally optimal at the given solution, the solution is a guaranteed feasible point for the original bi-level programming problem.

DOMINO is a flexible algorithm where any type of data-driven optimizer (i.e., local versus global or sample-based versus model-based algorithms) and deterministic solver (i.e., CPLEX,

ANTIGONE, BARON) can be incorporated depending on the problem definition. This flexibility allows DOMINO to benefit from the advantages of different approaches and does not impose a strict form on the single-level approximation of different bi-level optimization problems. The most important properties of the DOMINO framework are listed as remarks below.

Remark 1. The proposed framework is tailored to handle special classes of bi-level optimization problems that are given in the form of Equation 2.1.

Remark 2. DOMINO cannot guarantee ϵ-global optimality to the upper-level objective. Although commercially available optimization solvers such as CPLEX, ANTIGONE [90–92], and BARON [93] are incorporated within the framework for the deterministic optimization of the LLP, the ULP is treated as a grey-box, where the explicit analytical formulation and the convexity of the problem is assumed to be unknown.

Remark 3. Feasibility of the bi-level programming problem is guaranteed at convergence if and only if a feasible solution for the ULP is identified by DOMINO and the lower-level converges to a globally optimal solution at the given upper-level solution. The feasibility guarantee is achieved by formulating all the upper-level variable-containing constraints, $\boldsymbol{G}(\boldsymbol{x}, \boldsymbol{y})$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y})$, as black/grey-box constraints where their respective violations are tracked throughout the DFO procedure. As the LLP is solved to global optimality deterministically at every iteration, the constraints with only lower-level variables (i.e., $\boldsymbol{h}(\boldsymbol{y}) = \boldsymbol{0}$), are satisfied for a feasible solution of an ULP. In addition, the lower-level feasibility is verified through an *a posteriori* analysis for the returned bi-level solution.

Remark 4. DOMINO framework can handle a wide range of dimensionality, including several hundred variables, and constraints in both upper and lower-level problems, and provide feasible near-optimal solutions to varying bi-level programming problem types.

Remark 5. When the optimal solution of the LLP is not unique for the vector of optimal upper-level variables, the decision maker can take a pessimistic decision, an optimistic decision or any decision in between. Although many other bi-level approaches can guarantee and characterize

the solution type as pessimistic or optimistic, the proposed framework is not able to provide this characterization.

Remark 6. DOMINO does not impose any extra criterion for convergence or re-sampling. These decisions solely depend on the data-driven optimizer that is integrated within the DOMINO framework and vary from one data-driven methodology to another.

In our previous study [94], the basic idea of this data-driven approach was tested using a single data-driven optimizer for solving a B-MINLP problem in Food-Energy-Water Nexus considerations. In this chapter, the properties of the framework that are listed here are further demonstrated on an extended class of benchmark problems and the number of problems solved to global optimality is improved. The framework is extended to include an array of data-driven optimizers, which are presented in the following section. In addition, the full formulation of the Food-Energy-Water Nexus case study, its reformulation to B-MILP problem using Big-M constraints, as well as its detailed computational study with DOMINO is provided.

2.3 Computational Studies

The proposed data-driven methodology for solving bi-level optimization problems is tested on a challenging set of 100 test problems and a land allocation case study. In this work, 4 different constrained data-driven optimization strategies are identified to be implemented in the DOMINO framework: (1) Nonlinear Optimization by Mesh Adaptive Direct search (NOMAD) [95]; (2) Constrained Optimization BY Linear Approximations (COBYLA) [51]; (3) AlgoRithms for Global Optimization of coNstrAined grey-box compUTational problems (ARGONAUT) [28–30]; and (4) Improved Stochastic Ranking Evolution Strategy (ISRES) [96]. The selection of these solvers is based on their ability to perform constrained optimization on black/grey-box problems as well as their difference in solution methodology, where both local (NOMAD and COBYLA) and global (ARGONAUT and ISRES) optimization strategies are investigated. Each algorithm is briefly described in Table 2.1. These DFO solvers are available and/or implemented in R statistical software. ARGONAUT is implemented in R, the NLopt implementation of ISRES and COBYLA [97] is available in "nloptr" library in R, and the NOMAD software is available at [98]. All the tested case

Table 2.1: Descriptions and the convergence criteria of data-driven algorithms tested in this study.

Algorithm Name	Description
NOMAD	Local optimization based on pattern method (search, poll and update). Convergence criteria: maximum number of samples reached, mesh size tolerance reached [99].
COBYLA	Constraint handling via progressive barrier approach. Local optimization using linear approximations for the objective and constraints by interpolation at the vertices of a simplex. Convergence criteria: maximum number of samples reached, minimum trust region radius is exceeded/reached, an optimization step causes a relative change in the decision variables less than the set tolerance [51, 97].
ARGONAUT	Global optimization using surrogate model identification for the objective and constraints. Convergence criteria: maximum number of samples reached, no improvement of the incumbent solution over a consecutive set of iterations, all unknown functions are modeled with high accuracy (i.e., very low cross-validation mean squared error) and the incumbent solution is feasible [28].
ISRES	Global optimization via evolutionary method; couples mutation rule and differential variation. Constraint handling via stochastic ranking. Convergence criteria: maximum number of samples reached, an optimization step causes a relative change in the decision variables less than the set tolerance [97].

studies are modeled in GAMS and interfaced through R, where the input-output data collection on each grey-box problem is performed via text files.

All benchmark problems and high-dimensional case studies are executed 10 times on a High-Performance Computing (HPC) machine at Texas A&M High-Performance Research Computing facility using Ada IBM/Lenovo Intel Xeon E5-2670 v2 (Ivy Bridge-EP) HPC Cluster operated with Linux (CentOS 6). COBYLA, ISRES and NOMAD algorithms are executed using 1 node (1 core per node with 64 GB RAM), whereas the ARGONAUT algorithm is executed as a parallel job, using 1 node (20 cores per node with 64 GB RAM) on the supercomputer. Furthermore, for a fair comparison of results, the starting points of COBYLA, ISRES, and NOMAD are randomly generated, as well as the starting initial design of experiments for ARGONAUT is randomly determined for each run. In addition, all data-driven solvers are tested and implemented at their default setting provided from [97, 98], with the exception of ARGONAUT. By default, ARGONAUT sets the number of initial sampling points to $10k + 1$ for $k \leq 20$ and to 251 when $k > 20$, where k is the dimensionality of the problem (i.e., number of inputs). Since for $k \leq 2$, the number of

initial samples is not sufficient to reveal the input-output relationship for both levels in a bi-level programming problem, the number of initial points to be collected is increased to $40k + 1$. For problems with dimensionality $2 < k \leq 20$ and $k > 20$, the default values are implemented.

2.3.1 Benchmark Problems

The comprehensive test set from Mitsos and Barton [100] (Errata: from Paulavicius et al. [101]), as well as individual bi-level programming problems from Edmunds and Bard [102], Sahin and Ciric [103], Gümüş and Floudas [68], Colson [104], Mitsos [105], Kleniati and Adjiman [106], Woldemariam and Kassa [107], and Nie et al. [108] are used for assessing the performance of the DOMINO framework and for comparing the performance of different data-driven optimizers in finding the true global solution of the bi-level programming problems. In addition to this set, 61 benchmark studies are randomly generated using the bi-level random problem generator in B-POP toolbox [77] and are solved to global optimality, where the formulation of these are provided in Appendix A. The selection of the benchmark problems aim to cover various different types of bi-level optimization problems with varying dimensionalities in both upper and lower level problems. Especially for the problems generated by B-POP, the computational complexity of the test problems are limited to the dimensionalities that this solver can handle, so as to establish a basis for comparison and to be able to assess the performance of DOMINO accurately throughout the benchmark problems.

The dimensionality of each problem and their corresponding properties are provided in Tables 2.2, 2.3 and 2.4, where n represents the number of upper-level continuous variables, p and $r-p$ represents the lower-level dimensionality (continuous and integer, respectively) and n_g^{grey} represents the number of grey-box constraints for each problem. The number of grey-box constraints shown here is the sum of the number of the upper-level constraints and the lower-level constraints that include at least one upper-level variable in its mathematical form. This criterion is imposed since the LLP is solved deterministically within the framework, where the optimal solution already satisfies the constraints with only lower-level decision variables. This eliminates redundant model building or point search in the optimization phase, which speeds up the computational time required for

Table 2.2: Dimensionality of continuous bi-level linear benchmark problems tested with DOMINO.

Problem ID [Source]	Label	Problem Type (Upper-Lower)	n	p	$r-p$	n_g^{grey}	n_g^y
1 [103]	sc_1	LP-LP	1	2	0	3	0
2 [77]	LPLP1	LP-LP	2	2	0	2	0
3 [77]	LPLP2	LP-LP	2	2	0	5	2
4 [77]	LPLP3	LP-LP	5	5	0	2	0
5 [77]	LPLP4	LP-LP	10	10	0	4	0
6 [77]	LPLP5	LP-LP	20	500	0	350	0
7 [77]	LPLP6	LP-LP	20	20	0	4	0
8 [77]	LPLP7	LP-LP	20	30	0	5	0
9 [77]	LPLP8	LP-LP	20	50	0	7	0
10 [77]	LPLP9	LP-LP	20	80	0	7	0
11 [77]	LPLP10	LP-LP	40	150	0	10	0
12 [77]	LPLP11	LP-LP	50	200	0	20	0
13 [77]	LPLP12	LP-LP	80	90	0	3	0
14 [77]	LPLP13	LP-LP	200	200	0	200	0

convergence for all data-driven algorithms. In addition, an *a posteriori* analysis is performed on the LLP to ensure feasibility of the unmodeled constraints at convergence. The number of constraints with only the lower-level decision variables, hence not presented as grey-box constraints, are also provided in Tables 2.2, 2.3 and 2.4 under n_g^y.

The performance of each solver within DOMINO is assessed based on its efficiency and consistency in identifying the true global optimum of the benchmark studies over multiple repetitive runs. The accuracy and the consistency of each algorithm is evaluated by calculating the normalized mean absolute error (% MAE $= 100 \cdot |(F_{best} - F_{global})/F_{global}|$) of the best found solution with respect to the true global optimum and the standard deviation of this error over 10 runs, respectively. In the benchmark problems with $F_{global} = 0$, the percent absolute error (% MAE $= 100 \cdot |F_{best} - F_{global}|$) is calculated. It is important to note that 100% MAE is assigned for runs that returned an infeasible solution, constraint violation $\geq 10^{-6}$ and/or lower-level is not globally optimal (lower-level absolute optimality gap > 0 for LP, QP, MILP, MIQP-type lower-level problems and lower-level absolute optimality gap $\geq 10^{-6}$ for NLP and INLP-type lower-level problems),

Table 2.3: Dimensionality of continuous bi-level nonlinear benchmark problems tested with DOMINO.

Problem ID [Source]	Label	Problem Type (Upper-Lower)	n	p	$r-p$	n_g^{grey}	n_g^y
15 [100]	mb_1_1_06	LP-QP	1	1	0	0	0
16 [77]	LPQP1	LP-QP	30	60	0	10	0
17 [100]	mb_1_1_16	QP-QP	1	1	0	2	0
18 [107]	wk_2015_01	QP-QP	1	1	0	2	0
19 [68]	gf_4	QP-QP	1	1	0	3	0
20 [103]	sc_2	QP-QP	1	1	0	3	0
21 [68]	gf_2	NLP-QP	1	2	0	2	0
22 [100]	mb_2_3_02	NLP-QP	2	3	0	1*	2
23 [100]	mb_1_1_03	LP-NLP	1	1	0	0	0
24 [100]	mb_1_1_04	LP-NLP	1	1	0	0	0
25 [100]	mb_1_1_05	LP-NLP	1	1	0	0	0
26 [100]	mb_1_1_08	LP-NLP	1	1	0	0	0
27 [100]	mb_1_1_09	LP-NLP	1	1	0	0	0
28 [100]	mb_1_1_12	LP-NLP	1	1	0	0	0
29 [100]	mb_1_1_01	LP-NLP	1	1	0	0	2
30 [100]	mb_1_1_02	LP-NLP	1	1	0	1	0
31 [68]	gf_5	LP-NLP	1	2	0	1	1
32 [68]	gf_3	LP-NLP	2	3	0	2	1
33 [100]	mb_1_1_07	QP-NLP	1	1	0	0	0
34 [100]	mb_1_1_10	QP-NLP	1	1	0	0	0
35 [100]	mb_1_1_11	QP-NLP	1	1	0	0	0
36 [100]	mb_1_1_13	QP-NLP	1	1	0	0	0
37 [100]	mb_1_1_14	QP-NLP	1	1	0	0	0
38 [100]	mb_1_1_17	QP-NLP	1	1	0	0	0
39 [100]	mb_1_1_15	QP-NLP	1	1	0	1	0
40 [68]	gf_1	QP-NLP	1	1	0	2	0
41 [104]	c_2002_01	NLP-NLP	1	1	0	2	0
42 [104]	c_2002_03	NLP-NLP	1	1	0	2	0
43 [104]	c_2002_05	NLP-NLP	1	2	0	2	0
44 [108]	nwj_2017_02	NLP-NLP	2	3	0	1	2
45 [100]	mb_2_3_01	NLP-NLP	2	3	0	3	2
46 [107]	wk_2015_04	NLP-NLP	2	4	0	4	0
47 [107]	wk_2015_06	NLP-NLP	4	4	0	4	0
48 [106]	ka_2014_02	NLP-NLP	5	5	0	4	0
49 [100]	mb_5_5_01	NLP-NLP	5	5	0	4	2
50 [100]	mb_5_5_02	NLP-NLP	5	5	0	4	2

* This constraint is handled as “known" in ARGONAUT runs and as a grey-box constraint for other solvers.

Table 2.4: Dimensionality of bi-level mixed-integer benchmark problems tested with DOMINO.

Problem ID [Source]	Label	Problem Type (Upper-Lower)	n	p	$r-p$	n_g^{grey}	n_g^y
51 [105]	am_1_0_0_1_01	LP-ILP	1	0	1	0	0
52 [77]	LPMILP1	LP-MILP	10	10	10	4	0
53 [77]	LPMILP2	LP-MILP	10	10	10	4	0
54 [77]	LPMILP3	LP-MILP	20	20	10	2	0
55 [77]	LPMILP4	LP-MILP	30	30	30	4	0
56 [77]	QPMILP1	QP-MILP	5	5	5	4	1
57 [77]	QPMILP2	QP-MILP	10	5	5	5	0
58 [77]	QPMILP3	QP-MILP	10	10	6	3	0
59 [77]	QPMILP4	QP-MILP	20	10	5	2	3
60 [77]	QPMILP5	QP-MILP	22	12	7	5	0
61 [77]	QPMILP6	QP-MILP	25	20	15	3	0
62 [77]	QPMILP7	QP-MILP	25	25	10	6	0
63 [77]	QPMILP8	QP-MILP	30	120	120	120	0
64 [77]	QPMILP9	QP-MILP	30	200	200	250	0
65 [77]	NLPMILP1	NLP-MILP	5	8	6	9	1
66 [77]	NLPMILP2	NLP-MILP	10	10	10	10	0
67 [77]	NLPMILP3	NLP-MILP	15	15	15	14	1
68 [77]	NLPMILP4	NLP-MILP	20	20	20	20	0
69 [77]	NLPMILP5	NLP-MILP	25	30	30	30	0
70 [77]	NLPMILP6	NLP-MILP	25	50	50	50	0
71 [77]	NLPMILP7	NLP-MILP	30	70	70	70	0
72 [77]	NLPMILP8	NLP-MILP	30	100	100	100	0
73 [77]	NLPMILP9	NLP-MILP	30	200	200	200	0
74 [77]	LPMIQP1	LP-MIQP	7	7	6	1	0
75 [77]	LPMIQP2	LP-MIQP	7	7	6	1	0
76 [77]	LPMIQP3	LP-MIQP	10	7	6	1	0
77 [77]	LPMIQP4	LP-MIQP	10	7	6	1	0
78 [77]	LPMIQP5	LP-MIQP	10	10	6	1	0
79 [77]	LPMIQP6	LP-MIQP	10	13	6	1	0
80 [77]	LPMIQP7	LP-MIQP	10	13	6	1	0
81 [77]	LPMIQP8	LP-MIQP	12	13	6	1	0
82 [102]	eb_1	QP-IQP	1	0	1	3	0
83 [77]	QPMIQP1	QP-MIQP	5	20	10	1	0
84 [77]	QPMIQP2	QP-MIQP	6	5	2	3	0
85 [77]	QPMIQP3	QP-MIQP	6	5	3	4	0
86 [77]	QPMIQP4	QP-MIQP	6	5	5	4	0
87 [77]	QPMIQP5	QP-MIQP	10	3	3	3	0
88 [77]	QPMIQP6	QP-MIQP	10	30	7	1	0
89 [77]	QPMIQP7	QP-MIQP	10	40	7	1	0
90 [77]	NLPMIQP1	NLP-MIQP	5	5	2	0	3
91 [77]	NLPMIQP2	NLP-MIQP	7	5	3	3	0
92 [77]	NLPMIQP3	NLP-MIQP	9	6	3	2	0
93 [77]	NLPMIQP4	NLP-MIQP	11	7	5	2	0
94 [77]	NLPMIQP5	NLP-MIQP	12	10	10	1	0
95 [77]	NLPMIQP6	NLP-MIQP	12	11	10	0	1
96 [77]	NLPMIQP7	NLP-MIQP	12	11	5	1	0
97 [77]	NLPMIQP8	NLP-MIQP	12	12	6	1	0
98 [77]	NLPMIQP9	NLP-MIQP	13	9	8	1	0
99 [77]	NLPMIQP10	NLP-MIQP	15	15	4	1	0
100 [103]	sc_3	NLP-INLP	2	0	2	0	1

and their respective standard deviation of error is not calculated. Furthermore, the efficiency of the framework is evaluated based on the average elapsed time it takes for each solver to converge and based on the total number of function evaluations (i.e., samples) collected at convergence. The results for continuous linear, continuous nonlinear, mixed-integer linear and mixed-integer nonlinear bi-level programming problems are discussed in Sections 2.3.1.1, 2.3.1.2 and 2.3.1.3, respectively.

2.3.1.1 Results for Bi-Level Linear Programming Problems

The results of the bi-level linear benchmark problems are reported in Table 2.5. The overall performance of all grey-box solvers, tested as a part of the DOMINO framework, indicate that they return consistent feasible solutions with low errors to the bi-level linear programming (B-LP) problems. Specifically, it is observed that NOMAD, as a local sample-based grey-box optimization solver, outperforms the rest of the solvers in B-LP problems. Only in the benchmark problem with the highest number of upper-level variables, NOMAD returns an objective value with more than 5% average MAE. A similar trend is also observed in the ISRES algorithm, where at higher upper-level dimensionality benchmarks (i.e., 80 and 200 upper-level variables) the algorithm converges with high % MAE. One possible reason for this behavior in sample-based methodologies is reported in Figure 2.2B, where both NOMAD and ISRES algorithms converge and return the incumbent solution after hitting the maximum number of samples allowed (i.e., 10^5 samples) in all computational studies. Hence, by allowing these algorithms to collect more samples at high-dimensional B-LP problems, it is possible to get more consistent solutions with lower errors. Specifically, in problem 5 ("LPLP4"), it is observed that the ISRES algorithm hits the maximum number of samples even though a solution with 0.0000 average % MAE and 0.0000 average standard deviation of % MAE is found. This is due to the fact that the tolerance set for the criterion that defines the convergence with respect to the relative change in the decision variables is not met. Another optimization step taken by ISRES will result in a relative change in the decision variables that is greater than 10^{-6}. Hence for this specific case, ISRES algorithm terminates by reaching the maximum number of samples allowed. On the other hand, it is observed that model-based algorithms, such as COBYLA and ARGONAUT, can provide consistent near-optimal solutions

Table 2.5: Average % MAE and average standard deviation of % MAE for the bi-level linear programming problems. No infeasibility is reported by any of the grey-box solvers for this set of bi-level linear programming problems.

Problem	Average % MAE				Average Standard Deviation of % MAE			
ID	NOMAD	COBYLA	ARGONAUT	ISRES	NOMAD	COBYLA	ARGONAUT	ISRES
1	0.0000	16.1538	0.0007	0.0001	0.0000	26.0102	0.0007	0.0001
2	0.0000	0.0000	0.0000	0.0011	0.0000	0.0000	0.0000	0.0032
3	0.0000	8.0448	0.0000	0.0001	0.0000	10.3858	0.0000	0.0001
4	0.0000	0.0388	0.0000	0.0000	0.0000	0.1225	0.0000	0.0000
5	0.1044	6.4958	11.2746	0.0000	0.0960	8.8862	6.9450	0.0000
6	0.0000	4.6180	16.1528	0.0287	0.0000	9.3016	14.1927	0.0119
7	0.2804	6.4321	1.3349	0.1767	0.1462	4.6339	0.6668	0.0427
8	0.0000	0.0000	0.0000	0.1018	0.0000	0.0000	0.0000	0.0336
9	0.0000	0.0000	0.0000	0.0830	0.0000	0.0000	0.0000	0.0107
10	0.0000	0.0000	0.0000	0.1493	0.0000	0.0000	0.0000	0.0283
11	0.0000	0.0000	0.0000	1.4393	0.0000	0.0000	0.0000	0.1732
12	0.0000	0.0000	0.0000	1.8667	0.0000	0.0000	0.0000	0.1101
13	0.0001	0.0584	0.0664	30.2788	0.0004	0.1838	0.0975	1.3798
14	6.9641	0.0000	0.0779	57.6624	1.4234	0.0000	0.2463	1.0838

to these high-dimensional B-LPs. However in certain benchmark problems, these methodologies may return solutions with higher % MAE, where also a higher variability is observed among 10 repetitive runs of these test problems.

In addition to the solution accuracy of each grey-box solver tested as a part of the DOMINO framework, the computational performance of each methodology is compared with respect to the total elapsed time for convergence and the average number of samples collected at convergence (Figure 2.2). The overall computational performance of all solvers, shown in Figure 2.2A and B, indicates that the computational requirements for the DOMINO increases as the ULP dimensionality increases. This is an expected result since the computational efficiency of all grey-box solvers will highly depend on the number of decision variables and the grey-box constraints handled by these algorithms. Although the overall trend shows an increase in computational expense with increasing upper-level dimensionality, Figure 2.2A shows that the total elapsed time for DOMINO

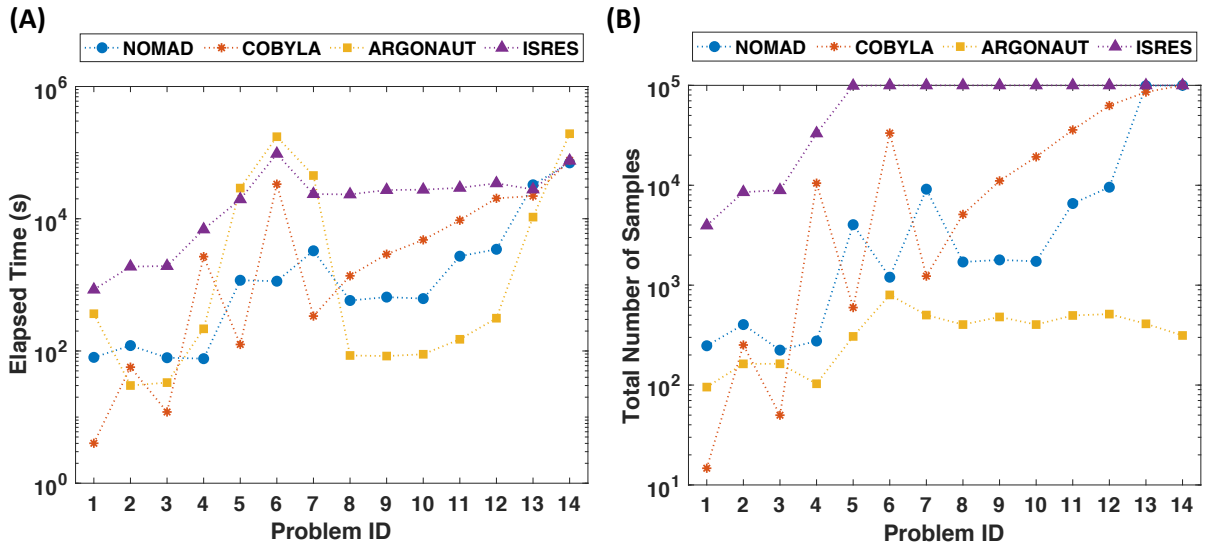

Figure 2.2: (A) Average elapsed time for solving bi-level linear programming problems; (B) Average total number of samples collected by each solver in bi-level linear programming problems.

is comparable when using NOMAD, COBYLA or ARGONAUT algorithms as the preferred grey-box solvers within the framework. On the contrary, the elapsed computational time for the ISRES algorithm is at least an order of magnitude higher for most of the B-LP benchmark problems when compared to other solvers. This is mainly because the solution strategy of the ISRES algorithm dictates significantly higher number of samples for convergence for all B-LP problems, where this in return increases the computational requirements for DOMINO, as shown in Figure 2.2B. It is also important to note that the computational time for solving the LLP in B-LP benchmark problems is minimal. On average, the amount of time required to solve the LLP took 0.013-0.065 seconds per sample. For example, for the ARGONAUT algorithm, it is observed that the total sampling time (i.e., total time spent to solve the LLP for a given B-LP benchmark problem) accounted for less than 9% of the total elapsed time spent for convergence. For this grey-box solver, the parameter estimation and the surrogate model optimization stages accounted for at least 59% of the total elapsed time, showing that the grey-box optimization stage was computationally much more expensive than solving the LLP at different sampling points.

The overall results demonstrate that NOMAD, as a sample-based local grey-box solver, is more

favorable to be incorporated in the DOMINO framework for solving B-LP problems. NOMAD is shown to achieve highly consistent solution accuracy with good computational efficiency compared to other methodologies. In spite of that, it is important to note that the incumbent solution obtained at convergence from all algorithms in the DOMINO framework are guaranteed feasible solutions to the B-LP problems, as all constraints, including the optimality of the LLP, are satisfied.

2.3.1.2 Results for Continuous Nonlinear Bi-Level Programming Problems

In addition to the B-LPs, the DOMINO framework is extensively tested with continuous bi-level nonlinear programming (B-NLP) problems. The results of this computational study are provided in Table 2.6. The overall results show that in B-NLP problems, the global methodologies outperform local solution strategies. Global grey-box solvers, namely ARGONAUT and ISRES, solve more benchmark problems with lower % MAE and with lower standard deviations of this error. ISRES solves 30 benchmark problems with less than 5% MAE and ARGONAUT solves 28 in the same error range out of the 36 benchmark problems tested. This number drops to 23 and 14 for NOMAD and COBYLA, respectively. Especially, the deteriorating performance of COBYLA is somewhat expected since this algorithm uses linear approximations for the objective function and constraints. In many of these B-NLP case studies, the linear approximations constructed by COBYLA are not sufficient to capture the nonlinear relationship in the input-output data. Hence, DOMINO is more prone to converging to sub-optimal solutions in B-NLP benchmark problems when COBYLA is preferred over other solvers.

Furthermore, Table 2.6 provides a more detailed overview on DOMINO's accuracy and consistency in solving many challenging B-NLP problems. In the LP-QP test problems, it is observed that for problem 16 ("LPQP1") NOMAD, COBYLA and ARGONAUT converge consistently to the true global solution over multiple repetitive runs, whereas ISRES converges to a near-optimal solution with less than 5% MAE. For benchmark 15 ("mb_1_1_06"), it is observed that DOMINO returns feasible solutions with high % MAE regardless of the grey-box solver of choice. The underlying reason for this inferior performance by DOMINO is due to the fact that the problem is degenerate. The optimal solution to the bi-level problem exists at $x = 0$, where all points for

$y \in [-1, 1]$ are trivially optimal [100]. However, for $-1 \leq x < 0$ the unique global solution exists at $y = -1$ and for $0 < x \leq 1$ the unique global minimum is at $y = x^2$. Hence, the data-driven algorithms tend to go to either unique optimal solution at the lower-level ($y = -1$ or $y = x^2$) due to the deterministic optimization step taken by the DOMINO at provided sampling points for x. As a result, higher deviations are observed in DOMINO solutions compared to the true global solution. It is also important to note that for this class of bi-level benchmark problems, all grey-box solvers provide guaranteed feasible solutions as the LLP returns the global optimum and a feasible solution to the grey-box problem is identified at convergence (Remark 3).

In the QP-QP problem set, the results indicate that global solvers can provide consistent near-optimal solutions to these benchmark problems. Especially, ISRES algorithm consistently converges to the true optimal solution in 3 out of 4 QP-QP benchmark problems. However, local methodologies (NOMAD and COBYLA) converge to sub-optimal solutions with high variability. Moreover, it is important to note that NOMAD's standard deviation of the % MAE for problem 17 ("mb_1_1_16") is not reported since this algorithm has returned an infeasible solution in 1 of the 10 random runs. In this case, the lower-level optimality is satisfied, however, one of the grey-box constraints is violated. In addition, it is important to highlight that a better solution for the problem 18 ("wk_2015_01") is identified by the DOMINO framework. Different decision variables are identified at the LLP with an improved objective function value compared to the ones reported by Woldemariam and Kassa [107]. Thus, the solution reported by this study [107] does not meet the optimality condition of the lower-level where the overall solution becomes infeasible for this B-NLP problem. The best found solution by DOMINO is reported in the Appendix A.

In the NLP-QP problem set, a similar trend is observed where global solvers outperform the local grey-box solution strategies. For problem 22 ("mb_2_3_02"), the global optimization step taken at the lower-level returned the optimal solution to all repetitive runs of the 4 grey-box solvers tested as a part of the DOMINO framework. However, due to the nonconvexity at the upper-level, it is observed that the local solvers converge to sub-optimal solutions and yield higher % MAE values with higher deviations. Hence, the global exploration of candidate sampling points by

ARGONAUT and ISRES leads to improved solution accuracy in this challenging B-NLP problem.

Similarly, in the LP-NLP problem set, the overall performance of ARGONAUT and ISRES show that these solvers are more favorable to be incorporated into the DOMINO framework for solving B-NLP problems, as they provide highly consistent and accurate solutions to these case studies. In several benchmark problems, it is observed that NOMAD and COBYLA return highly variable solutions with a high % MAE. Especially for problems 31 ("gf_5") and 32 ("gf_3"), COBYLA returns 1 infeasible solution out of 10 repetitive runs of these bi-level problems. In case of the NOMAD algorithm, an infeasible solution is returned for problem 32 ("gf_3"). In addition, it is important to note that for problem 24 ("mb_1_1_04") all grey-box solvers provide feasible solutions with more than 100% MAE with respect to the true global solution. In this case, the upper-level objective consists of the lower-level variable, y, and the inner objective is parametrized in x. As a result, the proposed data-driven approach can detect the unique global minimum for the inner objective, which is $y^* = 0.5$ for $x > 0$ and $y^* = 1$ for $x < 0$. However, none of the data-driven solvers can pinpoint the unique optimal solution of this bi-level problem at $x = 0$ where any $y \in [-0.8, 1]$ is trivially optimal. The main reason behind this issue is that the LLP is degenerate and the piecewise nature of the input-output data hinders the information collected at the sampling stage. Even though various points are sampled, with different x values, the corresponding upper-level objective is either 0.5 or 1. As a result, the solvers terminate the optimization procedure after several consecutive iterations, since there is no improvement to the best found objective as new sampling points are added. Hence, DOMINO fails to pinpoint the unique optimal solution to this benchmark problem.

Furthermore, in the QP-NLP problem set, the global grey-box solvers continue to provide optimal or near-optimal solutions consistently to many B-NLPs of this type. However, in problem 38 ("mb_1_1_17") all solvers consistently converge to the same sub-optimal solution. The main reason for this is that the LLP has two global minima with the objective function value of zero and $y = 1 + 0.1x \pm 0.5\sqrt{2 + 2x}$. By default, the negative counterpart is used for computing y, whereas the optimal solution reported in Mitsos and Barton [100] uses the positive counterpart for

the inner problem. Hence, all the grey-box solvers converge to the same sub-optimal solution and the results reported in Table 2.6 reflect the errors based on the negative counterpart of y. However, if y is strictly constrained to the positive counterpart, then all the grey-box solvers will identify a near-optimal solution with 0.0161 average % MAE and 0.0000 average standard deviation of % MAE. This observation is also consistent with Remark 5, where DOMINO cannot characterize pessimistic, optimistic and other types of decisions in the presence of multiple optima at the lower-level.

Finally for the NLP-NLP type bi-level problems, it is observed that global solvers return consistent feasible near-optimal solutions whereas the local solvers are prone to converging to sub-optimal solutions in a portion these nonconvex B-NLPs. This difference is also supported by the standard deviation values of the % MAE provided in Table 2.6, where high values of the deviation indicates that in a portion of the repeated test runs, these local solvers can find a feasible near-optimal solution, whereas in the rest they converge to feasible sub-optimal solutions that are distant to the true global solution. However, it is important to state that COBYLA struggles to find feasible solutions in 50% of the NLP-NLP type benchmark problems. As this algorithm uses linear approximations, using the COBYLA algorithm within the DOMINO framework is not favorable for solving nonconvex nonlinear bi-level programming problems. It is also observed that ARGONAUT returns an infeasible result for problems 46 ("wk_2015_04") and 47 ("wk_2015_06"), whereas NOMAD and ISRES return infeasible solutions to problem 47 ("wk_2015_06"). Both of these case studies are particularly challenging since they contain the absolute value function, where the derivative of the objective/constraints is discontinuous. Nonetheless, it is important to note that, for both of these benchmark problems, out of 10 random runs for each solver, a better objective function value is found than the solution reported in Woldemariam and Kassa [107]. This is possible since the lower-level optimality in this study [107] was not satisfied at the provided optimal solution, hence making the reported solution an infeasible point for both of these bi-level programming problems. The best found solutions by DOMINO for these benchmark problems are reported in the Appendix A.

Table 2.6: Average % MAE and average standard deviation of % MAE for continuous nonlinear bi-level benchmark problems. Number of infeasible solutions reported out of 10 runs: by NOMAD for problem 17 ("mb_1_1_16") is 1, for problem 32 ("gf_3") is 1, for problem 47 ("wk_2015_06") is 4; by COBYLA for problem 31 ("gf_5") is 1, for problem 32 ("gf_3") is 1, for problem 42 ("c_2002_03") is 2, for problem 44 ("nwj_2017_02") is 1, for problem 46 ("wk_2015_04") is 3, for problem 47 ("wk_2015_06") is 9, problem 48 ("ka_2014_02") is 1; by ARGONAUT for for problem 46 ("wk_2015_04") is 1, for problem 47 ("wk_2015_06") is 1; by ISRES for problem 47 ("wk_2015_06") is 8.

Problem	Average % MAE				Average Standard Deviation of % MAE			
ID	NOMAD	COBYLA	ARGONAUT	ISRES	NOMAD	COBYLA	ARGONAUT	ISRES
LP-QP								
15	90.0046	70.0139	100.0000	30.0175	31.6081	48.2822	0.0000	48.2929
16	0.0000	0.0000	0.0000	4.1468	0.0000	0.0000	0.0000	0.6435
QP-QP								
17	89.6785	56.8518	3.3533	0.0314	-	101.4962	3.0133	0.0007
18†	0.0000	0.0000	0.8891	0.0001	0.0000	0.0000	1.2213	0.0001
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
20	0.0000	78.4000	0.0077	0.0000	0.0000	28.6713	0.0161	0.0000
NLP-QP								
21	0.0000	0.0000	0.0000	0.0003	0.0000	0.0000	0.0000	0.0005
22	17.2004	59.1898	15.8680	6.3927	13.2570	32.6424	9.8335	3.0217
LP-NLP								
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
24	FS*	FS*	FS*	FS*	32.9404	32.2749	0.0000	0.1002
25	0.0141	0.0252	0.5802	0.0141	0.0000	0.0352	0.4652	0.0000
26	0.0000	10.0001	0.0000	0.0001	0.0000	31.6227	0.0000	0.0002
27	0.0000	5.0002	0.0000	0.0002	0.0000	15.8114	0.0000	0.0004
28	31.4656	FS*	0.1973	0.0293	47.3844	121.6939	0.3698	0.0000
29	1.0000	1.0000	1.0000	1.0000	0.0000	0.0000	0.0000	0.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001
31	0.0009	10.0007	0.8179	0.0007	0.0000	-	1.4561	0.0003
32	14.5205	29.9388	12.8843	0.0000	-	-	5.8265	0.0000
QP-NLP								
33	0.0000	90.0002	1.4847	0.0000	0.0000	144.9136	1.1691	0.0001
34	0.0000	0.0000	0.0035	0.0000	0.0000	0.0000	0.0069	0.0000
35	40.0000	20.0000	0.0000	0.0024	51.6398	42.1637	0.0000	0.0068
36	54.0004	FS*	2.6282	0.0005	88.4684	140.8542	2.1959	0.0003
37	0.0024	0.0024	0.0165	0.0024	0.0000	0.0000	0.0214	0.0000
38	83.3109	83.3109	83.3109	83.3109	0.0000	0.0000	0.0000	0.0000
39	0.0024	0.0024	0.0326	0.0024	0.0000	0.0000	0.0625	0.0000
40	1.1953	1.4353	0.0001	0.0000	1.2599	1.2353	0.0002	0.0000
NLP-NLP								
41	1.1490	1.1490	1.1490	1.1490	0.0000	0.0000	0.0000	0.0000
42	0.0000	20.0000	0.0000	0.0007	0.0000	-	0.0000	0.0008
43	0.0084	10.9125	0.0867	0.0084	0.0000	9.3847	0.1319	0.0000
44	9.9140	79.9494	5.8744	0.7041	19.5323	-	4.5579	0.0774
45	27.3509	37.5217	0.1481	0.0004	35.3098	40.7806	0.4682	0.0006
46‡	56.6928	62.3959	FS*	64.7918	39.1217	-	-	34.1483
47§	FS*	FS*	FS*	FS*	-	-	-	-
48	0.0000	16.0616	0.0054	2.8125	0.0000	-	0.0172	4.5286
49	0.0025	40.0776	4.0394	0.0025	0.0000	43.6221	2.0734	0.0000
50	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

*Feasible solution with more than 100% MAE on average is returned at convergence.

† % MAE calculated with respect to the best solution found by DOMINO ($F_{best} = 99.9955$).

‡ % MAE calculated with respect to the best solution found by DOMINO ($F_{best} = 0$).

§ % MAE calculated with respect to the best solution found by DOMINO ($F_{best} = 4.5078 \cdot 10^{-6}$).

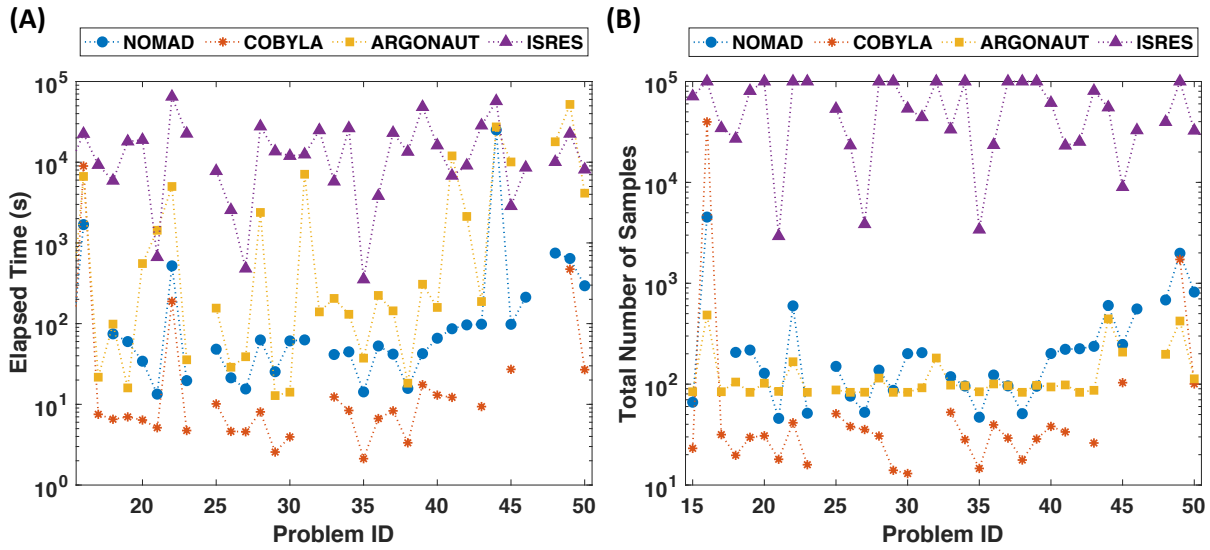

Figure 2.3: (A) Average elapsed time for solving continuous bi-level nonlinear programming problems; (B) Average total number of samples collected by each solver in continuous bi-level nonlinear programming problems.

Computational performance of DOMINO is also provided in Figure 2.3. As expected, the elapsed time for local solvers is significantly less than the global ones (Figure 2.3A). Specifically, ISRES stands out as the most computationally demanding methodology both in the time required to retrieve the optimal solution as well as the total number of samples required for convergence, where in many instances it hits the maximum number of function evaluations (10^5 samples) allowed for the algorithm, as shown in Figure 2.3B. This occurrence is due to the evolutionary nature of this method, as ISRES requires too many samples for convergence, even for the lower dimensional and relatively simpler benchmark problems. This is followed by the ARGONAUT algorithm where in certain benchmark problems the time required for convergence is higher, where in others the overall performance is comparable to local methodologies. The computation time required to solve the continuous nonlinear lower-level problems is minimal similar to the B-LP benchmark problems with the exception of problem 47 ("wk_2015_06"). On average, the computational expense for solving the lower-level varies between 0.0171-5.5514 seconds and the overall contribution of sampling to the total elapsed time varies between 0.03-18.9%. Specifically, in

problem 47 (“wk_2015_06”), the average computational time required to solve the LLP is 88.789 seconds with an overall contribution of 50.9% in total elapsed time. As this problem is more challenging to optimize due to the discontinuous derivatives at the lower-level, a higher contribution from the sampling phase is observed to the overall DFO procedure than the grey-box optimization phase. On the contrary, for the other B-NLP problems, the grey-box optimization phase (i.e., surrogate model building and its respective optimization) is the most computationally demanding step in ARGONAUT’s solutions. As for the sampling requirements, ARGONAUT collects fewer samples than the ISRES algorithm, since ARGONAUT is a model-based grey-box solver. The overall results show that COBYLA is the most computationally efficient methodology; however, this solver was unable to provide consistent feasible solutions to several B-NLP benchmark problems. Although the ARGONAUT and ISRES are computationally more expensive to execute, it is possible to retrieve optimal or near-optimal solutions more consistently through using these global data-driven solvers in DOMINO for B-NLP problems.

2.3.1.3 Results for Bi-Level Mixed-Integer Programming Problems

The results for the bi-level mixed-integer programming problems are summarized in Table 2.7. For this class of problems, it is observed that sample-based grey-box solvers outperform model-based methodologies. DOMINO can identify optimal or near-optimal solutions consistently to various types of bi-level mixed-integer programming problems when using NOMAD as the grey-box solver of choice. NOMAD almost perfectly returns solutions with low errors where only in one benchmark problem this algorithm returns a sub-optimal feasible solution. Likewise, the ISRES algorithm is very successful in finding near-optimal solutions, but struggles in finding near-optimal solutions in higher dimensional benchmark problems. It is also important to highlight that NOMAD, ARGONAUT and ISRES identify feasible solutions in all of the bi-level mixed-integer programming problems tested. However, COBYLA fails to identify a feasible solution in 1 of the 10 repetitive runs of benchmark 57 (“QPMILP2”).

Furthermore, the computational performance of DOMINO in solving bi-level mixed-integer programming problems is summarized in Figure 2.4. Figure 2.4A shows that ISRES requires an

Table 2.7: Average % MAE and average standard deviation of % MAE for bi-level mixed-integer benchmark problems. Infeasible solutions reported: by COBYLA for problem 57 ("QPMILP2") in 1 out of 10 runs.

Problem	Average % MAE				Average Standard Deviation of % MAE			
ID	NOMAD	COBYLA	ARGONAUT	ISRES	NOMAD	COBYLA	ARGONAUT	ISRES
LP-MILP								
51	0.0000	0.0000	0.0000	0.0009	0.0000	0.0000	0.0000	0.0018
52	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
53	0.3050	0.0000	0.8756	0.0000	0.8052	0.0000	1.3089	0.0000
54	0.0000	1.7135	8.3347	0.2276	0.0000	3.7096	5.2589	0.0846
55	0.0000	0.0004	0.0000	2.8790	0.0000	0.0011	0.0000	0.5561
QP-MILP								
56	0.0000	0.0028	0.0365	0.0000	0.0000	0.0088	0.1151	0.0000
57	0.0074	28.4286	2.3668	0.0002	0.0042	-	1.7631	0.0002
58	0.0000	18.9344	0.0000	0.0000	0.0000	57.5193	0.0000	0.0000
59	0.0000	FS*	FS*	FS*	0.0000	$> 10^5$	791.0469	$> 10^3$
60	0.0000	FS*	7.9741	2.5208	0.0000	220.4544	2.5821	1.3129
61	0.0000	FS*	55.6621	50.5639	0.0000	$> 10^3$	49.1259	27.5322
62	0.0000	2.8949	0.4577	0.5772	0.0000	5.3286	0.2089	0.1691
63	0.0000	FS*	FS*	36.5575	0.0000	364.1700	188.6630	5.9444
64	0.0000	26.7727	FS*	8.9426	0.0000	23.3885	116.9957	2.1176
NLP-MILP								
65	0.0000	27.5060	0.7382	0.0000	0.0000	44.6080	1.8132	0.0000
66	0.4039	0.4038	4.6050	0.4038	0.0000	0.0001	7.3684	0.0001
67	0.0000	1.2185	1.8888	0.0087	0.0000	3.8531	1.6111	0.0037
68	0.0026	6.9802	23.7610	0.5531	0.0049	10.8381	22.8133	0.1229
69	0.0000	23.3259	5.6201	0.5171	0.0001	32.5728	7.8602	0.2278
70	0.0039	0.2861	2.2180	0.8578	0.0079	0.9044	1.8165	0.1276
71	0.0006	0.0115	0.7633	1.1059	0.0016	0.0358	0.6795	0.1253
72	0.0023	1.7054	3.9910	1.1129	0.0074	3.2354	4.6783	0.1573
73	0.0030	1.3933	1.7064	1.1861	0.0068	1.3013	1.8636	0.1718
LP-MIQP								
74	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
75	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
76	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
77	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
78	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
79	0.1192	0.0000	0.0000	0.0000	0.3770	0.0000	0.0000	0.0000
80	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
81	0.0000	0.0000	0.0000	0.0012	0.0000	0.0000	0.0000	0.0008
QP-MIQP								
82	37.5000	25.0001	2.3111	0.0003	60.3807	52.7046	1.7433	0.0003
83	0.0000	3.7386	0.0000	0.0000	0.0000	11.6915	0.0000	0.0000
84	0.0000	13.5220	0.6207	0.0000	0.0000	36.4485	1.5634	0.0000
85	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
86	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
87	0.0000	6.9291	3.6101	0.0000	0.0000	21.9079	7.8285	0.0000
88	0.0000	FS*	0.3254	0.0000	0.0000	$> 10^4$	0.5022	0.0000
89	0.0000	FS*	0.0015	0.0000	0.0000	945.6580	0.0047	0.0000
NLP-MIQP								
90	0.0000	0.0000	0.2280	0.0000	0.0000	0.0000	0.2448	0.0000
91	0.0000	0.0055	0.0000	0.0000	0.0000	0.0056	0.0001	0.0000
92	0.0006	0.0024	0.6045	0.0004	0.0018	0.0021	0.8118	0.0007
93	0.1603	9.7658	0.8294	0.0774	0.3379	30.3412	0.7244	0.2442
94	0.0000	0.0076	28.2385	0.0004	0.0000	0.0184	16.3438	0.0006
95	0.0002	0.0126	0.4367	0.0007	0.0006	0.0393	0.3314	0.0009
96	0.0000	0.0000	0.0014	0.0004	0.0001	0.0000	0.0041	0.0004
97	0.0050	0.0022	1.5956	0.0072	0.0087	0.0046	4.7409	0.0053
98	0.0044	0.0131	1.0333	0.0074	0.0135	0.0180	1.5260	0.0142
99	0.0002	0.9624	2.1801	0.0071	0.0007	3.0432	2.2069	0.0022
NLP-INLP								
100	2.5628	32.4959	0.0000	0.0000	8.1043	52.6317	0.0000	0.0000

*Feasible solution with more than 100% MAE on average is returned at convergence.

order of magnitude higher time for convergence compared to other algorithms, and converges prematurely by hitting the maximum number of samples allowed in almost all tested case studies (Figure 2.4B). Moreover, it is important to note that for many of the bi-level mixed-integer benchmark problems both model-based methodologies (COBYLA and ARGONAUT) are recorded to have higher computational expense. Like in the other classes of bi-level programming problems, it is observed that the computation time to deterministically solve the LLP is small, between 0.016-0.067 seconds on average per sample. The overall contribution of solving the LLP deterministically to the total elapsed computation time was at most 15%, where the rest of the computational expense was sourced majorly from the grey-box optimization phase in the ARGONAUT results. Overall, NOMAD is computationally efficient both in terms of the computational time required for convergence as well as in terms of the total number of samples collected throughout the data-driven optimization step. Although in Figure 2.4B, ARGONAUT is shown to be the most sample efficient algorithm, the errors reported in Table 2.7 indicate that ARGONAUT converges to a sub-optimal feasible solution in high-dimensional problems, hindering the overall performance of DOMINO in finding the globally optimal solution to bi-level mixed-integer programming problems. The overall results show that NOMAD is more favorable to be incorporated in the DOMINO framework for solving bi-level mixed-integer programming problems. In the following section, the DOMINO framework is tested on a larger bi-level MINLP case study, which considers a land allocation problem under Food-Energy-Water Nexus considerations.

2.3.2 Land Allocation Problem in Food-Energy-Water Nexus

The sustainable development of an agricultural farming area is of critical importance for maintaining the interconnected elements, namely food, energy and water, that depend on the same land resources. Hence, the actions taken towards allocating land resources will essentially affect food production, which requires energy, in the form of fertilizers, and water for irrigation. On the other hand, clean water production requires energy (i.e., operating a filtration system) and energy can be produced through agriculture as biofuels. This interconnected relationship between these key resources is referred to as the Food-Energy-Water Nexus (FEW-N) and has recently gained a lot of

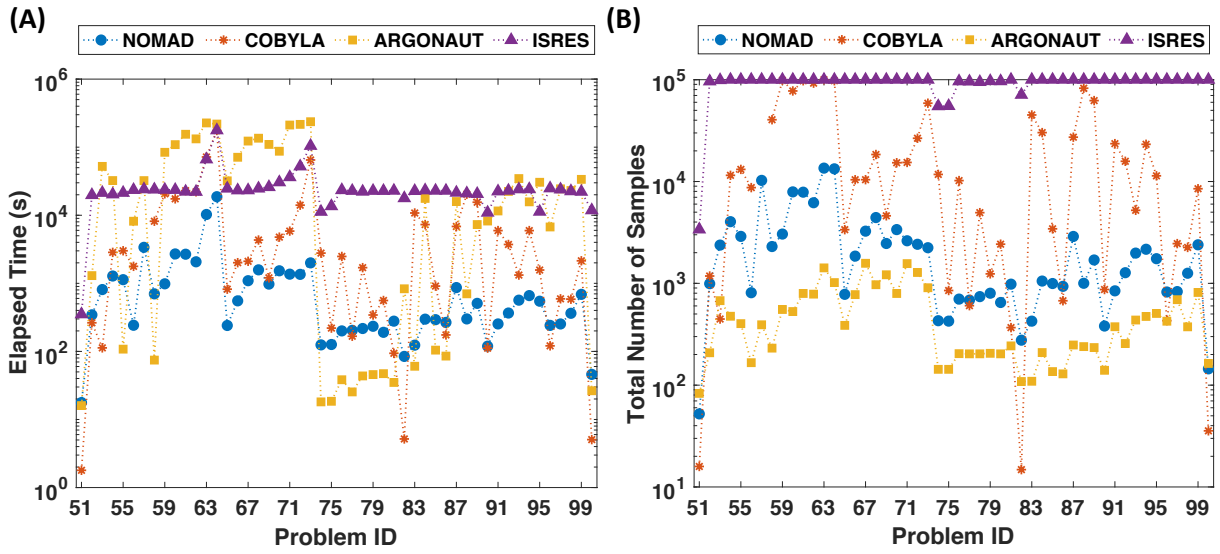

Figure 2.4: (A) Average elapsed time for solving bi-level mixed-integer programming problems; (B) Average total number of samples collected by each solver in bi-level mixed-integer programming problems.

attention for land use optimization in areas with resource scarcity [109–111].

While the government regulators would like to minimize the stress on the nexus in the long-term, many companies allocating and processing the land are concerned with short-term goals, such as maximizing profit. Thus, a formidable challenge exists in the optimization of the land allocation problem, where multiple stakeholders, each concerned with optimizing their own objective functions, are acting upon the optimal decision-making process. We have previously developed a hierarchical FEW-N approach to tackle this issue and to facilitate decision making under competition for these key resources while promoting the sustainable development of the land [94]. In this section, the data-driven optimization of the land allocation problem will be addressed by using the DOMINO framework.

The land allocation case study consists of two players: the government regulators and the agricultural developer. The goal of the agricultural developer is to maximize its profit whereas the government that regulates this piece of land aims to minimize the stress on the FEW-N, by offering subsidies to the agricultural producer or land developer. Hence, this can be viewed as a Stackelberg

game where the government regulators will lead, making the first move by assigning the subsidies, whereas the agricultural producer will follow the leader by reacting accordingly, taking optimal actions towards maximizing its own profit. This leads to the following hierarchical optimization problem [94],

$$\begin{array}{lll} min & & Stress\,on\,FEW\,Nexus \\ \text{s.t.} & & Government's\,Budget \\ & max & Developer's\,Profit \\ & \text{s.t.} & Land\,Properties \\ & & Land\,Process\,Models \end{array} \tag{2.2}$$

where the agricultural developer will invest on a piece of land to maximize its profit through a careful consideration of land properties, subsidies offered by the government and land process models at the lower-level. On the other hand, at the upper-level, the government agency that regulates this land will focus on sustainable development through minimizing the FEW-N stress, with respect to their allowed budget.

The detailed land allocation model (please see Appendix B for the model equations) is developed in GAMS and the lower-level problem is an MILP problem with 1,721 equations, 216 discrete variables and 772 continuous variables. The upper-level is an NLP problem consisting of 5 continuous variables with 165 grey-box constraints from the Big-M formulation (Equations B.19 and B.21). This large-scale bi-level NLP-MILP optimization problem is solved using the DOMINO framework and the performance of the 4 data-driven solvers are compared in the following section.

2.3.2.1 Computational Results of the FEW-N Case Study

The results of the hierarchical land allocation problem are summarized in Figures 2.5 and 2.6. The boxplot results in Figure 2.5A show that the DOMINO framework, when coupled with a global solver, consistently returns the same objective value over multiple repetitive runs, whereas some variability is observed in the returned solutions when the framework is coupled with local data-driven solvers. This result clearly indicates that the hierarchical FEW-N land allocation problem

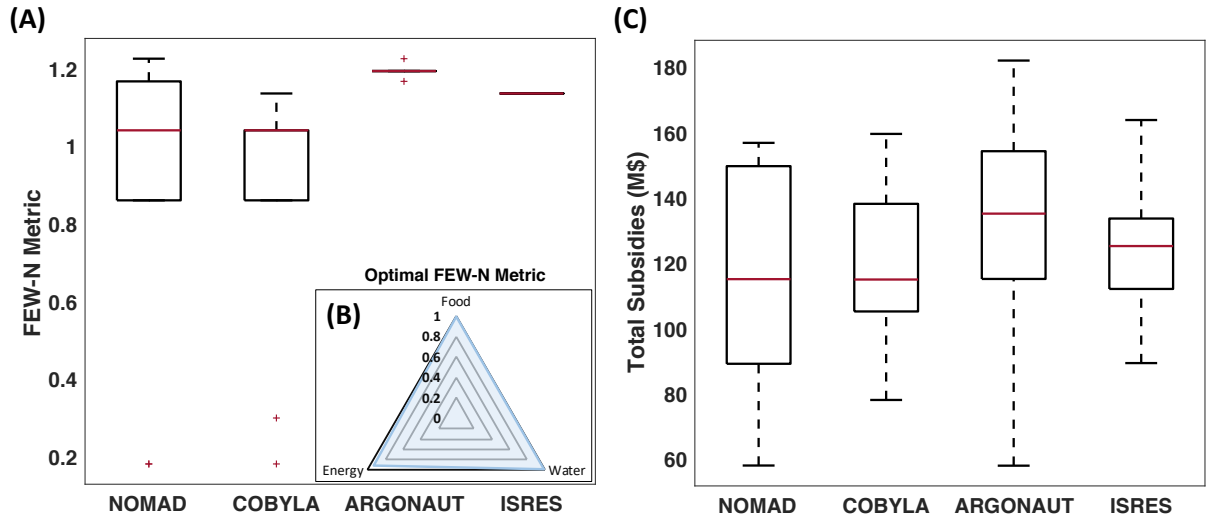

Figure 2.5: (A) Optimal FEW-N metric returned by DOMINO when coupled with local and global grey-box solvers; (B) Optimal nexus solution represented as the area of a triangle (Best solution found by ARGONAUT and NOMAD algorithms in DOMINO, $f_{best} = 1.2258$); (C) Boxplot of total amount of subsidies offered by the government for the solution of FEW-N land allocation problem over 10 runs.

is nonconvex and global optimization is necessary to find a superior solution. The maximum value for the FEW-N metric for this case study is identified by two algorithms, namely NOMAD and ARGONAUT. In addition, Figure 2.5B and C shows the globally optimal FEW-N metric found by the DOMINO framework and the distribution of the total amount of subsidies offered by the government for each solver over 10 runs, respectively. The radar plot in Figure 2.5B shows that the globally optimal solution can capture the food and water dimensions of the nexus almost perfectly (99.5% in food and 99% in water) with a small trade-off in the energy dimension (93%).

In addition, the boxplot in Figure 2.5C shows that all solvers are subject to some variability in finding the optimal set of decisions for the government regulators' objective. More specifically, the variability within the results of two global solvers, which returned consistent objective function values as shown in Figure 2.5A, is a clear indication of the multiplicity of solutions that exists in the problem. For the same optimal FEW-N metric value ($f_{best} = 1.2258$), NOMAD allocates a total of \$58.1M with a breakdown of \$0M for livestock grazing and solar energy, \$7.6M for wind energy,

Table 2.8: Computational performance of DOMINO with different grey-box solvers for the land allocation problem. The results are averaged over 10 runs.

Solver	Average Elapsed Time (s)	Average Total Number of Samples
NOMAD	138.6	283.9
COBYLA	23.6	67.1
ARGONAUT	$1.2 \cdot 10^4$	247.4
ISRES	$3.3 \cdot 10^4$	10^5

$37.8M and $12.7M for fruit and vegetable production, respectively. On the other hand, for the same optimal FEW-N metric value, ARGONAUT allocates a total of $115.2M with a breakdown of $0M for livestock grazing and solar energy, $15.2M for wind energy and $50M for both fruit and vegetable production. A clear difference between the solutions provided by these two algorithms is more apparent at the lower-level objective function value, where the solution provided by NOMAD enables the agricultural developer to have $3.47B profit, whereas this number increases by $500M to $3.97B profit with the ARGONAUT solution. This difference in profit values is captured in the optimal allocation results that are provided in Figure 2.6, where the allocation patterns for the same nexus solution differ as the subsidies offered by the government is lowered. Figures 2.6A and B show that the optimal allocation pattern for the land is exactly the same for the spring, summer and autumn seasons for both NOMAD and ARGONAUT, where a mix of wind energy and fruit production is preferred on the land. However, in winter, the optimal allocation for plot 7 changes to vegetable production for the ARGONAUT solution, while others remain the same. In the case of the NOMAD solution, the allocation pattern for plot 3 in winter changes from wind energy and fruit production to wind energy and vegetable production. Overall, both configurations are equally optimal and are sufficient to minimize the nexus stress, where the government will decide whether to subsidize the agricultural processes with a higher or a lower amount depending on their available budget and preferences.

The computational performance of each solver within the DOMINO framework for the FEW-N case study is also compared (Table 2.8). The average elapsed time and the average number of samples collected at convergence indicates that COBYLA is computationally very efficient. However,

(A)

Optimal Land Allocation for Autumn, Spring and Summer Seasons

1. Wind & Fruits	2. Wind & Fruits	3. Wind & Fruits	4. Wind & Fruits
5. Wind	6. Wind	7. Fruits	8. Fruits

Optimal Land Allocation for Winter

1. Wind & Fruits	2. Wind & Fruits	3. Wind & Fruits	4. Wind & Fruits
5. Wind	6. Wind	7. Veggie	8. Fruits

(B)

Optimal Land Allocation for Autumn, Spring and Summer Seasons

1. Wind & Fruits	2. Wind & Fruits	3. Wind & Fruits	4. Wind & Fruits
5. Wind	6. Wind	7. Fruits	8. Fruits

Optimal Land Allocation for Winter

1. Wind & Fruits	2. Wind & Fruits	3. Wind & Veggie	4. Wind & Fruits
5. Wind	6. Wind	7. Fruits	8. Fruits

Figure 2.6: (A) Optimal land allocation returned by ARGONAUT; (B) Optimal land allocation returned by NOMAD. Both solutions are equally optimal with the FEW-N metric value of 1.2258.

COBYLA was unable to locate the best solution found by NOMAD and ARGONAUT algorithms for the FEW-N problem, which is undesirable to a decision maker. NOMAD stands out as a grey-box solver of choice for this problem as this is the second most efficient algorithm that was able to locate the global solution. As mentioned earlier in this section, the optimal solution provided by NOMAD is more favorable for the government regulator as the total amount of subsidy offered is minimal. On the other hand, the solution offered by ARGONAUT is equally optimal with respect to the NOMAD solution, and favors the agricultural developer at the lower-level as this solution provides an additional $500M in their profit. However, ARGONAUT being a global model-based grey-box solver makes it more computationally demanding for this problem with respect to the elapsed time for convergence, since ARGONAUT explicitly constructs individual surrogate formulations for the 165 unknown grey-box constraints in this case study. In terms of sampling requirements, as shown in Table 2.8, it is observed that NOMAD and ARGONAUT are comparable as they both collect about equal number of samples on average over 10 repetitive runs. Finally, as observed in the results of many benchmark problems that are provided in Section 2.3.1, ISRES reaches the maximum number of samples allowed for the algorithm in all repetitive runs, which also leads to a more demanding computational time for the execution of this algorithm. Overall,

the results of the benchmark studies and the large-scale land allocation problem demonstrate that the DOMINO framework serves as an effective methodology for solving many large-scale bi-level MINLPs.

2.4 Concluding Remarks

In this chapter, the DOMINO framework is presented as an algorithmic advancement for solving bi-level mixed-integer nonlinear programming (B-MINLP) problems with guaranteed feasibility when the lower-level problem is solved to global optimality at convergence. A novel data-driven approach is followed to approximate a bi-level optimization problem into a single-level problem, where the upper-level decision variables are used to simulate the optimality of the lower-level problem. The resulting input-output data is further sent to a data-driven optimizer to retrieve the optimal solution to the bi-level problem, where the DOMINO framework is flexible to house any type of data-driven/grey-box optimizer. The accuracy, consistency and the computational performance of DOMINO is extensively investigated on a large set of benchmark problems consisting of bi-level linear, continuous nonlinear and mixed-integer programming problems. In addition, the effect of the data-driven solver on DOMINO's performance is investigated by incorporating a local sample-based, local model-based, global sample-based, and global model-based methodologies. Furthermore, the performance of the DOMINO framework is tested on a large-scale bi-level mixed-integer nonlinear case study in Food-Energy-Water Nexus (FEW-N). The results of the benchmark studies show that the DOMINO framework can identify the true global solution or a near-optimal solution for an extensive set of challenging bi-level optimization problems. Moreover, the results of the FEW-N case study demonstrate that DOMINO can handle large-scale bi-level mixed-integer nonlinear programming problems and provide superior feasible solutions consistently over multiple repetitive runs. Hence, DOMINO serves as a powerful computational algorithm for solving large-scale B-MINLPs which are traditionally difficult to solve using exact techniques.

3. CONSTRAINED GREY-BOX MULTI-OBJECTIVE OPTIMIZATION WITH APPLICATIONS TO ENERGY SYSTEMS DESIGN*

As discussed in Chapter 1, the global optimization of many engineering problems, which are commonly characterized by high-fidelity and large-scale complex models, poses a formidable challenge partially due to the high noise and/or computational expense associated with the calculation of derivatives. This complexity is further amplified in the presence of multiple conflicting objectives, for which the goal is to generate trade-off compromise solutions, commonly known as the *Pareto-optimal* solutions. In this chapter, an algorithmic advancement is presented for solving a special class of problems under mathematical programming that entail multiple competing objectives (i.e., multi-objective optimization) using a data-driven methodology. The presented framework uses the ϵ-constraint method to convert a multi-objective optimization problem into series of single objective sub-problems and uses a global constrained grey-box optimization algorithm to retrieve the optimal solution at each sub-problem. Computational results are reported for a number of benchmark multi-objective problems and a case study of an energy market design problem for a commercial building, while the performance of the framework is compared with other derivative-free optimization solvers.

This chapter is organized as follows. Section 3.1 provides a brief introduction to multi-objective optimization. In Section 3.2, an extensive literature review is provided on population-based and surrogate-based algorithms. Section 3.4 describes our methodology in detail, where Section 3.4.2 introduces the ϵ-constraint method for reformulating multi-objective optimization problems into a series of single objective sub-problems, and Section 3.4.3 demonstrates the steps of the proposed framework on a motivating example. The mathematical formulations of the computational studies are provided in Section 3.5. Finally, the results of the computational studies are presented in Section 3.6, along with concluding remarks in Section 3.7.

*Part of this chapter is reprinted with permission from "Optimal design of energy systems using constrained grey-box multi-objective optimization" by B. Beykal, F. Boukouvala, C.A. Floudas, E.N. Pistikopoulos, 2018. *Computers & Chemical Engineering*, vol. 116, pp. 488-502, Copyright [2018] by Elsevier and Copyright Clearance Center.

3.1 Multi-Objective Optimization

Multi-objective optimization (MOO) is a branch of mathematical programming where multiple competing objectives (i.e., economic, environmental, societal, political objectives) are present in the problem formulation. The general form of MOO problems is presented in Equation 3.1:

$$\begin{aligned} &\min_{\boldsymbol{x}} \left[f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), ..., f_N(\boldsymbol{x})\right] \\ &s.t.\ \boldsymbol{x} \in \boldsymbol{X} \end{aligned} \tag{3.1}$$

where $\boldsymbol{X}$ is a non-empty feasible region, $\boldsymbol{X} \subseteq \mathbb{R}^n$.

For this class of problems, it is not possible to locate a unique optimal solution since there are trade-offs between the conflicting objectives. As a result, MOO aims to find the best set of decisions that will simultaneously optimize multiple objectives in such a way that the solutions cannot be improved without degrading at least one of the other objectives [112]. In other words, the goal of MOO is to derive a set of trade-off optimal solutions, known as the *Pareto-optimal* solutions, that the decision makers can choose from, depending on their preferences.

3.2 Literature Review on Data-Driven Multi-Objective Optimization

While several methodologies exist in the open literature for MOO, this section only considers the ones that are linked to population-based and surrogate-based algorithms. Meta-heuristic (population-based) algorithms are advantageous since they do not require any reformulations, such as converting the multi-objective problem into a set of single objective sub-problems. These can simultaneously deal with a set of possible solutions without requiring series of separate runs, thus enabling the direct investigation of the multi-objective problem [113]. As a result, population-based algorithms have been a popular choice among many researchers for the MOO of various systems, including truss design [114], thermal system design [115], environmental economic power dispatch [116, 117], beam design [118], water distribution network design [119] and more recently the MOO of zeolite framework determination [120]. In addition to these, the books by Rangaiah and Bonilla-Petriciolet [121], and Coello et al. [113] demonstrate a plethora of applications of

evolutionary algorithms to numerous MOO problems.

Even though the population-based algorithms are widely studied in the open literature, their application to grey/black-box problems are rather limited. There are two main reasons for this: (1) most existing algorithms consider the box-constrained problem or handle general constraints via penalty functions, where the system is being continuously treated as a black-box, and (2) stochastic algorithms typically require a large number of function calls to reach the global optimality, which can be computationally prohibitive for expensive simulations. Several researchers have focused on hybrid implementations of surrogate modeling with stochastic algorithms to overcome such problems. Datta and Regis [122] have proposed a surrogate-assisted evolution strategy, which makes use of cubic radial basis surrogate models to guide the evolution strategy for the optimization of multi-objective black-box functions that are subject to black-box inequality constraints. Likewise, Bhattacharjee et al. [123] have used a well-known evolutionary algorithm, NSGA-II, as the baseline algorithm while using multiple local surrogates of different types to represent the objectives and the constraints.

Surrogate-based approaches, where the objectives and the grey/black-box constraints are approximated with simple tractable models, have also been investigated in the open literature in conjunction with derivative-free algorithms. Singh et al. [124] have proposed the Efficient Constrained Multi-objective Optimization (ECMO) algorithm to solve computer-intensive constrained multi-objective problems using kriging models for the objectives and the constraints. They make use of the hypervolume-based Probability of Improvement (PoI) criterion to handle multiple objectives along with the Probability of Feasibility (PoF) criterion to handle computationally expensive constraints and solve the final formulation using MATLAB's *fmincon* optimizer. Feliot et al. [125] have used an expected hypervolume improvement sampling criterion in their Bayesian Multi-Objective Optimization (BMOO) framework, where the nonlinear implicit constraints and the black-box objectives are handled via extended domination rule. In this algorithm, the authors use sequential Monte Carlo sampling technique for the computation and optimization of the expected improvement criterion. Martínez-Frutos and Herrero-Pérez [126] have introduced

the Kriging-based Efficient Multi-Objective Constrained Optimization (KEMOCO) algorithm that uses a kriging-based infill sampling strategy with DIRECT algorithm for constrained MOO of expensive black-box simulations. They combine the expected hypervolume improvement and the PoF to obtain the Pareto-front with minimum number of samples. Regis [127] has presented Multi-Objective Constrained Stochastic optimization using Response Surfaces (MOCS-RS) framework where the author uses radial basis surrogates as approximations for the objective and constraint functions. A more detailed overview on the existing methods for using surrogates in computationally expensive MOO problems can be found in an excellent survey by Tabatabaei et al. [128].

3.3 Novelty of the Proposed Data-Driven Multi-Objective Optimization Framework

Different than the studies discussed above, this work aims to implement a hybrid methodology that performs global parameter estimation coupled with k-fold cross-validation for individualized surrogate model identification on each unknown formulation (objective and constraints) in a given multi-objective programming problem. An algorithmic advancement is presented where a reformulation strategy and a global grey-box optimization solver is integrated for the global optimization of general constrained MOO problems. The methodological details are further described below.

3.4 Methodology

3.4.1 General Overview of the Data-Driven Multi-Objective Optimization Framework

Figure 3.1 demonstrates the workflow of the proposed data-driven multi-objective optimization methodology. Given a constrained MOO problem, the first step of the workflow is to reformulate it using the ϵ-constraint method. This reformulation will enable the discretization of the objective space, essentially creating a series of single objective sub-problems. Once the sub-problems are identified, a grey-box simulator is created for each sub-problem where the input-output data is generated. Finally, a global constrained grey-box optimization solver, namely the ARGONAUT algorithm [28, 29], is executed for finding the optimal solution of each sub-problem, through surrogate modeling and grey-box optimization of the input-output data. The detailed explanation of the methodology and its step-by-step demonstration on a motivating example is provided in the

following sections.

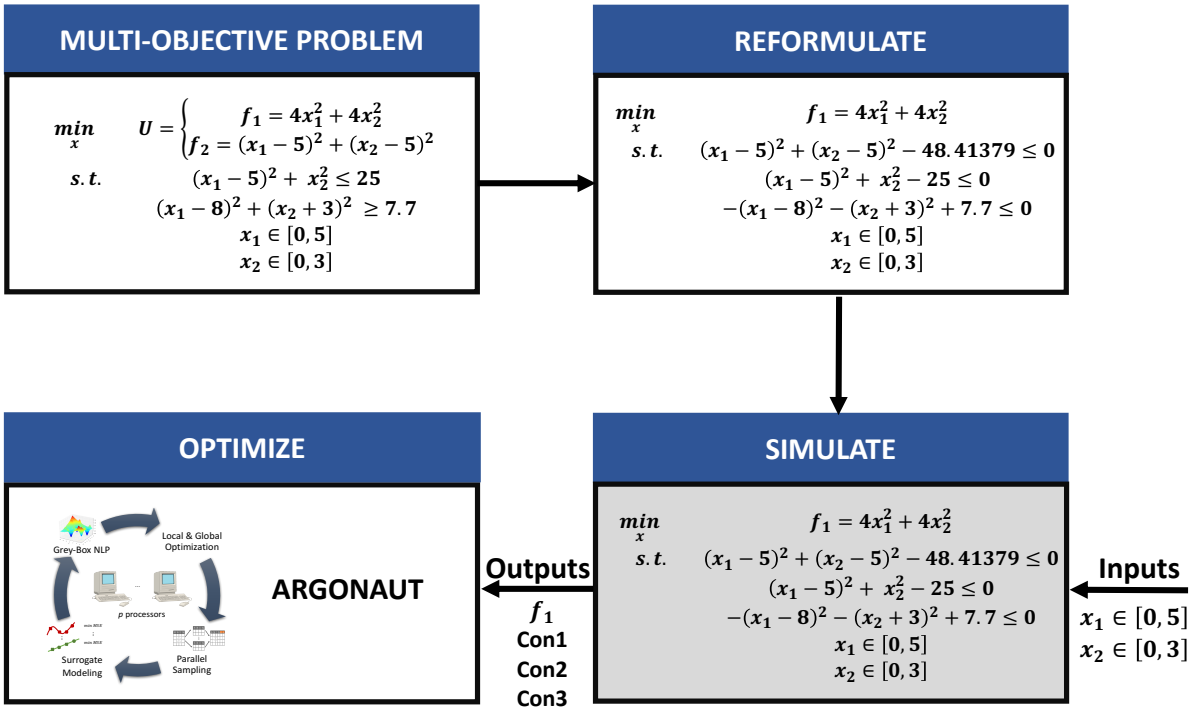

Figure 3.1: General workflow of the data-driven MOO framework using the ARGONAUT algorithm and the ϵ-constraint method.

3.4.2 ϵ-Constraint Method

The ϵ-constraint method is introduced by Clark and Westerberg [129] for converting multi-objective design problems into series of single objective sub-problems. Consider an optimization problem given in the form of Equation 3.1 with only 2 objectives (i.e., $N = 2$). The main idea behind ϵ-constraint method is to discretize the objective space into smaller sections, while obtaining the optimal solution at each discretization point to generate the Pareto-optimal curve. The discretization is done by moving one of the objectives into the constraints set, while setting an upper bound ($\boldsymbol{\epsilon}$) on the new constraint. This simply converts the MOO problem into a single objective optimization problem with an added expense of a single inequality constraint per discretized

problem, as shown in Equation 3.2.

$$\begin{aligned} &\min_{\boldsymbol{x}} f_1(\boldsymbol{x}) \\ &s.t.\ f_2(\boldsymbol{x}) \leq \boldsymbol{\epsilon} \\ &\boldsymbol{x} \in \boldsymbol{X} \\ &\boldsymbol{X} \subseteq \mathbb{R}^n \end{aligned} \tag{3.2}$$

The lower and upper bounds, $[\epsilon^L, \epsilon^U]$, on the discretization points can be derived by minimizing each of the objectives independently. The optimal solution resulting from the minimization of the first objective, $\boldsymbol{x_1^*}$, mathematically formulated in Equation 3.3, will give the maximum value of the second objective, $f_2(\boldsymbol{x_1^*})$, provided that increasing the value of f_2 beyond this maximum value will not affect the value of f_1. Thus, ϵ^U will be equal to $f_2(\boldsymbol{x_1^*})$.

$$\begin{aligned} &\min_{\boldsymbol{x}} f_1(\boldsymbol{x}) \\ &s.t.\ \boldsymbol{x} \in \boldsymbol{X} \\ &\boldsymbol{X} \subseteq \mathbb{R}^n \end{aligned} \tag{3.3}$$

Similarly, the lower bound on ϵ^L is derived by minimizing f_2 as a single objective optimization problem. The optimal solution to this problem, $\boldsymbol{x_2^*}$, gives the minimum possible value of f_2, which is also the minimum value of $\boldsymbol{\epsilon}$. Hence, ϵ^L will be equal to $f_2(\boldsymbol{x_2^*})$. Using these values of $[\epsilon^L, \epsilon^U]$, the objective region can now be divided into D equal intervals as follows:

$$\epsilon^q = \epsilon^{q-1} - \frac{\epsilon^U - \epsilon^L}{D-1} \quad \forall q = 2, \ldots, D, \epsilon^1 = \epsilon^U, \epsilon^D = \epsilon^L \tag{3.4}$$

Then, the final optimization problem becomes:

$$\begin{aligned} &\min_{\boldsymbol{x}} f_1(\boldsymbol{x}) \\ &s.t.\ f_2(\boldsymbol{x}) \leq \boldsymbol{\epsilon} \\ &\quad \boldsymbol{x} \in \boldsymbol{X} \\ &\quad \boldsymbol{\epsilon} \in [\epsilon^1, \ldots, \epsilon^D] \\ &\quad \boldsymbol{X} \subseteq \mathbb{R}^n \end{aligned} \tag{3.5}$$

Although a walk-through is provided for problems with two objectives, the ϵ-constraint method is a general partitioning strategy. For a system with N competing objectives, a similar procedure will be followed as the one shown in Equation 3.2, creating a minimization problem with $N-1$ constraints, which are added to the initial problem formulation. Then, the lower and upper bounds on $\boldsymbol{\epsilon}$ for the partitioned objectives, $[\epsilon^L, \epsilon^U]^1 \times \cdots \times [\epsilon^L, \epsilon^U]^{N-1}$, will define the boundaries of a Pareto-optimal surface when $N = 3$, and a Pareto-optimal polyhedron when $N \geq 3$.

3.4.3 Motivating Example

This section demonstrates the key steps of the solution methodology based on the integration of ARGONAUT with the ϵ-constraint method, on a 2-dimensional motivating example. The following multi-objective programming problem is considered:

$$\begin{aligned} &\min_{\boldsymbol{x}} U = \begin{cases} f_1 = 4x_1^2 + 4x_2^2 \\ f_2 = \big(x_1 - 5\big)^2 + \big(x_2 - 5\big)^2 \end{cases} \\ &s.t.\ \big(x_1 - 5\big)^2 + x_2^2 \leq 25 \\ &\quad \big(x_1 - 8\big)^2 + \big(x_2 + 3\big)^2 \geq 7.7 \\ &\quad x_1 \in [0, 5] \\ &\quad x_2 \in [0, 3] \end{aligned} \tag{3.6}$$

- *Step 1: Dissect the objective space using Equation 3.4*

As shown in Section 3.4.2, minimization of the first and second objectives gives the upper and lower bounds for the ϵ parameter ($\epsilon \in [4, 50]$), respectively. Within these bounds, the objective space is dissected into 30 equal points using Equation 3.4. Table 3.1 summarizes the values of ϵ corresponding to each point.

Table 3.1: Resulting values of ϵ from discretization of the objective space into 30 points.

Point number	ϵ	**Point number**	ϵ	**Point number**	ϵ
1	50	11	34.13793	21	18.27586
2	48.41379	12	32.55172	22	16.68966
3	46.82759	13	30.96552	23	15.10345
4	45.24138	14	29.37931	24	13.51724
5	43.65517	15	27.79310	25	11.93103
6	42.06897	16	26.20690	26	10.34483
7	40.48276	17	24.62069	27	8.75862
8	38.89655	18	23.03448	28	7.17241
9	37.31034	19	21.44828	29	5.58621
10	35.72414	20	19.86207	30	4

- *Step 2: Reformulate Equation 3.6 into single objective sub-problem*

Each point summarized in Table 3.1 is used to reformulate Equation 3.6 into the form of Equation 3.5. For demonstration purposes, only the reformulation of the second point is shown below.

$$\begin{aligned} &\min_{\boldsymbol{x}} \; 4x_1^2 + 4x_2^2 \\ &s.t. \; \big(x_1 - 5\big)^2 + \big(x_2 - 5\big)^2 - 48.41379 \leq 0 \\ &\quad \big(x_1 - 5\big)^2 + x_2^2 - 25 \leq 0 \\ &\quad - \big(x_1 - 8\big)^2 - \big(x_2 + 3\big)^2 + 7.7 \leq 0 \\ &\quad x_1 \in [0, 5] \\ &\quad x_2 \in [0, 3] \end{aligned} \tag{3.7}$$

It is important to realize that the optimization problem shown in Equation 3.7 has the exact same form as Equation 1.1, where set k, representing the known formulations, is assumed to be empty. Thus, it is assumed that the explicit forms of the objective function and the constraints are unknown as a function of the continuous variables, where their respective values are collected as outputs to the problem simulator, like in a true grey/black-box system. Once the constrained multi-objective problem is reduced to a constrained grey-box single objective problem, the simulation is passed on to the ARGONAUT algorithm for global optimization.

- *Step 3: Perform Latin Hypercube Design within the continuous variable bounds*

Initially, ARGONAUT utilizes Latin Hypercube Sampling to decide on the values of the input variables. Figure 3.2A shows the surface plot of the original objective function in Equation 3.7, and Figure 3.2B shows a sample design of experiments superimposed on the contour plot of the original objective.

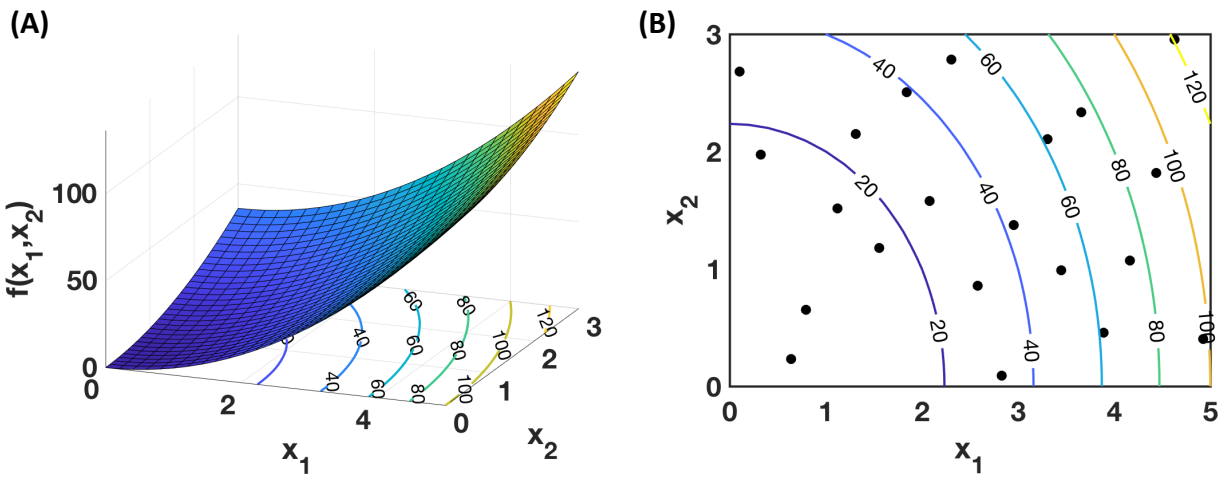

Figure 3.2: Original objective function; (A) shown in a surface plot and (B) shown in a contour plot superimposed on the initial sampling points to be collected by ARGONAUT.

- *Step 4: Perform global parameter estimation on each unknown equation*

In the next step, a subset of the collected samples is randomly chosen and passed on to the parameter estimation phase where the least-squares error between the predictions and the observed data is minimized to global optimality. This procedure is repeated k times (k-fold cross-validation), each starting with a random subset of samples, for all the unknown formulations. The surrogate identification is based on the cross-validation mean squared error (CVMSE) calculated across these repetitions and the surrogate with minimum CVMSE is selected. Table 3.2 summarizes the results of the first parameter estimation for the motivating example.

Table 3.2: Results from the first parameter estimation using ARGONAUT. In this case, quadratic surrogates are fitted to the initial sampling points.

Unknown equation	**Surrogate formulation from ARGONAUT**
Objective	$f(x_1, x_2) = 0.091 - 1.222x_1 - 0.038x_2 + 0.784x_1^2 - 2.254 \cdot 10^{-7}x_1x_2 + 0.276x_2^2$
Constraint #1	$1.478 - 1.385x_1 - 0.795x_2 + 0.621x_1^2 - 3.550 \cdot 10^{-8}x_1x_2 + 0.219x_2^2 \leq 0$
Constraint #2	$1.029 - 1.675x_1 - 0.037x_2 + 0.751x_1^2 - 8.010 \cdot 10^{-8}x_1x_2 + 0.265x_2^2 \leq 0$
Constraint #3	$0.325 - 1.098x_1 - 0.218x_2 - 0.316x_1^2 - 1.117 \cdot 10^{-7}x_1x_2 - 0.111x_2^2 \leq 0$

- *Step 5: Solve the resulting NLP, identify new sampling points, and cluster data*

Surrogate formulations presented in Table 3.2 are passed on to the optimization phase where multiple local solutions at pre-determined points are calculated alongside with the global optimum. These optimal results now become the new sampling points and this procedure is repeated until a convergence criteria is met.

Once the convergence is achieved, a session is completed and ARGONAUT clusters the data based on the Euclidean distance between the samples. Clustering of the samples for this problem is shown in Figure 3.3A, where the results are clustered into 6 different groups with the best cluster

shown in diamonds. Based on this clustering analysis, it is possible to further tighten the pre-defined variable bounds to focus in a specific region which provides the best objective. The new variable bounds are shown in Figure 3.3B. Once this region is determined, ARGONAUT resumes with the second session where the sample collection, modeling and optimization procedures are repeated within the reduced bounds. Once the desired accuracy is achieved in the second session, ARGONAUT will reach convergence and terminate the process.

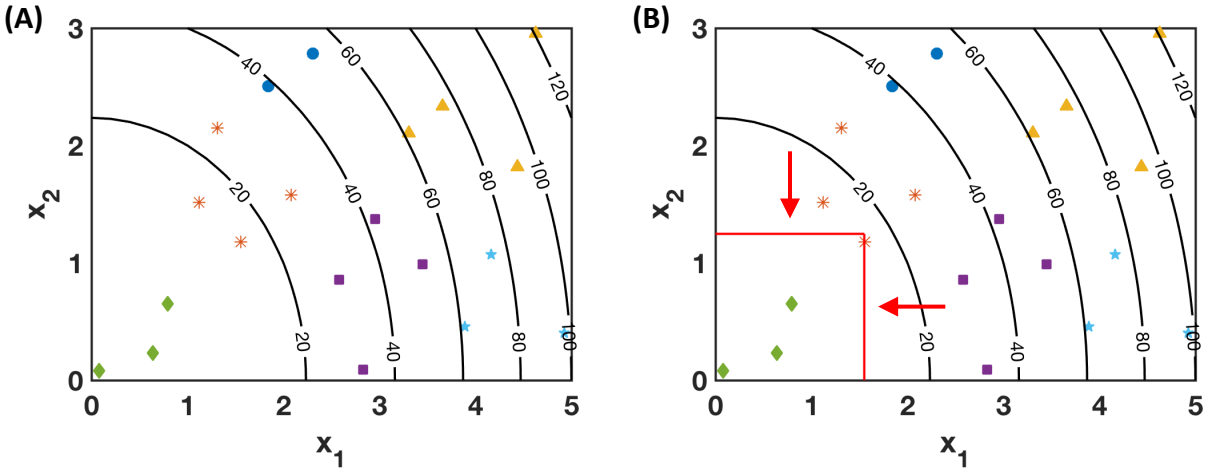

Figure 3.3: Clustering results for the motivating example; (A) Each cluster is represented with different shapes where the best cluster is given in diamonds; (B) Based on the best cluster, variable bounds are tightened and refined to the box marked with arrows. New iterations will now focus on this region for improved solutions.

- *Step 6: Final solution*

ARGONAUT returns the global solution as $x_1^* = 0.07995, x_2^* = 0.07995$ with the objective value of $f(x_1^*, x_2^*) = 0.051136$, which is significantly close to the actual deterministic solution $\left(x_1^* = 0.07995, x_2^* = 0.07995, f(x_1^*, x_2^*) = 0.051135\right)$. The plot of the final approximation generated by ARGONAUT is shown in Figure 3.4.

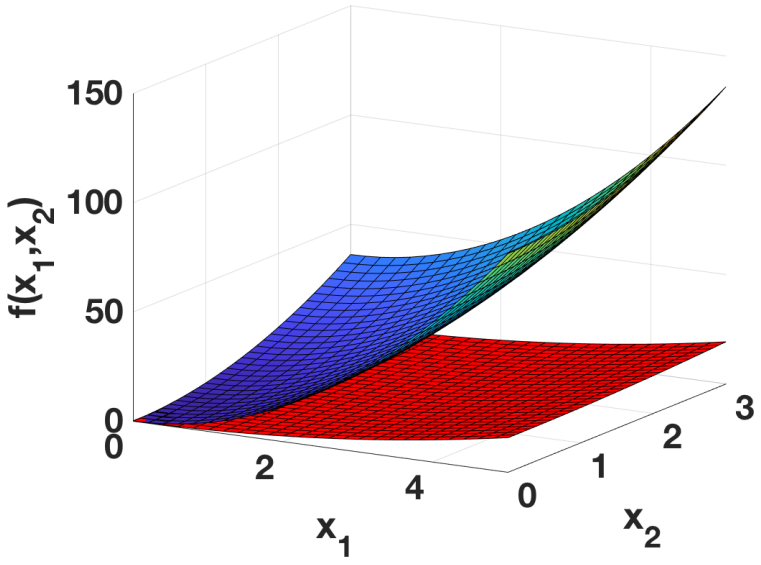

Figure 3.4: Comparison of the original objective function (top layer in blue) and its scaled surrogate formulation obtained using ARGONAUT (bottom layer in red).

3.5 Computational Studies

3.5.1 Benchmark Problems

Initially, the framework is tested on three constrained MOO benchmark problems, namely the Binh and Korn function (BNH), the CONSTR problem and the car-side impact test problem [130–132]. The BNH and CONSTR problems contain 2 objectives, 2 variables, and 2 constraints, whereas the car-side impact problem has 3 objectives, 7 variables and 10 constraints. Problem formulations are provided in Table 3.3.

3.5.2 Energy Systems Design Model for a Supermarket

In addition to the benchmark problems, the generic framework is extensively tested on a higher dimensional MOO problem, where an energy systems design model is chosen as the case study. This problem is initially investigated by Liu et al. [4], in which the authors proposed a superstructure and a mixed-integer model for the utilization of various available technologies for energy generation in a commercial building, as well as a multi-objective optimization strategy that minimizes the cost along with the environmental impact. In this work, the aforementioned relatively

Table 3.3: Multi-objective optimization test problems.

Test Problem	Mathematical Formulation
BNH	$\min_{\boldsymbol{x}} U = \begin{cases} f_1 = 4x_1^2 + 4x_2^2 \\ f_2 = (x_1 - 5)^2 + (x_2 - 5)^2 \end{cases}$ $s.t.\ (x_1 - 5)^2 + x_2^2 \leq 25$ $(x_1 - 8)^2 + (x_2 + 3)^2 \geq 7.7$ $x_1 \in [0, 5]$ $x_2 \in [0, 3]$
CONSTR	$\min_{\boldsymbol{x}} U = \begin{cases} f_1 = x_1 \\ f_2 = \frac{(1+x_2)}{x_1} \end{cases}$ $s.t.\ 9x_1 + x_2 - 6 \geq 0$ $9x_1 - x_2 - 1 \geq 0$ $x_1 \in [0.1, 1]$ $x_2 \in [0, 5]$
Car-side Impact	$\min_{\boldsymbol{x}} U = \begin{cases} f_1 = 1.98 + 4.9x_1 + 6.67x_2 + 6.98x_3 + 4.01x_4 \\ \qquad +1.78x_5 + 0.00001x_6 + 2.73x_7 \\ f_2 = F \\ f_3 = 0.5(V_{MBP} + V_{FD}) \end{cases}$ $s.t.\ 1.16 - 0.3717x_2x_4 - 0.0092928x_3 \leq 1$ $0.261 - 0.0159x_1x_2 - 0.06486x_1 - 0.019x_2x_7$ $+0.0144x_3x_5 + 0.0154464x_6 \leq 0.32$ $0.214 + 0.00817x_5 - 0.0587x_1 + 0.03099x_2x_6 - 0.018x_2x_7$ $+0.0304x_3 - 0.00364x_5x_6 - 0.018x_2^2 \leq 0.32$ $0.74 - 0.61x_2 - 0.031296x_3 - 0.031872x_7 + 0.227x_2^2 \leq 0.32$ $28.98 + 3.818x_3 - 4.2x_1x_2 + 1.27296x_6 - 2.68065x_7 \leq 32$ $33.86 + 2.95x_3 - 5.057x_1x_2 - 3.795x_2 - 3.4431x_7$ $+1.45728 \leq 32$ $46.36 - 9.9x_2 - 4.4505x_1 \leq 32$ $F = 4.72 - 0.5x_4 - 0.19x_2x_3 \leq 4$ $V_{MBP} = 10.58 - 0.674x_1x_2 - 0.67275x_2 \leq 9.9$ $V_{FD} = 16.45 - 0.489x_3x_7 - 0.843x_5x_6 \leq 15.7$ $x_1 \in [0.5, 1.5]$ $x_2 \in [0.45, 1.35]$ $x_3 \in [0.5, 1.5]$ $x_4 \in [0.5, 1.5]$ $x_5 \in [0.875, 2.625]$ $x_6 \in [0.4, 1.2]$ $x_7 \in [0.4, 1.2]$

high-dimensional model is used to test the constrained grey-box global optimization algorithm. The superstructure of the problem can be found in Figure 3.5.

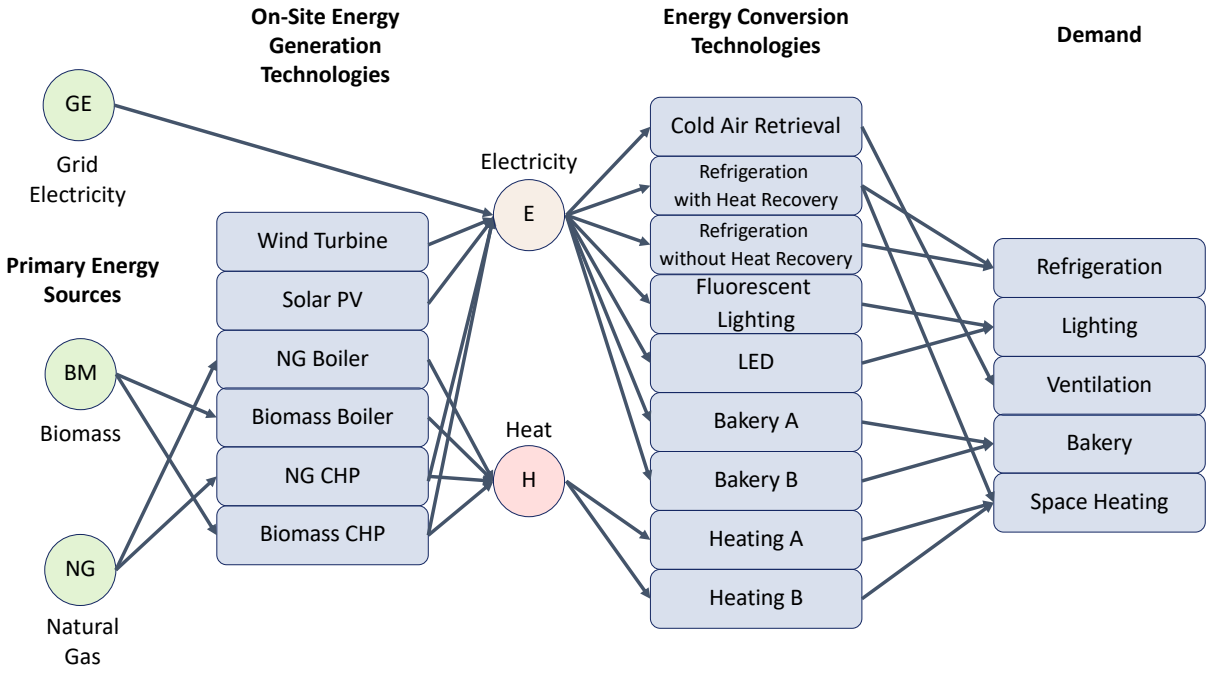

Figure 3.5: Superstructure for the energy design problem for a commercial building.

As demonstrated in Figure 3.5, the problem contains two primary energy sources, namely biomass and natural gas, which are converted into electricity and/or heat using the available on-site energy generation technologies. The total electricity (generation + supply from the electricity grid) and heat will then be converted into an output, using the energy conversion technologies shown in Figure 3.5, to meet the demand in refrigeration, lighting, ventilation, bakery and space heating. This can be mathematically modeled as follows. First, the conversion of primary sources to electricity and heat is subject to energy balance, where any generated capacity using the on-site energy generation technologies must be proportional to the efficiency of the technology and to the

energy provided by the primary source:

$$CAP_i^e \cdot t_i \cdot T = P_{ij} \cdot \eta_i^e \quad \forall i = 1, \ldots, 6, j = 1, 2 \tag{3.8}$$

$$CAP_i^h \cdot t_i \cdot T = P_{ij} \cdot \eta_i^h \quad \forall i = 1, \ldots, 6, j = 1, 2 \tag{3.9}$$

CAP_i^e and CAP_i^h denote the capacity of electricity and heat generated in (kW), respectively, t_i is the availability of the i^{th} on-site energy generation technology throughout the year given in (hr/yr), T is the total time of operation in years, P_{ij} is the amount of energy delivered by the utilization of the j^{th} energy source by the i^{th} on-site energy generation technology in kJ, and η_i^e and η_i^h denote the efficiency of the i^{th} on-site energy generation technology for electricity and heat generation, respectively. The availability of each technology is bounded (Equation 3.10), given that the technologies can be available for a certain amount of time during the year (τ_i).

$$t_i \leq \tau_i \quad \forall i = 1, \ldots, 6 \tag{3.10}$$

In addition, the capacity resulting from each energy generation technology is bounded, as modeled in Equation 3.11, and binary variables are included in the model to represent the selection of available technologies.

$$y_i \cdot CAP_i^L \leq CAP_i \leq y_i \cdot CAP_i^U \quad \forall i = 1, \ldots, 6, y_i \in \{0, 1\} \tag{3.11}$$

Here, CAP_i represents the total capacity of energy generated, both in the form of electricity and heat by a given technology.

$$CAP_i = CAP_i^e + CAP_i^h \quad \forall i = 1, \ldots, 6 \tag{3.12}$$

The total amount of electricity (E_{Total}), including any supply from the grid (e_{grid}), and heat

(H_{Total}) generated using the on-site energy generation technologies is defined as:

$$E_{Total} = T \cdot \sum_{i=1}^{6} CAP_i^e \cdot t_i + e_{grid} \tag{3.13}$$

$$H_{Total} = T \cdot \sum_{i=1}^{6} CAP_i^h \cdot t_i \tag{3.14}$$

The energy balance on total electricity and heat dictates that the total amount generated must be utilized in energy conversion technologies (CAP_k^{conv}) into an output. Here, it is assumed that there is no energy dissipation to the surroundings in on-site energy generation and conversion technologies. Thus, the total amount of energy generated is equal to the total energy utilized in the next step, as mathematically expressed in Equations 3.15 and 3.16.

$$E_{Total} = T \cdot \sum_{k=1}^{7} CAP_k^{conv} \tag{3.15}$$

$$H_{Total} = T \cdot \sum_{k=8}^{9} CAP_k^{conv} \tag{3.16}$$

It is important to note that only a portion of the energy conversion technologies take electricity or heat as an input. Hence, only the relevant conversion technologies are included in each balance. The details on the technical parameters for energy conversion technologies can be found in Table 3.6 in Section 3.6.2.

The final amount of output capacity ($Output_k^U$) generated using the appropriate energy conversion technologies is proportional to the efficiency of the corresponding technology (η_k^{conv}), as shown in Equation 3.17.

$$Output_k^U = CAP_k^{conv} \cdot \eta_k^{conv} \quad \forall k = 1, \ldots, 9 \tag{3.17}$$

Here U, is the set of end-uses for the generated output which significantly contribute to the energy consumption in a supermarket such as refrigeration, lighting, ventilation, bakery, and space

heating. Thus, in this supermarket case study, $U \in \{Refrigeration, \ldots, Space\ Heating\}$. The final output generated using each technology can only be utilized in a specific end-use and must meet the demand, as shown in Equation 3.18.

$$\sum_{k \in U} Output_k^U \geq Demand^U \quad \forall U = \{Refrigeration, \ldots, Space\ Heating\} \tag{3.18}$$

Given the energy balances, conversion equations, bounds and the demand constraints, the objectives in this case study is to minimize the cost of energy generation alongside with the total CO_2 emissions, explicitly defined in Equations 3.19 and 3.20.

$$\begin{aligned} Cost = &\sum_{i=1}^{6} INV_i \cdot CAP_i + T \sum_{i=1}^{6} OM_i \cdot CAP_i + \sum_{k=1}^{9} INV_k \cdot CAP_k^{conv} \\ &+T \sum_{k=1}^{9} OM_k \cdot CAP_k^{conv} + e_{grid} \cdot Price_{grid} + \sum_{j=1}^{2} Price_j \sum_{i=1}^{6} P_{ij} \end{aligned} \tag{3.19}$$

$$Emission = e_{grid} \cdot Emission_{grid} + \sum_{j=1}^{2} Emission_j \sum_{i=1}^{6} P_{ij} \tag{3.20}$$

INV and OM represent the investment cost (\$/kW), and operation and maintenance costs (\$/kW/yr) associated with each on-site energy generation or conversion technology, respectively. $Price_j$ and $Emission_j$ represents the price of the primary energy source per GJ of energy delivered and the amount of CO_2 emitted (kton CO_2/PJ) by each primary energy source, respectively. Subscript "grid" indicates the price and emissions related to the electricity supplied from the electricity grid.

3.6 Results of Computational Studies

Series of computational studies have been performed on the benchmark problems and on the energy systems design problem to test the accuracy and consistency of the proposed data-driven multi-objective optimization framework. Two other grey/black-box optimization solvers are also tested alongside the ARGONAUT algorithm to fully characterize the effect of different data-driven solvers on the integrated framework performance. The following solvers are tested as a part of

this framework: Improved Stochastic Ranking Evolution Strategy (ISRES) [96], and the Nonlinear Optimization by Mesh Adaptive Direct Search (NOMAD) algorithm [95]. The description of these solvers are provided in Chapter 2, Table 2.1.

In this work, an exhaustive comparison between all the recently published black-box algorithms is not performed, however, the performance of the proposed approach is compared with two widely accepted algorithms that can handle general black-box constraints. The criteria behind selecting these two algorithms for comparison is directly associated with their ability to handle nonlinear constraints and availability through user-friendly implementations [97, 98]. It should also be mentioned that these search algorithms have been executed without any tuning of their convergence parameters, which were left at their default settings. For all the benchmark problems, ARGONAUT is also tested at the default setting, assuming no *a priori* knowledge on the analytical forms of the equations, and let ARGONAUT perform model identification, parameter estimation and cross-validation. For the energy systems design problem, ARGONAUT is initially used at the default setting to perform numerous tests on the constrained problem. Through cross-validation of various types of functions in the library, individualized information regarding the optimal surrogate representation for each unknown function is gathered. Then, using this information, the parameter estimation problem is solved to global optimality for only these specific types of surrogates. Detailed information about the dimensionality of this case study as well as the surrogates used in modeling the energy systems design problem is provided in Table 3.9.

All the test problems are executed 10 times on a High-Performance Computing (HPC) machine at Texas A&M High-Performance Research Computing facility using Ada IBM/Lenovo x86 HPC Cluster operated with Linux (CentOS 6) using 1 node (20 cores per node with 64 GB RAM) for ARGONAUT runs, and on Intel Core i7-4770 CPU (3.4 GHz) operated with Linux (CentOS 7) for the other solvers. The average results for each solver across these 10 runs are reported in the following sections. It is also important to state that for fairness, the starting sampling design for ARGONAUT as well as the starting points for ISRES and NOMAD are randomly generated for each of the 10 executions of these solvers.

3.6.1 Pareto-optimal Solution for the Benchmark Problems

The Pareto-optimal curves resulting from this study are shown in scatter plots, given in Figures 3.6 and 3.7. Each row of figures represents a solver that is used to optimize the grey-box system, where Figures 3.6A, C, and E show the results for BNH, and Figures 3.6B, D, and F show the results for CONSTR benchmark problem. In addition, Figure 3.7 summarizes the results for car-

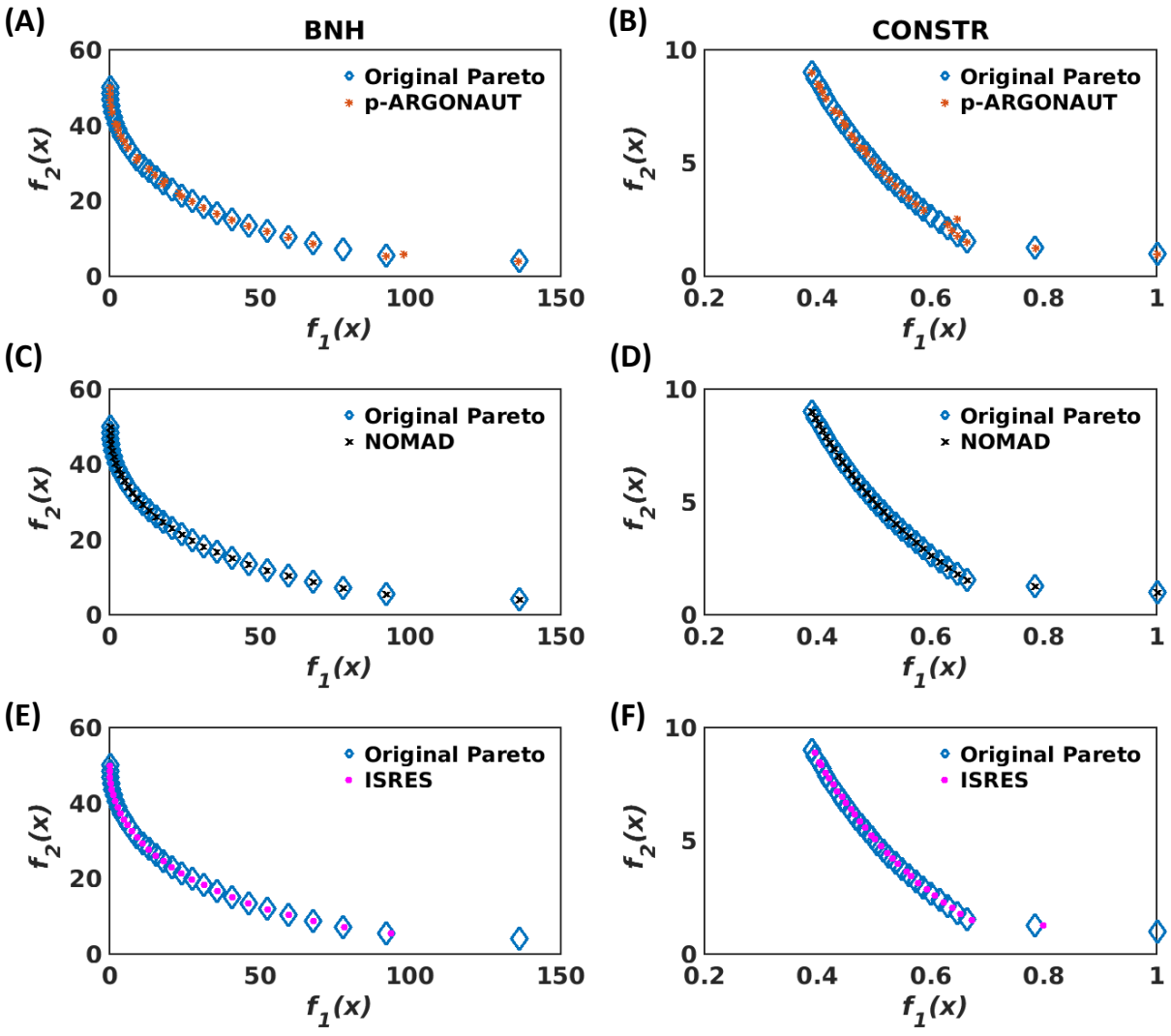

Figure 3.6: Pareto-optimal curves for the BNH and CONSTR benchmark problems. Diamonds represent the exact global solution for the fully deterministic problem.

side impact problem for all methods. All the results shown in Figures 3.6 and 3.7 are also compared with the exact global solution of the fully deterministic problem, which are shown in diamonds. Figure 3.6 demonstrates that all three optimization methods show good performance in locating the true global optimum at every point of the Pareto-optimal curve. In Figures 3.6E and F, it is observed that ISRES is unable to find a feasible solution for the very last point of the Pareto-curve over the course of 10 random runs. It is suspected the reason behind such a behavior is due to the stochastic nature of this algorithm [133]. The average results show that NOMAD outperforms ARGONAUT and ISRES algorithms in the BNH and CONSTR problems as shown in Figure 3.6C and D in locating the true global optimum. This increased performance of NOMAD can be explained in two-fold: (1) These two problems are relatively easy functions and the random initial starting point actually provides good solutions to the problem; (2) These good solutions are further refined towards the global solution due to NOMAD's detailed local exploration strategy which results in surpassing the performance of two global methods. It is worth mentioning that even though NOMAD, on average, seems to better locate the optimal point, the average performance does not consider the cases where NOMAD has failed to find a feasible solution. For the BNH benchmark problem, NOMAD returns highly infeasible solutions in 12% of the total number of runs. The performance is better for the CONSTR problem, where only in less than 1% of the executed runs, NOMAD terminates with an infeasible solution. This also shows that the location of the initial point provided for the algorithm plays a critical role in terms locating the global optimum and for identifying a feasible solution. On the contrary, ARGONAUT provides feasible solutions consistently for all the runs, which is a significant advantage of the algorithm compared to other methods.

Furthermore, Figure 3.7 shows the Pareto-front for the car-side impact benchmark problem where the trade-off solutions between three objective functions form the Pareto-optimal surface. Figures 3.7A and B show that both ARGONAUT and NOMAD on average perform well in locating the global solution in a higher dimensional problem. In total of 640 runs (64 points with 10 repetitive runs) executed to generate the Pareto-optimal surface, the NOMAD algorithm has returned an

infeasible solution in 10% of the runs. On the contrary, ARGONAUT was able to provide feasible solutions to all runs, where only in 2% of all cases the algorithm has returned a sub-optimal solution (a solution with an absolute error greater than 10^{-3} with respect to the true global solution). This clearly shows that ARGONAUT can sustain the solution accuracy over multiple repetitions, while being subject to variations at the initialization stage. Moreover, in Figure 3.7C, it is observed that ISRES is unable to locate any feasible solution in 36% of 640 runs whereas it converges to sub-optimal solutions in others. As expected, as the problem complexity increases, it is harder for all algorithms to find the optimal set of decision variables. Hence, compared to the results shown in Figure 3.6, ISRES and NOMAD algorithms have terminated with highly infeasible solutions in more runs than in lower dimensional problems, resulting in higher number of mismatches between the true global solution on Pareto-optimal curves. This result is compelling especially for problems with higher number of variables and constraints, where augmented number of failures in identifying the global solution would interfere with the shape of the Pareto-optimal curve and may alter the decision maker's ultimate judgment.

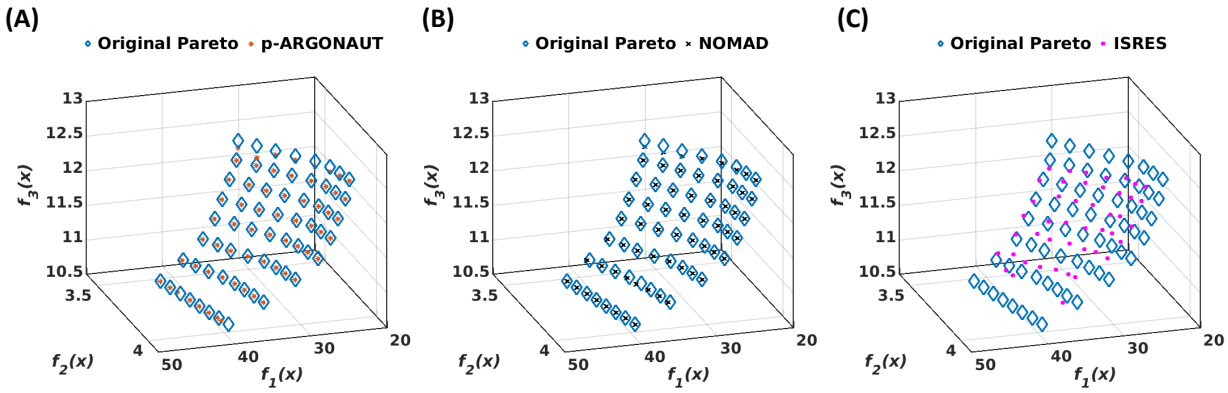

Figure 3.7: Pareto-optimal surfaces generated by different solvers for the car-side impact benchmark problem; (A) ARGONAUT; (B) NOMAD; (C) ISRES. Diamonds represent the exact global solution for the fully deterministic problem.

In addition to assessing the consistency and accuracy of different solvers, a comparison is established based on their computational performance, both in terms of sample collection and elapsed

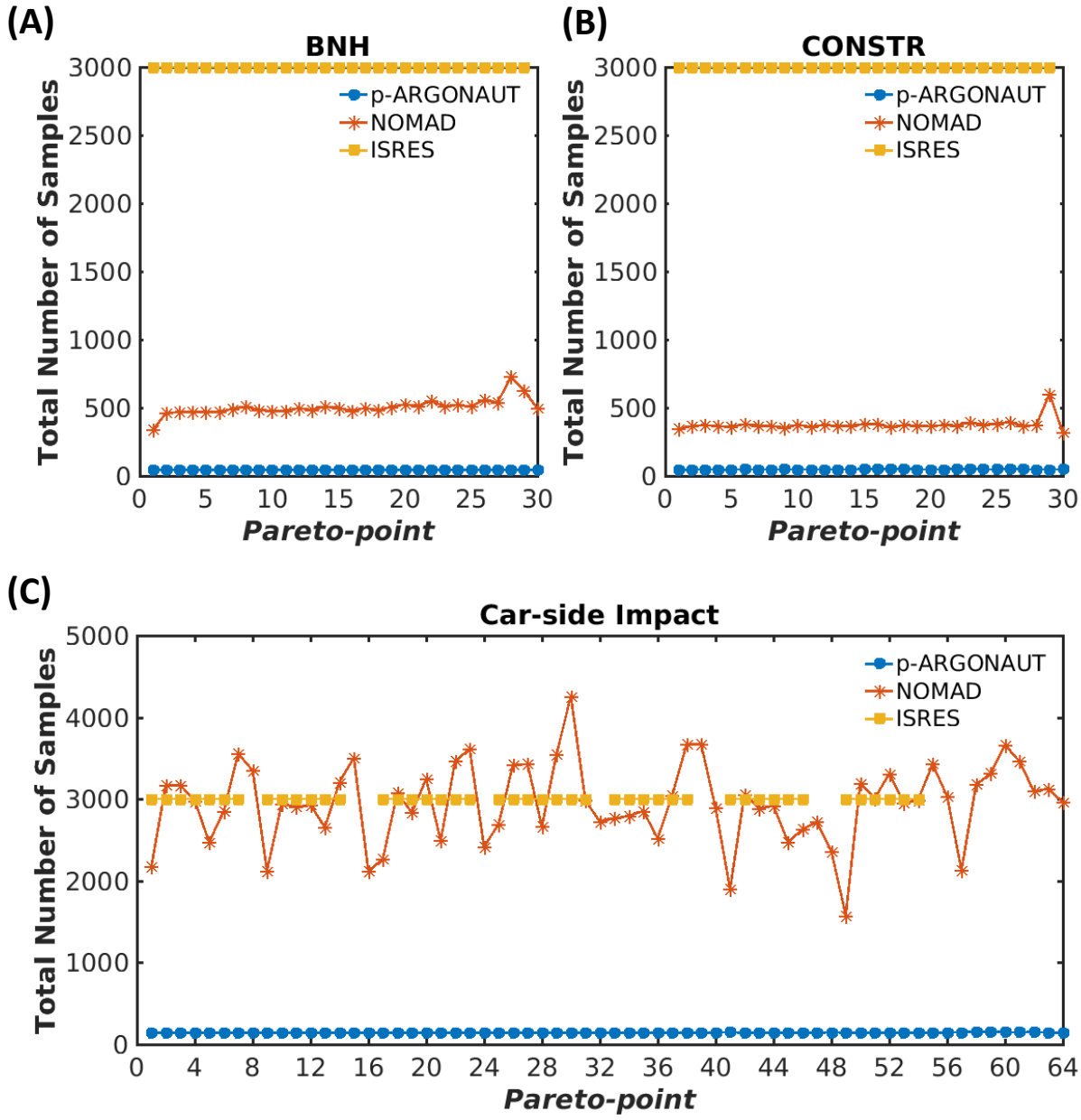

Figure 3.8: Comparison of average total number of samples collected by each solver in each benchmark problem. Results are shown for (A) BNH, (B) CONSTR and (C) car-side impact benchmark problems.

time, as shown in Figures 3.8 and 3.9, respectively. The infeasible results are excluded from both figures. In Figure 3.8, it is observed that ISRES collects 3000 samples for all benchmark problems which is also the maximum allowable number of function evaluations that was set. This observation may suggest that ISRES could have a better performance if more samples were collected, but the value of this limit is decided on by realizing that one of the main computational challenges of black-box optimization is convergence with a reasonable number of calls to the expensive black-box simulation. NOMAD algorithm on the other hand, collects about 500 samples in average on lower dimensional benchmark problems (Figure 3.8A and B), whereas the total number of samples collected significantly increases for the car-side impact benchmark (Figure 3.8C). However, ARGONAUT collects less than 100 samples on average for the BNH and CONSTR problems and less than 205 samples for the car-side impact problem while converging to globally optimal solutions. This feature of ARGONAUT is quite advantageous, especially for the problems with computationally expensive simulations, where the sample collection can significantly burden the whole optimization process.

Furthermore, Figure 3.9 shows the average elapsed time spent by each solver for the three different benchmark problems. As demonstrated in Figures 3.9A and C, ISRES and NOMAD algorithms take relatively longer time to converge to an optimum, as oppose to ARGONAUT. Especially for the NOMAD algorithm, the computational usage has increased at least by 5-fold with the increasing number of dimensions and problem complexity. This shows that NOMAD's refinement and detailed local search strategies comes with added number of function evaluations in higher dimensional problems, which in return increases the total amount of CPU time it takes for the algorithm to converge to an optimum. Interestingly, in Figure 3.9B, it is observed that ARGONAUT takes significant amount of time to converge to the global optimum in comparison to the other solvers. The reason behind this large difference in elapsed times across different benchmark problems is that the BNH and car-side impact benchmark problems are approximated via linear and/or quadratic surrogates within ARGONAUT, whereas the CONSTR problem is modeled via kriging and/or radial basis functions. As a result, the global optimization of convex surrogates

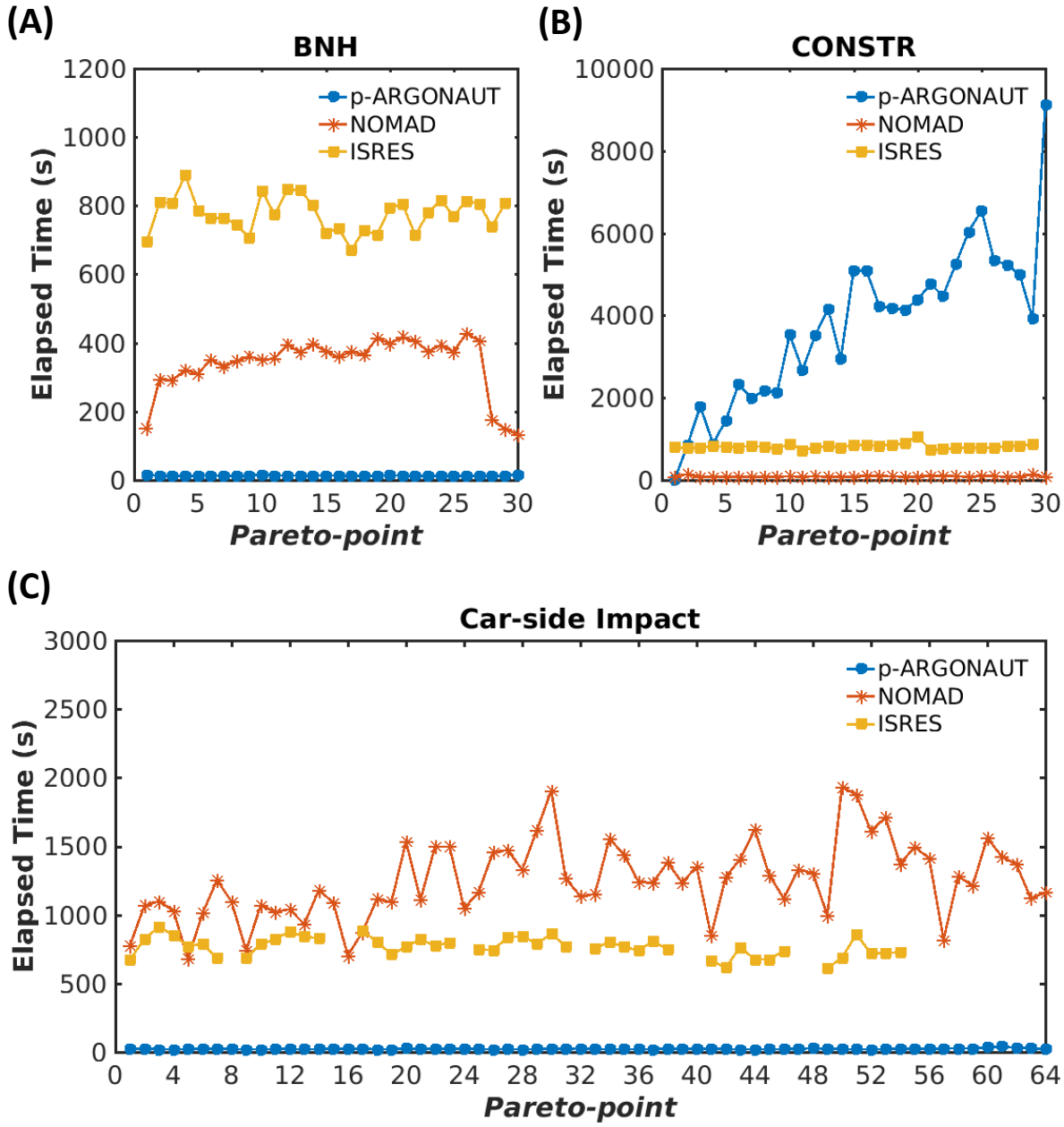

Figure 3.9: Average elapsed time for each solver across all the Pareto-points for (A) BNH; (B) CONSTR; (C) car-side impact benchmark problems.

that represent the BNH and car-side impact benchmark problems are much easier and much faster compared to the global optimization of nonconvex functions, which is the case in the CONSTR problem. Thus, the deterministic global optimization of nonconvex surrogate formulations representing the unknown objective and constraints adds up to the computational time it takes for ARGONAUT to converge to the optimum.

3.6.2 Pareto-optimal Solution for the Energy Systems Design Problem

In addition to the benchmark problems, the performance of the framework is extensively tested on a relatively high-dimensional energy systems design problem in a commercial building. The prices for the energy sources as well as the parameters associated with the costs, capacities and availabilities of each technology in the supermarket case study is summarized in Tables 3.4-3.6. In this case study, it is assumed that all the energy conversion technologies are available throughout the entire operation time horizon, which is set to 20 years.

Table 3.4: Prices and CO_2 emissions of energy sources and grid electricity [4].

	Natural Gas	**Electricity**	**Biomass**
Price ($/GJ)	8.89	36.11	9.72
CO_2 Emission (kton CO_2/PJ)	56	90	100

Table 3.5: Technical and economic parameters of on-site energy generation technologies [4].

Technology	η^e	η^h	CAP^L **(kW)**	CAP^U **(kW)**	τ **(hr/yr)**	**INV ($/kW)**	**O&M ($/kW/yr)**
Wind Turbine	-	-	10	30	1750	2000	1200
Solar PV	-	-	10	20	800	2000	500
NG Boiler	-	0.9	100	10^6	7000	200	10
Biomass Boiler	-	0.85	100	10^6	7000	250	15
NG CHP	0.35	0.55	800	10^6	7000	500	15
Biomass CHP	0.33	0.50	1000	10^6	7000	2000	30

Table 3.6: Technical and economic parameters of energy conversion technologies [4]. COP stands for coefficient of performance.

Technology	Input	Output	η^{conv}	INV ($/kW)	O&M ($/kW/yr)
Cold Air Retrieval	Electricity	Ventilation	6(COP)	50	3
Refrigeration with Heat Recovery	Electricity	Refrigeration, Space Heating	3,2(COP)	100	5
Refrigeration without Heat Recovery	Electricity	Refrigeration	3(COP)	70	4
Fluorescent Lighting	Electricity	Lighting	0.2	5	0.5
LED	Electricity	Lighting	0.8	10	1
Bakery A	Electricity	Bakery	0.7	30	3
Bakery B	Electricity	Bakery	0.75	40	4
Heating A	Heat	Space Heating	0.85	30	3
Heating B	Heat	Space Heating	0.9	40	4

In addition to the parameters taken from the original case study, this energy consumption problem is also investigated with the current updated values, which are shown in Tables 3.7 and 3.8, to observe the shift in the Pareto-optimal solution with changing prices. In the updated case, the technical and economic parameters regarding the energy conversion technologies are kept unchanged as in Table 3.6.

Table 3.7: Current prices and CO_2 emissions of energy sources [5–8].

	Natural Gas	Electricity	Biomass
Price ($/GJ)	7.056	28.694	8.137
CO_2 Emission (kton CO_2/PJ)	48.548	138.094	101.729

As it was shown previously in Equation 3.11, the selection of on-site energy generation technologies is handled via binary variables. This study does not enumerate all the possible combinations ($2^6 = 64$ possible combinations) but only show the results for 1 cost effective (natural gas-powered CHP) and 1 most environmentally benign (wind turbine and solar photovoltaic) set of

Table 3.8: Updated technical and economic parameters for on-site energy generation technologies [4, 9–12].

Technology	η^e	η^h	CAP^L **(kW)**	CAP^U **(kW)**	τ **(hr/yr)**	**INV ($/kW)**	**O&M ($/kW/yr)**
Wind Turbine	-	-	10	50	1750	6118	35
Solar PV	-	-	10	273	2500	2493	19
NG Boiler	-	0.85	88	10^6	8000	107	5
Biomass Boiler	-	0.80	100	10^6	8000	575	98
NG CHP	0.31	0.45	800	10^6	8000	1500	120
Biomass CHP	0.22	0.69	1000	10^6	8000	5792	98

technologies as suggested by Liu et al. [4]. It is also important to note that the equality constraints in the supermarket case study, resulting from the energy balances, shown in Section 3.5.2, further challenges the algorithmic framework to its greatest extents in locating the global optimum with highest accuracy. However, numerical issues may arise while satisfying these equality constraints in the derivative-free context. Thus, all the equality constraints are relaxed into two inequalities while being penalized with a small number (i.e., 1E-6), in order to set the numerical accuracy to 10^{-6}.

The Pareto-optimal curve resulting from the information provided in Table 3.9 as well as the parameters shown in Tables 3.4, 3.5 and 3.6, which reflect the prices and efficiencies reported in 2010, is presented in Figure 3.10. One of the most important characteristics of the curve shown in Figure 3.10A is that, each point represents an equally optimal design with different economic and environmental behaviors when different technologies are used as the on-site energy generation technology in a supermarket. For example, the most cost-effective design is achieved using natural gas-powered CHP system, shown as the very first point on the Pareto-frontier. However, this design completely neglects any constraints on the greenhouse gas emissions and possible impacts on the environment. As a result, CO_2 emissions are at its highest level when the cost is minimal. On the contrary, using wind turbine and solar PV provides an environmentally friendly alternative to the natural-gas powered CHP as an on-site energy generation technology in a supermarket. However, the cost of having this system on a supermarket is now at its maximum value, which is $11M.

Table 3.9: Dimensionality of the multi-objective energy systems design problem. The table also summarizes the types of surrogate used in the study for each grey-box constraint that was present in the problem formulation.

Type of on-site energy generation technology considered	**Number of Input Variables**	**Number of Grey-Box Constraints**	**Types of Surrogates Used**
Natural gas-powered CHP (NG CHP)	17	19	*Objective:* linear *Constraints 1, 6, 7, 10-19:* linear *Constraints 2-5, 8, 9:* quadratic
Wind Turbine & Solar Photovoltaics (WT + SPV)	16	12	*Objective:* quadratic *Constraints:* quadratic

This problem is also studied using the ISRES and the NOMAD algorithms, in which the results are summarized in Figure 3.10B. The complete Pareto-curve generated by ARGONAUT is now presented in squares whereas the results for the NOMAD algorithm are represented in circles, as shown in Figure 3.10B. It is important to note that the ISRES algorithm is unable to locate any feasible solutions within the maximum allowable number of samples (sample tolerance set to 3000) for this case study over the course of 10 runs for each Pareto-point. As a result, only the values found by the NOMAD algorithm are reported in comparison to the results obtained using ARGONAUT. Figure 3.10B demonstrates that the NOMAD algorithm can locate feasible solutions to the problem in the objective space. However, due to its local exploration strategy, the algorithm struggles to converge to the global optimum at each Pareto-point and can only return local feasible solutions. In addition, a fraction of the NOMAD runs is terminated with high infeasibility, where the algorithm is unable to satisfy all the constraints posed in the problem. On the contrary, ARGONAUT is able to report consistent feasible solutions for all the points that construct the Pareto-frontier reported in Figures 3.10A and B.

Furthermore, Figure 3.11 summarizes the computational performance of the two methods with respect to the elapsed computational time and number of samples collected by each method. Figures 3.11A and B show the elapsed time utilized by each solver for the case with natural gas-powered CHP and for the case with wind turbine and solar PV, respectively. For the natural gas-

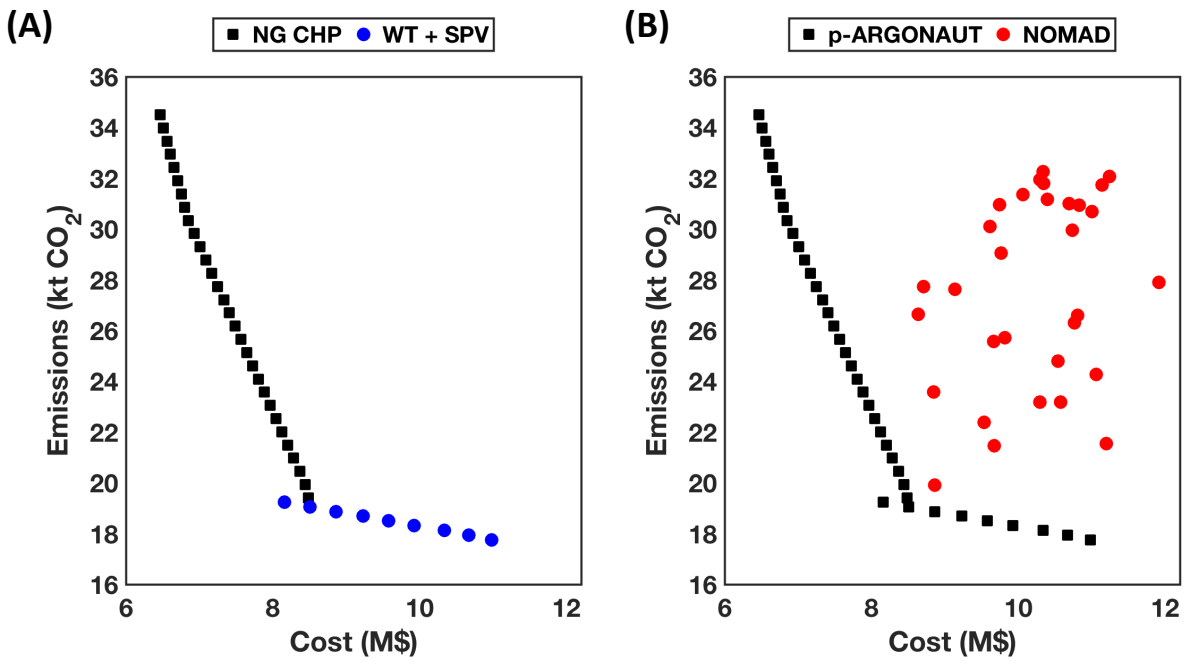

Figure 3.10: Pareto-frontier for the energy systems design problem in a supermarket obtained using ARGONAUT. (A) Pareto-frontier showing the cost-effective design using natural gas-powered CHP technology (NG CHP), and the environmentally friendly design using wind turbine (WT) and solar photovoltaics (SPV); (B) Comparison of results using ARGONAUT and the NOMAD algorithm.

powered CHP case, it is observed that both derivative-free solvers perform comparably with each other. However, Figure 3.11C shows that for the same case study NOMAD collects 5000 points on average per Pareto-point to converge to a feasible solution whereas ARGONAUT collects less than 700 samples per Pareto-point. Compared to the results summarized in Figure 3.8, ARGONAUT converges to the global optimal solution with higher number of samples at every Pareto-point. This is an expected result given that all the derivative-free solvers experience an increase in sampling requirements with increasing problem complexity. This trend is also reflected in NOMAD's results where there is a gradual increase in the total number of samples collected in each problem set, as shown in Figures 3.8 and 3.11.

Moreover, for the case with wind turbine and solar PV, it is observed that ARGONAUT collects significantly low number of samples to converge to global optimum, as shown in Figure 3.11D. It

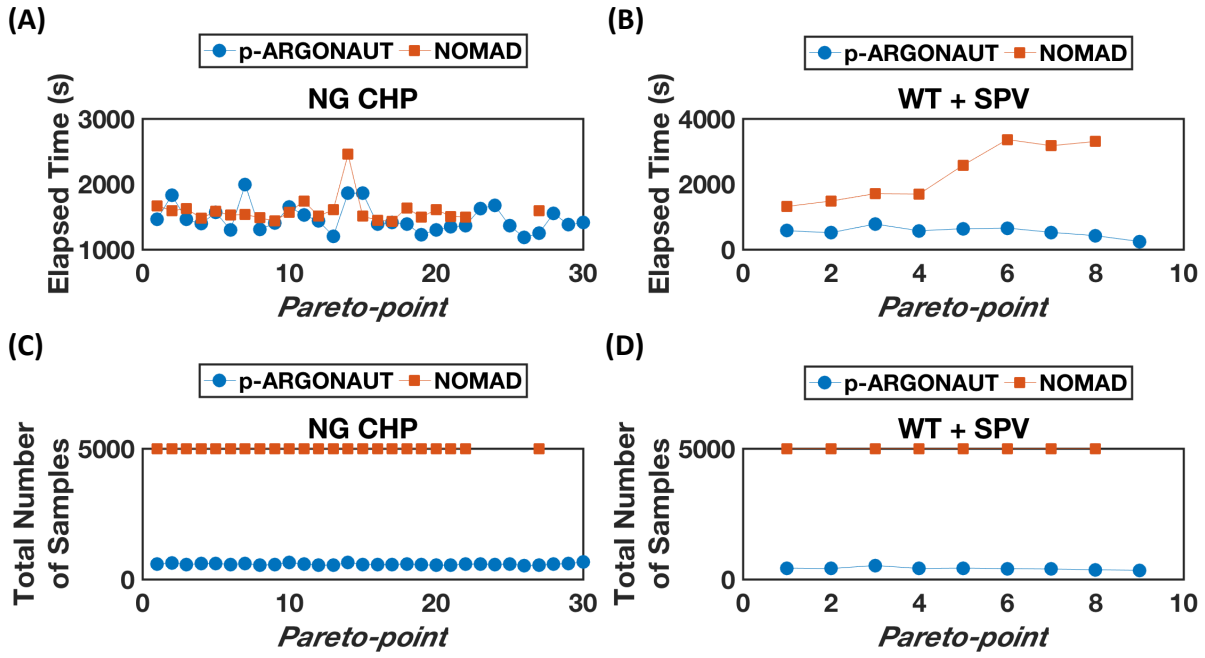

Figure 3.11: Comparison of computational performance of ARGONAUT and NOMAD; (A) Average elapsed time for the ARGONAUT and NOMAD algorithms per Pareto-point in natural gas-powered CHP (NG CHP) case; (B) Average elapsed time for the ARGONAUT and NOMAD algorithms per Pareto-point in wind turbine and solar PV (WT + SPV) case; (C) Average total number of samples collected by the ARGONAUT and NOMAD algorithms per Pareto-point in NG CHP case; (D) Average total number of samples collected by the ARGONAUT and NOMAD algorithms per Pareto-point in WT + SPV case.

is important to note that for both cases NOMAD consistently hits the tolerance set for maximum number of allowable samples and returns the best-found solution from these 5000 collected points. Furthermore, like in Figure 3.10, the results with infeasible solutions are not plotted in Figure 3.11. As a result, one can clearly see that for certain sub-problems in both natural gas-powered CHP and wind turbine and solar PV cases, NOMAD is unable to locate feasible solutions over 10 repetitive runs. Thus, it is safe to say that ARGONAUT outperforms other available derivative-free software, both in terms of computational performance and in accuracy for locating the global solution for the MOO of energy market design problem.

The same case study is repeated with the updated values for prices and efficiencies, where the values of these new parameters are summarized in Tables 3.7 and 3.8. The results for the energy

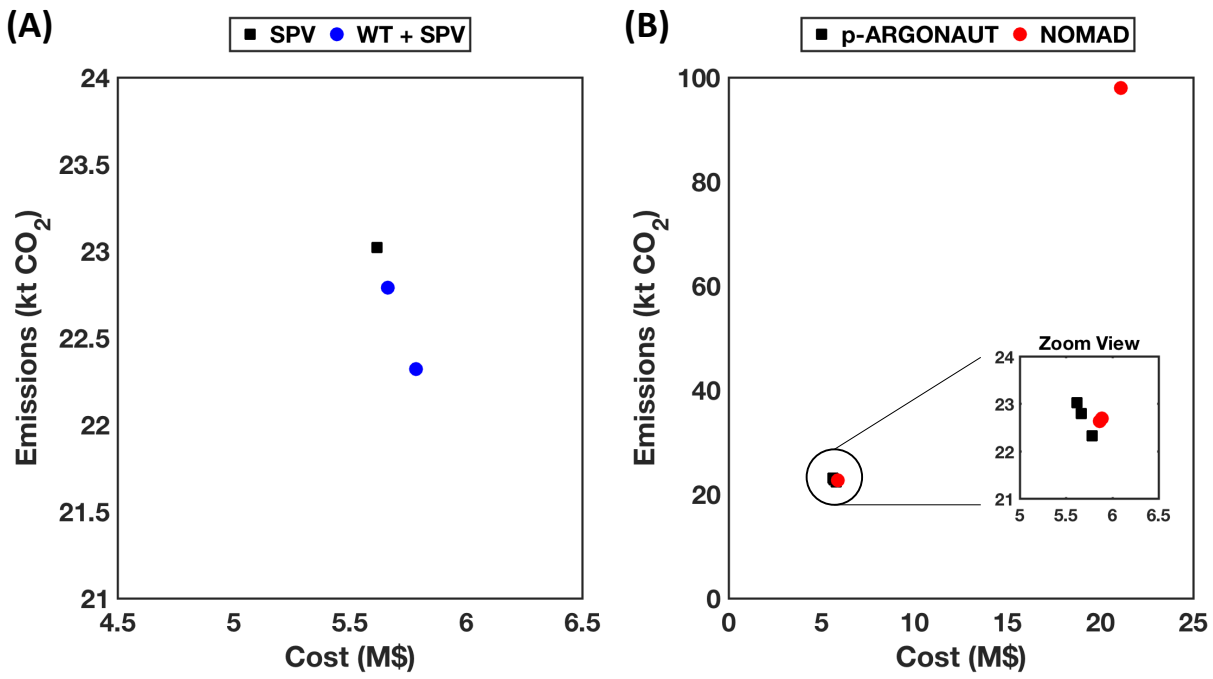

Figure 3.12: Multi-objective optimization results using the updated parameters; (A) Pareto-frontier obtained using ARGONAUT where the cost-effective design is achieved via solar photovoltaics (SPV), and the environmentally friendly design is achieved using wind turbine (WT) and solar photovoltaics (WT + SPV); (B) Comparison of results using ARGONAUT and the NOMAD algorithm.

market design problem with the updated parameters are summarized in Figure 3.12.

Interestingly, Figure 3.12A shows that the most economic on-site energy generation for a supermarket is achieved via solar PV rather than the natural-gas powered CHP. With the recent developments in the solar PV technology, the solar PV's are more available throughout the year with lower operating costs and higher capacities. As a result, the technology selection has shifted from natural gas-based to a renewable-based system for the supermarket. Thus, it is possible to minimize both the cost and the CO_2 emissions of on-site energy generation using solar PV, which also replaces the existing trade-off between the two objectives for this system, while shrinking the Pareto-curve into a single optimum. In addition, Figure 3.12A shows that using wind turbine and solar PV together as the on-site energy generation technologies result in lower CO_2 emissions. Yet again, as in Figure 3.10, as the CO_2 emissions decrease, the cost of having that technology increases. Moreover,

Figure 3.12B shows the comparison between the results obtained using ARGONAUT and the NOMAD algorithm. Like in the previous results, ISRES runs are terminated with high infeasibility for all three points constructing the Pareto-frontier hence, not included in the plots. The results show that NOMAD can locate local feasible solutions, but it struggles to find the Pareto-optimal solution for the current values of the energy market design problem where a fraction of runs has ended with high infeasibility. Especially for one of the points of the Pareto-curve, it is observed that the NOMAD solution is quite distant from the Pareto-optimal solution designated by ARGONAUT. Figure 3.12B also shows a zoomed view of the results that are close to each other. The zoomed picture shows that the NOMAD solutions are very close to the optimal solutions found by ARGONAUT but still does not perfectly capture the global solution.

3.7 Concluding Remarks

In this chapter, a hybrid framework is introduced for solving a class of mathematical programming problems, namely the general constrained multi-objective optimization problems, using a data-driven strategy. This hybrid framework integrates the ϵ-constraint methodology with a constrained grey-box optimization solver for the reformulation of multi-objective optimization problems into series of single objective sub-problems and for their respective optimization through a data-driven methodology. The performance of the framework is tested on three constrained multi-objective benchmark problems from the literature and on a case study of energy market design problem for a commercial building. The results show that ARGONAUT can consistently and efficiently identify the Pareto-frontier, which entails all the trade-off solutions that are equally optimal with respect to each other, under varying conditions and dimensions of constrained multi-objective problems. Furthermore, ARGONAUT outperforms other available derivative-free algorithms by providing consistent feasible solutions for the energy systems design case study, involving numerous equality constraints which are typically challenging for general derivative-free algorithms.

4. DATA-DRIVEN OPTIMIZATION OF STIFF DIFFERENTIAL ALGEBRAIC EQUATIONS WITH APPLICATIONS TO THERMAL CRACKING OF NATURAL GAS LIQUIDS

In this chapter, a Support Vector Machines (SVMs) based optimization framework is presented for the data-driven optimization of stiff Differential Algebraic Equations (DAEs) without the full discretization of the underlying first-principles model. By formulating the stability constraint of the numerical integration of a stiff DAE system as a supervised classification problem, it is demonstrated that SVMs can accurately map the feasible boundary of stiffness. The necessity of this data-driven approach is shown on a 2-dimensional motivating example, where highly accurate SVM models are trained, tested and validated using the data collected from the numerical integration of stiff DAEs. Furthermore, this methodology is extended and tested for a multi-dimensional case study from reaction engineering (i.e., thermal cracking of natural gas liquids). The data-driven optimization of this complex case study is explored through integrating the SVM models with a constrained global grey-box optimization algorithm, namely the ARGONAUT framework.

This chapter is organized as follows. First, the challenges with data-driven optimization in the presence of stiff DAEs or stiff Ordinary Differential Equations (ODEs) are discussed, and the stated challenges are demonstrated on a motivating example in Section 4.2. Next, in Section 4.3, the SVM-based filtering methodology is described and its implementation to a data-driven optimization algorithm for the global optimization of stiff DAEs is provided in Section 4.4. Finally, the algorithm is tested on a steam cracking model for ethylene and propylene production and the results for computational experiments are provided (Section 4.6) along with the concluding remarks (Section 4.7).

4.1 Differential Algebraic Equations and Dynamic Programming

The system of differential algebraic equations (DAEs) is ubiquitous in mathematical modeling of chemical engineering systems, as many first-principles models include differential equations like mass, energy, momentum balances along with process constraints, such as physical properties

and rate laws. DAE systems are commonly observed in the areas of process control, as well as chemical reactions and reactor design [134, 135].

The mathematical optimization of such systems is challenging since the direct implementation of deterministic optimization methods is prohibitive. Hence, many dynamic programming problems in the aforementioned application areas utilize commercial software like the gPROMS environment and Aspen Custom Modeler for first-principles modeling of DAE systems and their respective dynamic optimization [136–138]. Alternatively, full discretization of the DAE system and its incorporation into a nonlinear programming (NLP) formulation using orthogonal collocation on finite elements is also preferred for making DAEs amenable for optimization, specifically for unstable and ill-conditioned problems [139–141]. For example, Caballero et al. [142] investigated the optimization of ethylene production through one-dimensional plug-flow model at steady-state conditions with heat flux along the reactor length to be the only decision variable. In the problem formulation, the equality constraints governing the rate, mass, energy and momentum balance equations were expressed with stiff nonlinear DAEs, which inhibits the global optimization via direct deterministic methods. Hence, the authors implemented the orthogonal collocation on finite elements method that will spatially discretize the DAEs into a set of nonlinear equality constraints, while solving the resulting large-scale NLP problem to local optimality. In another study by Onel [17], the dynamic optimization of steam cracking of ethane, as well as the cracking of propane and butane with reactor coking considerations were investigated in detail. Similar to the aforementioned study, orthogonal collocation on finite elements was implemented to discretize the stiff DAEs and the resulting model was solved to local optimality using a multi-start approach to generate high-quality solutions. In addition, the reactor length was modeled using binary variables and the optimal length that maximizes the ethylene yield was also investigated by Onel [17].

A third alternative for dynamic optimization of the system of DAEs can be through the utilization of data-driven approaches and novel machine learning algorithms. The idea of representing highly complex engineering processes with simple tractable models using data (i.e., surrogate models) has gained accelerated attention in the last decade [36]. Although surrogate models were

primarily used as means of replacing detailed unit operations in flowsheet synthesis [50, 143–146], their application has also been expanding in different areas of process systems engineering including but not limited to dimensionality reduction in control [147], grey/black-box optimization [30, 31, 148, 149], bi-level programming [94, 150] and predictive modeling of environmental systems [151, 152]. In this work, a global constrained grey-box optimization algorithm, ARGONAUT [28–30], is utilized for the data-driven modeling and optimization of system of DAEs without the full discretization of the governing equations. Furthermore, a novel Support Vector Machine-based constraint handling scheme is introduced for handling the stiffness of multi-dimensional DAE systems, which further enables high-quality solution generation by rapidly eliminating the infeasible variable combinations, thus allowing the exploration of a wider range of decision variable space.

4.2 Challenges in Design of Experiments with Stiff Ordinary Differential Equations

Data-driven modeling and derivative-free optimization rely on different sampling strategies that provide an initial plan for the controlled experiments on problem simulators, which is commonly known as the Design of Experiments (DoE). The goal of DoE is to provide possible candidate locations for the input variables within the pre-defined box-constraints such that these experiments capture a variety of system dynamics. There are many different ways of constructing this initial set of candidate points including Latin Hypercube (LHD) and full factorial designs. The details on different types of DoE and the current developments in DoE research are discussed in a recent review article by Garud et al. [153], as well as in a notable textbook by Cavazzuti [154].

It is important to note that the DoE is a statistical procedure and not guided by the physical information that entails an engineering process. Thus, a subset of candidate initial points generated by the DoE may result in unphysical and/or undesirable outcomes, such as an early termination of the problem simulator due to failures or solving a numerically unstable problem (stiff DAEs/ODEs). This generally implies that a constraint should exist between the decision (input) variables, in which the explicit analytical formulation of this, as a function of the input variables, is unknown to the user. As a result, the global optimization of such a system using a data-driven methodology will be hindered since the returned optimal solution may not be a feasible one. Con-

sider the following initial value problem as a motivating example,

$$\dot{y} = y^2, \qquad y(0) = y_o > 0 \tag{4.1}$$

The analytical solution of this separable ODE is given in the following form,

$$y(t) = \frac{y_o}{1 - y_o \cdot t}, \qquad t < \frac{1}{y_o} \tag{4.2}$$

It is important to note that, the validity and the stability of solution in Equation 4.2 strictly depend on the condition between the time and the initial condition value. Specifically, at $t = 1/y_o$, the denominator will become zero and the solution will be undefined. If we wanted to explore the full space defined by t and y_o using DoE, the samples that violate this constraint are going to be removed *a priori* to sample collection, which will also prevent us from sampling in regions that won't yield a feasible or a numerically stable solution for the problem of interest. This is demonstrated in Figure 4.1.

Although in this motivating example it is rather easy to derive the constraint for a valid integration solution concerning the time horizon and the initial condition, in many complex engineering problems (i.e., reaction engineering), the analytical solution may not be trivial or may not even exist. Furthermore, in multi-dimensional problems where multiple variables are initialized for solving a system of ODEs, it is more challenging to postulate appropriate explicit constraints for the underlying relationships between the initial conditions. Hence, optimizing a black-box simulator with a system of multi-dimensional DAEs/ODEs that exert stiffness or contain implicit constraints (i.e., constraints that do not have an explicit mathematical or an analytical form) using a data-driven methodology is a challenging task.

Several approaches can be explored such as sampling in smaller regions or removing infeasible samples *a posteriori* to the simulator call. A smaller sampling region can be imposed such that all sampling points are feasible. For example, if we were to set the upper bound to be 0.5 for both t and y_o in the motivating example, all points collected within this new box will be feasible (Figure

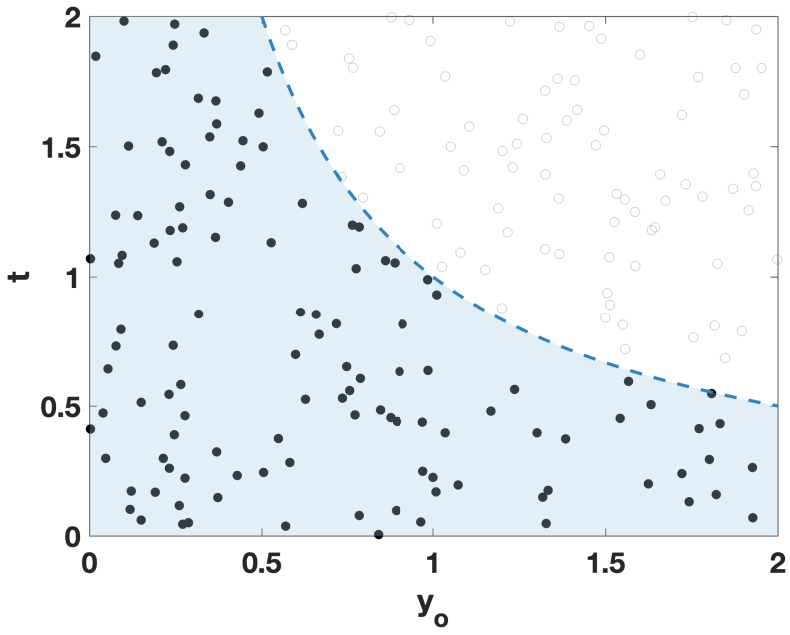

Figure 4.1: Design of experiments for the motivating example. Shaded area represents the feasible region defined by the constraint in Equation 4.2, $t < 1/y_o$. The sampling points that satisfy this constraint are represented with filled circles. Candidate points that violate this constraint are removed before calling the problem simulator. Removed samples are represented with hollow circles in the infeasible region.

4.1). However, in an optimization context, tighter bounds will yield a conservative decision variable space, where the global solution may lie outside these newly imposed bounds. Furthermore, *a posteriori* analysis on the input-output data can be computationally demanding since this will require all candidate points to be evaluated through the problem simulator. This is undesirable in many high-fidelity problems given that as the number of ODEs and the problem complexity increases, the numerical integration will become more time-consuming. Especially, evaluation of the infeasible candidate points that create numerical instability can take more than a couple of hundred seconds. For example, assuming a failed simulation takes 200 seconds per sample to evaluate, collecting the output of 150 numerically unstable candidate points will take more than 8 hours, where these points will not be viable for identifying the optimal solution.

Hence, this sampling challenge with stiff DAEs requires a systematic approach, where a wider range of system dynamics should be captured in a computationally efficient way. To this end, a su-

pervised machine learning methodology, namely the Support Vector Machines (SVMs), is used to assess the numerical stability of a given combination of initial conditions postulated at the several stages of a data-driven optimization process (i.e., initial sampling, and re-sampling) *a priori* to the simulator call. Previously, the idea of using SVMs to approximate the feasible region of optimization problems was explored in bi-level and mixed-integer programming problems [155, 156]. In this work, SVMs are used to handle the stiffness in a multi-dimensional system of DAEs such that they are amenable for data-driven modeling and optimization without the full discretization of the underlying first-principles model. Specifically, the SVMs are used to derive an implicit function that mimics the stability constraint for the solution of stiff DAEs in multi-dimensional space, rather than approximating the full feasible space of the problem as done in the aforementioned studies. Through this supervised machine learning approach, the nonlinear dependencies between the initial conditions and the independent variable of the differential equation that strictly defines the stability of the numerical integration are captured with high accuracy. The details of the approach are further explained in the following section.

4.3 Modeling Implicit Constraints with Support Vector Machines

In machine learning, SVMs are extensively used for classification and regression-type of analyses, spanning over several different application areas including but not limited to fault detection and diagnosis [157–159], improvement of process operations [160], and predictive modeling of complex substances [161]. In this work, an SVM model is used to mimic the implicit constraint imposed on the solution of the system of DAEs. Specifically, an SVM-based classification model is built in the offline phase by using a dataset of simulated samples with their outcome (feasible/infeasible). The obtained classification model acts as a filter and guides the sampling strategy of a data-driven optimizer such that the numerically unstable combinations of independent variables are eliminated *a priori* to sample collection.

If we now consider the initial value problem in Equation 4.1 from a data-driven perspective while assuming no knowledge on the stability constraint, we can numerically integrate Equation 4.1 for every combination of t and y_o values provided by the DoE. At the end of each simulation,

we can check whether the integration has failed or not and assign a label, “0” for “feasible” or a valid integration solution and “1" for “infeasible" or a failed integration solution. The resulting continuous input and the discrete output information can be used to formulate a nonlinear two-class classification problem using SVM, where this model will provide a decision boundary between feasible and infeasible combinations of t and y_o.

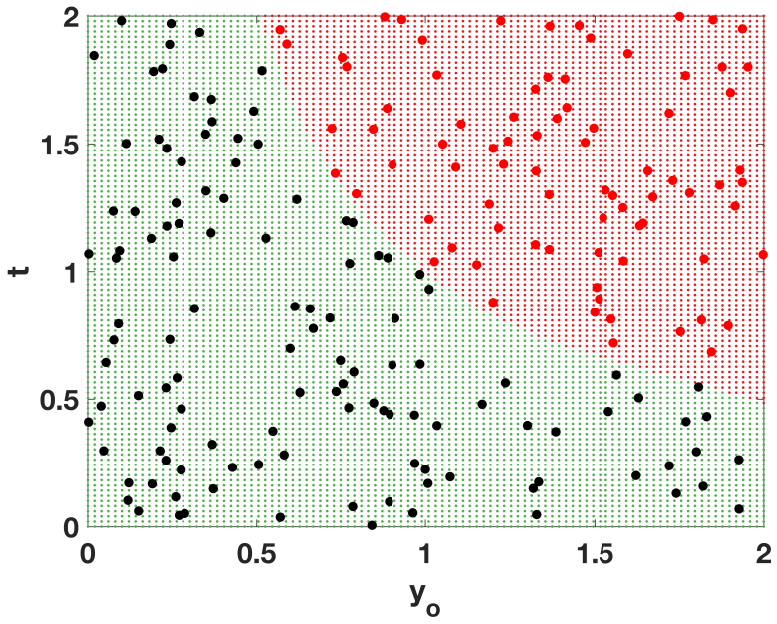

Figure 4.2: Nonlinear SVM model is trained to mimic the constraint, $1/y_o$, by only using the input-output data from the numerical integration of the initial value problem given in Equation 4.1. SVM classifier can model the boundary of the stability constraint with high accuracy, where the green area corresponds to the feasible, and the red area corresponds to the infeasible class, respectively.

Essentially, as shown in Figure 4.2, if the SVM classifier is properly trained, tested and validated, this separating nonlinear boundary will be the same as the constraint imposed on the input variables shown in Figure 4.1. As new samples are desired to be collected in this decision space, the SVM model can now be used to classify and filter the incoming combinations of input variables based on their probabilistic feasibility information provided by this model. This filtering step is essentially a function call that has minimal computational expense to execute and will allow us to

remove the infeasible combinations *a priori* to the problem simulator call, improving the stability and the computational speed of the data-driven optimization process with stiff DAEs. In the next section, the generalized framework for using and implementing SVM classifiers in data-driven optimization of stiff multi-dimensional DAE systems are described in detail.

4.4 Data-Driven Optimization Framework for Stiff Multi-Dimensional DAE Systems

The outline of the generalized framework for handling implicit constraints in data-driven optimization is provided in Figure 4.3. In phase 1, which is the offline phase of the framework, sampling is performed within the lower and upper bounds of the decision variables of a given optimization problem with a stiff DAE system. For each sampling point, a respective output class information (feasible/infeasible) is collected as described in the previous section. Using this continuous input and categorical output dataset, a nonlinear two-class classification problem is formulated and an SVM model is tuned for an accurate representation of the stability constraint of a multi-dimensional stiff DAE system. In phase 2, this trained, tested and validated SVM model is implemented to a grey-box optimization solver. In this phase, the SVM model filters the numerically unstable combination of input variables online as the grey-box optimization algorithm is executed. Each phase is further described in detail in the following sections.

4.4.1 Offline Phase: Data Collection and Tuning the SVM Model

- *Step 1: Sampling*

  For all the computational experiments performed in this study, maximin LHD is constructed for 2000 sampling points within the pre-defined bounds of each variable. The respective class information of each sample (i.e., feasible or infeasible) is collected from the problem simulator that performs the numerical integration. This offline sampling stage is done once and is solely used for the C-SVM model building stage.

- *Step 2: Data normalization and allocation*

  The input data is normalized by min-max scaling within the provided variable bounds and the collected data is split into train-test and validation sets. For the validation set, 10% of the

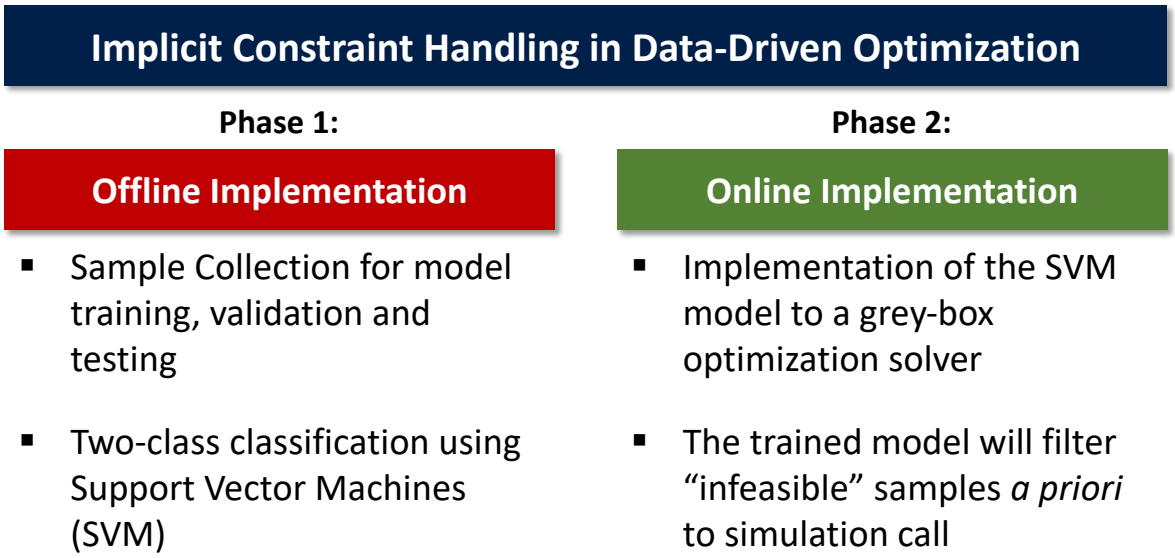

Figure 4.3: Outline of the SVM-based constraint handling framework for data-driven optimization with stiff DAEs.

data from each class is separated. This validation set is not used in any of the training and testing steps, hence allowing us to assess the unbiased performance of the trained C-SVM model. The remaining 90% of the data is used for model development.

- *Step 3: Model tuning and development*

  Here, 5-fold cross-validation is used to avoid the overfitting problem. This is a crucial step in achieving an accurate and generalizable C-SVM model simultaneously. Two important hyperparameters that require tuning in a C-SVM formulation are the γ and C parameters. In this study, the optimal γ and C parameters are obtained via grid search. In particular, the C parameter is tuned over the set of 2^{-10}, 2^{-9}, ..., 2^{10}, while the γ parameter is tuned over the $\frac{2^{-10}}{n}$, $\frac{2^{-9}}{n}$, ..., $\frac{2^{10}}{n}$, where n is the number of features of the dataset used in training. Normalization of the γ parameter based on the dataset density is performed to achieve an optimal separation of feasible and infeasible data points without overfitting. Finally, the model is developed by using the entire 90% of the dataset via the optimal C-SVM hyperparameters obtained during the tuning stage. The developed C-SVM model assigns probability to each

input sampling point. If the probability is higher than 0.5, the sampling point is classified as "feasible", otherwise "infeasible".

- *Step 4: Model performance assessment metrics*

  In this study, the classification model performance is quantified by calculating 5 different performance metrics on the validation dataset: (1) Accuracy; (2) Precision; (3) Recall; (4) Area under the Receiver Operating Characteristics (ROC) curve (AUC); (5) F_1 score. Accuracy is described as $\frac{TP+TN}{TP+TN+FP+FN}$, while precision and recall are defined as $\frac{TP}{TP+FP}$ and $\frac{TP}{TP+FN}$, respectively. Here, TP (TN) indicates the number of feasible (infeasible) sampling points that are correctly classified by the model. On the contrary, FP (FN) yields the number of infeasible (feasible) sampling points that are misclassified as feasible (infeasible) by the model. Note that F_1 score is the harmonic mean of precision and recall metrics.

The model developed with the described offline phase model building procedure has produced a perfect classifier (Validation scores: Accuracy = 100%, Precision = 100%, Recall = 100%, AUC = 100%, F_1 score = 100%) for the dataset provided in the motivating example (Figure 4.2). It is important to state that the normalization step is not performed for the motivating example as both t and y_o have the same upper and lower bounds.

Once the offline phase is completed, the validated SVM model is incorporated into a grey-box optimization solver. In this work, the ARGONAUT algorithm is utilized to demonstrate the effectiveness of this data-driven approach, outline the key steps of the framework and its integration with the SVM classifier in the following section.

4.4.2 Online Phase: Integration of the SVM Classifier with the ARGONAUT Framework

The ARGONAUT algorithm [28–30] is a constrained grey-box optimization solver that utilizes the input-output data to postulate appropriate surrogate formulations for the objective function and the unknown constraints, through solving the parameter estimation problem to global optimality. Initially, this framework has been developed to solve general constrained nonlinear grey/black-box optimization problems and was tested on a pressure swing adsorption example for CO_2 capture

[29] and numerous benchmark global optimization problems [28]. Later, several key stages of the framework are parallelized for utilizing distributed high-performance computing for improved computational efficiency [30]. The details of the parallelization are further described in Chapter 5.

The ARGONAUT algorithm starts with the DoE and sampling stages. In the presence of known constraints for the input variables, the algorithm will first run Optimality Based Bound Tightening (OBBT) to reduce the search space. OBBT cycles through each variable present in the known constraint by minimizing and maximizing their values, while being subject to this known constraint. This will allow the algorithm to update the current bounds on the variables and then generate a maximin LHD within the updated search space. When known constraints are present, the LHD is created with a large set of samples based on the input dimensionality (N_{dim}), where for $N_{dim} \leq 10$ the initial design will have $N_{sample} = 100 \cdot N_{dim}$ samples, whereas for $N_{dim} > 10$ the initial design will have $N_{sample} = 2000$. Among this large set of initial design points, the ones that do not satisfy the known constraint are removed from the initial design through an explicit function evaluation. The default version of ARGONAUT then continues with reducing the remaining set of feasible samples using the Optimal Scenario Reduction algorithm (OSCAR) [162] or by augmenting the LHD depending on the cardinality of the sampling set.

To handle stiff problems or problems with implicit constraints, an additional checkpoint is introduced before the OSCAR scenario reduction step using the developed C-SVM model in the offline phase (Figure 4.4). This C-SVM model filters the pre-determined values of input variables that potentially lead to numerical instability by classifying them as infeasible and removing these from the initial sampling set. Later, if the cardinality of the remaining numerically stable samples is higher than the intended size of the initial DoE, the algorithm proceeds with the scenario reduction step which leaves us with an appropriate set of sampling points to be executed in the problem simulator.

Once this feasible set of input variables are simulated and their corresponding outputs (i.e., objective function value and black-box constraint violations) are collected, this input-output data is passed onto the parameter estimation stage. Here, a distinction is made between different sources

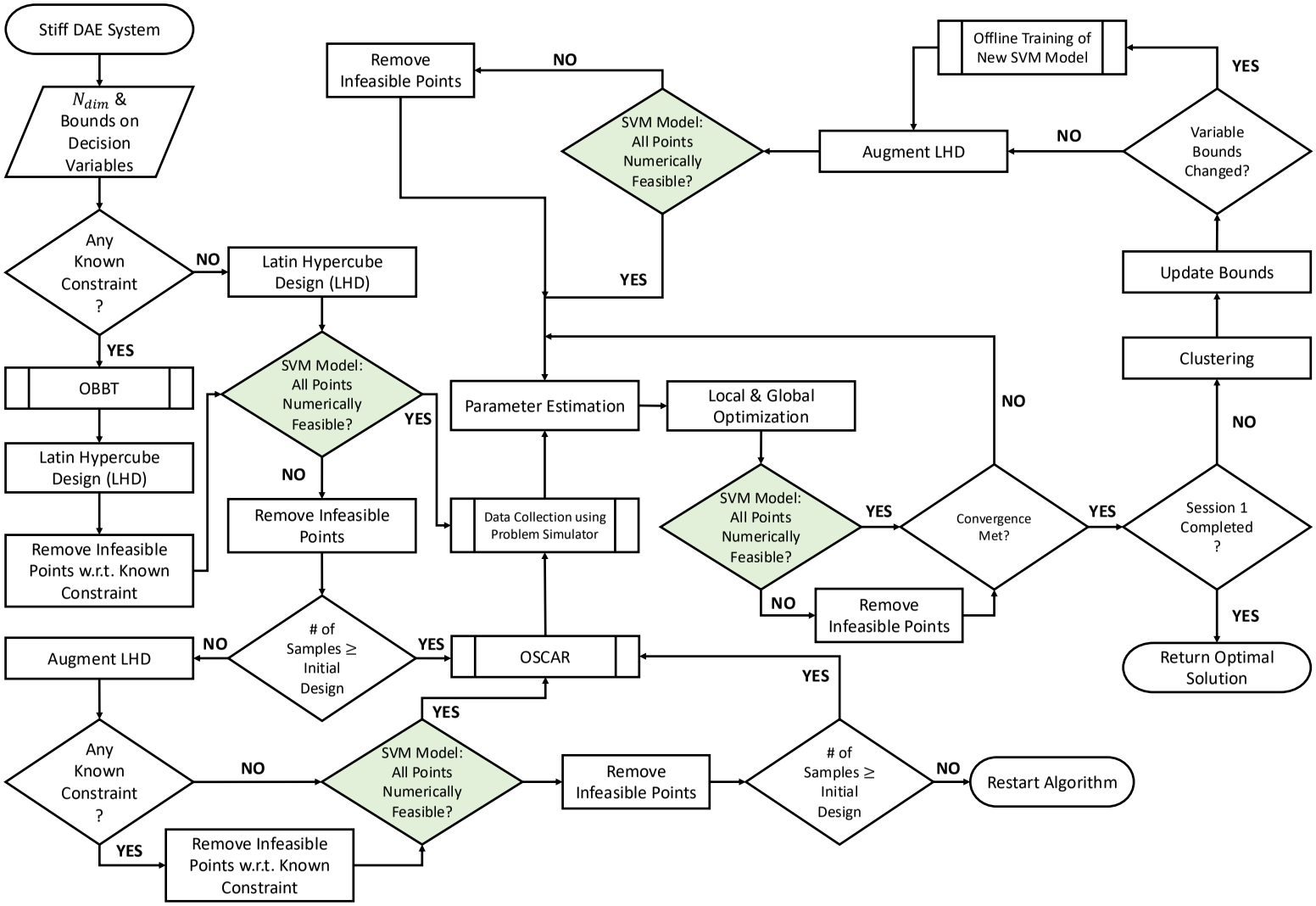

Figure 4.4: Integration of the SVM-based constraint handling with the ARGONAUT framework. The SVM model is incorporated to several sampling stages, including the initial design, adaptive sampling through local and global optimization of the grey-box surrogate model and LHD augmentation in updated bounds.

of infeasibility: (1) Infeasibility due to a violation of a known constraint which is available in closed-form; (2) Infeasibility due to a violation of the black-box output, where this information can only be extracted by running the full problem simulator; (3) Infeasibility due to a violation of implicit constraints or constraints that characterize the stability of the integration. Although samples that are feasible with respect to the known constraints and the integration stability are simulated, any of these combinations may still lead to an infeasible operation based on the process constraints, which requires the execution of the full problem simulation. Hence, ARGONAUT will keep track of this second type of infeasibility and construct individual surrogate models for these constraints at the parameter estimation stage. ARGONAUT contains multiple surrogate forms in its surrogate model library (i.e., linear, general quadratic, signomial, radial basis functions, kriging interpolation), where the algorithm can decide on the best surrogate form for a given input-output data through cross-validation. The algorithm is flexible in such a way that it can choose different surrogate forms for each unknown function. For example, for a problem with 3 unknown equations, ARGONAUT can construct a quadratic surrogate objective with 1 nonlinear (radial basis function) constraint and 1 linear constraint. As an alternative, the preferred surrogate form can also be specified for any unknown function at the start of the algorithm, where only this specific type of surrogate form will be explored. This exploration for both known and unknown forms are done through solving the parameter estimation problem to global optimality, which is one of the key properties of the algorithm to ensure accurate representations of the input-output information.

Once individual surrogate models are constructed for all unknown equations, a grey-box optimization problem is formulated using the surrogate functional forms for the objective and the unknown constraints. The known constraints are also included in this formulation to ensure the feasibility of the optimal solution. This formulation is then solved to global optimality and local optimality with a multi-start approach. The resulting high-quality solutions are then assigned as new sampling points on the next iteration to explore promising regions in the feasible space. Again, the C-SVM model will check the numerical stability of these candidate sampling points *a priori* to the simulator call and remove samples that are classified as infeasible. This procedure

will continue until one of the convergence criteria (Chapter 2, Table 2.1) are met.

Once a convergence criterion is met, a session of an ARGONAUT run will be completed, and the algorithm will perform clustering. Clustering will allow the algorithm to identify a promising sub-region of the sampling space based on the cluster with the best incumbent solution. Then, the bounds on the decision variables can be tightened around this cluster and the algorithm will proceed with the second session, where the number of sampling points in this reduced space is augmented. In the augmentation stage, as new combinations of decision variables are postulated, the C-SVM model needs to be called for a feasibility evaluation of these new combinations of candidate points. However, it is important to note that as the bounds on the decision variables are tightened at the end of the first session, the C-SVM model needs to be reconstructed again using the input-output relationship from the reduced decision variable space. Hence, the procedure described in the offline phase is repeated to generate the new C-SVM model within the new tightened bounds at the end of the first session. If the algorithm does not tighten the bounds, then the C-SVM model from the first session is still valid and the same model can be used to filter the numerically unstable combinations of variables prior to simulator call. After the new model is trained, tested, validated and incorporated in the framework, the algorithm restarts the iterative steps for sampling, parameter estimation and optimization of the grey-box formulation as described earlier. By default, the algorithm will reach full convergence after one of the aforementioned criteria is met and the second session is completed. The total number of sessions in the algorithm can be increased, however, as the C-SVM model requires reconstruction with changing variable bounds, the algorithm is used in the default mode, where the C-SVM models are only constructed twice.

As shown in Figure 4.4, this data-driven methodology to mimic the stability constraint through the use of SVM models is incorporated at every stage of ARGONAUT, where procedures regarding sampling are taken care of, including when the initial design is created, the existing design is augmented and when new samples are adaptively collected from the optimization step at every iteration. Although here the SVM-ARGONAUT integration is extensively discussed, the idea of using SVMs in this framework is generic and can be implemented to other data-driven solvers as

well. In the next section, a more complicated computational case study is described, namely the steam cracking of ethane and propane, where the data-driven modeling and optimization of these stiff DAE systems are explored using the SVM approach.

4.5 Data-Driven Dynamic Steam Cracking Optimization for Ethylene and Propylene Production

The rapid increase in shale gas production in the Appalachian and Permian Basins for the last decade has lead to significant growth in natural gas liquids (NGLs) production, as well as a projected increase for these petrochemical feedstocks in the upcoming years (Figure 4.5) [1–3, 163].

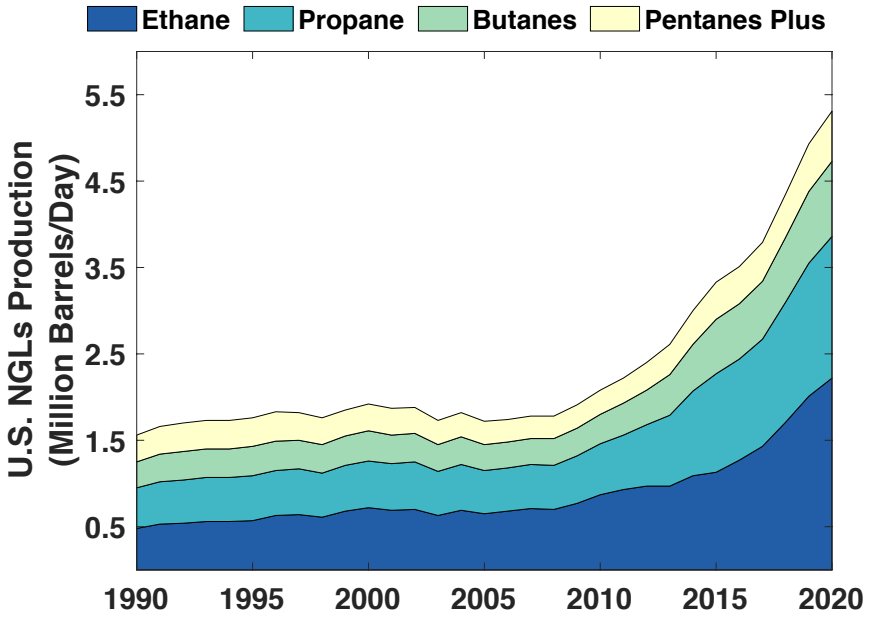

Figure 4.5: Historic natural gas liquids production in the U.S. and its short-term projection for the upcoming year [1–3].

Ethane and propane, being the major constituents of NGLs, are predominantly used for the production of ethylene and propylene, respectively, where ethylene consumption is expected to increase by 49% from the year 2017 to 2020 [2]. Naturally, the growing petrochemical feedstock

supply and the rising demand for light olefins sparks an interest in converting NGLs to olefins via the non-catalytic steam cracking process. In this perspective, many existing ethylene crackers have expanded capacity and new crackers are becoming online to benefit from this unique opportunity [164]. Hence, the mathematical optimization of this process emerges as a necessity to determine the optimal operating conditions for the steam cracker, in such a way that the profit from ethylene and propylene production is maximized.

To this end, the integrated SVM-ARGONAUT framework is utilized to handle the stiffness in the cracking model equations while exploring high-quality solutions for the optimal reactor length, inlet ethane/propane and steam flowrates, inlet temperature, inlet pressure, and heat flux profile along the optimal reactor length, through surrogate modeling and optimization. The steam cracker reactor model for ethylene and propylene production is adapted from [14–17, 165–168] and modeled as a one-dimensional plug flow reactor with coking effects (Figure 4.6). The detailed reactor model equations and parameters [18–20] are presented in Appendix C.

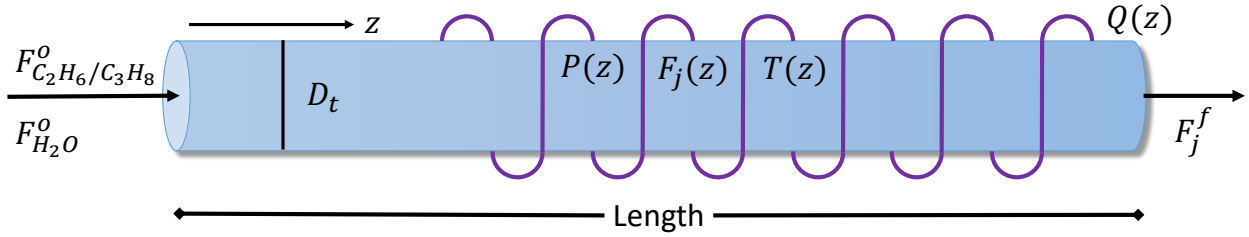

Figure 4.6: One-dimensional plug flow reactor for steam cracking (Tube diameter: D_t). $P(z)$, $F_j(z)$ and $T(z)$ represent the spatial change of pressure, species molar flowrate and temperature along the reactor length, respectively. Steam and feed (i.e., ethane or propane) is co-fed at the reactor inlet. Heat required for the endothermic cracking reactions is provided by the external heat flux, $Q(z)$.

The mathematical formulations presented in Appendix C for the steam cracking case studies are modeled in MATLAB and this problem simulator is used for the data-driven modeling and

optimization of the thermal cracking process with SVM-ARGONAUT. For both case studies, 10 input variables (the decision variables, Table C.7) are considered, 1 known constraint (Equation C.23 for ethane cracking model, Equation C.24 for propane cracking model), and 4 grey-box constraints (Equations C.19, C.20, 4.3, 4.4), where the objective is to maximize the profitability of operation (Equation C.21 for ethane, C.22 for propane). The detailed analysis of the results is provided in the following section.

4.6 Results of Computational Studies

The computational studies for the data-driven steam cracking optimization are performed on a High-Performance Computing (HPC) machine at Texas A&M High-Performance Research Computing facility (Ada HPC Cluster operated with Linux CentOS 6: Intel Xeon E5-2670 v2 10-core processor (Ivy Bridge-EP)). The supercomputer is used at both stages of the framework: (1) In the offline phase, for data collection and SVM model building; (2) In the online phase, for executing the ARGONAUT algorithm as a parallel job, using 1 node (20 cores per node with 64 GB RAM) on the supercomputer. Likewise, the data collection and SVM model-building phases are performed as a parallel job, using 1 node (20 cores per node with 64 GB RAM and 1 node (4 cores per node with 64 GB RAM), respectively. The results of the offline and the online phases of the framework are discussed in the following sections.

4.6.1 Offline Phase: Results of SVM Model Building

As a first step, the SVM model is built using the data generated from the steam cracking model which is subject to the known constraint (Equation C.23 for ethane cracking model, Equation C.24 for propane cracking model) and the provided bounds on the decision variables (Table C.7), following the methodology described in previous sections. As the bounds on the decision variables are the same in the first session of ARGONAUT runs, only 1 SVM model is built per case study. The model evaluation metrics on the validation data for both ethane and propane cracking case studies are provided in Table 4.1.

The validation results show that SVM models for these datasets can be generated in high ac-

Table 4.1: SVM model performance for the first session of runs with ARGONAUT.

Cracking Model	Accuracy (%)	Precision (%)	Recall (%)	AUC (%)	F_1 score (%)
Ethane	98.492	96.591	100	99.948	98.266
Propane	98.492	98.780	97.590	99.875	98.182

curacy (98.492%). It is important to note that the SVM model cannot guarantee feasibility to the selected sampling points as the model may misclassify 1.508% of the samples as "feasible" based on the validation dataset. When this model is implemented in the online phase, it may cause ARGONAUT to converge to a numerically unstable solution at the end of its iterations. To absolutely guarantee the feasibility of the solution, an extra grey-box constraint is added in the online phase that essentially confirms the assigned value for the length variable as a candidate point is equal to the simulation endpoint (i.e., simulation does not quit prematurely without reaching the endpoint of length). As ARGONAUT cannot directly handle equality constraints, this constraint is reformulated into two inequalities (Equations 4.3 and 4.4) with an added relaxation parameter.

$$L^{in} - L^{out} \leq 0.000001 \tag{4.3}$$

$$L^{in} - L^{out} \geq 0.000001 \tag{4.4}$$

Once the SVM model for the first session is established, the online phase of the SVM-ARGONAUT integrated approach is executed 20 times for each cracking model, each starting with a random LHD. After the convergence is reached, the first session is completed and the variable bounds are tightened, new SVM models are generated for each run such that they represent the numerical stability of the cracking models in the reduced space. The detailed validation performances of these SVM models under the tightened bounds are summarized in Appendix C (Tables C.8 and C.9). The overall results show that these models also have very high accuracy, with more than 97% and 95% correct classification performance among all tested samples for the ethane and propane cracking case studies, respectively. The other performance metrics are also satisfactory,

where the model precision is greater than or equal to 98% and 94%, the recall is greater than 98% and 95%, the AUC is greater than 99% and 99%, and the F_1 score is greater than 98% and 96% for ethane and propane case studies, respectively.

4.6.2 Online Phase: Results of the Grey-Box Optimization

In the online phase, the goal is to find the optimal solution to the steam cracking problem using the integrated approach for implicit modeling of the stability constraint and the constrained grey-box optimization of the problem of interest. For all case studies and their respective 20 repetitive runs, the number of initial sampling points is set to be $N_{sample} = 30 \cdot N_{dim} + 1$. The same rule-of-thumb is also used when performing sampling reduction via OSCAR and in the second session of the algorithm, when the LHD is augmented for exploring the most promising region for the optimal solution. For processes regarding the surrogate modeling and grey-box optimization, ARGONAUT is executed in the default mode, where the algorithm decides on the surrogate model form for the objective function and the grey-box constraints.

The thermal cracking models provided in Appendix C are used as grey-box problem simulators, where different combinations of decision variables are input to each simulator and the corresponding objective function value and the constraint violations are collected. The input combinations to the problem simulators are first evaluated to satisfy the known constraint and then evaluated by the SVM-feasibility checkpoint to ensure that this combination will yield a numerically stable solution. If the sampling point passes these two feasibility checks, then that sample is evaluated in the process models and its corresponding outputs are collected and further processed in the parameter estimation and data-driven optimization stages of the algorithm. Following this procedure, the best solution out of the 20 runs for the ethane cracking case study is summarized in Figure 4.7 and Table 4.2.

Figure 4.7A shows that the molar flowrate of the main products is increasing along the reactor length as the desirable reaction is taking place. Clearly, the ethane cracking shows a single-feed-single-product trend where ethylene is produced through the favorable reaction alongside with H_2. In addition, the molar flowrate of the byproducts are significantly limited compared to the desired

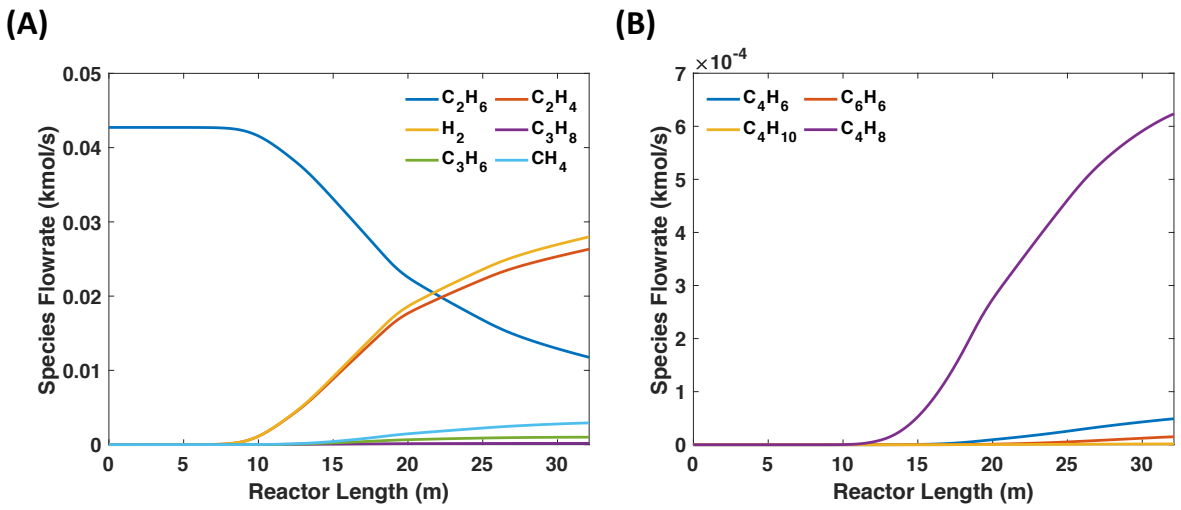

Figure 4.7: (A) The molar flowrate of species for the optimal configuration of an ethane thermal cracker. (B) The molar flowrate of C_4^+ species which lead to reactor coking in the optimal configuration.

products. Figure 4.7B shows the molar flowrate of the C_4^+ species which were previously identified as the coke precursors. It is observed that among these 4 species, 1-butene has the highest flowrate along the reactor length, hence contributing the most to the reactor coking. This observation is also consistent with the findings of Onel [17], demonstrating the validity of the presented data-driven approach for finding high-quality feasible solutions for the optimization of stiff DAEs.

Table 4.2: The results of the best solution found with SVM-ARGONAUT integration for the ethane cracking case study.

Decision Variables	Optimal Value	Decision Variables	Optimal Value	Results	Value
Q_1^o (kW/m^2)	507.393	$F^o_{C_2H_6}$ (kmol/s)	0.04272	Ethane Conversion	0.7247
Q_2^o (kW/m^2)	741.484	$F^o_{H_2O}$ (kmol/s)	0.00377	Ethylene Yield	0.6161
Q_3^o (kW/m^2)	954.021	T^o (K)	727.027	Ethylene Selectivity	0.8501
Q_4^o (kW/m^2)	462.987	P^o (kPa)	303.741	T^{out} (K)	1170.207
Q_5^o (kW/m^2)	260.125	L (m)	32.117	P^{out} (kPa)	131.289

Furthermore, Table 4.2 summarizes the results pertaining to the optimal decision variables achieved in the ethane cracking reactor using the SVM-ARGONAUT integrated framework. For

the thermal cracking of ethane, the SVM-ARGONAUT framework identifies the maximum profit as \$0.3359/s which corresponds to an overall annual profit of \$10.6M. The optimal decision variables show that a greater heat flux supply is required at the first 3/5 portion of the reactor where it later decreases gradually towards the exit of the reactor. This is consistent with the inlet temperature value as the reactor entrance temperature is low, a greater heat flux needs to be supplied to ensure endothermic cracking reactions take place. In addition, a higher ethane flowrate is established where the steam flowrate is relatively lower. This is an expected result; as ethane cracking being a single-feed-single-product system, the only positive contribution to the profit comes from ethylene production. It is also observed that the optimal value of the inlet temperature for the ethane cracker is lower than expected as higher temperatures will increase the reaction rates. However, as high temperature promotes faster reactions, the reactor coking will be enhanced due to the creation of more side products, hence leading to a loss of profit. As a result, the selected optimal inlet temperature value prevents early reactor coking and promotes a higher profit of operation. This is also supported by a lower steam flowrate for the ethane cracker where a minimal amount of steam will be required at minimal amounts of coking on the reactor wall. Moreover, a shorter reactor length is identified in the optimal configuration compared to a typical longer reactor lengths that are commonly reported in the literature. This is a key result showing that fine-tuning the reactor length will generate valid high-quality solutions with high profit values by decreasing the investment, heating and decoking costs of bigger reactors. In addition, it is observed that the reported optimal solution in Table 4.2 favors good ethane conversion and ethylene yield with high selectivity for ethylene.

Similarly, the optimal results of the thermal cracking of propane found using SVM-ARGONAUT integration are reported in Figure 4.8 and in Table 4.3. The results show that the profit obtained from thermal cracking of propane is \$0.0845/s which corresponds to an annual profit of \$2.67M. The reactor molar flowrate profiles in Figure 4.8A show that the favorable reaction starts taking place early at the reactor entrance, where propane flowrate depletes and products are produced along the reactor length. Figure 4.8A also shows that the thermal cracking of propane

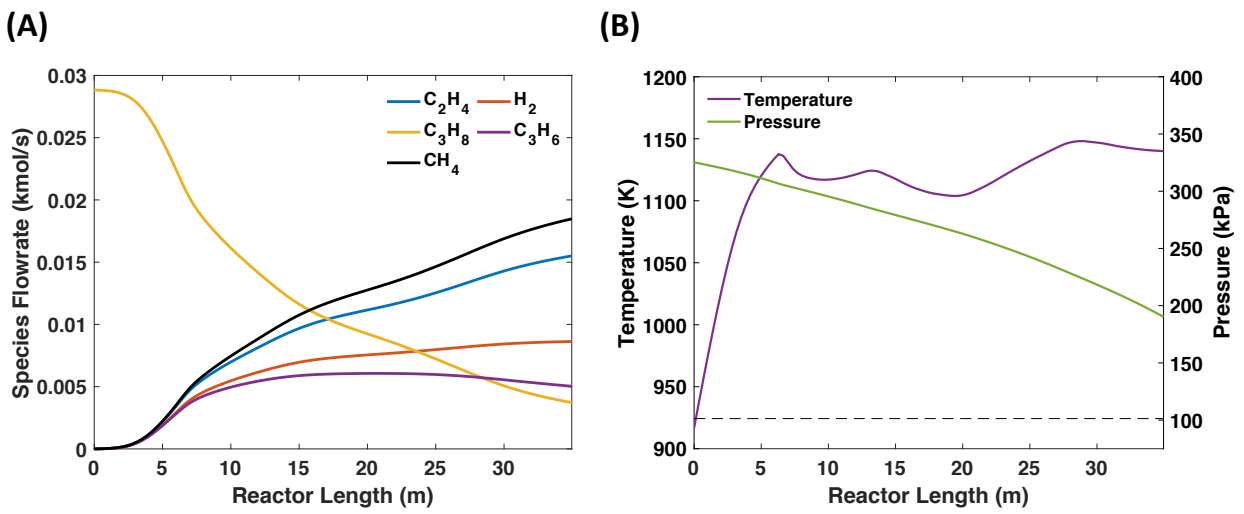

Figure 4.8: (A) The molar flowrate of the main products for the optimal configuration of a propane thermal cracker. (B) Reactor temperature and pressure profiles at the optimal configuration for the propane cracker. The dashed line represents the atmospheric pressure.

Table 4.3: The results of the best solution found with SVM-ARGONAUT integration for the propane cracking case study.

Decision Variables	Optimal Value	Decision Variables	Optimal Value	Results	Value
Q_1^o (kW/m^2)	910.298	$F^o_{C_3H_8}$ (kmol/s)	0.02883	Propane Conversion	0.8708
Q_2^o (kW/m^2)	282.332	$F^o_{H_2O}$ (kmol/s)	0.01004	Propylene Yield	0.1745
Q_3^o (kW/m^2)	55.025	T^o (K)	916.716	Ethylene Yield	0.5377
Q_4^o (kW/m^2)	177.221	P^o (kPa)	325.276	Propylene Selectivity	0.2004
Q_5^o (kW/m^2)	21.516	L (m)	34.957	Ethylene Selectivity	0.6175

enables the production of two main products, namely ethylene and propylene. As reported in Table 4.3, this optimal reactor configuration leads to a high propane conversion value with a larger yield and selectivity favored for ethylene. Hence, the optimal configuration of the propane cracker at maximum profit pushes for a greater ethylene production than propylene. Furthermore, Figure 4.8B shows the optimal temperature and pressure profiles for the propane cracker. Although higher inlet temperature and pressure are required for this case study where a slightly longer reactor length is also preferred for maximizing the profit, the solution is feasible with regards to the limits provided for outlet temperature and pressure.

Table 4.4: The profit breakdown for the optimal solution of ethane and propane cracking case studies.

Ethane Cracking		**Propane Cracking**	
Objective Variable	Value ($/s)	Objective Variable	Value ($/s)
Ethane Feed Cost	- 0.2991	Propane Feed Cost	- 0.6361
Steam Feed Cost	- 0.0006	Steam Feed Cost	- 0.0022
Heating Cost	- 0.0621	Heating Cost	- 0.0381
Investment Cost	- 0.0088	Investment Cost	- 0.0095
Decoking Cost	- 0.0855	Decoking Cost	- 0.0345
Ethylene Production	+ 0.7920	Propylene Production	+ 0.2581
-		Ethylene Production	+ 0.5468
Total	**+ 0.3359**	**Total**	**+ 0.0845**

Moreover, the overall profit breakdown in Table 4.4 shows that for both case studies the petrochemical feedstock costs, reactor heating and decoking costs take the most out of the profit, whereas the reactor investment cost is relatively small due to the optimized reactor length. In the thermal cracking of propane, the results show that ethylene production contributes more to the profit than the propylene, as the coking mechanism for this case study utilizes C_3H_6 as the coking precursor. Hence, this limits the propylene production while favoring ethylene production for profit and minimum coking generation on the reactor wall. It is important to note that the reaction mechanism for propane cracking allows flexibility in the mode of operation depending on the market demand or prices (i.e., maximizing ethylene or maximizing propylene production). However, it is important to note that exhaustive exploration of the full Pareto solution between maximizing ethylene versus maximizing propylene is possible, but out of the scope of this work. Nonetheless, the case studies presented in this work show that the SVM-based data-driven optimization algorithm is effective for optimizing process models with stiff DAEs and can generate high-quality feasible solutions.

Finally, the total elapsed computational time of the online phase with and without the SVM approach is compared. Figure 4.9 shows that the integrated SVM-based data-driven optimization algorithm is more computationally efficient than in the absence of this approach for both ethane and

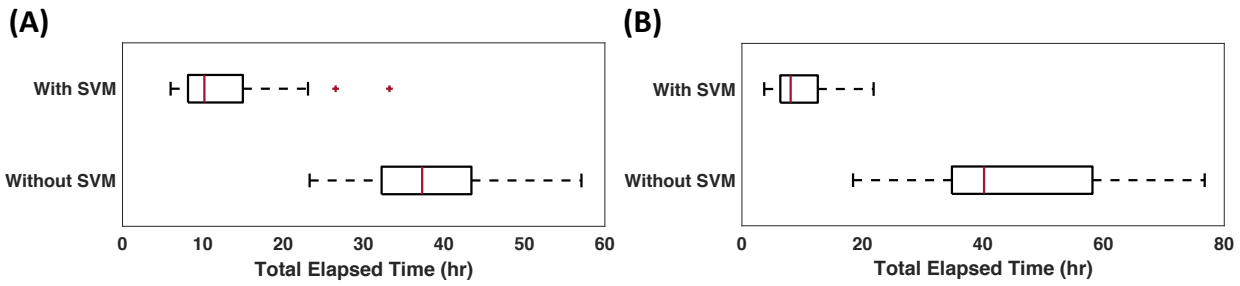

Figure 4.9: Boxplots for the total elapsed time in the online phase for the data-driven optimization of: (A) Ethane; and, (B) propane cracking case studies in the presence and absence of the SVM approach.

propane cracking case studies in the online phase. Furthermore, lower profit values are observed for both cracking case studies when the SVM approach is not implemented. For ethane cracking, the best-found profit over 20 random runs without the SVM approach is \$0.2970/s, whereas for propane cracking the best profit is \$0.0834/s. It is observed that the integrated approach can locate better solutions where the profit is improved by 13.1% for ethane cracking and 1.3% for the propane cracking problem. The overall results show that the SVM-based optimization algorithm can find superior feasible solutions to stiff multi-dimensional DAEs in a computationally efficient way.

4.7 Concluding Remarks

In this chapter, a data-driven optimization algorithm is presented using Support Vector Machines (SVMs) for systems with stiff Differential Algebraic Equations (DAEs). The numerical stability of a system of stiff DAEs is formulated as a nonlinear two-class classification problem, where the feasibility boundary of stiffness is implicitly modeled using an SVM model. Later by incorporating SVM models to a global constrained grey-box optimization solver, namely the ARGONAUT framework, any numerically unstable sampling points are filtered and removed *a priori* to simulator call and the optimal solution of the complex process model is explored using a data-driven approach. The fundamental idea behind this integrated approach is demonstrated on a 2-dimensional motivating example where the SVM approximation of the stability constraint is

shown to achieve high validation accuracy. Further, this approach is extended and tested on more challenging case studies, namely the thermal cracking of natural gas liquids. The results from thermal cracking case studies show that an SVM-based approach enables feasible, numerically stable, and high-quality solutions for the data-driven optimization of systems with stiff DAEs without the full discretization of the underlying first-principles process model.

5. DATA-DRIVEN NONLINEAR NONCONVEX OPTIMIZATION WITH APPLICATIONS TO HIGHLY CONSTRAINED OIL FIELD OPERATIONS*

This chapter presents algorithmic advances within the AlgoRithms for Global Optimization of coNstrAined grey-box compUTational problems (ARGONAUT) framework, developed for the global optimization of systems which lack analytical forms and/or derivative information. By taking advantage of high-performance computing, a new parallel version of ARGONAUT (p-ARGONAUT) is introduced to solve problems with high dimensionality and a large number of constraints. This framework is motivated by a complex case study, which pushes the boundaries of complexity of derivative-free optimization in terms of both dimensionality and number of constraints, namely the identification of the optimal operational control trajectories of an oilfield using water-flooding. The objective of this case study is the maximization of the Net Present Value of the operation over a five-year period by manipulating the pressures of the injection and production wells, while satisfying a set of complicating constraints related to water-cut limitations, platform capacity constraints and operational limits. First, a dimensionality reduction is performed via the parametrization of the pressure well control domain, which allows the efficient optimization of the constrained grey-box system by the proposed algorithm. Results are presented for various cases with increasing number of constraints and the performance of p-ARGONAUT is compared to other derivative-free optimization methods.

This chapter is organized as follows. Section 5.1 introduces water-flooding control optimization and provides an overview of the current state-of-the-art for addressing this challenging mathematical programming problem. Later in Section 5.2, the new parallel algorithm is described which enables a theoretical advancement in water-flooding control optimization by explicitly accounting for all the process constraints, while identifying superior guaranteed feasible solutions for these

*Part of this chapter is reprinted with permission from "Global optimization of grey-box computational systems using surrogate functions and application to highly constrained oil-field operations" by B. Beykal, F. Boukouvala, C.A. Floudas, N. Sorek, H. Zalavadia, E. Gildin, 2018. *Computers & Chemical Engineering*, vol. 114, pp. 99-110, Copyright [2018] by Elsevier and Copyright Clearance Center.

high-dimensional highly constrained grey/black-box problems. Furthermore, Section 5.3 outlines the Functional Control Method (FCM) for reducing the dimensionality of water-flooding control optimization via parametrization of the well control domain. Finally, the new parallel algorithm is tested on a realistic benchmark problem where the results of series of computational studies are provided in Section 2.3, along with concluding remarks in Section 5.6.

5.1 Optimization of Water-flooding Control Operations

Oil companies continuously strive to maximize oil recovery factors, using new technologies for enhanced oil recovery [169]. Primary oil recovery, uses the reservoir's initial pressure to transmit fluids to the production wellbore, however, as the reservoir depletes this initial pressure declines, and leads to the entrapment of significant amounts of oil in reservoirs. Water-flooding is a well-known and historically widely used secondary oil recovery (SOR) method, through which water is injected to the wells to displace and extract oil that is entrapped in the reservoir after primary oil recovery. SOR methods play an important role in the oil economy, since they are used to extract a significant amount of oil annually, using fluids such as water, CO_2 or hydrocarbon gases [170, 171], while water is one of the most inexpensive available options. Despite its popularity, water control during water-flooding poses significant challenges due to the amount of water required for the extraction, as well as costs for water handling [172]. In addition, uncertainty in the geological description of the reservoirs (e.g., unknown permeabilities and porosities) contribute to challenges in the operations, such as the fluid to bypass unswept regions, and thus detailed simulations are necessary to predict the behavior of these complex systems under different operating conditions.

During any oil extraction operation, reservoir management aims to find the optimal values for continuous operating variables, such as the well rates or the bottom hole pressures (BHP), to maximize the net present value (NPV), or the cumulative oil production of the operation over a specified period of time. Specifically in water-flooding, there are several constraints that must be taken into account when maximizing the profitability of an operation, such as the water-cut constraint, which directly affects the cost associated with water handling. Efficient optimization of the aforementioned constrained formulation is a challenging problem. First, the objective is a

nonlinear nonconvex function, which often displays many local optima. Second, the optimization problem is subject to constraints that can only be obtained as an output of a reservoir simulation. Third, the evaluation of the objective function and the constraints is costly since the reservoir simulations require the solution of a system of multi-dimensional partial differential equations. Lastly, the gradient information of the objective function and the constraints of such problems is often not available, due to the black-box, or proprietary nature of the simulators.

Generalized pattern search and global-search algorithms (i.e., genetic algorithms, particle swarm optimization) are commonly utilized in the literature for the optimization of water-flooding operations [173]. Recent studies have focused on box-constrained optimization [170, 174], as well as general constrained optimization, where the nonlinear constraints are treated using filter-based methods [175, 176], penalty functions or barrier methods [177–179] and an augmented Lagrangian approach [180, 181]. There are also studies concentrating on the incorporation of surrogate-based techniques to the water-flooding optimization problem. Queipo et al. [182] have investigated the global optimization of the box-constrained water-flooding problem by constructing a kriging surrogate function to represent the objective function. In addition, Horowitz et al. [183] have extensively studied the water-flooding optimization problem under general constraints using surrogate formulations. The authors build kriging surrogates for the objective function and for the nonlinear constraints, which they locally optimize using Sequential Quadratic Programming (SQP).

The main distinction of the approach followed in this work compared to the studies discussed above, is the use of adaptive sampling, and the optimal training and selection of hybrid surrogate formulations, which are solved to global optimality. The ARGONAUT framework [28, 29] trains and validates optimal approximations, selecting from a pool of potential surrogate functions, ranging from linear regression, to nonlinear interpolating functions for each of the unknown equations (objective and constraints) of the problem. ARGONAUT addresses many elements ranging from optimal sampling, optimal sampling reduction, model identification, bound refinement, variable selection to global optimization, which further amplifies the consistency and the performance of this framework. A detailed description of this algorithm is provided in Chapter 4 of this disserta-

tion. In the following section, the algorithmic parallelization developments for ARGONAUT are discussed, which allows the consideration of a variety of important constraints for water-flooding control that have a significant effect on the profitability of operations.

5.2 Parallelization of the ARGONAUT Algorithm

Even though the ARGONAUT algorithm is previously shown to find the global optimum for a large set of nonlinear optimization problems with up to 100 variables and constraints [28], the computational cost of it becomes a limiting step as the number of dimensions and the number of constraints of the problem formulation increases. There are three stages of the algorithm that contribute significantly to the computational cost of the method. First, the time required to collect samples from the simulation, has a large impact on the computational cost of the overall optimization, and this is directly linked to the computational cost of a single function call, which is often significant. Second, as the number of unknown constraints increases, the surrogate training, selection and validation stage becomes a limiting step since this procedure must be performed for each individual unknown function of Equation 1.1. Third, the final optimization of the hybrid grey-box formulations is performed multiple times to collect a diverse set of local optima as well as the global optimum as new promising sampling locations. This final stage can become computationally intensive, as the number of dimensions and/or the number of nonconvex terms in the optimization problem increase.

These stated challenges can be resolved by taking advantage of the fact that several stages of ARGONAUT can be independently performed in parallel. To achieve this, high-performance computing is employed to implement a fully parallel version of ARGONAUT (p-ARGONAUT). Specifically, three main stages of ARGONAUT are now performed on multiple processors: (1) sample collection; (2) model selection and validation; and (3) solution of multiple local and global optimization problems of surrogate formulations, as shown in Figure 5.1.

The parallelization of all these three stages is possible because of the following reasons. In the sample collection phase, shown in Figure 5.1A, each sample has a pre-determined location in the x-space. Thus, when p processors are available, it is possible to form p different subsets of

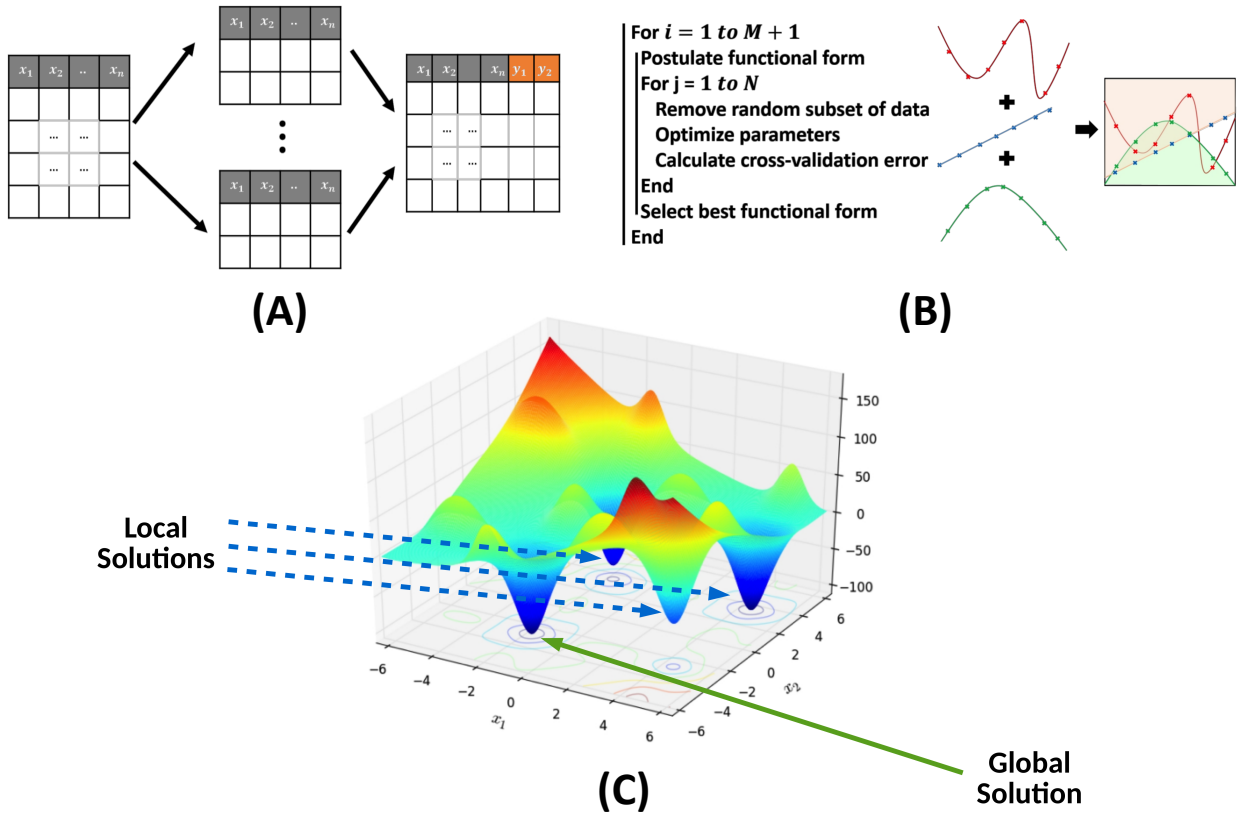

Figure 5.1: Parallelized sections of ARGONAUT: (A) Sample collection; (B) surrogate model identification and validation; (C) local and global optimization of surrogate formulations.

the initial sampling set, in order to run the simulation in parallel and collect the outputs accordingly. Moreover, each of the unknown equations is assumed to have a form that is unique and independent of the remaining formulations (Figure 5.1B). Consequently, the model identification, parameter estimation and cross-validation for each of the unknown constraints and the objective can be performed independently on multiple processors. Finally, at the end of one iteration, ARGONAUT collects multiple potential local optima, starting with multiple initial points using a local optimization solver (CONOPT) [184], and the global optimum of the surrogate formulation using a global optimization solver (ANTIGONE) [90–92], depicted in Figure 5.1C. Each of these optimization problems are independent from each other and can be solved in parallel for improved computational efficiency. The solutions from these parallel optimization problems are collected by p-ARGONAUT, which further identifies the unique solutions as new sampling points and proceeds

to the next iteration.

5.3 Dimensionality Reduction Using Functional Control Method

For the problem of well control optimization, the goal is to generate the optimal control trajectory of an oil-well for a given time horizon, and this tends to create very high-dimensional optimization formulations. As described in Section 5.1, the optimal water-flooding control problem can be tackled either by optimizing (1) the well rates or (2) the bottom hole pressures (BHPs). Using the BHP control approach, the simulation variables are the pressures for the producers and the injectors per time step, given in Equations 5.1 and 5.2, where n_I is the number of injectors, n_P is the number of producers and n_T is the number of time steps, respectively.

$$BHP(i,t) \qquad \forall i = 1, ..., n_I, t = 1, ..., n_T \tag{5.1}$$

$$BHP(p,t) \qquad \forall p = 1, ..., n_P, t = 1, ..., n_T \tag{5.2}$$

Naturally, the dimensionality of the (discretized) optimal control problem is dependent on the time step, the total time horizon, and the number of wells. Thus, as the time step gets smaller, or in other words the control over the wells is more frequent among a given time horizon, the number of variables of the optimization problem increases significantly. Likewise, in realistic scenarios, there are multiple injectors and producers that need to be controlled simultaneously, thus this leads to a very high-dimensional search space.

To overcome the problem, a dimensionality reduction technique (Functional Control Method, FCM) is employed to parameterize the well control domain using surrogate functions [147, 185, 186]. In this method, a known functional form is used to define the control trajectory of each well as a function of time, representing the control value at each time step. As a result, the optimization variables are reduced from the total number of pressure levels for every time step, to a set of surrogate function coefficients. This concept is illustrated with a simple example provided in Figure 5.2.

In Figure 5.2A, a typical BHP trajectory for a single well is plotted over a control interval. The

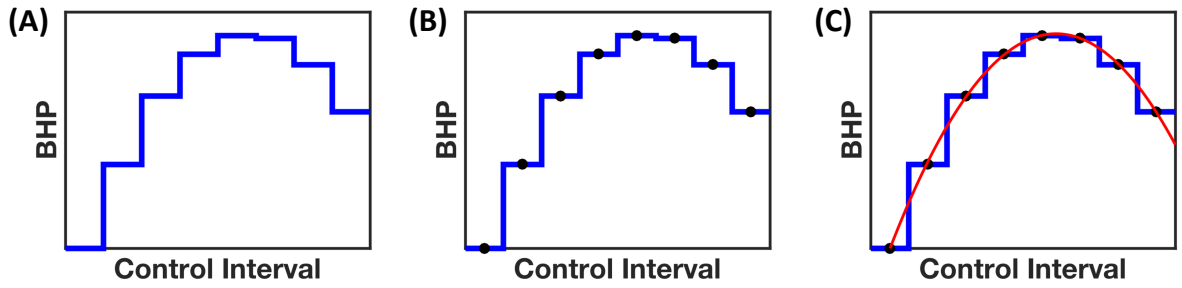

Figure 5.2: Simple illustration of the FCM: (A) An example of an BHP trajectory along a control interval; (B) midpoints of the BHPs at each control step are selected for the functional approximation, shown in black points; (C) second-order polynomial approximation is fitted through these points for approximating the original control trajectory, shown in red curve.

fundamental idea behind FCM is that this BHP trajectory can be approximated by a continuous surrogate approximation, which is a unidimensional function of time. The points that are used to train the approximation are obtained by taking the midpoints of each time step, as shown in Figure 5.2B. These points are used to fit a known surrogate function, which must be flexible enough to capture the typical trends of well control profiles, as shown in Figure 5.2C. Assuming that the number of parameters of the selected surrogate function is significantly less than the total number of well control pressures over the discretized time horizon, this approach leads to a significant reduction in the number of decision variables of the optimization problem. Specifically, using this approach, the optimization variables for the oil-well production optimization with BHP control become the parameters of the function $BHP(t)$, as described by Equations 5.3 and 5.4, where F is the maximum order of function $BHP(t)$, b_I and b_P are parameters of the injector and producer functions $BHP(t)$, respectively.

$$b_I(i,f) \qquad \forall i = 1,...,n_I, f = 0,...,F \tag{5.3}$$

$$b_P(p,f) \qquad \forall p = 1,...,n_P, f = 0,...,F \tag{5.4}$$

FCM dictates that any type of functional form that seems fit can be used to approximate the control trajectory. In this study, two functional forms are extensively tested: a second order polyno-

mial function (Equations 5.5-5.6), and a modified exponential function (Equations 5.7-5.8), which mimics an *s*-shaped trajectory within the control domain. The form of these functions was selected after carefully studying the form of many pressure control profiles in water-flooding applications.

$$BHP_I(t) = b_{I,0}(i) + b_{I,1}(i) \cdot t + b_{I,2}(i) \cdot t^2 \qquad \forall i = 1, ..., n_I \tag{5.5}$$

$$BHP_P(t) = b_{P,0}(p) + b_{P,1}(p) \cdot t + b_{P,2}(p) \cdot t^2 \qquad \forall p = 1, ..., n_P \tag{5.6}$$

$$BHP_I(t) = \frac{1}{1 + exp(b_{I,0}(i) + b_{I,1}(i) \cdot t + b_{I,2}(i) \cdot t^2)} \qquad \forall i = 1, ..., n_I \tag{5.7}$$

$$BHP_P(t) = \frac{1}{1 + exp(b_{P,0}(p) + b_{P,1}(p) \cdot t + b_{P,2}(p) \cdot t^2)} \qquad \forall p = 1, ..., n_P \tag{5.8}$$

In addition to dimensionality reduction, this approach results in optimal pressure control profiles that are relatively smooth, avoiding the occurrence of drastic changes in pressure levels from one time step to the next, which can cause operational difficulties. In fact, bounds on parameters of the surrogate functions can indirectly control the rate of change of the surrogate control profiles. In other words, upper and lower bounds on the new optimization variables, namely the surrogate function parameters b_I and b_P, can be inferred depending on the type of function used and known bounds on the control BHP variables. As an example, the effect of each parameter of the second-order polynomial expression on the trajectory characteristics is used to derive bounds that allow the proposed formulations to capture any possible control trajectory that the simulation might encounter. Hence, the bounds on the parameters are inferred by realizing that the zeroth-order parameter gives insights on the point where the polynomial intercepts the y-axis (initial pressure level), while the first and second-order parameters define the rate of change and the curvature of the polynomial, respectively. It is important to note that the pressure calculated using the polynomial approximation may exceed its bounds. In that case, the value of the pressure at that specific time step is set to the value at the nearest bound.

The second type of surrogate proposed is based on the fact that the pressure depletion in an oil reservoir is characterized by an exponential form, since the pressure control profiles have a

tendency to show a gradual increase or decrease towards their upper or lower bounds, after a certain amount of time has passed within the simulation [187]. To better capture this inherent trend in the pressure profiles, a unique *s*-shaped exponential functional form is introduced as an alternative way to approximate control trajectories within FCM, which contains the same number of parameters as the polynomial function (Equations 5.7-5.8). Through the results of this work, the ultimate aim is to quantify which of the two parametrization techniques is the most versatile and appropriate for optimization.

5.4 UNISIM Case Study Models

The proposed methodologies, namely the p-ARGONAUT coupled with an initial parametrization of the pressure space using FCM, have been used to solve the UNISIM case study, which is a complex oil reservoir benchmark problem. This benchmark problem is a realistic three-dimensional model developed by Gaspar et al. [188], and it has been widely used for identifying optimal oil exploitation strategies. The original model for this case study contains approximately 3.5 million active grid blocks based on the petrophysical characteristics of the Namorado Field, located in Campos Basin, Brazil. Avansi and Schiozer [189] developed a medium-scale reservoir model based on the UNISIM case study, to make it more applicable to the optimization of reservoir management operations, which may require many simulation calls. In this work, the latter reservoir model is used, which contains 20 layers with $100 \times 100 \times 8$ grid cell resolution and approximately 37,000 active grid blocks.

The problem contains 4 vertical production wells, 10 horizontal production wells and 11 horizontal injection wells [190]. Pore pressure, fracture pressure and minimum allowable pressure difference between an injector and a producer are dictating the bounds on the simulation variables which are given in bar in Equations 5.9-5.10. These bounds allow us to infer bounds on the parameters of the surrogate functions show in Equations 5.5-5.8.

$$190 \leq BHP(i) \leq 350 \qquad \forall i = 1, ..., n_I \tag{5.9}$$

$$35 \leq BHP(p) \leq 180 \qquad \forall p = 1, ..., n_P \tag{5.10}$$

In addition to the bound constraints, there are flowrate, platform capacity and economic limit constraints that are being considered in this work. The maximum flowrates (m^3/day) of water, $q_W(i,t)$, (Equation 5.11), and total liquid (water and oil, $q_{liq}(i,t)$), (Equation 5.12), which can be processed by each injector and producer, respectively are:

$$q_W(i,t) \leq 6000 \qquad \forall i = 1, ..., n_I, t = 1, ..., n_T \tag{5.11}$$

$$q_{liq}(p,t) \leq 3000 \qquad \forall p = 1, ..., n_P, t = 1, ..., n_T \tag{5.12}$$

Platform capacity constraints for water in producers, $Q_{P,W}(t)$, (Equation 5.13), oil in producers, $Q_{P,O}(t)$, (Equation 5.14) and water in injectors, $Q_{I,W}(t)$, (Equation 5.15) in m^3/day are:

$$Q_{P,W}(t) = \sum_{p=1}^{n_P} q_W(p,t) \leq 21240 \qquad \forall t = 1, ..., n_T \tag{5.13}$$

$$Q_{P,O}(t) = \sum_{p=1}^{n_P} q_O(p,t) \leq 21240 \qquad \forall t = 1, ..., n_T \tag{5.14}$$

$$Q_{I,W}(t) = \sum_{i=1}^{n_I} q_W(i,t) \leq 30680 \qquad \forall t = 1, ..., n_T \tag{5.15}$$

The constraints represented in Equations 5.11-5.15 are critical for obtaining realistic solutions in terms of water handling based on the capabilities and capacities of the field. These constraints seem like simple bound constraints, however, it must be stressed that these are complicating grey-box constraints, since the flowrates at each well and time step are outputs of the reservoir simulation, controlled by the original variables, namely the BHP and the solution of the discretized model.

Finally, the water-cut (WC) constraint is considered, which is critical to the economic viability of the field. The term water-cut is defined as the fraction of water produced in the total amount of liquid (water and oil) produced from all producer wells. The expression for the WC constraint is

obtained by setting the revenue on oil, R_O, to be greater than the costs of injecting and producing water (Equation 5.16). By enforcing this limit, the field cash flow is restricted to nonnegative values in each control time step, and thus this constrains the feasible space of the overall problem by taking into account the project's economic limit.

$$WC(t) \leq \frac{R_O - C_{I,W} \cdot VRR(t)}{C_{P,W}(t) + R_O} \qquad \forall t = 1, ..., n_T \tag{5.16}$$

Here, $C_{I,W}$ and $C_{P,W}$ are the costs of injecting and producing water, respectively, and VRR is the voidage replacement ratio, defined as the ratio of the volume of the injected fluid to the volume of the total produced fluid (Equation 5.17).

$$VRR(t) = \frac{Q_{I,W}(t)}{Q_{P,W}(t) + Q_{P,O}(t)} \tag{5.17}$$

The revenue, R_O, is given by the difference between the price of oil, PR_O, and cost of oil production, C_O. In this study, the cost of oil production is assumed to be zero and the revenue is taken to be equal to the price of oil, which provides an upper bound on the economic profitability of the operation. Given the bounds and constraints, the objective is to maximize the NPV, explicitly defined in Equation 5.18, using d as the discount rate of the project, Δt_j as the time interval at each step, $q_O^{k,j}$ and $q_W^{k,j}$ as the flowrate of oil and water for each injector/producer at each time step, respectively.

$$NPV = \sum_{j=1}^{n_T} \Delta t_j (1+d)^{-(\frac{t_j}{n_T})} \left(\sum_{k=1}^{n_P} R_O q_O^{k,j} - \sum_{k=1}^{n_P} C_{P,W} q_W^{k,j} - \sum_{k=1}^{n_I} C_{I,W} q_W^{k,j} \right) \tag{5.18}$$

5.5 Results of Computational Studies

The goal of this study is to (a) solve the UNISIM benchmark using the constrained formulation described in Equations 5.9-5.18, in order to provide valuable insights regarding the nature and complexity of all the constraints under consideration, and (b) test and compare various components

of the proposed methods. For this reason, series of computational studies were performed on the UNISIM benchmark problem to test the accuracy, efficiency and consistency of p-ARGONAUT coupled with FCM, for maximizing the NPV of oil production. The MATLAB Reservoir Simulation Toolbox (MRST) is used as the forward model simulator for the UNISIM benchmark problem [191, 192]. In this simulation, it is assumed that the reservoir pressure is above the bubble point and the fluids are immiscible and incompressible.

This water-flooding optimization problem is studied for a horizon time of 5 years, with control adjustment performed on a montly basis. As a result, the overall process time is discretized into 61 intervals, and for a total of 25 wells, the total number of original pressure control variables is 1525. It is important to note that the decision variables for the water-flooding optimization problem are not the original control variables at the simulation level, but they are the coefficients of the second-order polynomial (Equations 5.5-5.6) and exponential functions (Equations 5.7-5.8) postulated in FCM, which are directly linked to the BHP for each injector and producer. As a result, by using the FCM, the size of the input space is transformed from 1525 variables to 75 variables. Through the results of this work, the aim is to investigate whether the selection of the type of surrogate function for the FCM has an effect on the optimization, and if yes, to identify which surrogate function is optimal. For this reason, all the case studies are solved using both the polynomial and exponential functions, to represent the pressure control trajectories. The detailed list of the parameters used in the analysis of this problem, as well as a comparison of the dimensionality of the problem with and without the FCM approach are provided in Table 5.1.

Another key point is that the input space of the water-flooding optimization problem is defined as a function of the BHPs, whereas the constraints that are presented in Equations 5.11-5.16, as well as the objective in Equation 5.18, are functions of well flowrates. As a result, this problem is inherently a grey-box problem, with many unknown functions. In other words, the reservoir simulation requires the bottom-hole pressure profiles as inputs, and provides as outputs the flowrates at each well, which are in return used for the calculation of the constraints and the objective. Alternatively, one can choose to use flowrates as control variables for this optimization problem, in

Table 5.1: Values of the parameters used in the reservoir simulation and the dimensionality of the problem using traditional approach versus FCM.

Parameters	**Value**
R_O, PR_O	\$50/stb
$C_{I,W}$, $C_{P,W}$	\$1/stb
d	0.09
n_T	61
n_I	11
n_P	14
Dimensionality of the Optimization Problem	
Original Control at Simulation Level	Functional Control Method with $F = 2$
1525 variables	75 variables

which case, any constraints related to pressure-control and pressure bounds would be grey-box constraints.

One of the goals of this work is to start with the most comprehensive formulation, including all potential realistic constraints related to water-flooding operations, however, it was expected that some constraints may be more difficult to satisfy than others. This was proven after initial testing of the problem, which revealed that water-cut constraints, total liquid flowrate constraints and platform capacity constraints for the producers are satisfied easily for this case study. However, water flowrate constraints and platform capacity constraints for the injectors constrained the feasible region significantly. Based on this insight, a cascaded approach was followed, which involved first studying the problem with only bound constraints, and subsequently adding each set of grey-box constraints (Equations 5.11-5.16) to individual sub-problems. This approach allows (a) testing the performance of p-ARGONAUT on problems of increasing complexity, and (b) the identification of the set of complicating constraints that significantly affect the profitability of the operation, which is an aspect that is typically not studied simultaneously through a formalized optimization formulation. The case studies that have been solved are shown below:

- Case 0: No grey-box constraints.

- Case 1: 61 grey-box constraints: Equation 5.15.
- Case 2: 671 grey-box constraints: Equation 5.11.
- Case 3: 732 grey-box constraints: Equations 5.11 and 5.15.
- Case 4: 733 grey-box constraints: Equations 5.11, 5.15 and one penalty constraint calculated by summing the violations of Equations 5.12-5.14, 5.16.
- Case 5: 1769 grey-box constraints: Equations 5.11-5.16.

In addition to the selection of the appropriate surrogate function for the initial dimensionality reduction stage, there is a need to select a surrogate function to represent each of the objective and the constraints of the grey-box formulations of all six case studies. However, one of the advantages of the ARGONAUT framework is its ability to train, select and validate a function for each unknown correlation out of a pool of a library of functions using the minimum average cross-validation error. This aspect provides insight on the nonlinearity of each individual unknown constraint and the objective, which is reported in the results. Throughout the results that are presented in the following sections, the solution strategy relies on the framework's ability to select the most appropriate function to represent the objective and the different classes of constraints. Nonlinear functions, such as quadratic and kriging functions were selected most frequently to represent the objective function and the constraints in the formulation, indicating that the problem is in fact nonlinear. Surprisingly, a linear function was found to be optimal to represent a certain class of constraints, as described in detail in the next section. In previous work, it was shown that the selection of the surrogate function combination to represent the grey-box formulation has a significant effect on the quality of the optimal solution, the computational cost, and the required number of samples for convergence [29]. Although the same effects are observed in this study, this work does not present a thorough comparison between different types of surrogates for optimization, assuming that the framework has made the optimal selection.

The results for each case are also compared with other gradient-free methods: the local-search NOMAD method [95] and the global model-based constrained EGO (con-EGO) algorithm

[193, 194]. The NOMAD algorithm makes use of surrogate formulations to guide the search, implements a progressive barrier approach to handle general constraints, and requires an initial point to be provided. On the other hand, con-EGO models the objective and the constraints using kriging formulations and selects new points by maximizing the Expected Improvement function over the entire search space. For fairness, the first stage of dimensionality reduction is used for all comparisons that are performed, so all of the methods are tested on case studies with 75 variables. Each case is executed five times on a High-Performance Computing (HPC) machine at Texas A&M High Performance Research Computing facility, using Ada IBM/Lenovo x86 HPC Cluster operated with Linux (CentOS 6) using 1 node (20 cores per node with 256 GB RAM), where each time the global search algorithms (p-ARGONAUT and con-EGO) algorithms are initialized with different sampling sets, while NOMAD is initialized with a different initial point.

5.5.1 NPV Without the Grey-Box Constraints

In this first case study, none of the constraints are considered, and comparative results between the use of the polynomial and the *s*-shaped exponential function for the pressure profile parametrization are presented. The optimal NPV obtained from each method is provided in a boxplot in Figure 5.3. The overlaid plots show the pressure profiles for the first injector and eighth producer of the best solution out of five runs. The objective function within the p-ARGONAUT runs is fitted using a quadratic surrogate, which indicates that the objective is a relatively smooth function. Even though in several pressure control oil-field operation problems a flat objective function surface has been reported Zhao et al. [195], it is found that the global behavior of this problem is highly multimodal. Similar nonlinear behavior has also been reported in the literature by Fonseca et al. [196], which provides a plot of the undiscounted NPV projections using multidimensional scaling. The multimodal nature of this objective function is evident in the results provided in Figure 5.3, by observing that different methods converge to different local solutions. The results show that p-ARGONAUT provides a higher NPV with consistency, when compared to NOMAD for both functional forms (-poly and -exp). On the other hand, p-ARGONAUT and con-EGO provide comparable results when the problem is box-constrained. The high variability

of the results obtained by the local solver NOMAD, can be explained by the effect of the random initial point, which is a starting point of the local adaptive search. If a good initial point is known and is used, the performance of this method is expected to improve significantly.

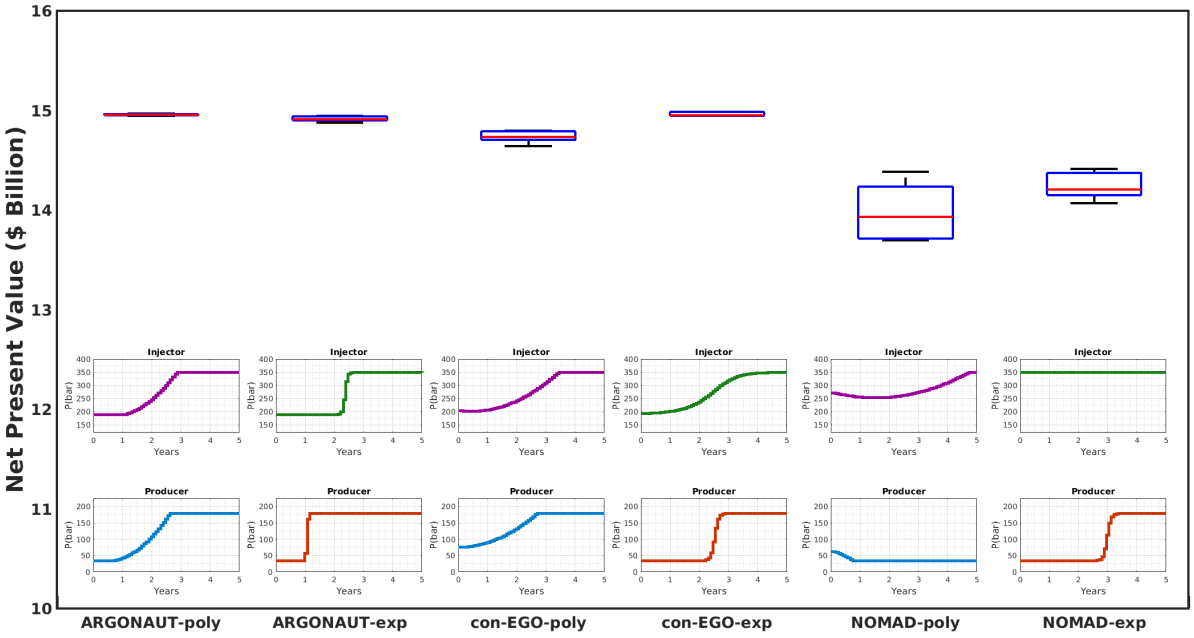

Figure 5.3: Optimal NPV for the box-constrained water-flooding optimization problem. -poly indicates that a second-order polynomial is used in the FCM formulation, given in Equations 5.5 and 5.6. -exp indicates that a modified exponential function is used in the FCM formulation, given in Equations 5.7 and 5.8. Overlaid pressure profiles, for the first injector and eighth producer, show the difference between the control trajectories that are approximated with polynomial versus exponential function.

In order to quantify the computational and qualitative gain achieved by parallelization, the results obtained by the original, sequential ARGONAUT framework, is compared with the results obtained by p-ARGONAUT for this case. By setting a CPU limit of 168 hours, it is observed that the non-parallelized algorithm often hits this limit, while the parallel version converges within 20-50 hours. Most importantly, better solutions are always obtained with the p-ARGONAUT framework.

5.5.2 NPV With the Grey-Box Constraints

All of the case studies presented in this Section contain different combinations of grey-box constraints, as described earlier. Case 1, represents a formulation with only the platform capacity constraints, which were identified to be active and highly nonlinear. For this reason, it is observed that p-ARGONAUT transitions to using kriging surrogate functions for their representation. In order to validate the framework's ability to select the most appropriate surrogate function, this case is specifically solved twice, first fixing the surrogate functions to quadratic and second to kriging type (Figure 5.4). Kriging surrogates were able to locate improved feasible solutions based on the NPV (Figure 5.4A). In addition, faster convergence is achieved by a simultaneous reduction in the number of required calls to the reservoir simulation when p-ARGONAUT uses kriging surrogates for the grey-box functions (Figure 5.4B). This is a highly desirable result, since reservoir simulations can have a significant computational cost.

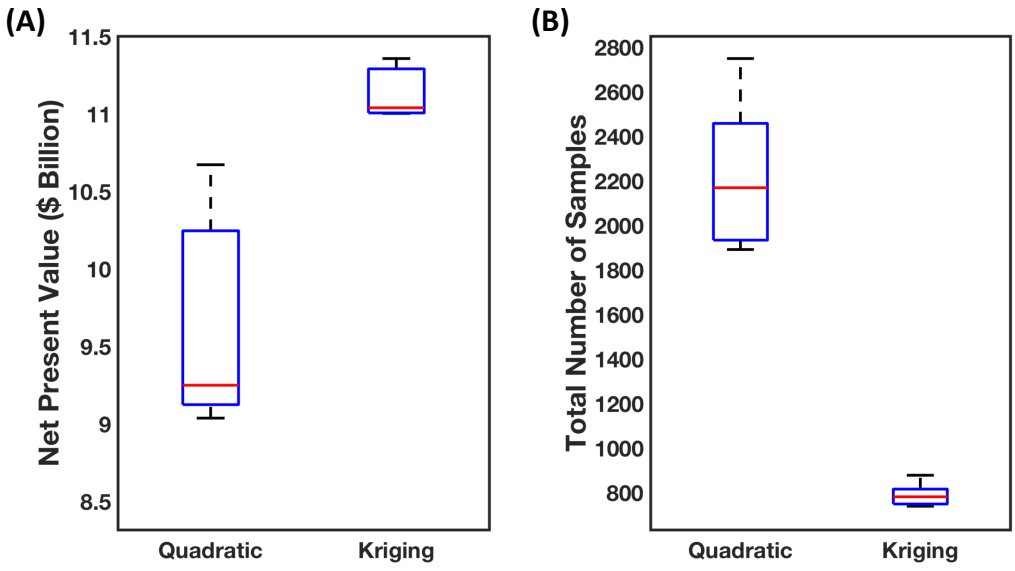

Figure 5.4: Using quadratic versus kriging surrogates as surrogate approximations for optimization within the p-ARGONAUT framework, for case 1 (61 highly nonlinear grey-box constraints). (A) Best obtained NPV values; (B) required number of samples from the simulation for convergence to solutions in (A).

The cumulative results for cases 1 through 5 are presented in Figure 5.5. In all cases, p-ARGONAUT obtains the best NPV for both functional forms, and by using kriging surrogate functions to represent most of the constraints of the investigated problems. Furthermore, it should be noted that only for case 4, all of the less active constraints are lumped into a single penalty function, which is the sum of the violations of the grey-box constraints given by Equations 5.12-5.14 and 5.16. For this case study, the penalty function is best approximated by a linear surrogate function, which is another indication that p-ARGONAUT can select the most appropriate and simplest function to represent correlations. In cases 2 through 5, it is important to state that a fraction of con-EGO and NOMAD runs are terminated with high infeasibility within the dedicated CPU time. In case 2, for the con-EGO runs, 2 out of 5 runs were infeasible for the second-order polynomial approximation and 4 out 5 runs were infeasible for the modified exponential approximation. Similarly, in the NOMAD runs, 1 out of 5 runs were infeasible for the second-order polynomial approximation and 5 out of 5 runs were infeasible for the modified exponential approximation. Since all the runs for NOMAD-exp were infeasible, their results are excluded from the boxplot in Figure 5.5. Likewise, in case 3, 4 out of 5 runs of con-EGO-poly and 1 out of 5 runs of con-EGO-exp were infeasible. Also, 1 out of 5 runs of NOMAD-poly and 5 out of 5 runs of NOMAD-exp were infeasible. This trend is also observed in the most complete cases 4 and 5, while p-ARGONAUT consistently identifies feasible solutions throughout all the runs for the highly constrained optimization problems.

It is important to note that in case 4, even though 4 out 5 runs of NOMAD for the exponential approximation were infeasible, the one single solution that this method finds is better than the average performance of p-ARGONAUT. This is an indication that a local solver can perform quite well, when a good, feasible initial point is provided. In addition, it is important to note that both con-EGO and NOMAD, are reliable and efficient tools, which have been used to solve many significant problems successfully. However, these methods are designed for problems with lower number of dimensions and constraints.

The details on the CPU times and number of samples collected are provided in Figures 5.6 and

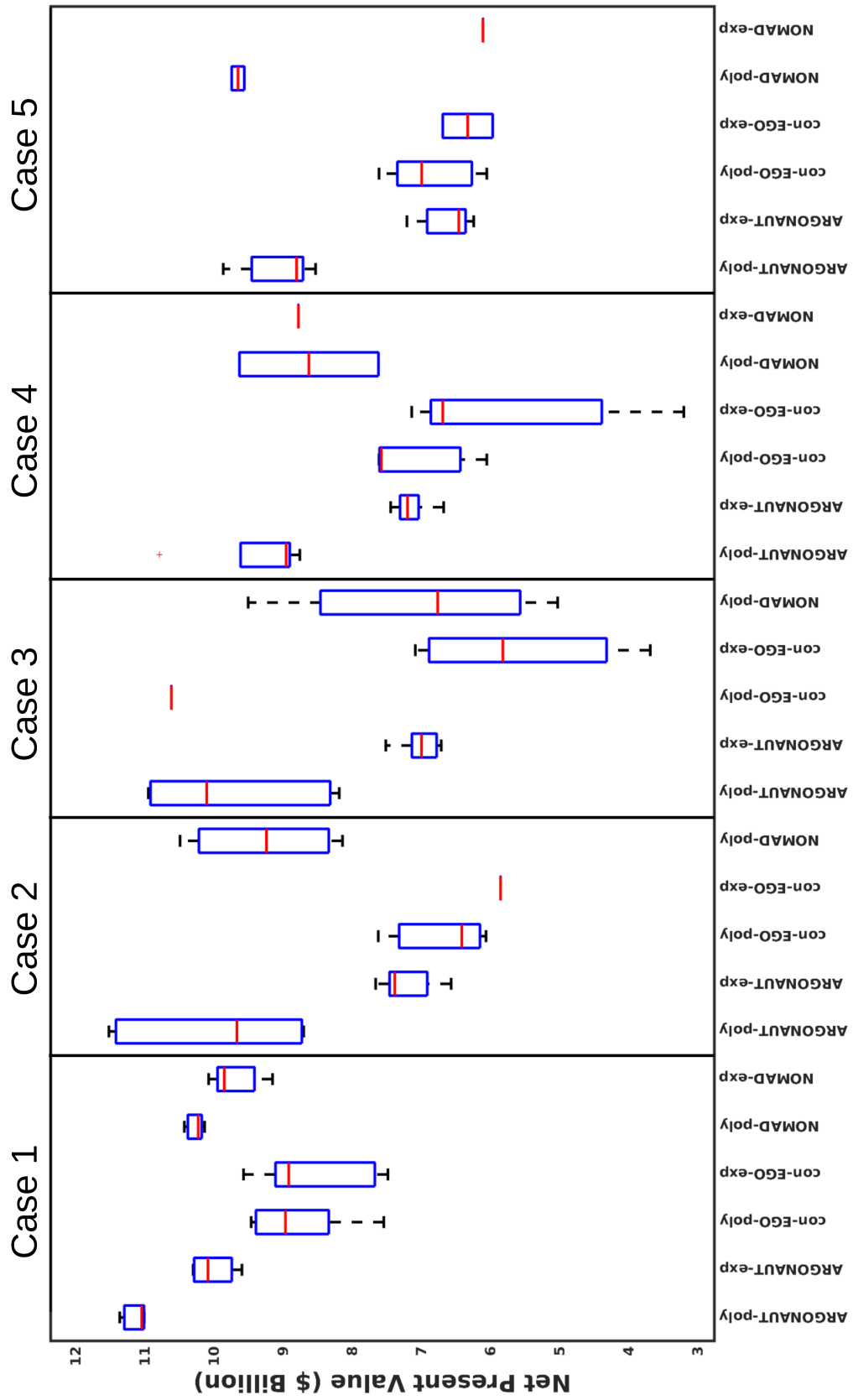

Figure 5.5: Best obtained NPV values for the cases 1 through 5. The number of grey-box constraints increases with increasing case number.

5.7, respectively. Comparing all the methods used in this work in terms of number of required samples and CPU times, it is evident that con-EGO converges to the solutions reported in Figure 5.5 with a fewer number of samples than p-ARGONAUT, while NOMAD consistently requires more samples for local convergence. However, by taking a fully parallelized approach in the new algorithm, consistent and reliable performance with less, or at least comparable CPU times was achieved. In addition, it is important to factor in the fact that the collection of more samples is accompanied by improved and consistent behavior in terms of locating better and feasible solutions.

Furthermore in Figure 5.7, it is observed that the total number of samples collected by p-ARGONAUT in the unconstrained problem (case 0) is significantly higher than the number of samples collected for the constrained problems in cases 1-5, which is an interesting finding. Studying the results in detail, it is found that this observation can be explained by the following two reasons. First, as more constraints are added, there is a significant reduction of the feasible region, which reduces the sampling search space, and thus this leads to faster convergence. This can be observed in case 1, where results are obtained before p-ARGONAUT reaches the maximum CPU limit, but with a fewer number of samples than case 0. Secondly, when the problem is unconstrained, p-ARGONAUT is generally able to complete more iterations within the maximum computational time that is enforced, which is directly connected to the number of samples collected. This affects some of the runs with even higher number of constraints, where the parameter estimation and global optimization of several hundred to thousand equations increases the time required per iteration.

Through these results, it is observed that the performance of all methods is affected by the type of the function used to approximate the pressure control profile. The best and most consistent results are obtained using the simpler second-order polynomial in FCM, which implies that there is no need to resort to the more complex *s*-shaped function to represent pressure control profiles. In addition, the difficult constraints which limit the feasible NPV can be identified as the maximum amount of water processed by the injectors, as well as the total amount of injection water that the platform can hold. In other words, if the water-related constraints were not considered, the optimal

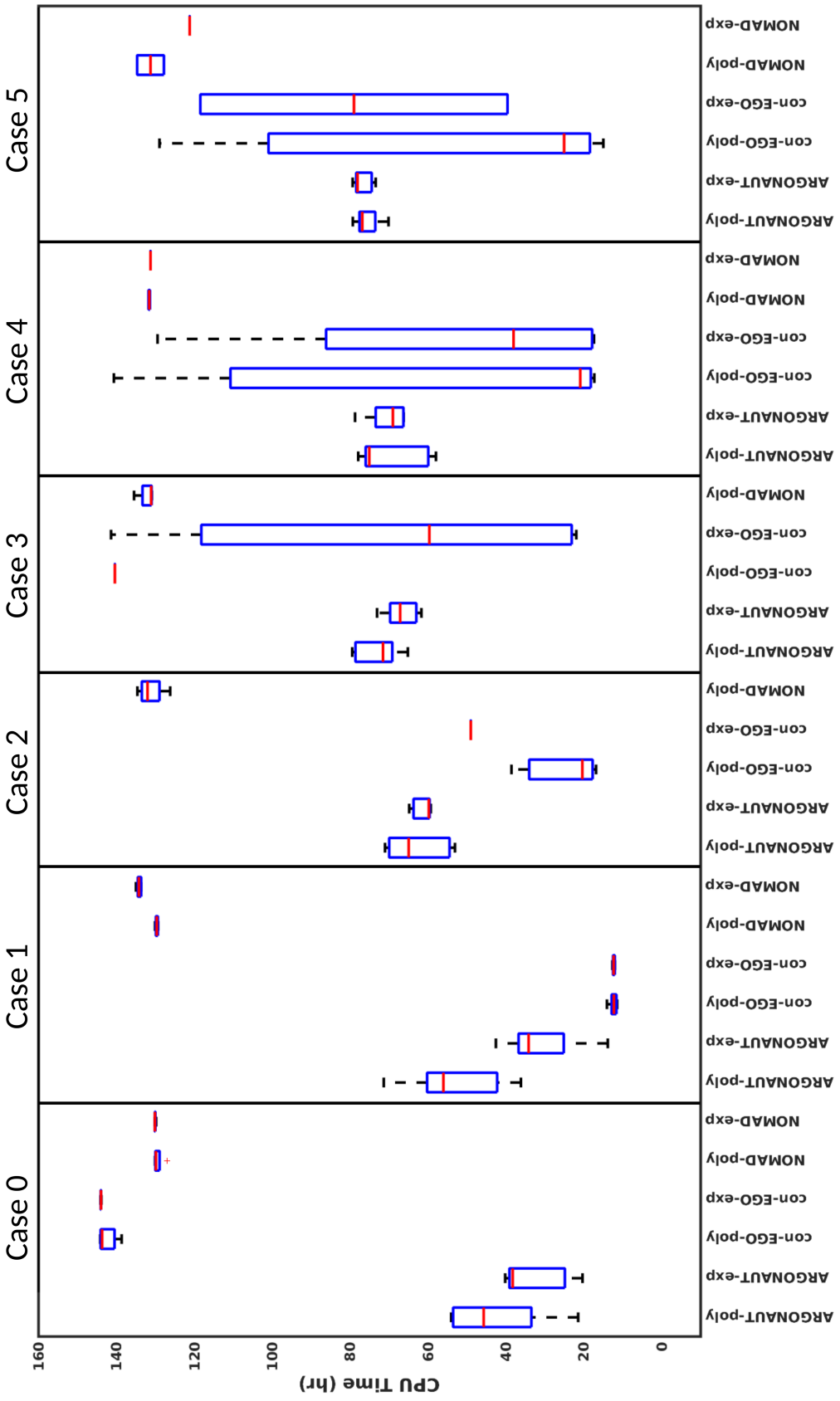

Figure 5.6: CPU times for each case and each solver.

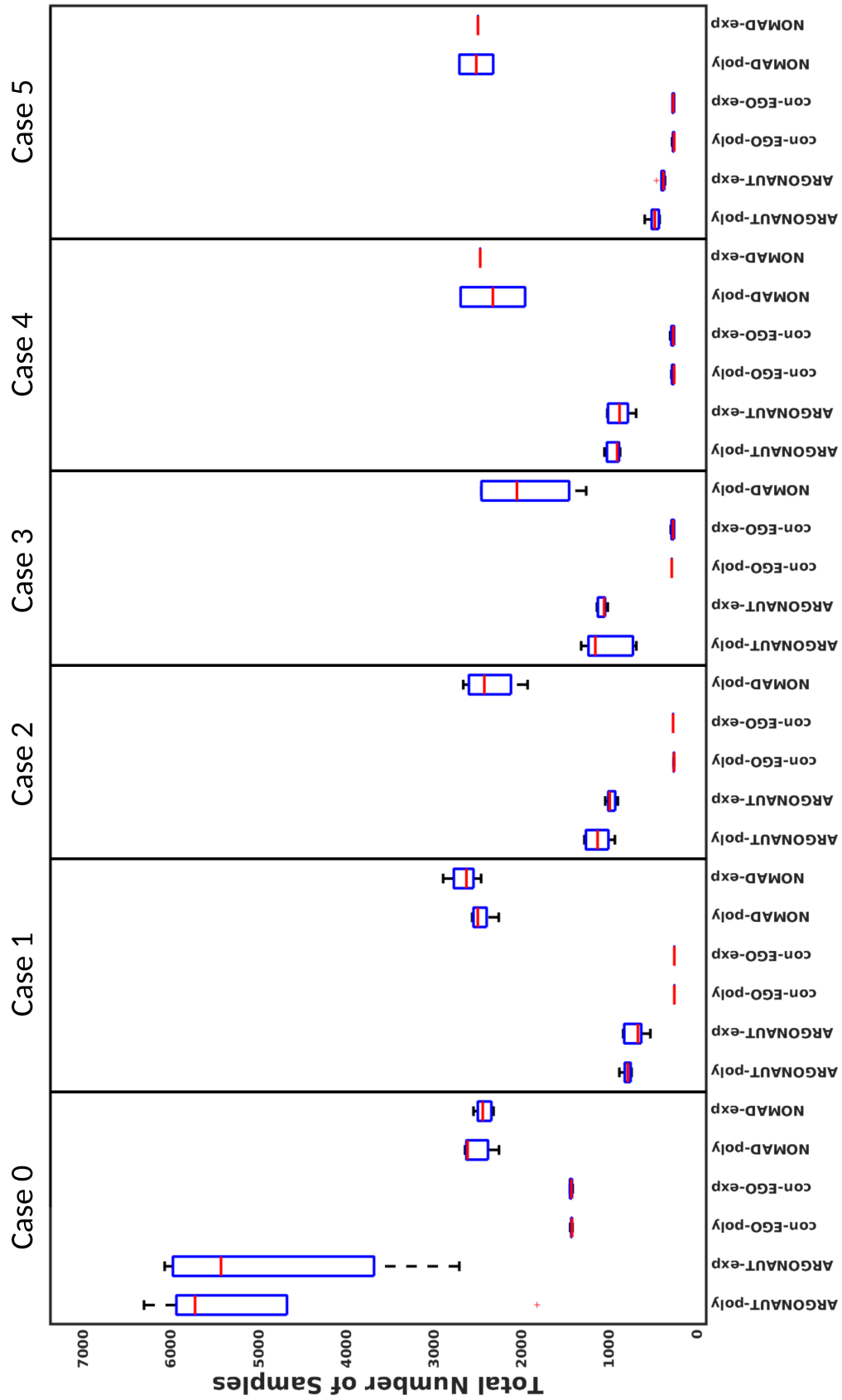

Figure 5.7: Number of samples collected by each solver for each case.

attainable NPV would be misleadingly overestimated. On the contrary, water-cut constraint that is considered in this case study, is satisfied easily without a significant effect on the NPV. The effect of each set of grey-box constraint on the cumulative oil and water production for the optimal solution obtained using p-ARGONAUT-poly is shown in Figure 5.8.

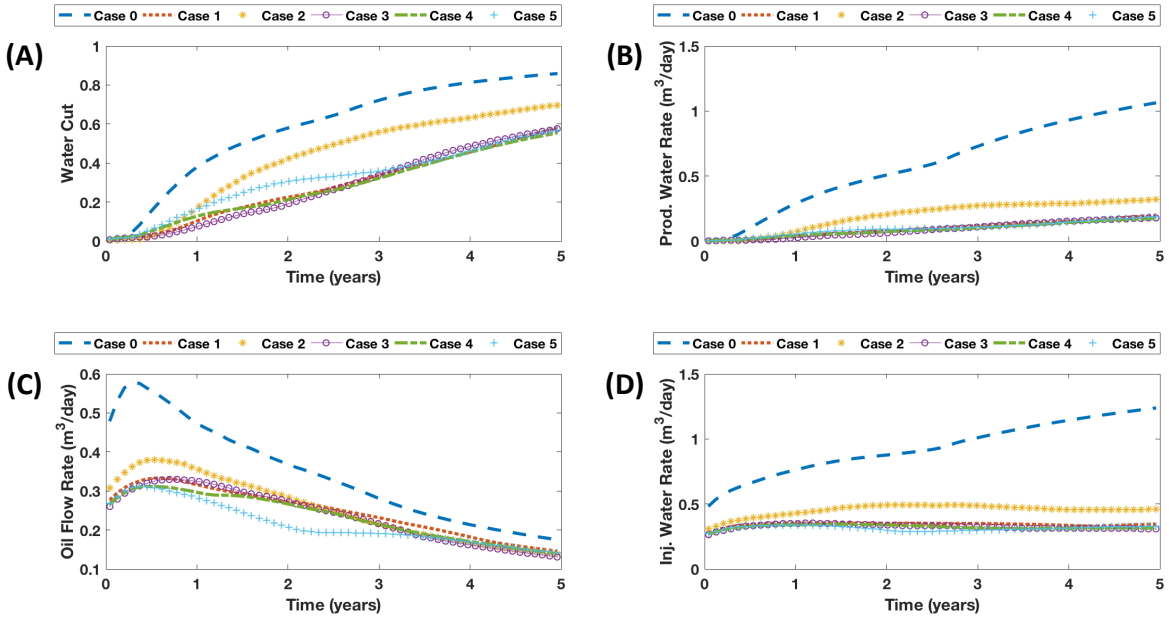

Figure 5.8: Production plots as a function of time for the best solution for all cases for p-ARGONAUT using polynomial approximation in FCM: (A) Water-Cut plots; (B) cumulative water production rate from the producer wells; (C) cumulative oil flowrate from producer wells; (D) cumulative water injection rate at the injection wells.

As shown in Figure 5.8, in the case where water constraints are not considered (Case 0), the cumulative oil production, as well as the injected and produced water is significantly overestimated compared to the Cases 4 and 5, where these constraints are taken into account. It is also observed that results from cases 1, 3, 4 and 5 almost overlap for all the production plots which suggests that cumulative water injection constraint captures most of the characteristics of the full formulation in the context of fluid flowrates throughout the time horizon. Even though case 2 limits the NPV and total fluid flowrates to some extent, it is important to include the necessary additional con-

straints that are included in cases 4 and 5. These results clearly show that implicit constraints that depend on the simulation output may have significant effects on the optimal objective, and thus it is essential to be able to solve highly constrained simulation-based problems.

5.6 Concluding Remarks

This chapter highlights new computational developments in the ARGONAUT framework and presents the performance of the new parallel algorithm (p-ARGONAUT) on a challenging nonlinear nonconvex programming case study of oil-well control operations using water-flooding. Through this work, it is shown that high-performance computing can be used to reduce the computational cost of the ARGONAUT framework significantly, which leads to also extending its capabilities towards solving high-dimensional, highly constrained problems. In addition, the usefulness of surrogate functions is shown within two steps of this work: (a) the reduction of the dimensionality of the water-flooding optimization problem via parametrization of the control domain; and, (b) the optimization of simulation-based grey-box problems through the p-ARGONAUT framework. For the first step, different functional control surrogate functions are studied and it is shown that a polynomial functional form leads to an improved performance of the overall optimization framework. More importantly, it is observed that the selection of the pressure control profile influences the shape, smoothness and gradient changes of the control trajectory, and is an important decision towards creating tractable optimization formulations, without limiting the solution space of the original problem. Overall, the results of this work show that compared to a few existing derivative-free optimization methods, p-ARGONAUT can locate feasible solutions with higher objective function values, in the presence of thousands of grey-box constraints.

6. DATA-DRIVEN MODELING OF ENVIRONMENTAL AND BIOMEDICAL SYSTEMS

In this chapter, the redistribution of toxic chemical compounds due to natural disasters (i.e., hurricanes) and their corresponding biological effect on human health due to chemical exposure is investigated using exploratory data analytics and data-driven modeling.

First, in Section 6.1, exploratory data analytics is employed to investigate the redistribution of contaminated soil samples, collected after Hurricane Harvey hit the Galveston coastline within the Manchester, TX area. These contaminated sediments were previously analyzed for trace metals, Polycyclic Aromatic Hydrocarbons (PAHs), Polybrominated Diphenyl Ethers (PBDEs), Polychlorinated Biphenyls (PCBs), and Organochlorine Pesticides (OCs) using series of experimental techniques to retrieve the concentrations of these pollutants [197]. In this work, the resulting dataset is visualized using boxplots and heatmaps, and the correlations between the geospatial location of sediments and the detected pollutant concentrations are investigated. Hierarchical clustering is performed on each dataset to explore their corresponding grouping information, where the clustering similarity with respect to their geospatial location is quantified using the Fowlkes-Mallows index. The studied visualization and data analysis techniques demonstrate an effective methodology for the interpretation of contaminants and enable the diagnosis of the potential pathways for the redistribution in a post-hurricane event.

Second, in Section 6.2, the biological impact of several benchmark chemicals is explored, as many environmental toxicants affect human health in various ways. This study focuses on a subclass of chemicals that impacts the estrogen receptor (ER), a pivotal transcriptional regulator in health and disease. The estrogenic or anti-estrogenic activity of compounds can be measured by many *in vitro* or cell-based high throughput assays that record various endpoints from large pools of cells, and increasingly at the single-cell level. More specifically, multiple mechanistic ER endpoints in individual cells that are affected by endocrine-disrupting chemicals (EDCs) can be captured simultaneously by using a sensitive high throughput/high content imaging assay that is based upon a stable cell line harboring a visible multicopy ER responsive transcription unit and

expressing a green fluorescent protein (GFP) fusion of ER [198–202]. This high content analysis generates voluminous multiplex data comprised of minable features that describe numerous mechanistic endpoints. In this work, a high content image analysis and machine learning pipeline are presented for rapid, accurate and sensitive assessment of the endocrine-disrupting potential of benchmark chemicals. The multi-dimensional high throughput/high content imaging data is used to train a classification model to ultimately predict the impact of unknown compounds on the ER, either as agonists or antagonists. To this end, both linear logistic regression and nonlinear Random Forest classifiers are benchmarked, evaluated and compared for predicting the estrogenic activity of unknown compounds. Furthermore, through feature selection, exploratory data visualization and model discrimination, the most informative features are identified for the classification of ER agonists/antagonists. The results of this data-driven study showed that highly accurate and generalized classification models with a minimum number of features can be constructed without loss of generality, where these statistical models serve as a means for rapid mechanistic/phenotypic evaluation of the estrogenic potential of many chemicals.

6.1 Understanding Contaminant Characteristics and Redistribution in Post-Harvey Soil Samples Through Data Visualization and Clustering Analysis

The ultimate goal of this work is to investigate the redistribution of contaminated sediments as a result of a natural environmental disaster. To this end, several different experimental characterization techniques are used, essentially generating diverse sets of data. However, these datasets are often hard to communicate solely using spreadsheets and/or tables. As a result, identifying an effective data-driven methodology that facilitates the dissemination and interpretation of the experimental results to a wider community is of critical importance for developing rapid detection, assessment, and evaluation tools.

In this section, exploratory data analytics is used for enabling the easy visualization and interpretation of varying types of environmental datasets. 4 different data visualization techniques are explored to represent the concentration profiles of sampled soil sediments. These include; (i) boxplots, (ii) heatmaps, (iii) pie charts, and (iv) scatter plots. In addition to the visualization

of experimental analysis, the correlation of the collected samples is investigated based on their concentration profiles and geospatial locations using unsupervised analysis. The details of the experimental data acquisition and the data-driven analysis are described in Sections 6.1.1 and 6.1.2, respectively.

6.1.1 Experimental Data Acquisition

Twenty-four soil samples are collected within the Manchester, TX area for their experimental characterization. Several different experimental data acquisition techniques are utilized to measure the concentrations of various environmental toxicants within these sediment samples. Inductively Coupled Plasma Mass Spectrometer (ICP-MS) is used for measuring the concentrations of trace metals (Hg is measured using cold vapor atomic absorption spectrometry) in soil samples. Gas Chromatography/mass spectrometry (GC-MS-MS) is used for measuring the concentrations of Polycyclic Aromatic Hydrocarbons (PAHs) and Polybrominated Diphenyl Ethers (PBDEs). Gas Chromatography Electron Capture Detection (GC-ECD) is used for measuring the concentrations of Polychlorinated Biphenyls (PCBs) and Organochlorine Pesticides (OCs). The detailed experimental procedures followed for the data acquisition are described in [197].

6.1.2 Data Visualization Techniques and Analysis

First, the experimental datasets for the 24 soil samples are pre-processed by scanning them for missing entries. If a missing value is detected, this entry is replaced with the value of zero. Later, the datasets are normalized following a series of scaling steps. The concentrations of all trace metals and their respective crustal abundances (CA) are normalized with respect to the detected Aluminum concentration (Equation 6.1 and 6.2).

$$Metal_{i,j}^{normal} = \frac{[Metal_{i,j}]}{[Metal_{i,Al}]} \qquad \forall i \in Samples, j \in Metals \tag{6.1}$$

$$CA_{j}^{normal} = \frac{[CA_{j}]}{mean([Metal_{Al}])} \qquad \forall j \in Metals \tag{6.2}$$

The organic pollutants (i.e., PAHs, PBDEs, PCBs, OCs) are normalized with respect to their total values.

$$Organic_{i,j}^{normal} = \frac{[Organic_{i,j}]}{[Total\ Organic_j]} \qquad \forall i \in Samples, j \in Organic\ Pollutants \tag{6.3}$$

The resulting normalized trace metal and organic compound datasets are used for exploratory data analytics and visualization. For this purpose, boxplots, heatmaps, pie charts and scatter plots are used for the effective communication of the observed patterns in environmental datasets.

Later, the standardized z-scores of the normalized data from Equations 6.1 and 6.3 are calculated prior to the clustering analysis using Equation 6.4 for trace metals and Equation 6.5 for organic pollutants.

$$zscore_{i,j}^{normal} = \frac{Metal_{i,j}^{normal} - mean(Metal_j^{normal})}{std.dev(Metal_j^{normal})} \qquad \forall i \in Samples, j \in Metals \tag{6.4}$$

$$zscore_{i,j}^{normal} = \frac{Organic_{i,j}^{normal} - mean(Organic_j^{normal})}{std.dev(Organic_j^{normal})} \qquad \forall i \in Samples, j \in Organic\ Pollutants \tag{6.5}$$

After the final normalization step, the resulting datasets are clustered using hierarchical clustering with average linkage and the Euclidean distance metric. The clustering on the geospatial locations is performed using hierarchical clustering with the Haversine distance metric. The grouping of the samples on the map of the studied area along with the clustering dendrogram is shown in Figure 6.1. The quantitative comparison of the resulting dendrograms is calculated using the Fowlkes-Mallows (FM) index [161]. The clustering analysis is performed in R (version 3.6.0) using the "hclust" function under the "stats" library, the Fowlkes-Mallows index is calculated using the "Bk" function under the "dendextend" library and the Haversine distance of the geospatial locations are calculated using the "distHaversine" function under the "geosphere" library.

Moreover, the Mantel test is used to evaluate the correlation between geospatial distance matrix and chemical/concentration profile distance matrices under the null hypothesis. In this study, the null hypothesis is that any observed relationship between the tested two matrices could have been

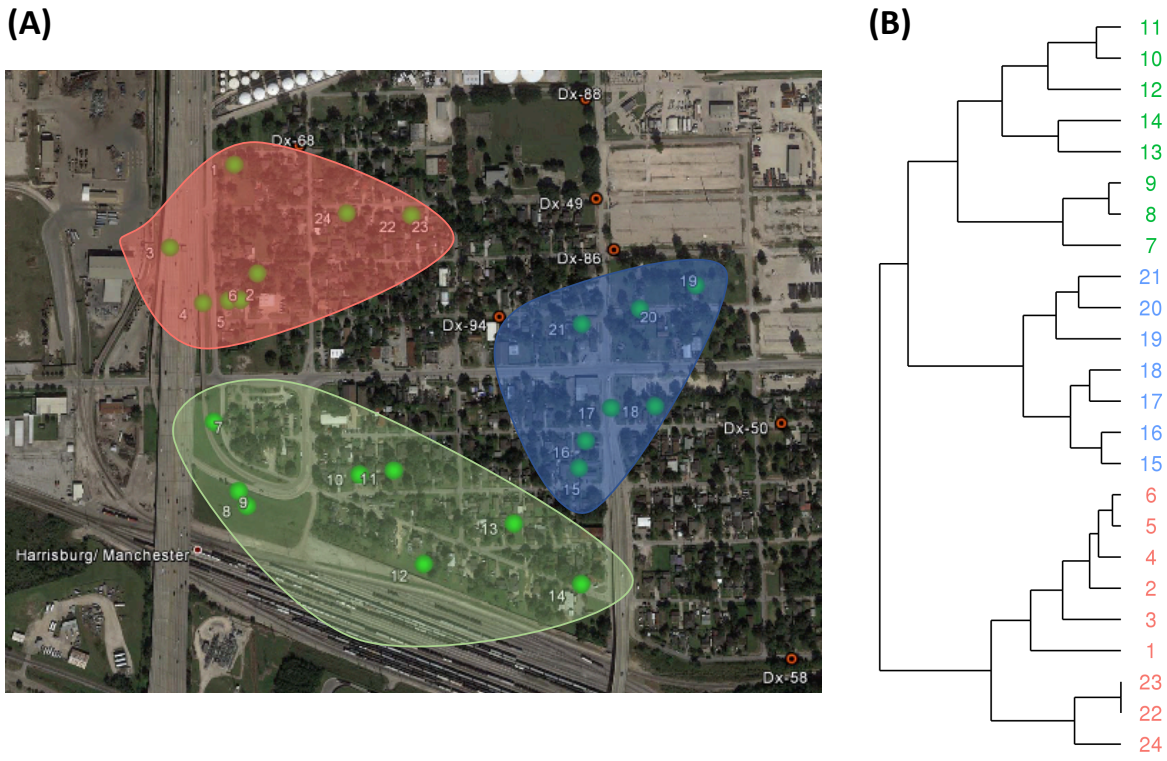

Figure 6.1: Geospatial location-based clustering analysis of the 24 soil samples collected from the Manchester, TX area. (A) Samples are divided into 3 distinct groups shown on the map. (B) The 3 groups of samples are shown on the dendrogram.

obtained by random arrangement. Hence, the statistical significance of any observed relationship between the geospatial locations and the chemical/concentration profiles are reported where the strength of the correlation is quantified using the Pearson correlation coefficient (r). The Mantel test is also performed in R (version 3.6.0) using the "mantel.test" function under the "cultevo" library.

6.1.3 Results

6.1.3.1 Visualizing Trace Metal Concentrations

The results of the overall distribution of trace metal concentrations across all samples are shown in Figure 6.2 using boxplots. Boxplots provide basic statistical analysis for a given dataset, including median, outliers, range, interquartile range. In addition, the crustal abundance information is

also provided with the boxplots, where the overall distribution of the detected metal concentrations can be evaluated for their environmental availability. Results show that several toxic trace metals are above their crustal abundance. Specifically, it is observed that Zinc, Lead, Mercury and Arsenic have higher concentrations than their crustal abundance in the analyzed soil samples.

The trace metal concentration dataset is also visualized using a heatmap to get sample-specific information. To aid the visualization, the normalized datasets from Equations 6.1 and 6.2 are used to calculate a relative normalized concentration value. If a sample is above its CA, then the following formula is used to calculate the relative concentration:

$$Conc^{+}_{relative} = \frac{Metal^{normal}_{i,j} - CA^{normal}_{j}}{max(Metal^{normal}_{i,j} - CA^{normal}_{j})} \qquad \forall i \in Samples, j \in Metals \tag{6.6}$$

If a sample is below its CA, then the following formula is used to calculate the relative concentration:

$$Conc^{-}_{relative} = -\frac{Metal^{normal}_{i,j} - CA^{normal}_{j}}{min(Metal^{normal}_{i,j} - CA^{normal}_{j})} \qquad \forall i \in Samples, j \in Metals \tag{6.7}$$

If the sample is at its CA, then the value of the relative normalized concentration is zero. This relative concentration dataset is then normalized once again prior to the clustering analysis using Equation 6.4. The resulting processed data is finally clustered and the results are visualized using a heatmap as provided in Figure 6.3.

The results show that the detected Zinc, Mercury and Selenium levels are the highest in sample 4, whereas sample 22 has elevated levels of Copper, Lead and Arsenic, and sample 10 has elevated levels of Thallium, Antimony, Cadmium, and Silver. Through the use of intuitive colors and relative scaling in this analysis, it is safe to conclude that heatmaps serve as useful visualization tools for seeing sample-specific information, thus enabling the rapid diagnosis of the elevated levels of trace metal content.

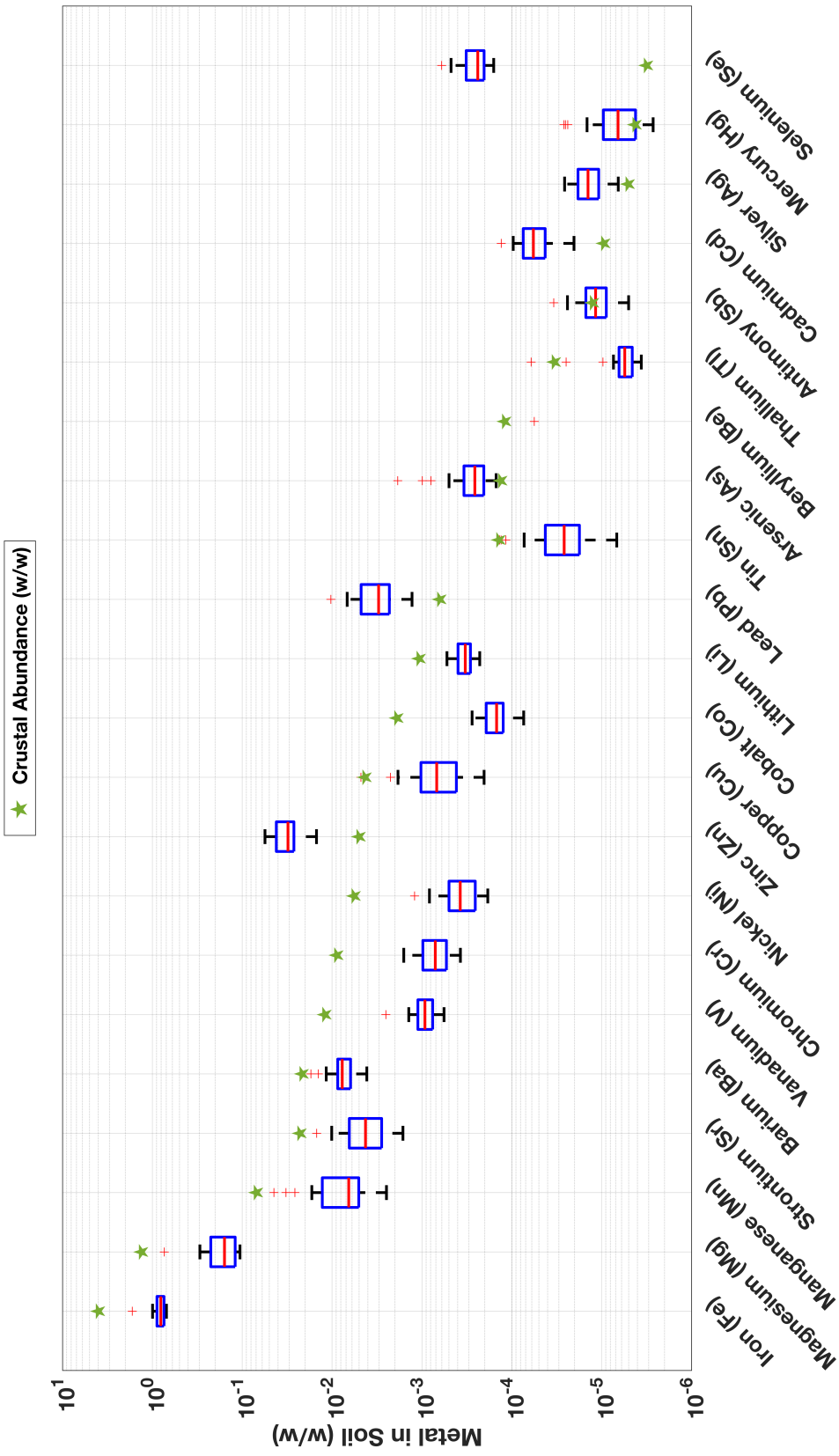

Figure 6.2: Fraction of toxic metals in collected sediment samples shown in boxplots. The star indicates the crustal abundance of the metal.

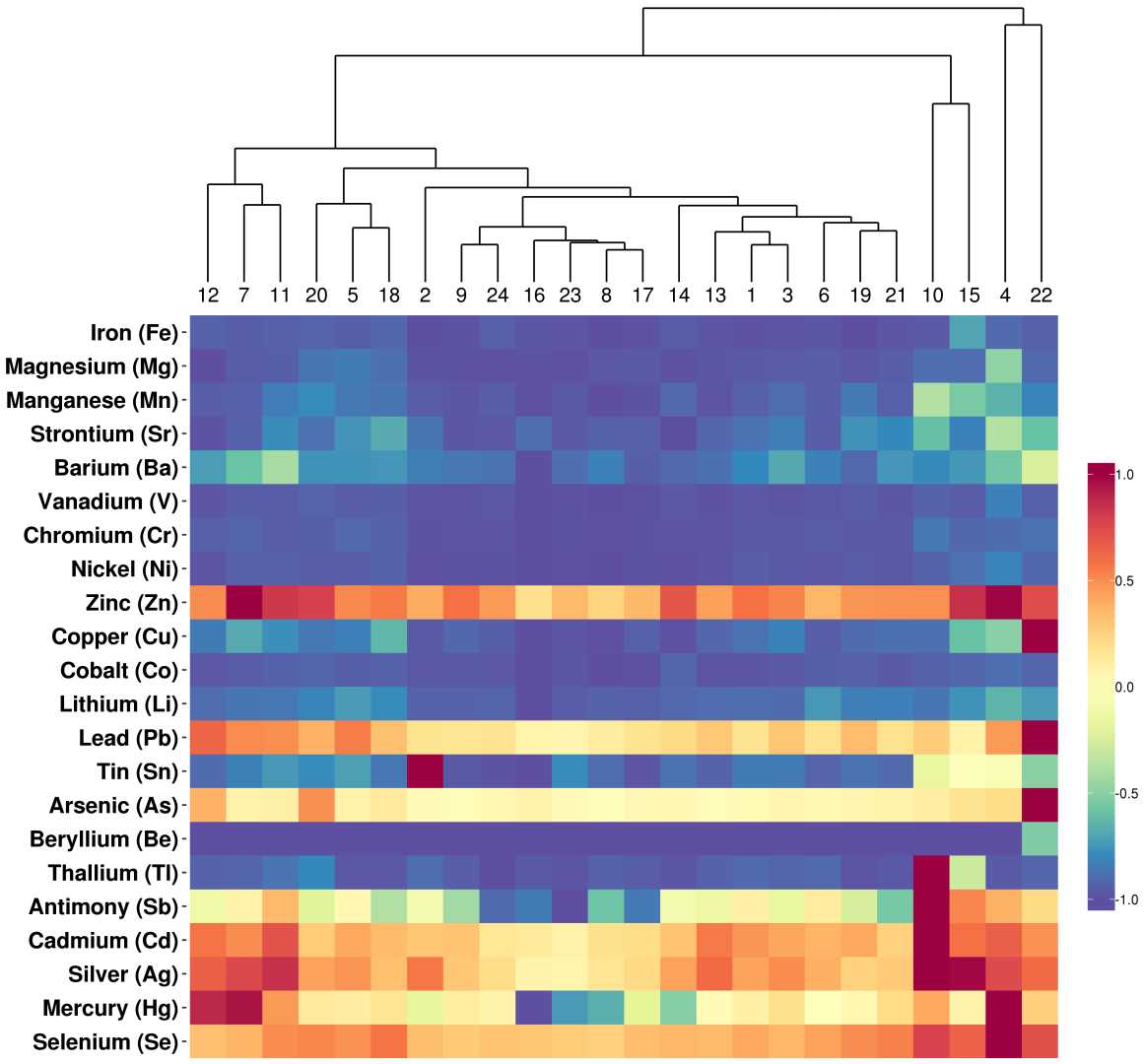

Figure 6.3: Heatmap of relative trace metal concentrations of each soil sample. The heatmap is coupled with a dendrogram to show the grouping of samples with respect to their relative concentrations. Red indicates highest level of detection, yellow indicates the CA level, and purple indicates lowest level of detection.

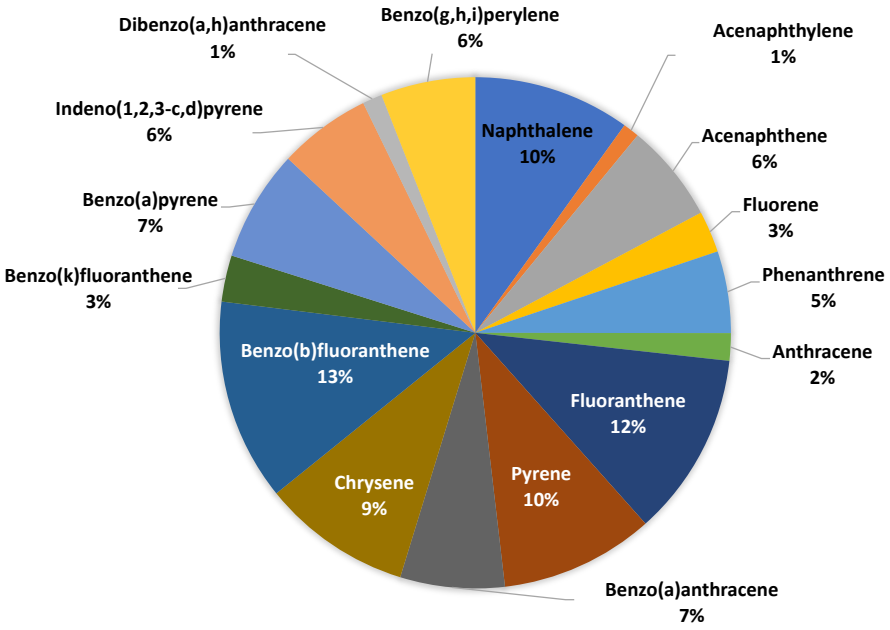

Figure 6.4: Pie chart showing the distribution of 16 priority pollutant PAHs in the collected sediment samples.

6.1.3.2 Visualizing 16 Priority Pollutant Polycyclic Aromatic Hydrocarbon Concentrations

Furthermore, the distribution of the 16 priority pollutant PAH content detected across all samples is shown in the pie chart provided in Figure 6.4. The results indicate that the sediment samples contain high levels of Benzo(b)fluoranthene, Fluoranthene, Naphthalene, Pyrene, and Chrysene compared to the other priority pollutants.

In addition, the pyrogenic and petrogenic sources of the sediment samples are explored via scatter plots. Figure 6.5A shows the Indeno(1,2,3-cd)pyrene (InP) to Benzo(ghi)perylene (BgP) ratio, indicating that all 24 samples come from a pyrogenic source. On the other hand, Figure 6.5B shows the Fluoranthene (FLA) to Pyrene (PYR) ratio which indicates that 3 of the 24 samples come from a petrogenic source. As a result, scatter plots facilitate the visualization of petrogenic or pyrogenic sources of sediments and prominently display the corresponding sample-specific information.

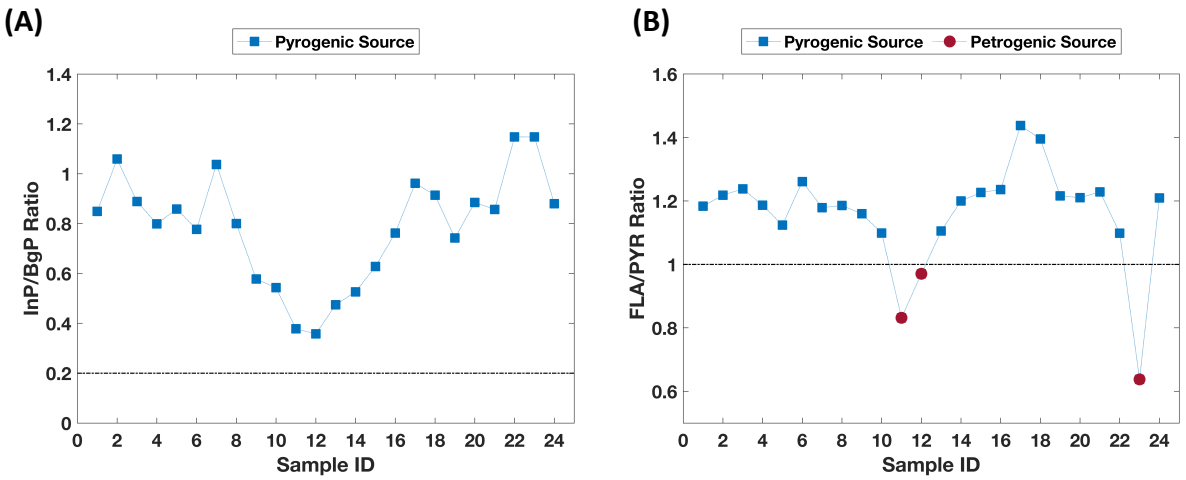

Figure 6.5: Scatter plot showing samples corresponding to pyrogenic and petrogenic sources. The pyrogenic/petrogenic cutoff is shown with a dashed line.

6.1.3.3 Clustering and Correlations with Geospatial Locations

Finally, the hierarchical clustering results of the sediments based on their pollutant concentrations and their corresponding similarity to the geospatial location-based grouping are reported. Table 6.1 provides a summary of the findings along with the results of two statistical tests. The individual clustering dendrograms generated in this analysis are provided in Appendix D (Figures D.1-D.5).

Table 6.1: The results of the clustering analysis and the similarity calculation with respect to the geospatial location grouping. Null hypothesis test is performed over 10,000 permutations.

Comparison	FM Index	Null FM Index	Mantel r	p-value
Geospatial Location - Trace Metals	0.50	0.51	0.112	0.162
Geospatial Location - 16 Priority PAHs	0.41	0.43	-0.002	0.486
Geospatial Location - PBDEs	0.51	0.51	-0.004	0.499
Geospatial Location - PCBs	0.48	0.49	0.074	0.135
Geospatial Location - OCs	0.51	0.51	0.041	0.258

The clustering results show that the trace metal content of the soil samples from the Manchester

area does not group similarly with respect to their geospatial locations. The similarity between these two clustering dendrograms is moderate as indicated by the FM index. Similarly, the comparison of clustering analysis with respect to the geospatial locations and the organic compound content shows that there is no strong correlation between the detected concentrations of organic toxicants and geospatial locations. As the area has been subject to high volumes of rain and floodwater after Hurricane Harvey, the incoming water to the area may have caused the environmental pollutants to randomly disperse over the sampled area. The correlation analysis also indicates that there is no point source of contamination and the observed correlation coefficient (r) between the samples is close to zero (i.e., no correlation). This observation is further supported by two statistical analyses: (1) the true value of the FM index for all pollutants is equal or worse than the null FM index; and, (2) the p-value of the permuted results is high. This indicates that there is little evidence against the null hypothesis and the observed grouping similarity with respect to the geospatial locations is due to the random arrangement. The next section discusses the biological impact of environmental toxicants due to chemical exposure.

6.2 Classification of Estrogenic Compounds Through Image Analysis Using Machine Learning Algorithms

Characterization and prediction of the endocrine disruptive potential of complex chemical mixtures are essential to prevent their adverse effects on human health while understanding the biological pathways that lead to such undesirable health outcomes [203]. A key target of endocrine-disrupting chemicals (EDCs) is the Estrogen Receptor (ER), a modulator of important physiological and pathological states, including reproduction, metabolism, hormone-sensitive cancers and obesity. There are many natural and man-made compounds that are capable of binding to the ER interfering with its activity, either as agonists, which activate a biological response (i.e., genistein, bisphenol A); or as antagonists, which generally compete with the endogenous hormones (i.e., 17β-Estradiol (E2)) to suppress the receptor function (i.e., 4-hydroxytamoxifen, fulvestrant). Mechanistically, E2 activates the ER pathway cascade through enabling a specific ER conformational change, receptor dimerization, DNA binding to regulatory elements in the genome, coregu-

lator recruitment and gene transcription activation/repression [204–206].

The estrogenic potential of different chemicals can be measured using cell-based or cell-free *in vitro* assays by recording several facets of the ER mechanism of action (i.e., ligand binding, cell proliferation, gene expression, etc.) [13, 198, 199]. Previously, a high content/high throughput microscopy-based assay in HeLa cells, engineered to harbor a visible multicopy integration of the ER responsive unit present within the prolactin promoter/enhancer, was developed to capture several mechanistic steps of the ER pathway by imaging [198–202]. Coupled with stable expression of GFP-ER, this high content analysis-based approach facilitates the characterization of ligands based upon their effect on ER activity when compared to known agonists and antagonists.

Furthermore, recent efforts have also focused on coupling high throughput experimentation with computational methods for enabling the rapid diagnosis of the estrogenic potential of various chemicals via *in silico* predictions [13, 207–212]. Judson et al. [13] used a linear model to predict the estrogenic activity of 1812 commercial and environmental chemicals based on the activity patterns across *in vitro* assays. The accuracy of this linear model is further tested by Browne et al. [209] for evaluating the ER agonist bioactivity, in which the authors postulated an integrated methodology to discriminate bioactivity from assay-specific interference. Similarly, Kleinstreuer et al. [211] used high throughput screening data of 1855 chemicals along with a linear additive model to predict the Androgen Receptor (AR) activity. Furthermore, Li and Gramatica [210] used AR data to develop quantitative structure-activity relationship (QSAR) models to classify binders as AR agonist or antagonist. The authors also investigated the performance of 4 different classification models, namely k-nearest neighbors (kNN), local lazy method (lazy IB1), alternating decision tree (ADTree) and an integrated consensus model [210]. In another study by Chierici et al. [212], deep learning and support vector machine (SVM) models were developed using the Collaborative Estrogen Receptor Activity Prediction Project (CERAPP) ToxCast dataset for predicting the effects of EDCs on ER binding activity. A further detailed overview of *in silico* toxicity predictions using machine learning algorithms is provided in the notable review by Idakwo et al. [213].

Different from the aforementioned studies, an integrated data-driven framework is presented

for characterizing the endocrine-disrupting potential of chemicals that affect ER functions. In this framework, the high throughput/high content image analysis data, which provides hundreds of intensity and geometry-based features per cell, are used to generate classification models for promptly detecting the endocrine disruptive potential of unknown compounds as ER agonists or antagonists. This approach is benchmarked using a group of control chemicals and presents a systematic computational approach for predicting the estrogenic potential of unknown chemicals. Furthermore, by incorporating feature selection steps in this framework, the most informative image-based features that enable a highly accurate separation between an ER agonist and antagonist are identified without the loss of generality.

6.2.1 Methodology

6.2.1.1 Benchmark Chemicals

Forty-five chemicals (Table 6.2) with varying estrogenic potentials were obtained from the United States Environmental Protection Agency (EPA) and were utilized for benchmarking the data-driven framework. The same compounds have been used by NIEHS/EPA as a set for developing computational models of the ER pathway [13].

6.2.1.2 Experimental Data Generation

High throughput microscopy and high content analysis-based experiments were performed using the GFP-ERα:PRL-HeLa cell line model following the experimental methodology described previously [198–202]. 384 multiwell plates were treated for 2 hrs with a six-point dose-response of 45 reference compounds provided by the EPA. Control compounds included the agonist 17β-estradiol (E2) and the antagonist 4-hydroxytamoxifen (4HT). Experiments with these compounds were repeated 8 times, resulting in 392 different observations (8 technical replicates of the controls and media, and 8 biological replicates for each of the 45 compounds) and 40 different image descriptors. The single-cell descriptors capture GFPα-ER fluorescence intensity (i.e., pixel intensity-PI) and morphology features of each cell, nucleus and PRL array that are identified using myImage Analysis (mIA) automated image analysis pipelines [214]. The single-cell population

Table 6.2: Summary of benchmark chemicals analyzed in this work. The ER activity information is adapted from Judson et al. [13].

CASRN	Compound Name	ER Activity [13]	Potency [13]
140-66-9	4-(1,1,3,3-Tetramethylbutyl)phenol	Agonist	Weak
599-64-4	4-Cumylphenol	Agonist	Weak
521-18-6	5α-Dihydrotestosterone	Agonist	Weak
57-91-0	17α-Estradiol	Agonist	Moderate
57-63-6	17α-Ethinyl estradiol	Agonist	Strong
58-18-4	17α-Methyltestosterone	Agonist	Very weak
50-28-2	17β-Estradiol	Agonist	Strong
520-36-5	Apigenin	Agonist	Very weak
85-68-7	Butylbenzyl phthalate	Agonist	Very weak
80-05-7	Bisphenol A	Agonist	Weak
77-40-7	Bisphenol B	Agonist	Weak
480-40-0	Chrysin	Agonist	Very weak
486-66-8	Daidzein	Agonist	Weak
117-81-7	Diethylhexyl phthalate	Agonist	Very weak
84-74-2	Di-n-butyl phthalate	Agonist	Very weak
115-32-2	Dicofol	Agonist	Very weak
56-53-1	Diethylstilbestrol	Agonist	Strong
53-16-7	Estrone	Agonist	Moderate
120-47-8	Ethylparaben	Agonist	Very weak
60168-88-9	Fenarimol	Agonist	Very weak
446-72-0	Genistein	Agonist	Weak
520-18-3	Kaempferol	Agonist	Very weak
143-50-0	Kepone	Agonist	Weak
84-16-2	meso-Hexestrol	Agonist	Strong
72-43-5	Methoxychlor	Agonist	Very weak
789-02-6	o,p'-DDT	Agonist	Weak
104-40-5	p-n-Nonylphenol	Agonist	Very weak
72-55-9	p,p'-DDE	Agonist	Very weak
68392-35-8	4-Hydroxytamoxifen	Antagonist	-
82640-04-8	Raloxifene Hydrochloride	Antagonist	-
10540-29-1	Tamoxifen	Antagonist	-
54965-24-1	Tamoxifen citrate	Antagonist	-
1912-24-9	Atrazine	Inactive	-
50-22-6	Corticosterone	Inactive	-
66-81-9	Cycloheximide	Inactive	-
13311-84-7	Flutamide	Inactive	-
52-86-8	Haloperidol	Inactive	-
52806-53-8	Hydroxyflutamide	Inactive	-
65277-42-1	Ketoconazole	Inactive	-
330-55-2	Linuron	Inactive	-
57-30-7	Phenobarbital sodium	Inactive	-
32809-16-8	Procymidone	Inactive	-
57-83-0	Progesterone	Inactive	-
50-55-5	Reserpine	Inactive	-
52-01-7	Spironolactone	Inactive	-

is filtered to remove artifacts generated from cell toxicity, cell clusters, and incorrectly segmented cells. The remaining cell population data is averaged per sample to yield a data matrix size of 392 observations x 40 features, where the categorical output information for classification is provided in Table 6.2 in the "ER Activity" column. A full list of experimental features is provided in [202].

6.2.1.3 Computational Methodology

The computational methodology follows a similar approach described in [215] where key steps of the framework are summarized in Figure 6.6. First, a series of pre-processing steps are executed to ensure accurate *in silico* predictions of ER activity with classification models. Once the pre-processing is completed, the dataset is then passed on to the feature selection phase, where collinear features are eliminated from the analysis using hierarchical clustering. Later, a two-class classification problem is formulated using a subset of the features that are identified as independent and biologically relevant in the previous step. Finally, model validation is performed, and the predictive capability of the resulting classification model is quantified using model performance metrics. A detailed description of each step is provided below.

Data Preprocessing

The pre-processing steps used in this analysis are: (1) missing data handling, (2) data cleaning, (3) outlier detection via unsupervised analysis, and (4) data normalization. The experimental data is first analyzed for missing data entries. If any missing data is detected, several procedures can be followed including, deletion of the entire row, deletion of the entire column, or data imputation [216]. As the experimental data from the image analysis did not have any missing points, no action is taken at this step and the data matrix size of 392 observations x 40 features are retained.

In the next preprocessing step, the dataset is cleaned by removing the observations corresponding to inactive compounds (Table 6.2) and technical replicates. After this cleaning step, the data matrix size is reduced to 256 observations x 40 features. For outlier detection, the replicate observations of each compound are averaged, yielding a data matrix size of 32 average observations x 40 features. Hierarchical clustering is performed on the Euclidean distance-based dissimilarity

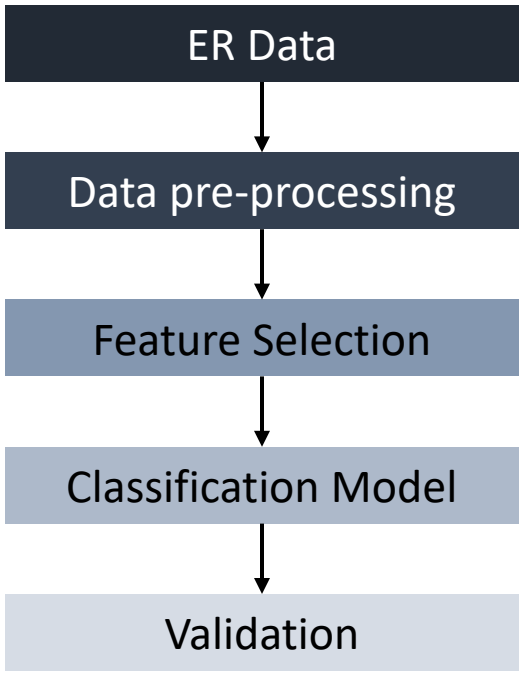

Figure 6.6: Classification framework for characterizing the estrogenic potential of chemical compounds.

matrix of this aggregate data with complete linkage. The clustering analysis is visualized using a dendrogram tree as shown in Figure 6.7. The results of the clustering analysis indicate that there are no global outliers present in the dataset as none of the compounds significantly differ from each other. It is observed that the active compounds are generally clustered under two groups based on their feature-specific patterns and are not presenting themselves on a separate branch at the root node of the dendrogram tree. As a result, the imaging data for all 32 compounds are viable for further analysis. The clustering is performed in R (version 3.6.0) using the "hclust" function under the "stats" library.

In the final preprocessing step, the remaining 32 active compounds are normalized using column-wise mean absolute deviation with respect to the control agonist E2 (Equation 6.8). The normalization is performed on the complete cleaned dataset with biological replicates (Data matrix size: 256 observations x 40 features). In Equation 6.8, i represents the rows in the dataset (i.e.,

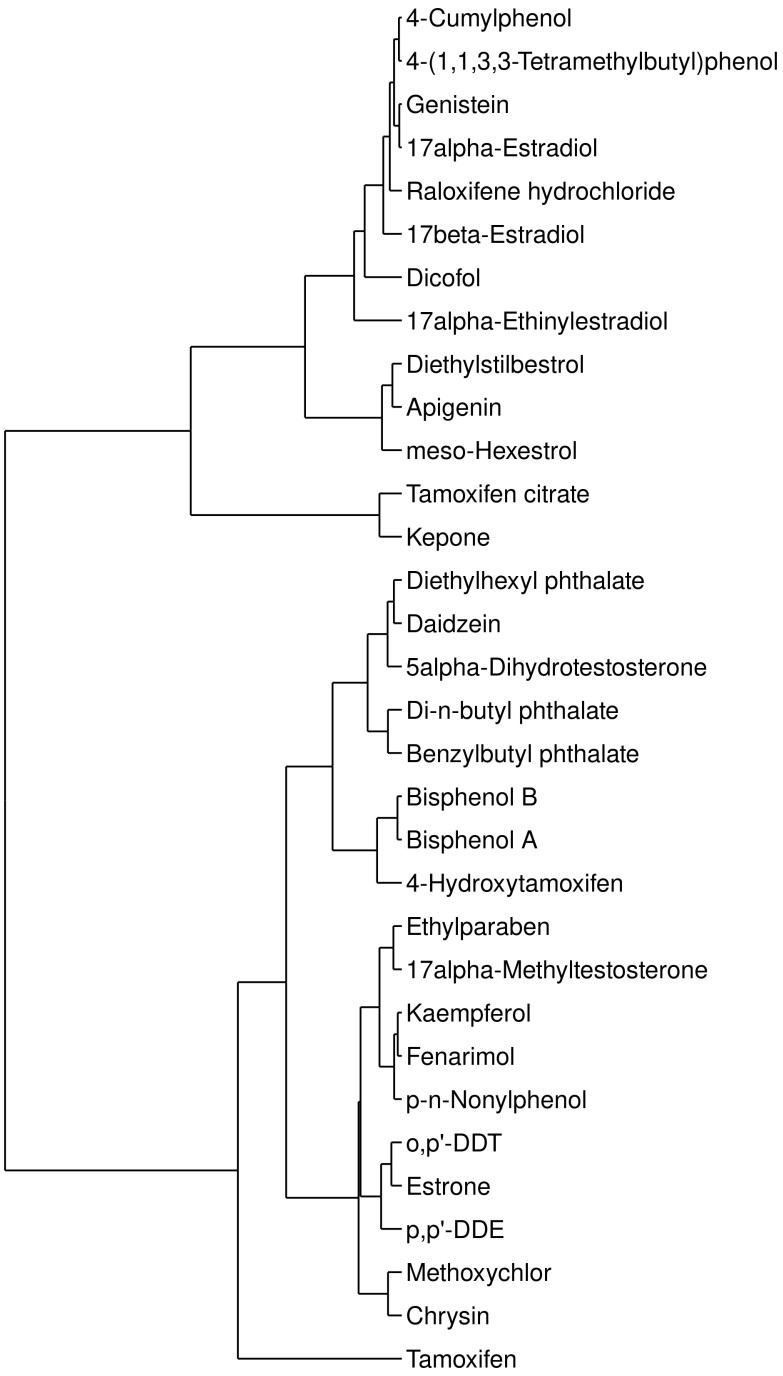

Figure 6.7: Outlier analysis via hierarchical clustering shown on a dendrogram tree.

observations) and j represents the columns in the dataset (i.e., features).

$$ER\ Data_{i,j}^{normal} = \frac{ER\ Data_{i,j}^{original} - median(E2_j)}{mean(|E2_{i,j} - mean(E2_j)|_j)} \qquad \forall i,j \tag{6.8}$$

Feature Selection

The ultimate goal of this work is to present a data-driven methodology that integrates high content, high throughput image analysis-based data with machine learning algorithms for developing a robust, generalized classification model that accurately predicts the estrogenic potential of unknown chemicals. Within the scope of this work, as the image analysis provides numerous fluorescence intensity and morphology features, several challenges come to rise in classification model development: (1) Only a subset of experimental features may provide valuable knowledge for the separation of agonist/antagonist ER activity and identification of those is a challenging task; (2) A subset of the features may be highly correlated, and may cause bias, leading to loss of generality, precision and accuracy in the predictive capability of the data-driven model; (3) Modeling with a high number of features without an adequate amount of samples may lead to overfitting.

In this work, the aforementioned challenges are addressed by incorporating a feature selection step in our data-driven modeling framework. Feature selection or variable selection is one of the key processes in machine learning model building, where the aim is to identify a subset of features among many others that are uncorrelated and the most informative set of descriptors, for a given data-driven modeling problem. There is a growing interest within various fields of engineering and sciences for developing computationally efficient feature selection algorithms that enable the identification of the minimum number of features for maximum predictive capabilities in data-driven models [158, 159, 217–219]. Here, the feature selection is done in two steps: (1) Through hierarchical clustering for identifying the groups of similar and correlated features, and (2) Through a heuristic feature selection step, in which a single feature is selected from each cluster based on the ER pathway model presented in [13].

In step 1, hierarchical clustering is performed on the pairwise similarity of experimental fea-

tures, calculated using the Pearson correlation, with complete linkage. From the clusters of correlated features, groups that possess less than 5% similarity are identified as unique and uncorrelated for classification analysis. The clustering outcome is shown in Figure 6.8 with 20 independent feature groups of which we can select a subset of these for analysis. Like in the outlier analysis, the clustering for feature selection is performed in R (version 3.6.0) using the "hclust" function under the "stats" library.

In step 2, the goal is to further reduce the number of features for the classification analysis such that they are: (1) selected from the independent groups of features following the clustering analysis (Figure 6.8); and, (2) the selected features are biologically relevant. The biological relevance of features is assessed through cross-referencing the image-based features to the ER pathway nodes presented in Judson et al. [13]. So, from 20 independent groups of features, the top 5 biologically relevant features (one shape and four PI-related descriptors) that are closely associated with a node on the ER signaling pathway, are selected. A summary of these features along with their descriptions are provided in Table 6.3. This selection yields a data matrix size of 256 observations x 5 features that are passed on to the model development stage of the presented framework.

Table 6.3: A subset of experimental features identified as uncorrelated and biologically significant for the classification analysis.

Feature Name	**Image-Based Property of the Feature**	**ER Pathway Node [13]**	**Biological Relevance**
Array Area	Size in pixels of visible promoter array	A4	Describes the chromatin remodeling of promoter array
Array Mean PI	Average intensity of the ER-GFP signal at the visible promoter array	A3, A5	Describes the level of ER-GFP binding to the promoter array
Array PI Variance	Statistical variance of ER-GFP pixel intensity at the visible promoter array	A3, A5	Describes the ER-GFP intensity distribution at the visible promoter array
Array Total PI	Total intensity of the ER-GFP signal at the visible promoter array	A3, A5	Describes the level of ER-GFP binding to the promoter array
Array to Nucleoplasm Intensity Ratio	Ratio of ER-GFP intensity at visible promoter array to ER-GFP intensity in the surrounding nucleoplasm	A5	Describes the efficiency of ER-GFP binding the promoter array

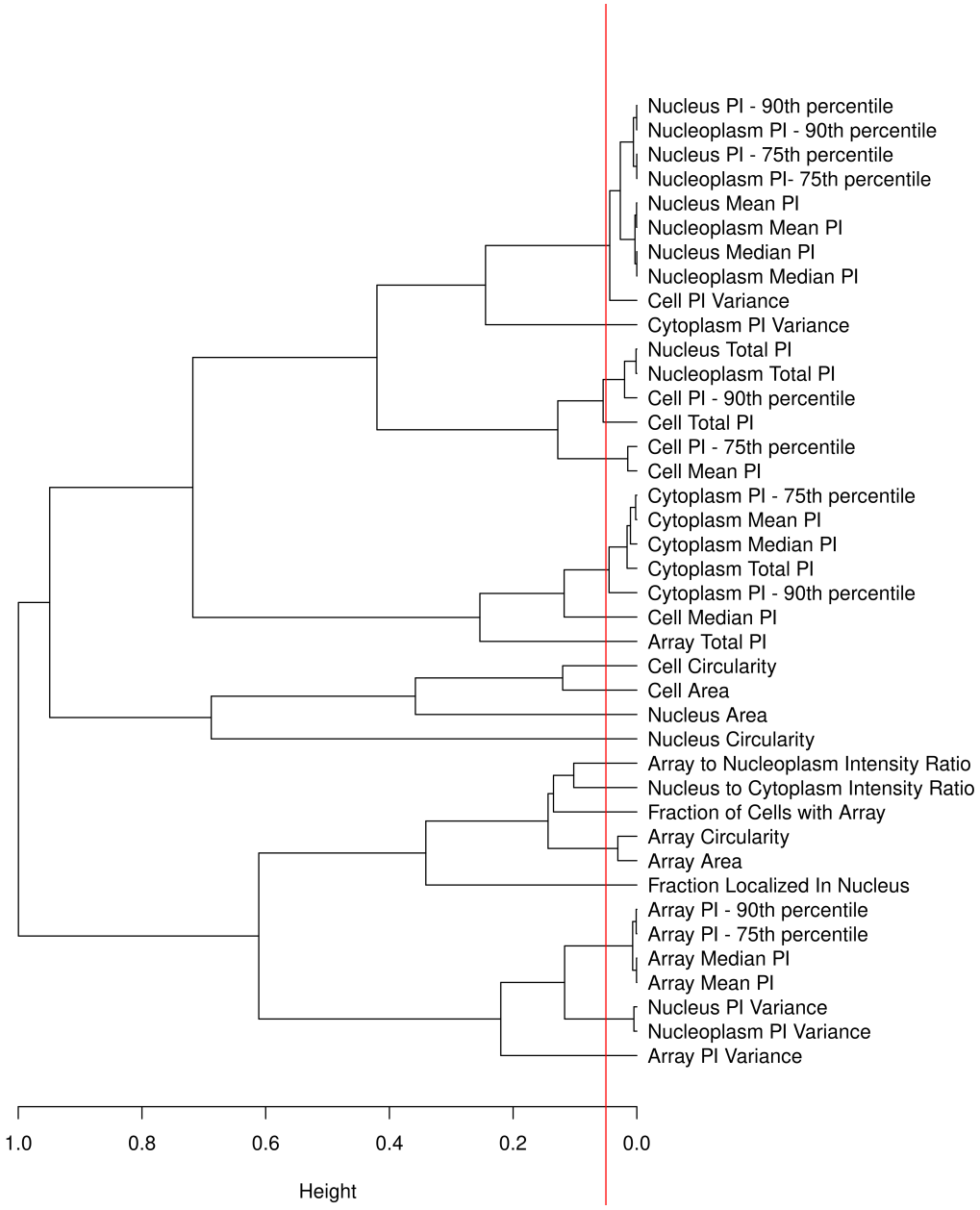

Figure 6.8: Uncorrelated feature selection using hierarchical clustering on pairwise feature similarity. The red line indicates the 5% similarity cutoff used for identifying independent feature groups.

Classification Model Development Using Logistic Regression and Random Forest Classifier

Once the feature selection step is completed, the clean normalized dataset is split into training, testing and validation sets, and the training data are passed on to the model building phase for supervised analysis. Supervised learning algorithms are widely studied in many fields of engineering and sciences primarily in classification and regression-type problems for predicting either a categorical output or a continuous output, respectively [30, 148, 151, 161, 218]. Classification is the problem of finding the categorical output of a new observation and distinguishing between different classes of information via statistical recognition of patterns in a training dataset. In this study, classification models are developed to predict the endocrine disruptor activity of a set of benchmark chemicals. In this effort, both linear and nonlinear models are tested and their predictive performance on unknown chemicals is shown for comparison.

Linear classification is performed using the logistic regression model and the variables are selected using the Akaike Information Criterion (AIC). The logistic regression model with one predictor is provided in Equation 6.9,

$$P(antagonist) = \frac{1}{1 + exp(-\beta_o - x \cdot \beta_1)} \tag{6.9}$$

where x is the value of a predictor, $P(antagonist)$ is the probability that the outcome is an "antagonist", and β_1 and β_o are the parameters of the linear model where their values are estimated using the training data. The goal of the logistic regression training stage is two-fold: (1) To create a highly accurate and precise linear separating boundary between ER agonist and antagonist compounds; and, (2) to identify the most descriptive feature out of the 5 selected in the feature selection step such that the *in silico* distinction between an ER agonist and antagonist is achieved without loss of generality. To this end, an exhaustive search is performed where individual logistic regression models are constructed for all possible combinations of single features. The best performing model in this training phase with the minimum AIC, the highest CV training accuracy, and the highest testing accuracy is selected. In addition, the most informative feature for the linear

classification problem is identified through analyzing the β_1 parameter given that $exp(-\beta_1)$ quantifies the increase in the odds of a compound being an antagonist. As a result, an important feature with a larger weight in the closed-form equation will have a larger impact on the classification predictions.

For nonlinear classification, the random forest (RF) algorithm is used with built-in feature ranking. RF is a nonparametric, tree-based ensemble learning method that uses multiple decision trees, independently constructed with a bootstrap sample of training data, to predict an outcome based on a majority vote [220]. The algorithm can identify "strong features" that causes a larger mean decrease in accuracy and display the relevance of features used in the training stage via the "Gini index" score [220, 221]. In this work, RF classifiers are constructed with 500 decision trees on the training data. The data-driven models for linear and nonlinear classification are implemented in R (version 3.6.0) using "glm" function in "stats" library, and "randomForest" function in "randomForest" library, respectively.

Model Validation and Performance Metrics

Model validation is done with the validation dataset that the model has not been trained or tested on. As a result, the validation set will enable the quantification of the unbiased predictive performance of the trained classification model. The classification model performances are assessed using several evaluation metrics. These include accuracy, sensitivity (i.e., true positive rate or recall), specificity and balanced accuracy. Definitions of accuracy and sensitivity are provided in Chapter 4 in Section 4.4.1. Specificity is defined as $\frac{TN}{TN+FP}$ and the balanced accuracy is defined as the average of sensitivity and specificity, $\frac{1}{2} \cdot \left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right)$. For this study, a "true positive" (TP) is defined as an agonist being correctly identified as an agonist and a "false positive" (FP) is defined as an agonist being misclassified as an antagonist. On the contrary, a "true negative" (TN) is defined as an antagonist being correctly classified as an antagonist and a "false negative" (FN) is defined as an antagonist being misclassified as an agonist. The classification results of the integrated data-driven framework are presented in the following section.

6.2.2 Results and Discussion

During and after environmental emergencies (i.e., hurricanes), humans are exposed to a number of chemicals, which in return creates an urgent need for the precise identification of their estrogenic potentials using rapid assessment techniques. Towards this goal, 18 experimental analyses are performed (each containing 392 observations x 40 features) on the 45 benchmark compounds for generating their high throughput, high content image analysis data, using the methodology described in Section 6.2.1.2. The aim is to construct robust, generalized data-driven models that can accurately predict the endocrine-disrupting potential of unknown compounds from a limited number of experimental observations. To this end, one experimental dataset is randomly selected for constructing the classification models among the 18 repeated image analysis experiments. The remaining 17 datasets serve as the validation sets and are reserved for quantifying the classification model performance of estrogenic potential of chemicals subject to experimental noise.

The selected dataset is first preprocessed, and the uncorrelated features are identified using the computational methodology described in Section 6.2.1.3. Then, the clean data is split into training and test sets. Although it is common to split the dataset using 80-20 or 70-30 rules (i.e., 80% training - 20% testing), the experimental analysis on the 45 benchmark chemicals yields an unbalanced dataset due to the limited number antagonist versus agonist compounds. Hence, five agonist compounds (Table 6.4) are randomly selected for analysis with varying potency such that the classification models are trained on data where the distinct characteristics of the two classes of estrogenic activity are learned precisely. Furthermore, identification of a balanced dataset is critical, as the primary goal of this study is to predict the endocrine-disrupting potential of unknown chemicals on the ER; and, the remaining active compounds serve as the test set, enabling fair assessment of the classification accuracy and other performance metrics. As a result, the final training data matrix size becomes 72 observations x 5 features, the final testing data matrix size becomes 184 observations x 5 features and the validation set matrix size which is comprised of the other experimental replicates is 17 experiments x 184 observations x 5 features.

Table 6.4: The agonist and antagonist compounds with varying ER potency selected for classification model training.

CASRN	Compound Name	ER Activity [13]	Potency [13]
115-32-2	Dicofol	Agonist	Very weak
56-53-1	Diethylstilbestrol	Agonist	Strong
53-16-7	Estrone	Agonist	Moderate
60168-88-9	Fenarimol	Agonist	Very weak
789-02-6	o,p'-DDT	Agonist	Weak
68392-35-8	4-Hydroxytamoxifen	Antagonist	-
82640-04-8	Raloxifene Hydrochloride	Antagonist	-
10540-29-1	Tamoxifen	Antagonist	-
54965-24-1	Tamoxifen citrate	Antagonist	-

6.2.2.1 Linear Classification Results

For the logistic regression model, the computational methodology described in Section 6.2.1.3 is followed. The five biologically relevant features that were previously identified in the feature selection step are used to construct individual linear classification models with a single descriptor. The best performing model is then selected out of these five logistic regression classifiers based on their AIC value, 5-fold training CV accuracy, and testing accuracy. The results of linear classification training with the logistic regression model are provided in Table 6.5.

Table 6.5: Linear classification model results with 1 experimental feature. The bootstrap confidence intervals (CI) for β_1 are presented alongside with AIC, training CV accuracy and testing accuracy results.

Experimental Feature in Model	β_1	95% CI of β_1	AIC	CV Accuracy	Testing Accuracy
Array to Nucleoplasm Intensity Ratio	7.12	(6.65, 7.44)	4.00	1.00	0.96
Array PI Variance	8.25	(5.05, 8.78)	4.00	1.00	0.87
Array Area	- 0.65	(-0.68, -0.60)	4.00	1.00	0.87
Array Mean PI	0.20	(0.14, 0.31)	51.83	0.87	0.84
Array Total PI	- 0.11	(-0.17, 0.06)	89.79	0.78	0.70

The results show that a logistic regression model with a single image analysis feature can accu-

rately map the separation between agonist and antagonist compounds in the training phase. Specifically, it is observed that linear classifiers trained with "Array PI Variance," "Array to Nucleoplasm Intensity Ratio" and "Array Area" descriptors can classify the compounds with 100% training CV accuracy. The linear models with "Array Mean PI" and "Array Total PI" features have an inferior training performance, as the AIC values for these two models are higher and the CV accuracies are lower compared to the other models. Furthermore, the results show that "Array Area" and "Array Total PI" features have a negative effect on the linear classifier whereas the rest of the features have a positive effect. Specifically, values of the β_1 parameter for "Array PI Variance" and "Array to Nucleoplasm Intensity Ratio" are the highest, respectively, indicating that a compound with higher values of these two features has an increased probability of being an antagonist. In addition, among these two most prominent features for the linear classification of estrogenic potentials of unknown chemicals, it is observed that the model parameters of "Array to Nucleoplasm Intensity Ratio" and "Array PI Variance" have a relatively wider range of 95% confidence intervals. Finally, the testing accuracy of trained models is evaluated using the remaining 23 active compounds in this experimental replicate. The testing accuracy results show that although "Array PI Variance" has a larger weight in the linear classifier compared to the rest of the descriptors, "Array to Nucleoplasm Intensity Ratio" has a higher testing accuracy for predicting the class information of the unseen chemicals. Table 6.5 shows that the linear classifier with "Array to Nucleoplasm Intensity Ratio" has a testing accuracy of 96% where this number drops to 87% when "Array PI Variance" is used as the sole predictor in the linear model. As a result, both predictors can perfectly map the separating linear boundary between the agonistic and antagonistic behaviors of chemicals in the training phase, whereas the linear model with "Array to Nucleoplasm Intensity Ratio" has a superior testing performance with a higher potential for achieving generality.

6.2.2.2 Nonlinear Classification Results

The nonlinear classification analysis results are summarized in Table 6.6 where it shows the ranking of the experimental features based on the mean decrease in the Gini index score. The mean decrease in Gini index score is a measure of how strong a feature is for separating different

classes of information, where prominent features lead to a larger decrease in this index. The results of the Random Forest (RF) model indicate that "Array to Nucleoplasm Intensity Ratio" is the top informative feature followed by "Array Area" and "Array PI Variance." The mean decrease in Gini index for these 3 descriptors are very close to each other, showing that they are equally important for modeling the estrogenic potential of chemicals. The nonlinear classification results are consistent with the linear model, where these 3 features had 100% training CV accuracy and minimum AIC. Through careful consideration of the model parameters and the testing accuracy in linear models, "Array to Nucleoplasm Intensity Ratio" and "Array PI Variance" are distinguished as the top two informative features for linear classification of agonist and antagonist compounds. Different than the linear analysis, it is observed that the "Array Area" is the second most important feature for the nonlinear classification of estrogenic compounds whereas in the linear model the second-best feature was identified as "Array PI Variance." Moreover, the initial model performance assessment with the training and testing data for the RF showed 100% and 93% classification accuracy, respectively. This high performance on the training data is expected as the model has learned the patterns within this set with high precision. The high testing accuracy of this model, on the other hand, shows that RF retains its predictive capability over a set of compounds that the model has not seen. As these initial tests show satisfactory results, further characterization of the model performance over different experimental replicates is provided in the following section.

Table 6.6: Experimental features ranked with respect to their mean decrease in the Gini index.

Experimental Feature	Mean decrease in Gini index
Array to Nucleoplasm Intensity Ratio	11.28
Array Area	11.26
Array PI Variance	10.15
Array Mean PI	1.94
Array Total PI	0.47

In addition to the classification model development and using their mathematical properties

to extract valuable information on the experimental features, additional insights on the separation between agonist and antagonist compounds are obtained through exploratory data analytics. To this end, the density distributions of agonist/antagonist compounds are plotted for all experimental replicates using the top important features identified by both linear and nonlinear classification analysis, namely the "Array to Nucleoplasm Intensity Ratio" and "Array PI Variance." The density plots are provided in Figure 6.9 and 6.10 where the separation between agonistic and antagonistic behaviors of the chemicals, based on the values of the aforementioned descriptors, are visualized. The results in Figure 6.9 and 6.10 show that the "Array PI Variance" and "Array to Nucleoplasm Intensity Ratio" lead to a clear distinction between an agonist and antagonist for all experimental replicates. To clearly distinguish between these two prominent features, the separation between the agonist and antagonist density distributions are quantified by calculating the Hellinger Distance (HD). This metric provides a measure of the distance between probability distributions and takes the values between 0 and 1, where smaller HD indicates that the two distributions are similar and the separation between them through the use of this feature is not statistically significant.

In Figure 6.9, the results show that the HD between the density distributions of agonist and antagonist compounds based on the "Array to Nucleoplasm Intensity Ratio" is high (min = 0.63, max = 0.90). Specifically, for experimental replicates 3, 5, 9, 11, 15 and 17 there is a clear separation between agonistic and antagonistic behavior based on this descriptor, hence a linear classifier enables a highly accurate separation between these two estrogenic potential classes. However, in experimental replicates 4, 10 and 16, a portion of the density distributions of these two estrogenic activities overlap and may lead to misclassification of compounds if the normalized "Array to Nucleoplasm Intensity Ratio" value of an agonist/antagonist fall into this overlapping region. Overall, it is observed that ER antagonists possess strong signals for this experimental feature, thus enabling the separation of these two classes via a linear logistic regression model.

Similarly, in Figure 6.10, it is observed that the HD distance of "Array PI Variance" measurements for all experimental replicates is high (min = 0.66, max = 0.87), indicating that this feature is a valid descriptor for agonist versus antagonist separation. Different from Figure 6.9, in this

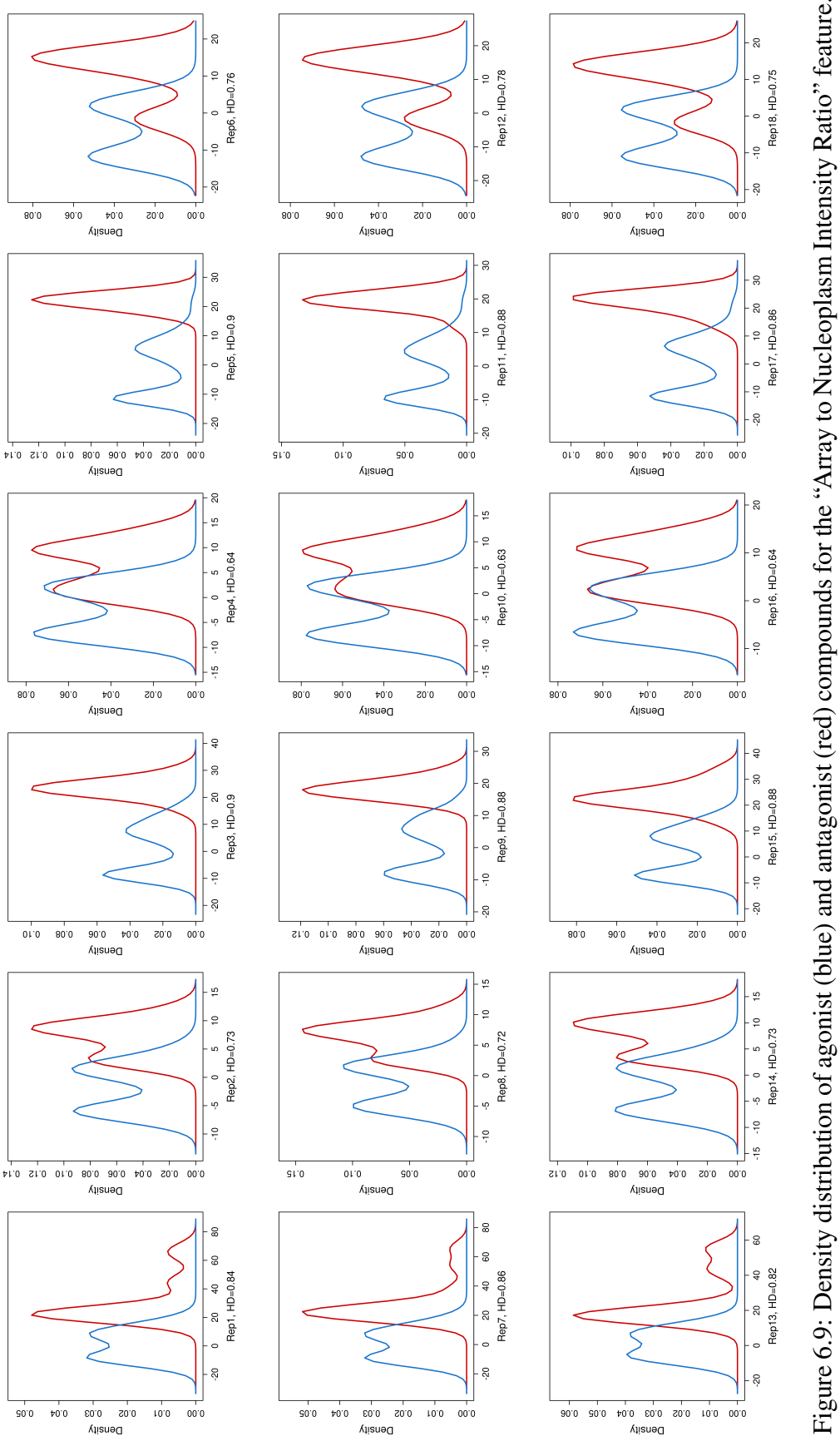

Figure 6.9: Density distribution of agonist (blue) and antagonist (red) compounds for the "Array to Nucleoplasm Intensity Ratio" feature.

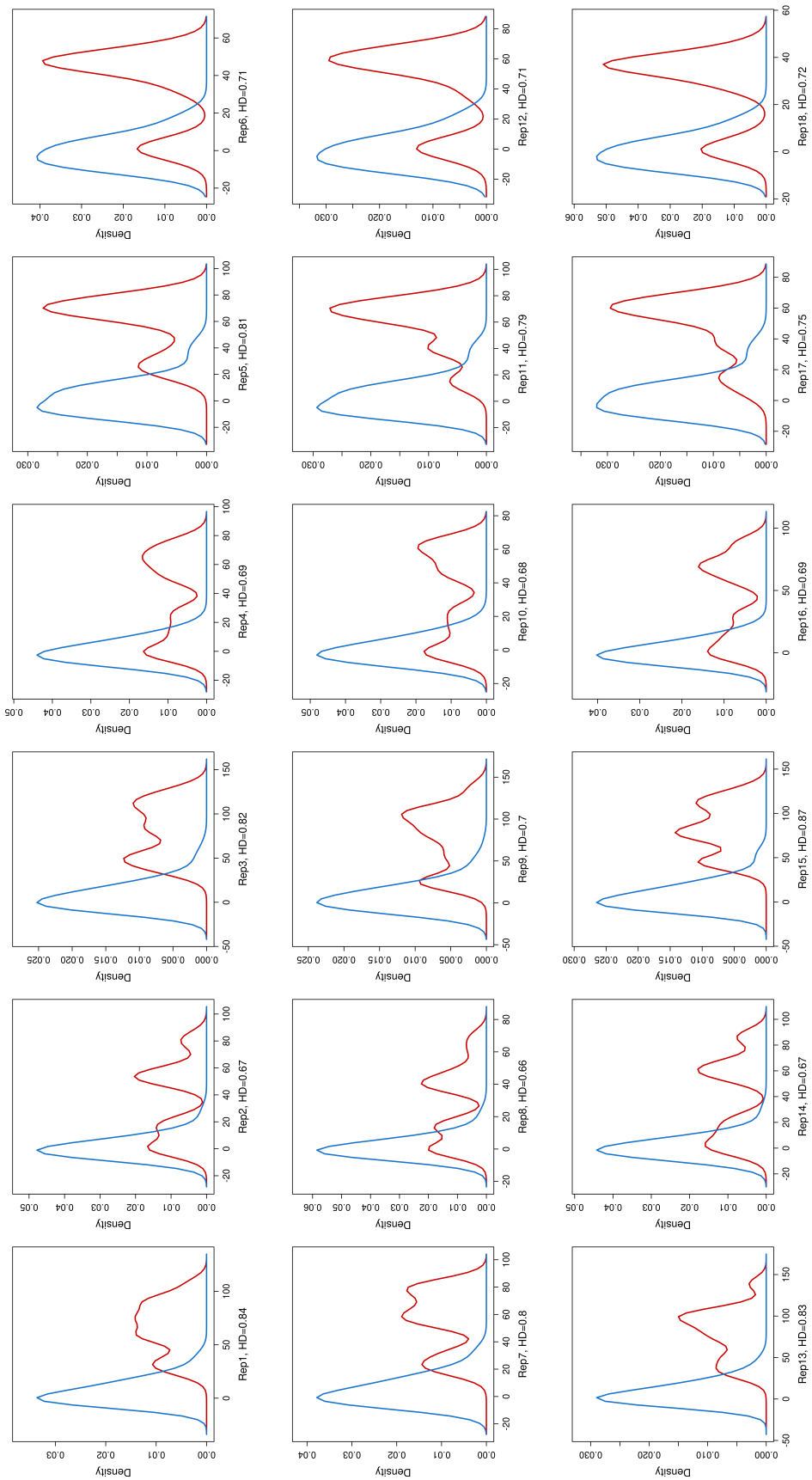

Figure 6.10: Density distribution of agonist (blue) and antagonist (red) compounds for the "Array PI Variance" feature.

case it is observed that ER agonists have strong signals for this experimental feature in 12 out of 18 experimental replicates. Furthermore, although the range of HD values are similar for both features, the number of experimental replicates with HD > 0.8 is higher for "Array to Nucleoplasm Intensity Ratio" compared to "Array PI Variance." This result indicates that "Array to Nucleoplasm Intensity Ratio" is more favorable for classification model building as this feature provides a clear distinction between ER agonists and antagonists over multiple experimental replicates. Overall, visualization results are also consistent with findings in the supervised analysis phase, essentially validating the importance of "Array to Nucleoplasm Intensity Ratio" for robust and precise modeling of estrogenic potentials of compounds using high throughput microscopy and high content analysis-based assays.

6.2.2.3 Model Validation Results

In addition to the HD calculations, the predictive capabilities of the trained and tested logistic regression (linear) and RF (nonlinear) classifiers are validated with a set of new experimental replicates, and their corresponding predictive performance is quantified. The results for the model performance evaluation are presented in Figure 6.11 and Tables 6.7-6.9. In Figure 6.11, the model performance is quantified using 17 experimental replicates comprised of 24 agonist compounds that the model has not been trained on, hence allowing us to test the prediction accuracy of the constructed models. In Tables 6.7-6.9, an overall accuracy, sensitivity, and specificity of different classification models are reported for the same 17 experimental replicates, but with all 32 active compounds.

The "blind" validation accuracy results in Figure 6.11 shows that the predictive performance of the logistic regression model with "Array PI Variance" feature is inferior to the logistic regression model with "Array to Nucleoplasm Ratio" and the RF classifier. In 4 out of 17 experimental replicates, the validation accuracy of this model is below 80% whereas, for the logistic regression model with "Array to Nucleoplasm Ratio" and the RF classifier, the validation accuracies exceed 90% for all experimental replicates. Furthermore, the predictive capability of the latter two models is comparable to each other, where only in experimental replicates 1 and 6, a relatively high

difference between the validation accuracies of these two models is observed. In replicates 1 and 6, the validation accuracy of the RF model is 95% and 97%, respectively, whereas for the logistic regression model with "Array to Nucleoplasm Ratio" has a validation accuracy of 91% for both replicate testing. The lowest prediction accuracy for these two models is 95% for the RF classifier and 91% for the logistic regression model with "Array to Nucleoplasm Ratio" whereas this number drops to 75% for the logistic regression model with "Array PI Variance." Moreover, the 95% CI of the validation accuracy is also provided in Figure 6.11. The RF and logistic regression models using "Array to Nucleoplasm Ratio" have tighter CI around the validation accuracy whereas the other logistic regression model has a wider CI on the prediction accuracy for all experimental replicates. Overall, the blind validation accuracy results indicate that the nonlinear RF and linear logistic regression models with "Array to Nucleoplasm Ratio" are more favorable for predicting the estrogenic potential of unknown compounds as they have a more robust performance and can sustain their predictive capabilities over multiple experimental replicates. The other performance metrics are also computed on all the active compounds, and their analysis is provided in Tables 6.7-6.9.

The results in Tables 6.7-6.9 indicate that all three classification models have high accuracy and sensitivity for predicting the estrogenic potential of all active compounds considered in this study. Specifically, all models predict > 90% accuracy in 11 out of 17 experimental replicates. Moreover, it is observed that specificity and balanced accuracy of the logistic regression model with "Array PI Variance" is higher overall, when compared to other models, whereas the sensitivity of the RF classifier and logistic regression model with "Array to Nucleoplasm Ratio" is higher across different replicates. For the latter two models, it is observed that the specificity value is 0 for 5 experimental replicates, indicating that these models identified all compounds as agonists and failed to classify the 4 antagonist compounds correctly. As a result, their balanced accuracy is also lower (50%), as this performance metric averages specificity and sensitivity values.

The biphasic performance of each model on the identification of antagonist compounds suggests an underlying feature of the replicate datasets that determines model performance. An ex-

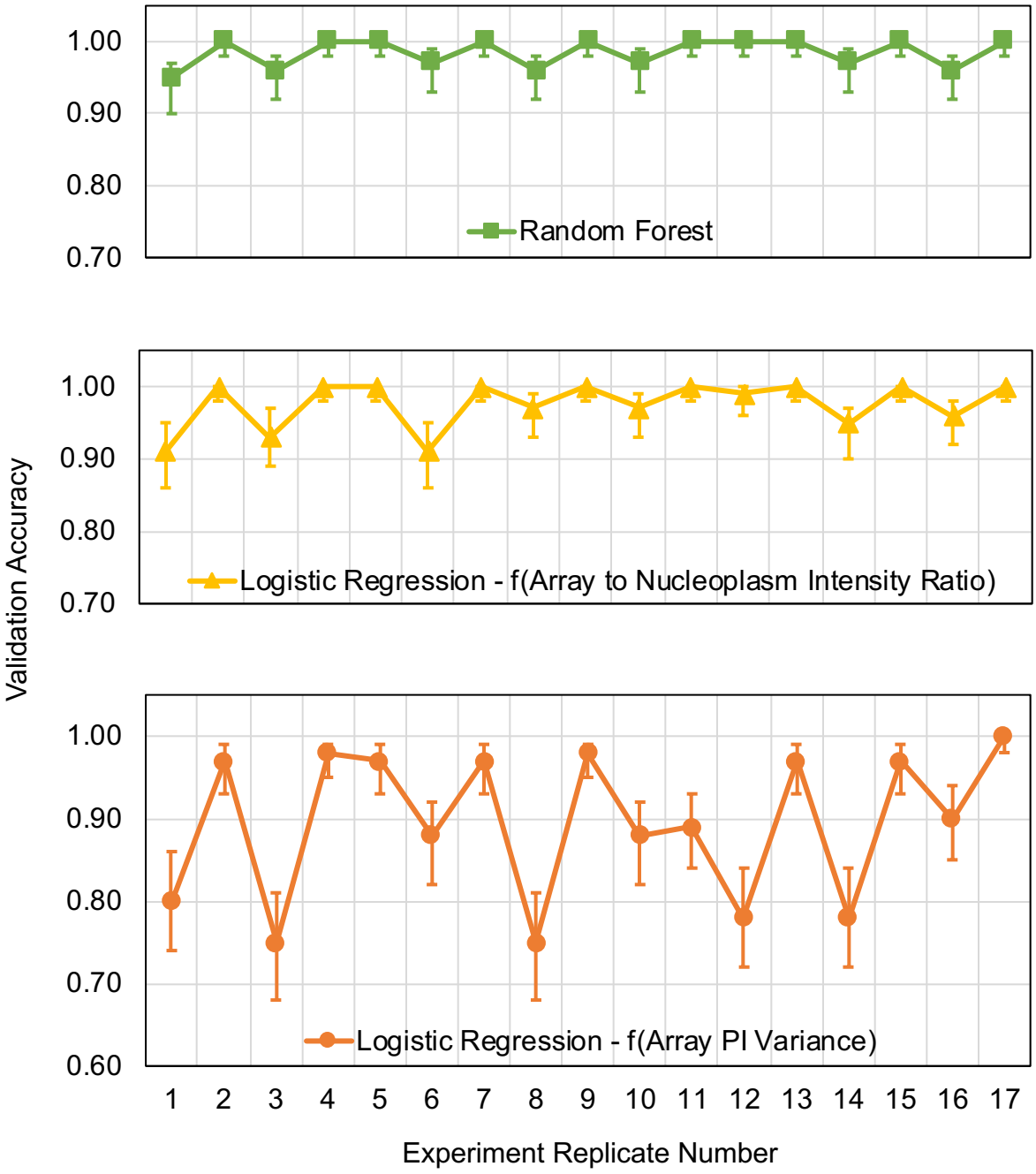

Figure 6.11: Model validation results with 24 unseen agonist compounds over 17 experimental replicates for the logistic regression model as a function of "Array PI Variance", the logistic regression model as a function of "Array to Nucleoplasm Intensity Ratio" and the Random Forest classifier.

Table 6.7: Logistic regression model validation results with all active compounds for 17 experimental replicates with "Array PI Variance" as the model predictor.

Experimental Replicate	Overall Accuracy	95% CI	Sensitivity	Specificity	Balanced Accuracy
1	0.84	(0.79, 0.89)	0.82	1.00	0.91
2	0.93	(0.89, 0.96)	0.97	0.62	0.80
3	0.80	(0.74, 0.84)	0.77	1.00	0.88
4	0.94	(0.90, 0.96)	0.98	0.62	0.80
5	0.95	(0.91, 0.97)	0.97	0.75	0.86
6	0.91	(0.87, 0.95)	0.90	1.00	0.95
7	0.92	(0.88, 0.95)	0.97	0.56	0.77
8	0.80	(0.74, 0.84)	0.77	1.00	0.88
9	0.94	(0.90, 0.96)	0.98	0.62	0.80
10	0.91	(0.86, 0.94)	0.90	0.94	0.92
11	0.89	(0.85, 0.93)	0.91	0.75	0.83
12	0.82	(0.77, 0.87)	0.79	1.00	0.90
13	0.93	(0.89, 0.96)	0.97	0.62	0.80
14	0.82	(0.77, 0.87)	0.79	1.00	0.90
15	0.93	(0.89, 0.96)	0.97	0.62	0.80
16	0.91	(0.87, 0.95)	0.92	0.88	0.90
17	0.97	(0.94, 0.99)	1.00	0.75	0.88
Average	**0.89**	**-**	**0.90**	**0.81**	**0.86**

Table 6.8: Logistic regression model validation results with all active compounds for 17 experimental replicates with "Array to Nucleoplasm Intensity Ratio" as the model predictor.

Experimental Replicate	Overall Accuracy	95% CI	Sensitivity	Specificity	Balanced Accuracy
1	0.92	(0.88, 0.95)	0.91	1.00	0.96
2	0.88	(0.83, 0.91)	1.00	0.00	0.50
3	0.94	(0.90, 0.96)	0.93	1.00	0.96
4	0.88	(0.83, 0.91)	1.00	0.00	0.50
5	0.95	(0.91, 0.97)	1.00	0.56	0.78
6	0.93	(0.89, 0.96)	0.92	1.00	0.96
7	0.88	(0.83, 0.91)	1.00	0.00	0.50
8	0.96	(0.93, 0.98)	0.97	0.88	0.92
9	0.88	(0.83, 0.91)	1.00	0.00	0.50
10	0.97	(0.94, 0.99)	0.97	0.94	0.96
11	0.95	(0.92, 0.98)	1.00	0.62	0.81
12	0.98	(0.95, 0.99)	0.99	0.88	0.93
13	0.88	(0.83, 0.91)	1.00	0.00	0.50
14	0.94	(0.90, 0.96)	0.94	0.94	0.94
15	0.89	(0.85, 0.93)	1.00	0.12	0.56
16	0.96	(0.93, 0.98)	0.96	0.94	0.95
17	0.91	(0.87, 0.95)	1.00	0.31	0.66
Average	**0.92**	**-**	**0.98**	**0.54**	**0.76**

Table 6.9: Random Forest model validation results with all active compounds for 17 experimental replicates.

Experimental Replicate	Overall Accuracy	95% CI	Sensitivity	Specificity	Balanced Accuracy
1	0.95	(0.91, 0.97)	0.94	1.00	0.97
2	0.88	(0.83, 0.91)	1.00	0.00	0.50
3	0.95	(0.92, 0.98)	0.95	1.00	0.97
4	0.88	(0.83, 0.91)	1.00	0.00	0.50
5	0.95	(0.91, 0.97)	1.00	0.56	0.78
6	0.98	(0.95, 0.99)	0.97	1.00	0.99
7	0.88	(0.83, 0.91)	1.00	0.00	0.50
8	0.95	(0.92, 0.98)	0.96	0.88	0.92
9	0.88	(0.83, 0.91)	1.00	0.00	0.50
10	0.97	(0.94, 0.99)	0.97	0.94	0.96
11	0.95	(0.92, 0.98)	1.00	0.62	0.81
12	0.98	(0.96, 1.00)	1.00	0.88	0.94
13	0.88	(0.83, 0.91)	1.00	0.00	0.50
14	0.95	(0.92, 0.98)	0.96	0.94	0.95
15	0.89	(0.85, 0.93)	1.00	0.12	0.56
16	0.95	(0.92, 0.98)	0.96	0.88	0.92
17	0.91	(0.87, 0.95)	1.00	0.31	0.66
Average	**0.93**	**-**	**0.98**	**0.54**	**0.76**

haustive analysis of the correlation between dataset features and model performance identified cell density (number of cells per microscopic image) as having a strong negative correlation (- 0.54 to -0.71) with model balanced accuracy. Cell density varies by 12.6% across replicates (Figure 6.12A). Using a threshold of 257.5 cells/well to divide replicates into "Low Density" and "High Density", it is observed that the superior performance by all three models in the "Low Density" replicates (Figure 6.12B) with average balanced accuracy exceeding 0.9 or 0.95. This result is not surprising based on the technical limitations of the assay and features used by the model. Increasing cell density increases the likelihood of any individual cell in a field being slightly out of focus. Since both "Array PI Variance" and "Array to Nucleoplasm Ratio" are contrast based features, they are dependent on focus quality. While the original optimization of cell density was based simply on the ability to detect the presence of a nuclear spot, slightly lower cell densities may be required to produce higher quality data required for high classification performance.

Overall, the nonlinear classification model, namely the RF algorithm, and the logistic regression model with "Array to Nucleoplasm Ratio" are found to be highly accurate and robust for predicting the endocrine-disrupting potential of compounds on the ER. Among 40 different experimental features studied in this work, "Array to Nucleoplasm Ratio" is found to be the top informative feature through a series of supervised and unsupervised analyses, and the results indicate that it is essential for predicting the ER activity of compounds through generalized predictive models.

6.3 Concluding Remarks

In this chapter, data-driven modeling and exploratory data analytics are used for: (1) understanding the redistribution of toxic chemical compounds after natural disasters (i.e., hurricanes); and, (2) characterizing the biological effects of toxic chemical compounds on human health. In Section 6.1, various environmental datasets are visualized and analyzed using exploratory data analytics. The results of this study indicate that boxplots and heatmaps are complementary to each other when visualizing environmental datasets, where boxplots provide information on the overall distribution whereas heatmaps display sample-specific patterns. Furthermore, clustering analysis

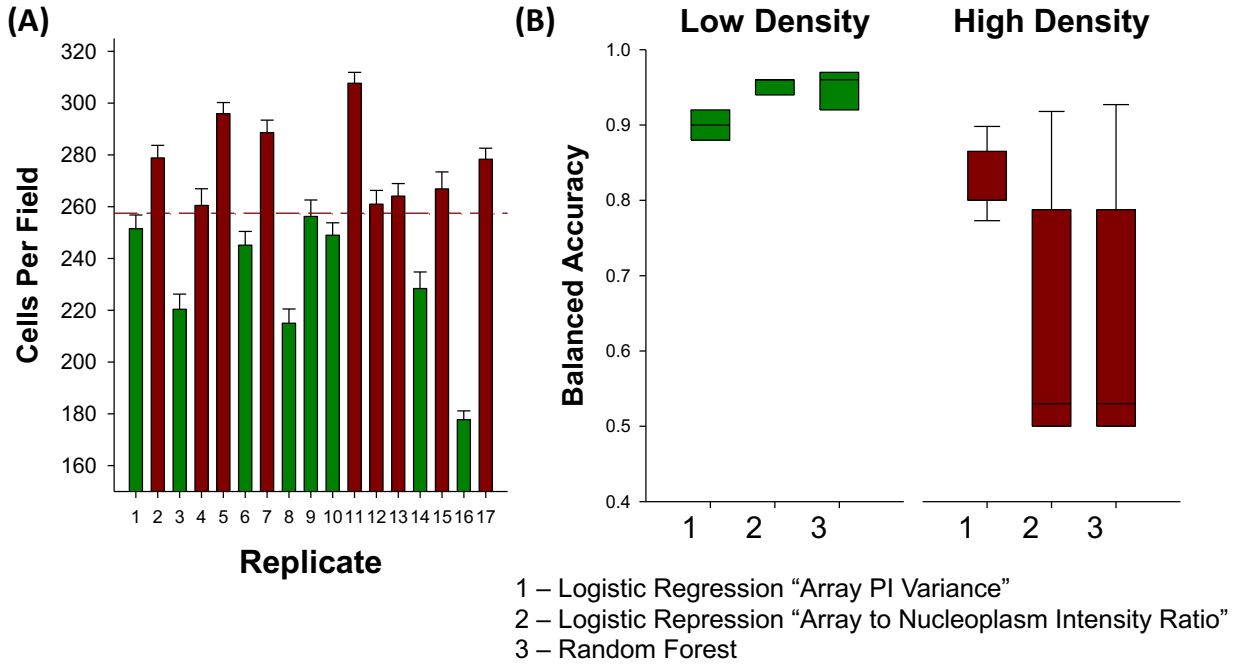

Figure 6.12: Cell density negatively affects model performance. (A) The average number of cells per microscopic field for replicates. The dashed line indicates the threshold for low- and high-density replicates. (B) Box plots of model balanced accuracy performance observed in low- and high-density replicates.

is performed to investigate the grouping patterns of the sediment samples with respect to their pollutant concentrations. These results are also compared to the grouping of sediment samples based on their geospatial locations and the similarities between pollutant concentration patterns and geospatial location of the sediment samples are investigated. The similarity between two different groupings is evaluated using the Fowlkes-Mallows (FM) index and the statistical significance of the results is assessed under the null hypothesis. The clustering results show that the detected concentrations of environmental pollutants do not group similarly with respect to their geospatial locations, indicating that there is no point source of contamination. The statistical significance of results is further investigated by two tests: (1) Null FM index calculation; and, (2) the Mantel test. The results of these statistical tests indicate that the true value of the FM index for all pollutants is equal or worse than the null FM index and the p-value of the permuted results is high. As a result, the null hypothesis cannot be rejected which confirms that the observed grouping similarities are

due to the random arrangement.

In Section 6.2, an integrated data-driven framework is developed that enables the rapid identification of unknown pure chemicals that affect the estrogen receptor (ER) pathway as either agonists or antagonists. High throughput microscopy and high content analysis-based data are utilized to formulate highly accurate classification models by following a series of preprocessing, visualization, unsupervised, and supervised analysis steps. The framework is benchmarked with a set of chemicals with known ER activity. In the presented framework, a detailed preprocessing step is executed where: (1) experimental image analysis data is scanned for missing data points; (2) data is cleaned by removing the inactive compounds; (3) dataset outliers are detected via hierarchical clustering; and, (4) experimental features are normalized via mean absolute deviation. Following preprocessing, the framework continues with a two-step feature selection methodology where uncorrelated features are first identified by hierarchical clustering using the pairwise similarity of the descriptors; secondly, the biologically relevant descriptor(s) are selected for analysis. Both linear and nonlinear classifiers are tested as a part of this framework for modeling endocrine-disrupting potentials of chemicals that affect ER functions, and their predictive performances are quantified via evaluation metrics. The linear and nonlinear classification model results show that high throughput microscopy and high content analysis-based experimental data can be used to train robust, highly accurate classifiers with a minimum number of features and sampling points (i.e., one feature for linear classification and five features for nonlinear classification). Through the results of this framework, one can identify the topmost important feature for the classification of ER agonists/antagonists without loss of generality and provide recommendations for the appropriate model selection. In addition, the presented data-driven framework serves as a guideline for rapidly scanning unknown chemicals and obtaining their estrogenic potentials with high accuracy. This property of the framework will be profound during environmental emergencies, where it is of the utmost importance to rapidly identify the potential biological risks of unknown chemicals.

7. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

7.1 Conclusions

In this dissertation, theoretical and algorithmic advances are presented for a number of challenging classes of optimization problems, including: (1) Bi-level mixed-integer nonlinear programming; (2) general constrained multi-objective programming; (3) stiff differential algebraic equations; and, (4) general constrained nonlinear nonconvex programming. Furthermore, this dissertation addresses the key modeling challenges in environmental and biomedical systems using advanced data analytics methods.

In Chapter 2, the Data-driven Optimization of bi-level Mixed-Integer NOnlinear problems (DOMINO) framework is established for the optimization of bi-level mixed-integer nonlinear programming (B-MINLP) problems. In this framework, bi-level optimization problems are approximated as single-level optimization problems by collecting samples of the upper-level objective and solving the lower-level problem to global optimality at those sampling points. The accuracy, consistency, and the computational performance of DOMINO are extensively investigated on 100 benchmark problems, consisting of bi-level linear, continuous nonlinear and mixed-integer programming problems, and on a bi-level mixed-integer nonlinear land allocation problem in Food-Energy-Water Nexus. The results of the computational studies show that the DOMINO framework can identify the true global solution or a near-optimal solution for an extensive set of challenging bi-level optimization problems.

Chapter 3 presents an algorithmic advancement for solving a special class of problems under mathematical programming that entails multiple competing objectives. The presented framework uses the ϵ-constraint method to convert a multi-objective optimization problem into a series of single objective sub-problems and uses a global constrained grey-box optimization algorithm to retrieve the optimal solution at each sub-problem. Computational results are reported for a number of benchmark multi-objective problems and a case study of an energy market design prob-

lem for a commercial building, while the performance of the framework is compared with other derivative-free optimization solvers. The results show that ARGONAUT can consistently and efficiently identify the Pareto-frontier, which entails all the trade-off solutions that are equally optimal with respect to each other, under varying conditions and dimensions of constrained multi-objective problems.

Furthermore, Chapter 4 proposes a Support Vector Machines (SVMs) based optimization framework for the data-driven optimization of stiff Differential Algebraic Equations (DAEs) without the full discretization of the underlying first-principles model. By formulating the stability constraint of the numerical integration of a stiff DAE system as a supervised classification problem, it is demonstrated that SVMs can accurately map the feasible boundary of stiffness. The fundamental idea behind this integrated approach is demonstrated on a 2-dimensional motivating example where the SVM approximation of the stability constraint is shown to achieve high validation accuracy. Further, this approach is extended and tested on more challenging case study, namely the thermal cracking of natural gas liquids.

In Chapter 5, new computational developments in the ARGONAUT framework are highlighted and the performance of the new parallel algorithm (p-ARGONAUT) is presented on a challenging nonlinear nonconvex programming case study of oil-well control operations using water-flooding. Through this work, it is shown that high-performance computing can be used to reduce the computational cost of the ARGONAUT framework significantly, which leads to also extending its capabilities towards solving high-dimensional, highly constrained problems. Data-driven approximations are used within two steps of this work: (a) the reduction of the dimensionality of the water-flooding optimization problem via parametrization of the control domain; and, (b) the optimization of simulation-based grey-box problems through the p-ARGONAUT framework.

Finally, in Chapter 6, the redistribution of toxic chemical compounds due to natural disasters (i.e., hurricanes) and their corresponding biological effect on human health due to chemical exposure is investigated using exploratory data analytics and data-driven modeling. Exploratory data analytics is employed to investigate the redistribution of contaminated soil samples, collected after

the Hurricane Harvey hit the Galveston coastline within the Manchester, TX area. The sediment samples are experimentally characterized and the resulting datasets are visualized using boxplots and heatmaps, and the correlations between geospatial locations of the sediments and the detected pollutant concentrations are investigated. Moreover, the biological impact of several benchmark chemicals is explored, as many environmental toxicants affect human health in various ways. In this chapter, a high content image analysis and machine learning pipeline are presented for rapid, accurate and sensitive assessment of the endocrine-disrupting potential of benchmark chemicals. The results of this data-driven study show that highly accurate and generalized classification models with a minimum number of features can be constructed without the loss of generality. The presented data-driven framework serves as a guideline for rapidly scanning unknown chemical compounds and obtaining their estrogenic potential with high accuracy. This property of the framework will be profound during environmental emergencies, where identifying the potential biological risks of chemical compounds is of utmost importance.

Overall, data-driven hybrid modeling and optimization tools presented in this dissertation address special classes of mathematical programming problems and key modeling challenges in environmental and biomedical systems. These computational tools are used for finding solutions to a diverse set of problems faced in the engineering and sciences domain including, food-energy-water nexus considerations, energy systems design with economic and environmental considerations, thermal cracking of natural gas liquids, oil production optimization, pollutant redistribution in environmental studies, and the biological impact of toxic chemicals in biomedical sciences.

7.2 Key Contributions

Key contributions of this dissertation are provided below.

1. DOMINO framework is presented as an algorithmic advancement for solving bi-level mixed-integer nonlinear programming problems with guaranteed feasibility when the lower-level problem is solved to global optimality at convergence. A novel data-driven approach is followed to approximate a bi-level optimization problem into a single-level problem, where the upper-level decision variables are used to simulate the optimality of the lower-level problem

(Chapter 2).

2. A hybrid framework is introduced for the optimization of general constrained multi-objective programming problems. This framework integrates the ϵ-constraint methodology with a constrained grey-box optimization solver for the reformulation of multi-objective optimization problems into series of single objective sub-problems and for their respective optimization through a data-driven methodology (Chapter 3).

3. A theoretical advancement is presented for the data-driven optimization of stiff Differential Algebraic Equations (DAEs) without the full discretization of the underlying first-principles model. Support Vector Machines (SVMs) are used to formulate the stability constraint of the numerical integration of a stiff DAE system as a nonlinear two-class classification problem. Through this approach, high-quality solutions are generated by rapidly eliminating the numerically unstable variable combinations, thus allowing the exploration of a wider range of decision variable space for improved solutions (Chapter 4).

4. A new parallel version of the ARGONAUT algorithm (p-ARGONAUT) is introduced for solving high-dimensional highly constrained nonlinear programming problems. Through this work, it is shown that high-performance computing can be used to reduce the computational cost of the ARGONAUT framework significantly, which leads to also extending its capabilities towards solving high-dimensional, highly constrained problems (Chapter 5).

5. An effective data-driven methodology is presented for understanding the redistribution of toxic chemical compounds after being mobilized via natural disasters, and for characterizing the biological effects of these compounds on human health due to chemical exposure. First, through a systematic study of various visualization and data analysis techniques, it is shown that the potential pathways of environmental pollutant redistribution can be effectively communicated, interpreted and diagnosed using exploratory data analytics in a post-hurricane event. Second, through developing an integrated data-driven framework, the endocrine-disrupting potential of chemical compounds is characterized using two-class classification

models which enable the rapid evaluation of the estrogenic potential of many chemical compounds (Chapter 6).

7.3 Future Work

7.3.1 Data-Driven Bi-level Optimization for Integrated Planning and Scheduling

The DOMINO algorithm presented in Chapter 2 can be extended to handle multiple followers at the lower-level and utilized to solve integrated planning and scheduling problems. The production planning and scheduling formulations typically use a sequential approach, where higher level decisions are made first (i.e., planning) and implemented at the lower level (i.e., scheduling). However, the sequential approach may lead to an infeasible lower-level solution given that there is a natural hierarchy between different levels of planning. This natural hierarchy between planning and scheduling problems can be formulated as a bi-level programming problem where the DOMINO algorithm can be used to solve this challenging optimization problem. The bi-level formulation of the integrated planning and scheduling problem will consider the minimization of the total cost of planning subject to inventory and balance equations at the upper-level. The lower-level will entail the scheduling problem where the cost for each planning period is minimized subject to the scheduling constraints. The production targets for the given products in an integrated planning and scheduling problem will be the inputs to the data-driven algorithm where each schedule is solved to global optimality over the entire planning period. The total cost and inventories at each schedule will be accounted and the total cost of planning will be calculated (output data). This input-output data can be used by DOMINO to find the optimal allocation of inventories and assignment of production goals to meet the demand in products. Further extensions to the conventional formulation are also possible. For example, planning and scheduling variable costs are typically approximated as linear functions whereas, in reality, the variable cost is nonlinear in nature. As DOMINO can efficiently handle nonlinearities in the problem formulation through its data-driven strategy, quadratic and cubic variable cost functions can also be explored. As DOMINO can provide guaranteed feasibility of a given bi-level programming problem, it can provide feasible solutions to integrated planning and scheduling problems.

7.3.2 Extensions to the DOMINO Algorithm for Solving Tri-level Mixed-Integer Programming Problems

The DOMINO framework, presented in Chapter 2, can be extended to handle tri-level mixed-integer programming problems. The tri-level problems can be reformulated into a single-level problem by integrating the DOMINO algorithm with B-POP, where the exact solution from the two lower-level problems will be recovered via parametric programming. The dimensionality information of the leader problem can be used to generate a random initial point or a random design of experiments. These pre-determined candidate solutions can then be simulated at the constraining bi-level problem where B-POP will retrieve the exact solution of the bi-level integer programming problem for a given set of upper level variables. This input-output data can then be used by a derivative-free optimization solver to retrieve the guaranteed feasible solution of the original problem as discussed in Chapter 2.

7.3.3 Multi-class Classification Models for Characterizing the Biological Potential of Toxic Compounds

The proposed modeling framework in Chapter 6 can be extended for the characterization and prediction of the endocrine disruptive potential of complex chemicals on other nuclear hormone receptors, such as the androgen receptor and thyroid hormone receptor. Furthermore, the two-class classification model presented in Chapter 6 can be extended to handle multiple classes of information on the estrogen receptor activity. Specifically, the separation between the three classes of estrogenic potential activity (i.e., agonist, antagonist and inactive) can be explored using the Random Forest algorithm. Moreover, the different agonist potency levels (i.e., strong, moderate, weak, very weak) can be distinguished through multi-class classification algorithms.

7.4 List of Publications

Journal Articles:

1. **B. Beykal**, M. Onel, O. Onel, E.N. Pistikopoulos. A Data-Driven Optimization Algorithm for Stiff Differential Algebraic Equations. AIChE Journal, 2019 (Under Review).

2. R. Mukherjee*, **B. Beykal***, M. Onel, A.T. Szafran, F. Stossi, M.G. Mancini, D. Lloyd, F.A. Wright, L. Zhou, M.A. Mancini, E.N. Pistikopoulos. Classification of Estrogenic Compounds by Coupling High Content Analysis and Machine Learning Algorithms. PLOS Computational Biology, 2020 (Under Review - * Equally Contributing First Authors).

3. **B. Beykal**, S. Avraamidou, I.P.E. Pistikopoulos, M. Onel, E.N. Pistikopoulos. DOMINO: Data-driven Optimization of bi-level Mixed-Integer NOnlinear Problems. Journal of Global Optimization, 2020, DOI: https://doi.org/10.1007/s10898-020-00890-3.

4. M. Onel, **B. Beykal**, K. Ferguson, W.A. Chiu, T.J. McDonald, L. Zhou, J.S. House, F.A. Wright, D.A. Sheen, I. Rusyn, E.N. Pistikopoulos. Grouping of Complex Substances Using Analytical Chemistry Data: A Framework for Quantitative Evaluation and Visualization. PLOS One, 2019, 14 (10), e0223517.

5. **B. Beykal**, F. Boukouvala, C.A. Floudas, E.N. Pistikopoulos. Optimal Design of Energy Systems Using Constrained Grey-Box Multi-Objective Optimization. Computers & Chemical Engineering, 2018, 116, 488-502.

6. **B. Beykal**, F. Boukouvala, C.A. Floudas, N. Sorek, H. Zalavadia, E. Gildin. Global Optimization of Grey-Box Computational Systems Using Surrogate Functions and Application to Highly Constrained Oil-Field Operations. Computers & Chemical Engineering, 2018, 114, 99-110.

7. N. Sorek, E. Gildin, F. Boukouvala, **B. Beykal**, C.A. Floudas. Dimensionality Reduction for Production Optimization Using Polynomial Approximations. Computational Geosciences, 2017, 21, 247-266.

Conference Proceedings:

1. R. Mukherjee, M. Onel, **B. Beykal**, A.T. Szafran, F. Stossi, M.A. Mancini, L. Zhou, F.A. Wright, E.N. Pistikopoulos. Development of the Texas A&M Superfund Research Program

Computational Platform for Data Integration, Visualization, and Analysis. Computer Aided Chemical Engineering, 2019, 46, 967-972.

2. S. Avraamidou, **B. Beykal**, I.P.E. Pistikopoulos, E.N. Pistikopoulos. A Hierarchical Food-Energy-Water Nexus (FEW-N) Decision-Making Approach for Land Use Optimization. Computer Aided Chemical Engineering, 2018, 44, 1885-1890.

3. M. Onel, **B. Beykal**, M. Wang, F.A. Grimm, L. Zhou, F.A. Wright, T.D. Phillips, I. Rusyn, E.N. Pistikopoulos. Optimal Chemical Grouping and Sorbent Material Design by Data Analysis, Modeling and Dimensionality Reduction Techniques. Computer Aided Chemical Engineering, 2018, 43, 421-426.

REFERENCES

[1] U.S. Energy Information Administration, "Annual energy outlook 2019 with projections to 2050." https://www.eia.gov/outlooks/aeo/pdf/aeo2019.pdf, 2019.

[2] U.S. Energy Information Administration, "Short-term energy outlook. u.s. hydrocarbon gas liquids (hgl) and petroleum refinery balances." https://www.eia.gov/outlooks/steo/data/browser/, 2019.

[3] U.S. Energy Information Administration, "Petroleum & other liquids natural gas plant field production." https://www.eia.gov/dnav/pet/pet_pnp_gp_dc_nus_mbblpd_a.htm, 2019.

[4] P. Liu, E. N. Pistikopoulos, and Z. Li, "An energy systems engineering approach to the optimal design of energy systems in commercial buildings," *Energy Policy*, vol. 38, no. 8, pp. 4224–4231, 2010.

[5] U.S. Energy Information Administration, "State electricity profiles." https://www.eia.gov/electricity/state/unitedstates/, November 2016.

[6] U.S. Energy Information Administration, "Monthly densified biomass fuel report." https://www.eia.gov/biofuels/biomass/, May 2017.

[7] U.S. Energy Information Administration, "Monthly energy review." https://www.eia.gov/totalenergy/data/monthly/archive/00351705.pdf, May 2017.

[8] U.S. Environmental Protection Agency, "Center for corporate climate leadership ghg emission factors hub." https://www.epa.gov/climateleadership/center-corporate-climate-leadership-ghg-emission-factors-hub, November 2015.

[9] BASIS - Biomass Availability and Sustainability Information System, "Report on conversion efficiency of biomass." http://www.basisbioenergy.eu/fileadmin/BASIS/D3.5_Report_on_conversion_efficiency_of_biomass.pdf, July 2015.

[10] U.S. Energy Information Administration, "Updated buildings sector appliance and equipment costs and efficiencies." https://www.eia.gov/analysis/studies/buildings/equipcosts/, November 2016.

[11] U.S. Environmental Protection Agency, "Biomass combined heat and power catalog of technologies." https://www.epa.gov/sites/production/files/2015-07/documents/biomass_combined_heat_and_power_catalog_of_technologies_v.1.1.pdf, September 2007.

[12] National Renewable Energy Laboratory, "Distributed generation renewable energy estimate of costs." http://www.nrel.gov/analysis/tech_lcoe_re_cost_est.html, February 2016.

[13] R. S. Judson, F. M. Magpantay, V. Chickarmane, C. Haskell, N. Tania, J. Taylor, M. Xia, R. Huang, D. M. Rotroff, D. L. Filer, *et al.*, "Integrated model of chemical perturbations of a biological pathway using 18 in vitro high-throughput screening assays for the estrogen receptor," *Toxicological Sciences*, vol. 148, no. 1, pp. 137–154, 2015.

[14] K. M. Sundaram and G. F. Froment, "Modeling of thermal cracking kinetics I: Thermal cracking of ethane, propane and their mixtures," *Chemical Engineering Science*, vol. 32, no. 6, pp. 601–608, 1977.

[15] K. M. Sundaram and G. F. Froment, "Modeling of thermal cracking kinetics -II: Cracking of iso-butane, of n-butane and of mixtures ethane-propane-n-butane," *Chemical Engineering Science*, vol. 32, no. 6, pp. 609–617, 1977.

[16] P. Kumar and D. Kunzru, "Modeling of naphtha pyrolysis," *Industrial & Engineering Chemistry Process Design and Development*, vol. 24, no. 3, pp. 774–782, 1985.

[17] O. Onel, *Advances in Modeling, Synthesis, and Global Optimization of Hybrid Energy Systems Toward the Production of Liquid Fuels and Olefins*. PhD thesis, Princeton University, New Jersey, 2017.

[18] NIST, "National institute of standards and technology chemistry webbook, srd 69, gas phase thermochemistry data." DOI: https://doi.org/10.18434/T4D303, 2018.

[19] J. M. Smith, H. C. Van Ness, and M. M. Abbott, *Introduction to chemical engineering thermodynamics*. New York: McGraw-Hill, 7th edition ed., 2005.

[20] Design Institute for Physical Properties AIChE, *DIPPR Project 801 Evaluated Standard Thermophysical Property Values*. Design Institute for Physical Property Research/American Institute of Chemical Engineers, 2019.

[21] F. Boukouvala, R. Misener, and C. A. Floudas, "Global optimization advances in mixed-integer nonlinear programming, MINLP, and constrained derivative-free optimization, CDFO," *European Journal of Operational Research*, vol. 252, pp. 701–727, 2016.

[22] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to Derivative-Free Optimization*. Philadelphia: Society for Industrial and Applied Mathematics, 2009.

[23] C. A. Kieslich, F. Boukouvala, and C. A. Floudas, "Optimization of black-box problems using smolyak grids and polynomial approximations," *Journal of Global Optimization*, vol. 71, no. 4, pp. 845–869, 2018.

[24] E. Newby and M. M. Ali, "A trust-region-based derivative free algorithm for mixed integer programming," *Computational Optimization and Applications*, vol. 60, no. 1, pp. 199–229, 2015.

[25] F. Boukouvala and M. G. Ierapetritou, "Derivative-free optimization for expensive constrained problems using a novel expected improvement objective function," *AIChE Journal*, vol. 60, no. 7, pp. 2462–2474, 2014.

[26] I. Bajaj, S. S. Iyer, and M. M. F. Hasan, "A trust region-based two phase algorithm for constrained black-box and grey-box optimization with infeasible initial point," *Computers & Chemical Engineering*, 2017.

[27] J. P. Eason and L. T. Biegler, "A trust region filter method for glass box/black box optimization," *AIChE Journal*, vol. 62, no. 9, pp. 3124–3136, 2016.

[28] F. Boukouvala and C. A. Floudas, "ARGONAUT: Algorithms for Global Optimization of coNstrAined grey-box compUTational problems," *Optimization Letters*, vol. 11, pp. 895–913, 2017.

[29] F. Boukouvala, M. M. F. Hasan, and C. A. Floudas, "Global optimization of general constrained grey-box models: new method and its application to constrained PDEs for pressure swing adsorption," *Journal of Global Optimization*, vol. 67,, pp. 3–42, 2017.

[30] B. Beykal, F. Boukouvala, C. A. Floudas, N. Sorek, H. Zalavadia, and E. Gildin, "Global optimization of grey-box computational systems using surrogate functions and application

to highly constrained oil-field operations," *Computers & Chemical Engineering*, vol. 114, pp. 99–110, 2018.

[31] A. Cozad, N. V. Sahinidis, and D. C. Miller, "Learning surrogate models for simulation-based optimization," *AIChE Journal*, vol. 60, no. 6, pp. 2211–2227, 2014.

[32] Z. T. Wilson and N. V. Sahinidis, "The ALAMO approach to machine learning," *Computers & Chemical Engineering*, vol. 106, pp. 785–795, 2017.

[33] J. Müller, C. A. Shoemaker, and R. Piché, "SO-MI: A surrogate model algorithm for computationally expensive nonlinear mixed-integer black-box global optimization problems," *Computers & Operations Research*, vol. 40,, pp. 1383–1400, 2013.

[34] T. G. Kolda, R. M. Lewis, and V. Torczon, "Optimization by direct search: New perspectives on some classical and modern methods," *SIAM Review*, vol. 45, no. 3, pp. 385–482, 2003.

[35] L. M. Rios and N. V. Sahinidis, "Derivative-free optimization: a review of algorithms and comparison of software implementations," *Journal of Global Optimization*, vol. 56, pp. 1247–1293, 2013.

[36] A. Bhosekar and M. Ierapetritou, "Advances in surrogate based modeling, feasibility analysis, and optimization: A review," *Computers & Chemical Engineering*, vol. 108, pp. 250–267, 2018.

[37] K. K. Vu, C. D'Ambrosio, Y. Hamadi, and L. Liberti, "Surrogate-based methods for black-box optimization," *International Transactions in Operational Research*, vol. 24, no. 3, pp. 393–424, 2017.

[38] R. M. Lewis and V. Torczon, "A globally convergent augmented lagrangian pattern search algorithm for optimization with general constraints and simple bounds," *SIAM Journal on Optimization*, vol. 12, pp. 1075–1089, 2002.

[39] M. A. Diniz-Ehrhardt, J. M. Martínez, and L. G. Pedroso, "Derivative-free methods for nonlinear programming with general lower-level constraints," *Computational & Applied Mathematics*, vol. 30, pp. 19–52, 2011.

[40] R. B. Gramacy, G. A. Gray, S. Le Digabel, H. K. H. Lee, P. Ranjan, G. Wells, and S. M.

Wild, "Modeling an augmented lagrangian for blackbox constrained optimization," *Technometrics*, vol. 58, pp. 1–11, 2016.

[41] C. Audet and J. E. Dennis Jr, "Mesh adaptive direct search algorithms for constrained optimization," *SIAM Journal on optimization*, vol. 17, no. 1, pp. 188–217, 2006.

[42] C. Audet and J. E. Dennis Jr, "A progressive barrier for derivative-free nonlinear programming," *SIAM Journal on Optimization*, vol. 20, pp. 445–472, 2009.

[43] G. Liuzzi, S. Lucidi, and M. Sciandrone, "Sequential penalty derivative-free methods for nonlinear constrained optimization," *SIAM Journal on Optimization*, vol. 20, pp. 2614–2635, 2010.

[44] S. Gratton and L. N. Vicente, "A merit function approach for direct search," *SIAM Journal on Optimization*, vol. 24, pp. 1980–1998, 2014.

[45] J. M. Martínez and F. Sobral, "Constrained derivative-free optimization on thin domains," *Journal of Global Optimization*, vol. 56, pp. 1217–1232, 2013.

[46] M. B. Arouxét, N. E. Echebest, and E. A. Pilotta, "Inexact restoration method for nonlinear optimization without derivatives," *Journal of Computational and Applied Mathematics*, vol. 290, pp. 26–43, 2015.

[47] G. Di Pillo, G. Liuzzi, S. Lucidi, V. Piccialli, and F. Rinaldi, "A DIRECT-type approach for derivative-free constrained global optimization," *Computational Optimization and Applications*, vol. 65, pp. 361–397, 2016.

[48] G. Liuzzi, S. Lucidi, and F. Rinaldi, "A derivative-free approach to constrained multiobjective nonsmooth optimization," *SIAM Journal on Optimization*, vol. 26, pp. 2744–2774, 2016.

[49] R. G. Regis and C. A. Shoemaker, "Constrained global optimization of expensive black box functions using radial basis functions," *Journal of Global Optimization*, vol. 31,, pp. 153–171, 2005.

[50] J. A. Caballero and I. E. Grossmann, "An algorithm for the use of surrogate models in modular flowsheet optimization," *AIChE journal*, vol. 54, no. 10, pp. 2633–2650, 2008.

[51] M. J. D. Powell, "A direct search optimization method that models the objective and constraint functions by linear interpolation," in *Advances in optimization and numerical analysis*, pp. 51–67, Springer, 1994.

[52] A. March and K. Willcox, "Constrained multifidelity optimization using model calibration," *Structural and Multidisciplinary Optimization*, vol. 46,, pp. 93–109, 2012.

[53] A. R. Conn and S. Le Digabel, "Use of quadratic models with mesh-adaptive direct search for constrained black box optimization," *Optimization Methods and Software*, vol. 28, pp. 139–158, 2013.

[54] J. Müller and C. A. Shoemaker, "Influence of ensemble surrogate models and sampling strategy on the solution quality of algorithms for computationally expensive black-box global optimization problems," *Journal of Global Optimization*, vol. 60,, pp. 123–144, 2014.

[55] S. Avraamidou and E. N. Pistikopoulos, "A multiparametric mixed-integer bi-level optimization strategy for supply chain planning under demand uncertainty," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 10178–10183, 2017.

[56] A. Gupta and C. D. Maranas, "A two-stage modeling and solution framework for multisite midterm planning under demand uncertainty," *Industrial & Engineering Chemistry Research*, vol. 39, no. 10, pp. 3799–3813, 2000.

[57] Z. Li and M. Ierapetritou, "Integrated production planning and scheduling using a decomposition framework," *Chemical Engineering Science*, vol. 64, pp. 3585–3597, 08 2009.

[58] S. Avraamidou and E. N. Pistikopoulos, "A bi-level formulation and solution method for the integration of process design and scheduling," in *Proceedings of the 9th International Conference on Foundations of Computer-Aided Process Design* (S. G. Munoz, C. D. Laird, and M. J. Realff, eds.), vol. 47 of *Computer Aided Chemical Engineering*, pp. 17–22, Elsevier, 2019.

[59] S. Avraamidou and E. N. Pistikopoulos, "A novel algorithm for the global solution of mixed-integer bi-level multi-follower problems and its application to planning & scheduling integration," in *2018 European Control Conference (ECC)*, pp. 1056–1061, IEEE, 2018.

[60] J. F. Bard, J. Plummer, and J. C. Sourie, "Determining tax credits for converting nonfood crops to biofuels: An application of bilevel programming," in *Multilevel Optimization: Algorithms and Applications* (A. Migdalas, P. M. Pardalos, and P. Värbrand, eds.), pp. 23–50, Boston, MA: Springer US, 1998.

[61] M. Labbé and A. Violin, "Bilevel programming and price setting problems," *Annals of Operations Research*, vol. 240, no. 1, pp. 141–169, 2016.

[62] M. Fampa, L. A. Barroso, D. Candal, and L. Simonetti, "Bilevel optimization applied to strategic pricing in competitive electricity markets," *Computational Optimization and Applications*, vol. 39, no. 2, pp. 121–142, 2008.

[63] A. Sinha, P. Malo, A. Frantsev, and K. Deb, "Multi-objective stackelberg game between a regulating authority and a mining company: A case study in environmental economics," in *2013 IEEE Congress on Evolutionary Computation*, pp. 478–485, IEEE, 2013.

[64] S. Avraamidou and E. N. Pistikopoulos, "Adjustable robust optimization through multiparametric programming," *Optimization Letters*, pp. 1–15, 2019.

[65] J. Lu, J. Han, Y. Hu, and G. Zhang, "Multilevel decision-making: A survey," *Information Sciences*, vol. 346, pp. 463–487, 2016.

[66] H. v. Stackelberg, *Theory of the market economy*. Oxford University Press, 1952.

[67] M. Simaan and J. B. Cruz, "On the stackelberg strategy in nonzero-sum games," *Journal of Optimization Theory and Applications*, vol. 11, no. 5, pp. 533–555, 1973.

[68] Z. H. Gümüş and C. A. Floudas, "Global optimization of nonlinear bilevel programming problems," *Journal of Global Optimization*, vol. 20, no. 1, pp. 1–31, 2001.

[69] J. F. Bard and J. T. Moore, "A branch and bound algorithm for the bilevel programming problem," *SIAM Journal on Scientific and Statistical Computing*, vol. 11, no. 2, pp. 281–292, 1990.

[70] P. M. Kleniati and C. S. Adjiman, "A generalization of the branch-and-sandwich algorithm: from continuous to mixed-integer nonlinear bilevel problems," *Computers & Chemical Engineering*, vol. 72, pp. 373–386, 2015.

[71] P. Garcia-Herreros, L. Zhang, P. Misra, E. Arslan, S. Mehta, and I. E. Grossmann, "Mixed-integer bilevel optimization for capacity planning with rational markets," *Computers & Chemical Engineering*, vol. 86, pp. 33–47, 2016.

[72] A. Mitsos, P. Lemonidis, and P. I. Barton, "Global solution of bilevel programs with a non-convex inner program," *Journal of Global Optimization*, vol. 42, no. 4, pp. 475–513, 2008.

[73] N. P. Faísca, V. Dua, B. Rustem, P. M. Saraiva, and E. N. Pistikopoulos, "Parametric global optimisation for bilevel programming," *Journal of Global Optimization*, vol. 38, no. 4, pp. 609–623, 2007.

[74] N. P. Faísca, P. M. Saraiva, B. Rustem, and E. N. Pistikopoulos, "A multi-parametric programming approach for multilevel hierarchical and decentralised optimisation problems," *Computational management science*, vol. 6, no. 4, pp. 377–397, 2009.

[75] L. F. Domínguez and E. N. Pistikopoulos, "Multiparametric programming based algorithms for pure integer and mixed-integer bilevel programming problems," *Computers & Chemical Engineering*, vol. 34, no. 12, pp. 2097–2106, 2010.

[76] R. Oberdieck, N. A. Diangelakis, S. Avraamidou, and E. N. Pistikopoulos, "On unbounded and binary parameters in multi-parametric programming: applications to mixed-integer bilevel optimization and duality theory," *Journal of Global Optimization*, vol. 69, no. 3, pp. 587–606, 2017.

[77] S. Avraamidou and E. N. Pistikopoulos, "B-POP: Bi-level parametric optimization toolbox," *Computers & Chemical Engineering*, vol. 122, pp. 193–202, 2019.

[78] S. Avraamidou and E. N. Pistikopoulos, "Multi-parametric global optimization approach for tri-level mixed-integer linear optimization problems," *Journal of Global Optimization*, vol. 74, no. 3, pp. 443–465, 2019.

[79] S. Avraamidou and E. N. Pistikopoulos, "A multi-parametric optimization approach for bilevel mixed-integer linear and quadratic programming problems," *Computers & Chemical Engineering*, vol. 125, pp. 98–113, 2019.

[80] S. Avraamidou and E. N. Pistikopoulos, "A global optimization algorithm for the solution

of tri-level mixed-integer quadratic programming problems," in *WCGO 2019: Optimization of Complex Systems: Theory, Models, Algorithms and Applications* (H. A. Le Thi, H. M. Le, and T. Pham Dinh, eds.), (Cham), pp. 579–588, Springer, 2019.

[81] A. Sinha, P. Malo, and K. Deb, "A review on bilevel optimization: from classical to evolutionary approaches and applications," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 2, pp. 276–295, 2017.

[82] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics and Engineering*, vol. 186, no. 2-4, pp. 311–338, 2000.

[83] A. Homaifar, C. X. Qi, and S. H. Lai, "Constrained optimization via genetic algorithms," *Simulation*, vol. 62, no. 4, pp. 242–253, 1994.

[84] K. Sedlaczek and P. Eberhard, "Using augmented Lagrangian particle swarm optimization for constrained problems in engineering," *Structural and Multidisciplinary Optimization*, vol. 32, no. 4, pp. 277–286, 2006.

[85] S. D. Handoko, L. H. Chuin, A. Gupta, O. Y. Soon, H. C. Kim, and T. P. Siew, "Solving multi-vehicle profitable tour problem via knowledge adoption in evolutionary bi-level programming," *2015 IEEE Congress on Evolutionary Computation, CEC 2015 - Proceedings*, pp. 2713–2720, 2015.

[86] I. Nishizaki and M. Sakawa, "Computational methods through genetic algorithms for obtaining stackelberg solutions to two-level integer programming problems," *Cybernetics and Systems*, vol. 36, no. 6, pp. 565–579, 2005.

[87] L. Hecheng and W. Yuping, "Exponential distribution-based genetic algorithm for solving mixed-integer bilevel programming problems," *Journal of Systems Engineering and Electronics*, vol. 19, no. 6, pp. 1157–1164, 2008.

[88] J. M. Arroyo and F. J. Fernández, "A genetic algorithm approach for the analysis of electric grid interdiction with line switching," in *2009 15th International Conference on Intelligent System Applications to Power Systems*, pp. 1–6, IEEE, 2009.

[89] ILOG and IBM, *IBM ILOG CPLEX Optimization Studio Getting Started with CPLEX*. IBM

Corporation, 2017.

[90] R. Misener and C. A. Floudas, "Global optimization of mixed-integer models with quadratic and signomial functions: a review," *Applied and Computational Mathematics*, vol. 11, no. 3, pp. 317–336, 2012.

[91] R. Misener and C. A. Floudas, "GloMIQO: Global mixed-integer quadratic optimizer," *Journal of Global Optimization*, vol. 57, no. 1, pp. 3–50, 2013.

[92] R. Misener and C. A. Floudas, "ANTIGONE: algorithms for continuous/integer global optimization of nonlinear equations," *Journal of Global Optimization*, vol. 59, no. 2-3, pp. 503–526, 2014.

[93] M. Tawarmalani and N. V. Sahinidis, "A polyhedral branch-and-cut approach to global optimization," *Mathematical programming*, vol. 103, no. 2, pp. 225–249, 2005.

[94] S. Avraamidou, B. Beykal, I. P. E. Pistikopoulos, and E. N. Pistikopoulos, "A hierarchical food-energy-water nexus (FEW-N) decision-making approach for land use optimization," in *13th International Symposium on Process Systems Engineering (PSE 2018)* (M. R. Eden, M. G. Ierapetritou, and G. P. Towler, eds.), vol. 44 of *Computer Aided Chemical Engineering*, pp. 1885–1890, Elsevier, 2018.

[95] S. Le Digabel, "Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm," *ACM Transactions on Mathematical Software (TOMS)*, vol. 37, no. 4, pp. 1–15, 2011.

[96] T. P. Runarsson and X. Yao, "Search biases in constrained evolutionary optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 35, no. 2, pp. 233–243, 2005.

[97] S. G. Johnson, "The NLopt nonlinear-optimization package," 2014. (accessed 16 January 2018).

[98] M. A. Abramson, C. Audet, G. Couture, J. E. Dennis Jr, S. Le Digabel, and C. Tribes, "The NOMAD project," 2015. (accessed 16 January 2018).

[99] S. Le Digabel, C. Tribes, V. R. Montplaisir, and C. Audet, "NOMAD user guide version

3.9.1," 2019. (accessed 14 July 2019).

[100] A. Mitsos and P. I. Barton, "A test set for bilevel programs," 2007. (accessed 16 January 2018).

[101] R. Paulavicius, P. M. Kleniati, and C. S. Adjiman, "A library of nonconvex bilevel test problems with the corresponding ampl input files (version v1.0)," 2016. [Data set].

[102] T. A. Edmunds and J. F. Bard, "An algorithm for the mixed-integer nonlinear bilevel programming problem," *Annals of Operations Research*, vol. 34, no. 1, pp. 149–162, 1992.

[103] K. H. Sahin and A. R. Ciric, "A dual temperature simulated annealing approach for solving bilevel programming problems," *Computers & Chemical Engineering*, vol. 23, no. 1, pp. 11–25, 1998.

[104] B. Colson, "BIPA(bilevel programming with approximation methods)(software guide and test problems)," *Cahiers du GERAD*, 2002. (accessed 16 January 2018).

[105] A. Mitsos, "Global solution of nonlinear mixed-integer bilevel programs," *Journal of Global Optimization*, vol. 47, no. 4, pp. 557–582, 2010.

[106] P.-M. Kleniati and C. S. Adjiman, "Branch-and-sandwich: a deterministic global optimization algorithm for optimistic bilevel programming problems. part ii: Convergence analysis and numerical results," *Journal of Global Optimization*, vol. 60, no. 3, pp. 459–481, 2014.

[107] A. T. Woldemariam and S. M. Kassa, "Systematic evolutionary algorithm for general multilevel stackelberg problems with bounded decision variables (seamsp)," *Annals of Operations Research*, vol. 229, no. 1, pp. 771–790, 2015.

[108] J. Nie, L. Wang, and J. J. Ye, "Bilevel polynomial programs and semidefinite relaxation methods," *SIAM Journal on Optimization*, vol. 27, no. 3, pp. 1728–1757, 2017.

[109] Y. Nie, S. Avraamidou, X. Xiao, E. N. Pistikopoulos, J. Li, Y. Zeng, F. Song, J. Yu, and M. Zhu, "A food-energy-water nexus approach for land use optimization," *Science of The Total Environment*, vol. 659, pp. 7–19, 2019.

[110] Y. Nie, S. Avraamidou, X. Xiao, E. N. Pistikopoulos, and J. Li, "Two-stage land use optimization for a food-energy-water nexus system: A case study in texas edwards region," in

Computer Aided Chemical Engineering, vol. 47, pp. 205–210, Elsevier, 2019.

[111] Y. Nie, S. Avraamidou, J. Li, X. Xiao, and E. N. Pistikopoulos, "Land use modeling and optimization based on food-energy-water nexus: a case study on crop-livestock systems," in *Computer Aided Chemical Engineering*, vol. 44, pp. 1939–1944, Elsevier, 2018.

[112] K. Miettinen, *Nonlinear Multiobjective Optimization*. Springer, 1998.

[113] C. A. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary algorithms for solving multi-objective problems*. Springer, 2007.

[114] T. Ray, K. Tai, and K. C. Seow, "Multiobjective design optimization by an evolutionary algorithm," *Engineering Optimization*, vol. 33, no. 4, pp. 399–424, 2001.

[115] A. Toffolo and A. Lazzaretto, "Evolutionary algorithms for multi-objective energetic and economic optimization in thermal system design," *Energy*, vol. 27, no. 6, pp. 549–567, 2002.

[116] D. Gong, Y. Zhang, and C. Qi, "Environmental/economic power dispatch using a hybrid multi-objective optimization algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 32, no. 6, pp. 607–614, 2010.

[117] L. Wang and C. Singh, "Environmental/economic power dispatch using a fuzzified multi-objective particle swarm optimization algorithm," *Electric Power Systems Research*, vol. 77, no. 12, pp. 1654–1664, 2007.

[118] J. Sanchis, M. A. Martínez, and X. Blasco, "Integrated multiobjective optimization and a priori preferences using genetic algorithms," *Information Sciences*, vol. 178, no. 4, pp. 931–951, 2008.

[119] F. Di Pierro, S. T. Khu, D. Savić, and L. Berardi, "Efficient multi-objective optimal design of water distribution networks on a budget of simulations using hybrid algorithms," *Environmental Modelling & Software*, vol. 24, no. 2, pp. 202–213, 2009.

[120] O. Abdelkafi, L. Idoumghar, J. Lepagnot, J. L. Paillaud, I. Deroche, L. Baumes, and P. Collet, "Using a novel parallel genetic hybrid algorithm to generate and determine new zeolite frameworks," *Computers & Chemical Engineering*, vol. 98, pp. 50–60, 2017.

[121] G. P. Rangaiah and A. Bonilla-Petriciolet, *Multi-objective optimization in chemical engineering: developments and applications*. John Wiley & Sons, 2013.

[122] R. Datta and R. G. Regis, "A surrogate-assisted evolution strategy for constrained multi-objective optimization," *Expert Systems with Applications*, vol. 57, pp. 270–284, 2016.

[123] K. S. Bhattacharjee, H. K. Singh, and T. Ray, "Multi-objective optimization with multiple spatially distributed surrogates," *Journal of Mechanical Design*, vol. 138, no. 9, p. 091401, 2016.

[124] P. Singh, I. Couckuyt, F. Ferranti, and T. Dhaene, "A constrained multi-objective surrogate-based optimization algorithm," in *2014 IEEE Congress on Evolutionary Computation (CEC)*, pp. 3080–3087, IEEE, 2014.

[125] P. Feliot, J. Bect, and E. Vazquez, "A bayesian approach to constrained single-and multi-objective optimization," *Journal of Global Optimization*, vol. 67, no. 1-2, pp. 97–133, 2017.

[126] J. Martínez-Frutos and D. Herrero-Pérez, "Kriging-based infill sampling criterion for constraint handling in multi-objective optimization," *Journal of Global Optimization*, vol. 64, no. 1, pp. 97–115, 2016.

[127] R. G. Regis, "Multi-objective constrained black-box optimization using radial basis function surrogates," *Journal of Computational Science*, vol. 16, pp. 140–155, 2016.

[128] M. Tabatabaei, J. Hakanen, M. Hartikainen, K. Miettinen, and K. Sindhya, "A survey on handling computationally expensive multiobjective optimization problems using surrogates: non-nature inspired methods," *Structural and Multidisciplinary Optimization*, vol. 52, no. 1, pp. 1–25, 2015.

[129] P. A. Clark and A. W. Westerberg, "Optimization for design problems having more than one objective," *Computers & Chemical Engineering*, vol. 7, no. 4, pp. 259–278, 1983.

[130] D. Chafekar, J. Xuan, and K. Rasheed, "Constrained multi-objective optimization using steady state genetic algorithms," in *Genetic and Evolutionary Computation Conference*, pp. 813–824, Springer, 2003.

[131] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using

reference-point based nondominated sorting approach, part ii: Handling constraints and extending to an adaptive approach.," *IEEE Trans. Evolutionary Computation*, vol. 18, no. 4, pp. 602–622, 2014.

[132] K. Zielinski, D. Peters, and R. Laur, "Constrained multi-objective optimization using differential evolution," in *Third International Conference on Computational Intelligence, Robotics and Autonomous Systems*, Citeseer, 2005.

[133] Z. Fan, J. Liu, T. Sorensen, and P. Wang, "Improved differential evolution based on stochastic ranking for robust layout synthesis of mems components," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 4, pp. 937–948, 2009.

[134] L. T. Biegler, "Optimization of differential-algebraic equation systems." https://www.lehigh.edu/~wes1/apci/26may00.pdf, 2000.

[135] P. Daoutidis, "Daes in model reduction of chemical processes: An overview," in *Surveys in Differential-Algebraic Equations II*, pp. 69–102, Springer, 2015.

[136] N. A. Diangelakis, B. Burnak, J. Katz, and E. N. Pistikopoulos, "Process design and control optimization: A simultaneous approach by multi-parametric programming," *AIChE Journal*, vol. 63, no. 11, pp. 4827–4846, 2017.

[137] M. Berreni and M. Wang, "Modelling and dynamic optimization of thermal cracking of propane for ethylene manufacturing," *Computers & Chemical Engineering*, vol. 35, no. 12, pp. 2876–2885, 2011.

[138] B. Burnak, J. Katz, N. A. Diangelakis, and E. N. Pistikopoulos, "Simultaneous process scheduling and control: a multiparametric programming-based approach," *Industrial & Engineering Chemistry Research*, vol. 57, no. 11, pp. 3963–3976, 2018.

[139] L. T. Biegler, *Nonlinear programming: concepts, algorithms, and applications to chemical processes*, vol. 10. Siam, 2010.

[140] Y.-D. Lang and L. T. Biegler, "A software environment for simultaneous dynamic optimization," *Computers & Chemical engineering*, vol. 31, no. 8, pp. 931–942, 2007.

[141] W. R. Esposito and C. A. Floudas, "Global optimization for the parameter estimation

of differential-algebraic systems," *Industrial & Engineering Chemistry Research*, vol. 39, no. 5, pp. 1291–1310, 2000.

[142] D. Y. Caballero, L. T. Biegler, and R. Guirardello, "Simulation and optimization of the ethane cracking process to produce ethylene," in *Computer Aided Chemical Engineering*, vol. 37, pp. 917–922, Elsevier, 2015.

[143] C. A. Henao and C. T. Maravelias, "Surrogate-based superstructure optimization framework," *AIChE Journal*, vol. 57, no. 5, pp. 1216–1232, 2011.

[144] I. Fahmi and S. Cremaschi, "Process synthesis of biodiesel production plant using artificial neural networks as the surrogate models," *Computers & Chemical Engineering*, vol. 46, pp. 105–123, 2012.

[145] O. Onel, A. M. Niziolek, H. Butcher, B. A. Wilhite, and C. A. Floudas, "Multi-scale approaches for gas-to-liquids process intensification: Cfd modeling, process synthesis, and global optimization," *Computers & Chemical Engineering*, vol. 105, pp. 276–296, 2017.

[146] C. D. Demirhan, W. W. Tso, J. B. Powell, and E. N. Pistikopoulos, "Sustainable ammonia production through process synthesis and global optimization," *AIChE Journal*, vol. 65, no. 7, 2019.

[147] N. Sorek, E. Gildin, F. Boukouvala, B. Beykal, and C. A. Floudas, "Dimensionality reduction for production optimization using polynomial approximations," *Computational Geosciences*, vol. 21, no. 2, pp. 247–266, 2017.

[148] B. Beykal, F. Boukouvala, C. A. Floudas, and E. N. Pistikopoulos, "Optimal design of energy systems using constrained grey-box multi-objective optimization," *Computers & Chemical engineering*, vol. 116, pp. 488–502, 2018.

[149] I. Bajaj, S. S. Iyer, and M. M. F. Hasan, "A trust region-based two phase algorithm for constrained black-box and grey-box optimization with infeasible initial point," *Computers & Chemical Engineering*, vol. 116, pp. 306–321, 2018.

[150] B. Beykal, S. Avraamidou, I. P. E. Pistikopoulos, M. Onel, and E. N. Pistikopoulos, "DOMINO: Data-driven Optimization of bi-level Mixed-Integer NOnlinear problems,"

Journal of Global Optimization, pp. 1–36, 2020.

[151] M. Onel, B. Beykal, M. Wang, F. A. Grimm, L. Zhou, F. A. Wright, T. D. Phillips, I. Rusyn, and E. N. Pistikopoulos, "Optimal chemical grouping and sorbent material design by data analysis, modeling and dimensionality reduction techniques," in *Computer Aided Chemical Engineering*, vol. 43, pp. 421–426, Elsevier, 2018.

[152] C. D. Demirhan, W. W. Tso, G. S. Ogumerem, and E. N. Pistikopoulos, "Energy systems engineering-a guided tour," *BMC Chemical Engineering*, vol. 1, no. 1, p. 11, 2019.

[153] S. S. Garud, I. A. Karimi, and M. Kraft, "Design of computer experiments: A review," *Computers & Chemical Engineering*, vol. 106, pp. 71–95, 2017.

[154] M. Cavazzuti, *Optimization methods: from theory to design scientific and technological aspects in mechanics*. Springer Science & Business Media, 2012.

[155] L. S. Dias and M. G. Ierapetritou, "Data-driven feasibility analysis for the integration of planning and scheduling problems," *Optimization and Engineering*, vol. 20, no. 4, pp. 1029–1066, 2019.

[156] A. Basudhar, C. Dribusch, S. Lacaze, and S. Missoum, "Constrained efficient global optimization with support vector machines," *Structural and Multidisciplinary Optimization*, vol. 46, no. 2, pp. 201–221, 2012.

[157] L. H. Chiang, M. E. Kotanchek, and A. K. Kordon, "Fault diagnosis based on fisher discriminant analysis and support vector machines," *Computers & Chemical Engineering*, vol. 28, no. 8, pp. 1389–1401, 2004.

[158] M. Onel, C. A. Kieslich, and E. N. Pistikopoulos, "A nonlinear support vector machine-based feature selection approach for fault detection and diagnosis: Application to the tennessee eastman process," *AIChE Journal*, vol. 65, no. 3, pp. 992–1005, 2019.

[159] M. Onel, C. A. Kieslich, Y. A. Guzman, C. A. Floudas, and E. N. Pistikopoulos, "Big data approach to batch process monitoring: Simultaneous fault detection and diagnosis using nonlinear support vector machine-based feature selection," *Computers & Chemical engineering*, vol. 115, pp. 46–63, 2018.

[160] G. T. Jemwa and C. Aldrich, "Improving process operations using support vector machines and decision trees," *AIChE Journal*, vol. 51, no. 2, pp. 526–543, 2005.

[161] M. Onel, B. Beykal, K. Ferguson, W. A. Chiu, T. J. McDonald, L. Zhou, J. S. House, F. A. Wright, D. A. Sheen, I. Rusyn, and E. N. Pistikopoulos, "Grouping of complex substances using analytical chemistry data: A framework for quantitative evaluation and visualization," *PloS one*, vol. 14, no. 10, 2019.

[162] Z. Li and C. A. Floudas, "Optimal scenario reduction framework based on distance of uncertainty distribution and output performance: I. single reduction via mixed integer linear optimization," *Computers & Chemical Engineering*, vol. 70, pp. 50–66, 2014.

[163] U.S. Energy Information Administration, "Appalachia region drives growth in U.S. natural gas production since 2012." https://www.eia.gov/todayinenergy/detail.php?id=33972, 2017.

[164] U.S. Energy Information Administration, "U.S. ethane consumption, exports to increase as new petrochemical plants come online." https://www.eia.gov/todayinenergy/detail.php?id=35012, 2018.

[165] G. P. Froment, B. O. Van de Steene, P. S. Van Damme, S. Narayanan, and A. G. Goossens, "Thermal cracking of ethane and ethane-propane mixtures," *Industrial & Engineering Chemistry Process Design and Development*, vol. 15, no. 4, pp. 495–504, 1976.

[166] A. Tarafder, B. C. S. Lee, A. K. Ray, and G. P. Rangaiah, "Multiobjective optimization of an industrial ethylene reactor using a nondominated sorting genetic algorithm," *Industrial & Engineering Chemistry Research*, vol. 44, no. 1, pp. 124–141, 2005.

[167] K. M. Sundaram, P. S. Van Damme, and G. F. Froment, "Coke deposition in the thermal cracking of ethane," *AIChE Journal*, vol. 27, no. 6, pp. 946–951, 1981.

[168] C. Wilke, "A viscosity equation for gas mixtures," *The Journal of Chemical Physics*, vol. 18, no. 4, pp. 517–519, 1950.

[169] A. Muggeridge, A. Cockin, K. Webb, H. Frampton, I. Collins, T. Moulds, and P. Salino, "Recovery rates, enhanced oil recovery and technological limits," *Philosophical Transactions of the Royal Society A*, vol. 372, 2014.

[170] M. Asadollahi, G. Nævdal, M. Dadashpour, and J. Kleppe, "Production optimization using derivative free methods applied to brugge field case," *Journal of Petroleum Science and Engineering*, vol. 114, pp. 22–37, 2014.

[171] M. S. Tavallali, I. A. Karimi, and D. Baxendale, "Process systems engineering perspective on the planning and development of oil fields," *AIChE Journal*, vol. 62, pp. 2586–2604, 2016.

[172] B. Bailey, M. Crabtree, J. Tyrie, J. Elphick, F. Kuchuk, C. Romano, and L. Roodhart, "Water control," *Oilfield Review*, vol. 12, no. 1, pp. 30–51, 2000.

[173] D. E. Ciaurri, T. Mukerji, and L. J. Durlofsky, "Derivative-free optimization for oil field operations," in *Computational Optimization and Applications in Engineering and Industry* (X. Yang and S. Koziel, eds.), pp. 19–55, Springer Berlin Heidelberg, 2011.

[174] C. Wang, G. Li, and A. C. Reynolds, "Production optimization in closed-loop reservoir management," *SPE Journal*, vol. 14, pp. 506–523, 2009.

[175] O. J. Isebor, L. J. Durlofsky, and D. E. Ciaurri, "A derivative-free methodology with local and global search for the constrained joint optimization of well locations and controls," *Computational Geosciences*, vol. 18, pp. 463–482, 2014.

[176] O. J. Isebor, D. E. Ciaurri, and L. J. Durlofsky, "Generalized field-development optimization with derivative-free procedures," *SPE Journal*, vol. 19, pp. 891–908, 2014.

[177] D. E. Ciaurri, O. J. Isebor, and L. J. Durlofsky, "Application of derivative-free methodologies to generally constrained oil production optimisation problems," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 2, pp. 134–161, 2011.

[178] T. D. Humphries and R. D. Haynes, "Joint optimization of well placement and control for nonconventional well types," *Journal of Petroleum Science and Engineering*, vol. 126, pp. 242–253, 2015.

[179] E. Suwartadi, S. Krogstad, and B. Foss, "Nonlinear output constraints handling for production optimization of oil reservoirs," *Computational Geosciences*, vol. 16, pp. 499–517, 2012.

[180] C. Chen, Y. Wang, G. Li, and A. C. Reynolds, "Closed-loop reservoir management on the brugge test case," *Computational Geosciences*, vol. 14, pp. 691–703, 2010.

[181] X. Liu and A. C. Reynolds, "Augmented lagrangian method for maximizing expectation and minimizing risk for optimal well-control problems with nonlinear constraints," *SPE Journal*, vol. 21, pp. 1830–1842, 2016.

[182] N. V. Queipo, J. V. Goicochea, and S. Pintos, "Surrogate modeling-based optimization of SAGD processes," *Journal of Petroleum Science and Engineering*, vol. 35, pp. 83–93, 2002.

[183] B. Horowitz, S. M. B. Afonso, and C. V. P. de Mendonça, "Surrogate based optimal waterflooding management," *Journal of Petroleum Science and Engineering*, vol. 112, pp. 206–219, 2013.

[184] A. Drud, "CONOPT–a large-scale GRG code," *ORSA Journal on Computing*, vol. 6, pp. 207–216, 1994.

[185] N. Sorek, H. Zalavadia, and E. Gildin, "Model order reduction and control polynomial approximation for well-control production optimization," in *SPE Reservoir Simulation Conference*, Montgomery, TX: Society of Petroleum Engineers, 2017.

[186] N. Sorek, *Reservoir Flooding Optimization by Control Polynomial Approximations*. PhD thesis, Texas A&M University, College Station, 2017.

[187] J. Lee, J. B. Rollins, and J. P. Spivey, *Pressure Transient Testing*. Richardson: Society of Petroleum Engineers, 2003.

[188] A. T. Gaspar, G. D. Avansi, A. A. d. S. dos Santos, J. C. von Hohendorff Filho, and D. J. Schiozer, "UNISIM-ID: benchmark studies for oil field development and production strategy selection," *International Journal of Modeling and Simulation for the Petroleum Industry*, vol. 9, pp. 47–55, 2015.

[189] G. D. Avansi and D. J. Schiozer, "UNISIM-I: synthetic model for reservoir development and management applications," *International Journal of Modeling and Simulation for the Petroleum Industry*, vol. 9, pp. 21–30, 2015.

[190] M. A. S. Pinto, M. Ghasemi, N. Sorek, E. Gildin, and D. J. Schiozer, "Hybrid optimization

for closed-loop reservoir management," in *SPE Reservoir Simulation Symposium*, Houston, TX: Society of Petroleum Engineers, 2015.

[191] S. Krogstad, K. A. Lie, O. Møyner, H. M. Nilsen, X. Raynaud, and B. Skaflestad, "MRST-AD–an open-source framework for rapid prototyping and evaluation of reservoir simulation problems," in *SPE Reservoir Simulation Symposium*, Houston, TX: Society of Petroleum Engineers, 2015.

[192] K. A. Lie, *An introduction to reservoir simulation using MATLAB: User guide for the Matlab Reservoir Simulation Toolbox (MRST)*. SINTEF ICT, 2016.

[193] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, vol. 13, pp. 455–492, 1998.

[194] M. J. Sasena, P. Papalambros, and P. Goovaerts, "Exploration of metamodeling sampling criteria for constrained global optimization," *Engineering Optimization*, vol. 34, pp. 263–278, 2002.

[195] H. Zhao, C. Chen, S. Do, D. Oliveira, G. Li, and A. C. Reynolds, "Maximization of a dynamic quadratic interpolation model for production optimization," *SPE Journal*, vol. 18, pp. 1012–1025, 2013.

[196] R. M. Fonseca, A. S. Stordal, O. Leeuwenburgh, P. M. J. Van den Hof, and J. D. Jansen, "Robust ensemble-based multi-objective optimization," in *ECMOR XIV-14th European conference on the mathematics of oil recovery*, vol. 2014, pp. 1–14, European Association of Geoscientists & Engineers, 2014.

[197] G. Bera, K. Camargo, J. L. Sericano, Y. Liu, S. T. Sweet, J. Horney, M. Jun, W. Chiu, I. Rusyn, T. L. Wade, and A. H. Knap, "Baseline data for distribution of contaminants by natural disasters: results from a residential houston neighborhood during hurricane harvey flooding," *Heliyon*, vol. 5, no. 11, p. e02860, 2019.

[198] A. T. Szafran, F. Stossi, M. G. Mancini, C. L. Walker, and M. A. Mancini, "Characterizing properties of non-estrogenic substituted bisphenol analogs using high throughput microscopy and image analysis," *PloS one*, vol. 12, no. 7, 2017.

[199] F. Stossi, M. J. Bolt, F. J. Ashcroft, J. E. Lamerdin, J. S. Melnick, R. T. Powell, R. D. Dandekar, M. G. Mancini, C. L. Walker, J. K. Westwick, and M. A. Mancini, "Defining estrogenic mechanisms of bisphenol a analogs through high throughput microscopy-based contextual assays," *Chemistry & Biology*, vol. 21, no. 6, pp. 743–753, 2014.

[200] Z. D. Sharp, M. G. Mancini, C. A. Hinojos, F. Dai, V. Berno, A. T. Szafran, K. P. Smith, T. T. Lele, D. E. Ingber, and M. A. Mancini, "Estrogen-receptor-α exchange and chromatin dynamics are ligand-and domain-dependent," *Journal of cell science*, vol. 119, no. 19, pp. 4101–4116, 2006.

[201] M. J. Bolt, F. Stossi, A. M. Callison, M. G. Mancini, R. Dandekar, and M. A. Mancini, "Systems level-based rnai screening by high content analysis identifies ubr5 as a regulator of estrogen receptor-α protein levels and activity," *Oncogene*, vol. 34, no. 2, pp. 154–164, 2015.

[202] F. J. Ashcroft, J. Y. Newberg, E. D. Jones, I. Mikic, and M. A. Mancini, "High content imaging-based assay to classify estrogen receptor-α ligands based on defined mechanistic outcomes," *Gene*, vol. 477, no. 1-2, pp. 42–52, 2011.

[203] D. J. Dix, K. A. Houck, M. T. Martin, A. M. Richard, R. W. Setzer, and R. J. Kavlock, "The toxcast program for prioritizing toxicity testing of environmental chemicals," *Toxicological sciences*, vol. 95, no. 1, pp. 5–12, 2007.

[204] S. Nilsson, S. Makela, E. Treuter, M. Tujague, J. Thomsen, G. Andersson, E. Enmark, K. Pettersson, M. Warner, and J.-Å. Gustafsson, "Mechanisms of estrogen action," *Physiological reviews*, vol. 81, no. 4, pp. 1535–1565, 2001.

[205] J. M. Hall, J. F. Couse, and K. S. Korach, "The multifaceted mechanisms of estradiol and estrogen receptor signaling," *Journal of biological chemistry*, vol. 276, no. 40, pp. 36869–36872, 2001.

[206] J. M. Hall and D. P. McDonnell, "Coregulators in nuclear estrogen receptor action," *Molecular interventions*, vol. 5, no. 6, p. 343, 2005.

[207] T. M. Martin, "Prediction of in vitro and in vivo oestrogen receptor activity using hierar-

chical clustering," *SAR and QSAR in Environmental Research*, vol. 27, no. 1, pp. 17–30, 2016.

[208] Y. Chen, F. Cheng, L. Sun, W. Li, G. Liu, and Y. Tang, "Computational models to predict endocrine-disrupting chemical binding with androgen or oestrogen receptors," *Ecotoxicology and environmental safety*, vol. 110, pp. 280–287, 2014.

[209] P. Browne, R. S. Judson, W. M. Casey, N. C. Kleinstreuer, and R. S. Thomas, "Screening chemicals for estrogen receptor bioactivity using a computational model," *Environmental science & technology*, vol. 49, no. 14, pp. 8804–8814, 2015.

[210] J. Li and P. Gramatica, "Classification and virtual screening of androgen receptor antagonists," *Journal of chemical information and modeling*, vol. 50, no. 5, pp. 861–874, 2010.

[211] N. C. Kleinstreuer, P. Ceger, E. D. Watt, M. Martin, K. Houck, P. Browne, R. S. Thomas, W. M. Casey, D. J. Dix, D. Allen, S. Sakamuru, M. Xia, R. Huang, and R. Judson, "Development and validation of a computational model for androgen receptor activity," *Chemical research in toxicology*, vol. 30, no. 4, pp. 946–964, 2017.

[212] M. Chierici, M. Giulini, N. Bussola, G. Jurman, and C. Furlanello, "Machine learning models for predicting endocrine disruption potential of environmental chemicals," *Journal of Environmental Science and Health, Part C*, vol. 36, no. 4, pp. 237–251, 2018.

[213] G. Idakwo, J. Luttrell, M. Chen, H. Hong, Z. Zhou, P. Gong, and C. Zhang, "A review on machine learning methods for in silico toxicity prediction," *Journal of Environmental Science and Health, Part C*, vol. 36, no. 4, pp. 169–191, 2018.

[214] A. T. Szafran and M. A. Mancini, "The myimageanalysis project: a web-based application for high-content screening," *Assay and drug development technologies*, vol. 12, no. 1, pp. 87–99, 2014.

[215] R. Mukherjee, M. Onel, B. Beykal, A. T. Szafran, F. Stossi, M. A. Mancini, L. Zhou, F. A. Wright, and E. N. Pistikopoulos, "Development of the texas a&m superfund research program computational platform for data integration, visualization, and analysis," in *Computer Aided Chemical Engineering*, vol. 46, pp. 967–972, Elsevier, 2019.

[216] C. K. Enders, *Applied missing data analysis*. Guilford press, 2010.

[217] R. Mukherjee, D. Sengupta, and S. K. Sikdar, "Parsimonious use of indicators for evaluating sustainability systems with multivariate statistical analyses," *Clean Technologies and Environmental Policy*, vol. 15, no. 4, pp. 699–706, 2013.

[218] R. Mukherjee, "Selection of sustainable process and essential indicators for decision making using machine learning algorithms," *Process Integration and Optimization for Sustainability*, vol. 1, no. 2, pp. 153–163, 2017.

[219] J. Rogers and S. Gunn, "Identifying feature relevance using a random forest," in *International Statistical and Optimization Perspectives Workshop" Subspace, Latent Structure and Feature Selection"*, pp. 173–184, Springer, 2005.

[220] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[221] B. H. Menze, B. M. Kelm, R. Masuch, U. Himmelreich, P. Bachert, W. Petrich, and F. A. Hamprecht, "A comparison of random forest and its gini importance with standard chemometric methods for the feature selection and classification of spectral data," *BMC bioinformatics*, vol. 10, no. 1, p. 213, 2009.

[222] S. Avraamidou, A. Milhorn, O. Sarwar, and E. N. Pistikopoulos, "Towards a quantitative food-energy-water nexus metric to facilitate decision making in process systems: A case study on a dairy production plant," in *28th European Symposium on Computer Aided Process Engineering* (A. Friedl, J. J. Klemes, S. Radl, P. S. Varbanov, and T. Wallek, eds.), vol. 43 of *Computer Aided Chemical Engineering*, pp. 391–396, Elsevier, 2018.

APPENDIX A

DOMINO SOLUTIONS AND BENCHMARK PROBLEMS

A.1 Best Found Solutions for Benchmark Problems 18, 46 and 47

Problem 18 ("wk_2015_01"):

$x^* = 9.999776$, $y^* = 9.9998$, $f_{best} = 4.5443471 \cdot 10^{-7}$, $F_{best} = 99.9955201008$.

Lower Level Relative Gap: 0 (Retrieved from CPLEX version 12.8.0.0)

Problem 46 ("wk_2015_04"):

$x_1^* = 0, x_2^* = 0, y_1^* = 0, y_2^* = 0, y_3^* = 0, y_4^* = 0, f_{best} = 0, F_{best} = 0$.

Lower Level Relative Gap: $1 \cdot 10^{-9}$ (Retrieved from ANTIGONE version 1.1)

Problem 47 ("wk_2015_06"):

$x_1^* = 0.000984369218350, x_2^* = -0.001021751016379, x_3^* = 1.663984077237546, x_4^* = -0.076938496530056, y_1^* = -1.0187598163, y_2^* = 1.0574476104, y_3^* = -0.0004531744, y_4^* = 0, f_{best} = -5, F_{best} = 0.0000045078$.

Lower Level Relative Gap: $1.76 \cdot 10^{-7}$ (Retrieved from BARON version 18.11.12)

A.2 Randomly Generated Benchmark Problems Using B-POP

The randomly generated 61 benchmark studies using the bi-level random problem generator in B-POP toolbox [77] are provided online at http://parametric.tamu.edu/POP/. For example, for the following bi-level optimization problem, a corresponding text file (Figure A.1) is provided.

$$
\begin{aligned}
&\min_{x} \qquad F(x,y) = 2y_1/5 + 2y_2/5 + 3x_1 + 2x_2/5 \\
&\text{s.t.} \\
&\quad \min_{y} \qquad f(x,y) = 5 \cdot y_1/2 + 5 \cdot y_2 + x_2 \\
&\quad \text{s.t.} \\
&\quad -(202^{0.5} \cdot (3y_1 - 12x_1 - 7x_2 + 13))/202 \leq 0 \\
&\quad -(373^{0.5} \cdot (2y_1 - 15x_1 - 12x_2 + 6))/373 \leq 0 \\
&\qquad x \in [-10, 10]^2, \qquad y \in [-10, 10]^2
\end{aligned}
\tag{A.1}
$$

```
=====================================================================
================================
Supplementary Material
DOMINO: Data-driven Optimization of bi-level Mixed-Integer NOnlinear
Problems
Burcu Beykal, Styliani Avraamidou, Ioannis P.E. Pistikopoulos, Melis
Onel, Efstratios N. Pistikopoulos
=====================================================================
================================
Benchmark Problem Definitions from B-POP: LPLP1

Upper Level Continuous Variables: x1, x2;

Lower Level Continuous Variables: y1, y2;

Lower Level Binary Variables:;

min F(x,y) =  (2*y1)/5 + (2*y2)/5 + 3*x1 + (2*x2)/5;
 x
s.t.
         min f(x,y) = (5*y1)/2 + 5*y2 + x2;
          y
         s.t.
             -(202^(1/2)*(3*y1 - 12*x1 - 7*x2 + 13))/202 <= 0;
             -(373^(1/2)*(2*y1 - 15*x1 - 12*x2 + 6))/373 <= 0;

Bounds on the Upper Level Variables:
x1.lo = -10;
x1.up = 10;

x2.lo = -10;
x2.up = 10;

Bounds on the Lower Level Variables:
y1.lo = -10;
y1.up = 10;

y2.lo = -10;
y2.up = 10;
```

Figure A.1: An example problem definition file for the "LPLP1" benchmark problem.

APPENDIX B

FOOD-ENERGY-WATER NEXUS MODEL FOR LAND ALLOCATION

B.1 Notation for the Food-Energy-Water Nexus Case Study

e	efficiency
$energy$	energy
max	maximum
min	minimum
$profit$	profit
$total$	total
$trans$	transportation
H_2O	water

B.2 List of Land Processes Considered in the Food-Energy-Water Nexus Case Study

Energy Land Processes

1. Solar Energy
2. Wind Energy

Agricultural Processes

3. Fruit Production
4. Vegetable Production
5. Livestock Grazing

B.3 Agricultural Developer's Problem

The chosen land allocation problem considers a piece of land which will be processed by an agricultural developer over 4 seasons in a climate similar to that of Texas, U.S. and is divided into 8 equal (1 km^2) plots. The nomenclature for this problem is provided in Table B.1. On each piece of land, a subset of agricultural and energy land processes can occur, where fruit production, vegetable production, and livestock grazing are representatives of agricultural processes defined by the subset T_A, whereas solar energy and wind energy are representatives of energy land processes, defined by the subset T_E. Two important properties regarding these subsets are given in Equations B.1 and B.2.

$$T_A \cup T_E = T_L \tag{B.1}$$

$$T_A \cap T_E = \emptyset \tag{B.2}$$

Table B.1: Nomenclature for the Food-Energy-Water Nexus case study.

Type	Name	Description
Indices	$i \in \{1, 2, ..., I\}$	land processes ($card(i) = 5$)
	$j \in \{1, 2, ..., J\}$	land plot square number ($card(j) = 8$)
	$k \in \{1, 2, ..., K\}$	seasons in a Texas-type climate ($card(k) = 4$)
Sets	T_L	land use types
	$T_A \subset T_L$	agriculture land use type ($card(T_A) = 3$)
	$T_E \subset T_L$	energy land use type ($card(T_E) = 2$)
Binary Variables	$y_{i,j,k}$	activates the i^{th} process that occurs on the j^{th} plot in the k^{th} season
	$y_j^{H_20}$	activates water availability on the j^{th} plot
	$y_{i,j,k}^{trans,H_2O}$	activates water transportation that is required for the i^{th} process on the j^{th} plot in the k^{th} season, where $i \in T_A$
Parameters	$P_{i,k}^{e}$	efficiency multiplier of the i^{th} land process in the k^{th} season
	$P_{i,k}^{profit}$	profit multiplier of the i^{th} land process for the k^{th} season, where $i \in T_E$
	$D_k^{H_2O}$	multiplier of minimum water required for the k^{th} season
	C_k^{trans,H_2O}	water transportation cost multiplier for the k^{th} season

Table B.1: Continued.

Type	Name	Description
	$L_i^{H_2O}$	lower bound on water transportation and consumption for the i^{th} land process in kg, where $i \in T_A$
	$U_i^{H_2O}$	upper bound on water transportation and consumption for the i^{th} land process in kg, where $i \in T_A$
	L_i^{energy}	lower bound on energy consumption for the i^{th} land process in kWh, where $i \in T_A$
	U_i^{energy}	upper bound on energy consumption for the i^{th} land process in kWh, where $i \in T_A$
	M_i^{energy}	metric ton of yield per kWh energy consumed for the i^{th} land process, where $i \in T_A$
	$M_i^{H_2O}$	metric ton of yield per kg of water consumption in the i^{th} land process, where $i \in T_A$
	M_i^{profit}	profit made from the i^{th} land process per unit energy produced in k$/kWh when $i \in T_E$ and profit made from i^{th} land process per unit yield obtained in k$/ton when $i \in T_A$
	B_i	government budget allocated for supporting the i^{th} land process type in k$
	BM	Big-M parameter
Continuous Variables	$EP_{i,j,k}$	energy produced by the i^{th} land process type on the j^{th} plot during the k^{th} season in kWh, where $i \in T_E$
	$EC_{i,j,k}$	energy consumed by the i^{th} land process type on the j^{th} plot during the k^{th} season in kWh, where $i \in T_A$
	$W_{i,j,k}$	water consumed from an existing source by the i^{th} land process type on the j^{th} plot during the k^{th} season in kg, where $i \in T_A$
	$W_{i,j,k}^{trans}$	water consumed from a transported source by the i^{th} land process type on the j^{th} plot during the k^{th} season in kg, where $i \in T_A$
	$Y_{i,j,k}$	yield produced by the i^{th} land process type on the j^{th} plot during the k^{th} season in metric tonnes, where $i \in T_A$
	$G_{i,j,k}^{profit}$	profit gained by the i^{th} land process type on the j^{th} plot during the k^{th} season in k$
	S_i	subsidies offered by the government for using the i^{th} land process
	$\acute{S}_{i,j,k}$	variable introduced in the Big-M formulation for replacing the bi-linear term $S_i \cdot y_{i,j,k}$
	E^{total}	total energy gained from the land in kWh
	Y^{total}	total yield gained from the land in metric tonnes
	W^{total}	total water consumed on the land in kg
	$G^{profit,total}$	total profit gained from the land in k$

Table B.2: Land properties for the case study. These limit the processes that can occur on each plot over 4 seasons, defined by the binary variable $y_{i,j,k}$. The water availability is defined by the binary variable $y_j^{H_20}$. 1 indicates existence and 0 indicates absence of that property.

Land Properties for all seasons ($\forall k$)	Land Plot Number (j)							
	1	2	3	4	5	6	7	8
Good Soil ($y_{i,j,k}\ \forall i \in T_A$)	1	1	1	1	0	0	1	1
Adequate Sun ($y_{1,j,k}$)	0	0	1	1	1	1	1	1
Adequate Wind ($y_{2,j,k}$)	1	1	1	1	1	1	0	0
Water Available ($y_j^{H_20}$)	0	0	0	0	1	1	0	1

The agricultural producer will be subject to various constraints regarding the properties of the land, the properties of the agricultural and energy production processes while making an optimal decision towards its own objective. First, the land characteristics will affect the selection of any process that can occur in each land plot. If good soil is not available in a plot section, agricultural processes are restricted to not to take place in that land section for all seasons. If the adequate sun is not available in a plot section, solar energy will not be implemented in that land section for all seasons. Finally, if a plot section does not have access to the adequate amount of wind, wind energy production will not be implemented in that land section for all seasons. These characteristics are summarized in Table B.2. Based on this information, constraints regarding water transportation can be defined for the problem such as water must be transported to the land if there is no water on a plot and an agricultural process is selected to occur on that plot:

$$y_{i,j,k}^{trans,H_2O} \leq y_{i,j,k} + y_j^{H_20} \qquad \forall i \in T_A, j, k \tag{B.3}$$

No water will be transported, if water is already available on the plot:

$$y_{i,j,k}^{trans,H_2O} \leq 1 - y_j^{H_20} \qquad \forall i \in T_A, j, k \tag{B.4}$$

No water should be transported, if there is no water on the plot and no agricultural process is

selected to occur on that plot:

$$y_{i,j,k}^{trans,H_2O} \geq y_{i,j,k} - y_j^{H_20} \qquad \forall i \in T_A, j, k \tag{B.5}$$

In addition to the land properties, there are other constraints that further influence the selection of land processes and restrict the feasible space for this case study. The constraints regarding the selection of land processes is imposed such that at least one land process must be allocated on each plot.

$$\sum_{i \in I} y_{i,j,k} \geq 1 \qquad \forall j, k \tag{B.6}$$

Furthermore, it is not practical to have solar panels and agricultural production on the same plot. Thus, at most one out of solar energy, fruit, vegetables and livestock can be allocated in one plot:

$$\sum_{i \neq 2, i \in T_L} y_{i,j,k} \leq 1 \qquad \forall j, k \tag{B.7}$$

Wind energy will occupy minimal space on the land plot, compared to solar energy production systems, hence both wind energy and either fruit or vegetable production can be allocated on the same plot:

$$\sum_{i=2}^{4} y_{i,j,k} \leq 2 \qquad \forall j, k \tag{B.8}$$

Moreover, only one energy process is allowed on a plot:

$$\sum_{i \in T_E} y_{i,j,k} \leq 1 \qquad \forall j, k \tag{B.9}$$

If an energy process is selected in a plot, the type of energy production will stay the same throughout the year, since it is too expensive to move equipment over seasons:

$$y_{i,j,k+1} \geq y_{i,j,k} \qquad \forall i \in T_E, j, k \leq card(k) - 1 \tag{B.10}$$

Second, the seasonal differences must be considered, as these can impact the energy demand, water transportation cost, water availability for irrigation and efficiency of energy production processes. For example, in seasons with rainfall, such as winter, spring and fall, the transportation cost for water will be less and less water will be required for irrigation. On the other hand, the solar systems will have lower efficiency due to the reduced amount of sunshine throughout these seasons. A similar analysis is also done for the summer, where there is going to be greater demand for energy and water, and higher transportation costs for water will be in effect. However, the solar systems will have greater efficiency since there will be plenty of sunshine during summer. Hence, both spatial and time scenarios are considered and their respective parameters are included in the model equations (for the parameters please see Tables B.3-B.6).

The land processes will be quantified on the amount of energy produced or agricultural yield, if an energy or an agricultural process is selected, respectively. It is important to note that, if an energy process is selected for a given plot in a given season, a fixed amount of energy can be produced from these technologies:

$$\begin{aligned} EP_{1,j,k} &= P^e_{1,k} \cdot 50 \cdot y_{1,j,k} \qquad \forall j,k \\ EP_{2,j,k} &= P^e_{2,k} \cdot 1000 \cdot y_{2,j,k} \qquad \forall j,k \end{aligned} \tag{B.11}$$

Likewise, the yield for agricultural processes can be calculated as a function of water and energy consumption. The parameter $P^e_{i,k}$ is used to take in consideration the changes in efficiency of land processes over different seasons.

$$Y_{i,j,k} = P^e_{i,k}\big(M_i^{energy} \cdot EC_{i,j,k} + M_i^{H_2O} \cdot W_{i,j,k}\big) \qquad \forall i \in T_A, j, k \tag{B.12}$$

The amount of energy consumption and water consumption (from an already existing source) by agricultural processes, which are used to calculate the yield in Equation B.12, are bounded. Note that the lower bound on the water consumption depends on seasonal effects (dry seasons versus

seasons with rainfall), hence multiplied by its respective parameter, $D_k^{H_2O}$.

$$\begin{aligned} L_i^{energy} \cdot y_{i,j,k} \leq EC_{i,j,k} \leq U_i^{energy} \cdot y_{i,j,k} \qquad \forall i \in T_A, j, k \\ D_k^{H_2O} \cdot L_i^{H_2O} \cdot y_{i,j,k} \leq W_{i,j,k} \leq U_i^{H_2O} \cdot y_{i,j,k} \qquad \forall i \in T_A, j, k \end{aligned} \tag{B.13}$$

In addition to the box-constraints, it is important to supply adequate amount of water to each plot in each season for the agricultural land processes. Thus, the amount of water consumption (source-based and transportation-based) is set to be at least 200 times greater than the energy consumption in each plot and in each season:

$$\sum_{i \in T_A} W_{i,j,k} + D_k^{H_2O} \cdot \sum_{i \in T_A} W_{i,j,k}^{trans} \geq 200 \cdot \sum_{i \in T_A} EC_{i,j,k} \qquad \forall j, k \tag{B.14}$$

The amount of water transported for agricultural processes is also bounded and affected by the seasonal differences:

$$D_k^{H_2O} \cdot L_i^{H_2O} \cdot y_{i,j,k}^{trans,H_2O} \leq W_{i,j,k}^{trans} \leq U_i^{H_2O} \cdot y_{i,j,k}^{trans,H_2O} \qquad \forall i \in T_A, j, k \tag{B.15}$$

As described previously in Chapter 2, Section 2.3.2, the objective of the agricultural developer is to maximize its profit. The profit calculation for all land processes includes the money made from energy production and the yield from the agricultural processes, if an energy or an agricultural process is selected, respectively. For energy producing land processes profit is given as:

$$G_{i,j,k}^{profit} = M_i^{profit} \cdot P_{i,k}^{profit} \cdot EP_{i,j,k} + \acute{S}_{i,j,k} \qquad \forall i \in T_E, j, k \tag{B.16}$$

For agricultural processes, the profit is given as:

$$G_{i,j,k}^{profit} = M_i^{profit} \cdot Y_{i,j,k} + \acute{S}_{i,j,k} \qquad \forall i \in T_A, j, k \tag{B.17}$$

The profit calculations also considers the relevant subsidies ($\acute{S}_{i,j,k}$) offered by the government

agencies for developing different processes on the land, where these subsidies should only be considered in the profit when their respective land process is activated.

$$\acute{S}_{i,j,k} = S_i \cdot y_{i,j,k} \qquad \forall i, j, k \tag{B.18}$$

To avoid this bilinear term that appears in the profit equation, the variable $\acute{S}_{i,j,k}$ and its Big-M formulation is introduced in Equations B.18-B.21, where BM is the Big-M parameter.

$$S_i \leq BM \cdot \sum_j \sum_k y_{i,j,k} \qquad \forall i \tag{B.19}$$

$$\acute{S}_{i,j,k} \leq BM \cdot y_{i,j,k} \qquad \forall i, j, k \tag{B.20}$$

$$\acute{S}_{i,j,k} \leq S_i \qquad \forall i, j, k \tag{B.21}$$

Moreover, the agricultural developer is interested in maximizing the total profit, which is a function of the total energy production, total yield from agricultural production and total water consumption. The total energy, E^{total}, is defined as the difference between total energy produced from energy land processes and total energy consumed by the agricultural processes in all plots throughout the 4 seasons.

$$E^{total} = \sum_{i \in T_E} \sum_j \sum_k EP_{i,j,k} - \sum_{i \in T_A} \sum_j \sum_k EC_{i,j,k} \tag{B.22}$$

Similarly, the total yield, Y^{total}, is the summation of yield of all agricultural processes over all plots and 4 seasons.

$$Y^{total} = \sum_{i \in T_A} \sum_j \sum_k Y_{i,j,k} \tag{B.23}$$

The total water consumption, W^{total}, includes both the amount of water consumed from a natural source (i.e. water already existing as in the land properties, given in Table B.2) and from a transported source. The transported total water also considers seasonal demand, defined by the

parameter $D_k^{H_2O}$.

$$W^{total} = \sum_{i \in T_A} \sum_{j} \sum_{k} W_{i,j,k} + \sum_{k} D_k^{H_2O} \sum_{i \in T_A} \sum_{j} W_{i,j,k}^{trans} \tag{B.24}$$

The total profit, $G^{profit,total}$, is calculated by subtracting the total water transportation cost throughout all plots, all seasons and all agricultural land processes from the cumulative profit from all land processes. The cost of water transportation is assumed to be \$10/kg of water. In addition, the cost of transportation is impacted by seasonal differences, as explained previously, hence the formulation includes the $C_k^{H_2O,trans}$ parameter to account for such effects. The objective function of the LLP is given as:

$$G^{profit,total} = \sum_{i} \sum_{j} \sum_{k} G_{i,j,k}^{profit} - 0.01 \cdot \sum_{k} C_k^{H_2O,trans} \sum_{i \in T_A} \sum_{j} W_{i,j,k}^{trans} \tag{B.25}$$

Finally, the continuous variables defined in Equations B.22-B.25 are bounded and their respective values are obtained through minimizing and maximizing each variable as the sole objective to the land allocation problem.

$$\begin{aligned} 0 &\leq W^{total} \leq 2.46 \cdot 10^9 \\ 0 &\leq Y^{total} \leq 13860 \\ 0 &\leq E^{total} \leq 21945 \\ G^{profit,total} &\geq 0 \end{aligned} \tag{B.26}$$

The variables defined in Equations B.22-B.25 as well as their respective bounds, provided in Equation B.26, are used to enumerate the upper level objective function of the government regulators. The ULP is discussed in detail in the following section.

B.4 Government Regulators' Problem

As shown in Equation 2.2, the objective of the government regulators is to minimize the nexus stress. However, the mathematical quantification of the nexus, which will take in consideration of the trade-offs between food, energy and water, has not yet been fully established. Recently,

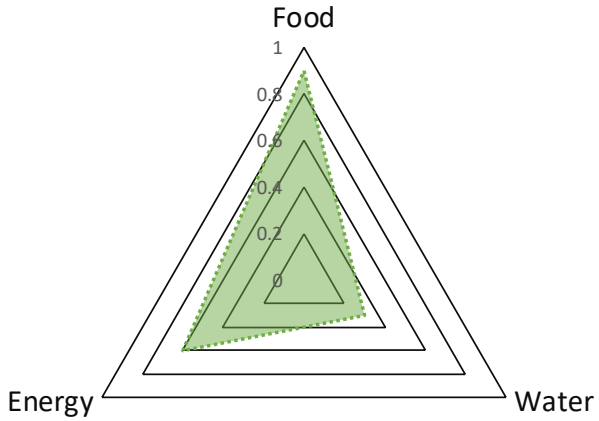

Figure B.1: FEW-N metric represented as the area of a triangle. Shaded area demonstrates an example solution to FEW-N.

Avraamidou et al. [222] has introduced a methodology to develop a FEW-N metric, which brings relevant decision elements and their respective quantification together through rth order averaging. In this work, we adopt this idea through a similar methodology where a single geometric metric, i.e. the area of a triangle, is used to represent the FEW-N metric as the government regulators' objective. An illustration of the FEW-N metric is provided in Figure B.1.

In Figure B.1, the corners of the triangle represent the scaled quantities of each FEW-N element, where their respective values lie between 0 and 1. In this case, a value of 1 represents the best possible scenario and 0 represents the worst. The objective of the government regulators is to maximize the best possible scenario for each element, namely minimizing the total water consumed and maximizing the total energy and food produced, which essentially translates into maximizing the area of the triangle. The explicit formulation of this objective is provided in Equation B.27.

$$FEW_{metric} = \left[\frac{E^{total} - E^{min}}{E^{max} - E^{min}} \cdot \left(1 - \frac{W^{total} - W^{min}}{W^{max} - W^{min}} \right) + \frac{E^{total} - E^{min}}{E^{max} - E^{min}} \cdot \frac{Y^{total} - Y^{min}}{Y^{max} - Y^{min}} + \left(1 - \frac{W^{total} - W^{min}}{W^{max} - W^{min}} \right) \cdot \frac{Y^{total} - Y^{min}}{Y^{max} - Y^{min}} \right] \cdot \frac{\sin 120^{\circ}}{2} \tag{B.27}$$

Note that E^{total}, Y^{total}, and W^{total} is obtained through solving the agricultural producer's problem, explicitly defined in Equations B.22-B.24, respectively.

In this case study, the government is offering subsidies (S_i) to the land developers for each nexus element, as much as their budget (B_i) allows.

$$0 \leq S_i \leq B_i \qquad \forall i \tag{B.28}$$

These subsidies further motivate the land owner to properly allocate and utilize the land to maximize their own profit (Equations B.16-B.17). The upper bound on the total governmental budget is set to be \$250M where this is allocated equally among all land processes. Essentially, the goal of the government agency is to decide on the amount of subsidies to be offered to the agricultural producer in such a way that the objective function defined in Equation B.27 is maximized.

B.5 Parameters

Parameter values are tabulated in Tables B.3-B.6, where 4 seasons (autumn, winter, spring, and summer) are considered for the FEW-N case study with production starting in autumn and ending after summer. These parameters are used as multipliers to capture seasonal differences among technological efficiencies, water demand and transportation costs. The efficiency of the solar energy production process is lower in autumn and winter whereas it is higher in the summer. Likewise, the efficiency of agricultural processes is lower in winter as shown in Table B.3.

Table B.3: Parameter values for $P^e_{i,k}$

	k			
i	1 (Autumn)	2 (Winter)	3 (Spring)	4 (Summer)
1 (solar energy)	0.85	0.70	0.90	1.00
2 (wind energy)	0.90	1.00	0.90	0.80
3 (fruit production)	1.00	0.85	1.00	1.00
4 (vegetable production)	1.00	0.85	1.00	1.00
5 (livestock grazing)	1.00	0.85	1.00	1.00

Table B.4: Parameter values for $P_{i,k}^{profit}$

	k			
i	1 (Autumn)	2 (Winter)	3 (Spring)	4 (Summer)
1 (solar energy)	1.00	1.20	1.00	1.20
2 (wind energy)	1.00	1.20	1.00	1.20

Table B.5: Parameter values for $D_k^{H_2O}$ and C_k^{trans,H_2O}

k	1 (Autumn)	2 (Winter)	3 (Spring)	4 (Summer)
$D_k^{H_2O}$	1.00	0.70	1.00	1.20
$C_k^{H_2O,trans}$	1.00	1.00	1.00	1.30

The profit from energy production during winter and summer should be higher since there would be higher demand for energy in very cold and hot weathers. Hence, higher multipliers are assigned for both energy production land processes, which are summarized in Table B.4.

Table B.5 summarizes the multipliers for the minimum amount of water required as well as the cost of transporting water over 4 seasons. Both the required amount of water and the cost of transportation is expected to be higher in summertime due to elevated temperatures and higher demand for water in agricultural production. Finally, Table B.6 summarizes other parameters used in the FEW-N case study.

Table B.6: Parameter values for $L_i^{H_2O}$, $U_i^{H_2O}$, L_i^{energy}, U_i^{energy}, M_i^{energy}, $M_i^{H_2O}$ and M_i^{profit}

i	1 (solar energy)	2 (wind energy)	3 (fruit production)	4 (vegetable production)	5 (livestock grazing)
$L_i^{H_2O}$	-	-	100	100	10^4
$U_i^{H_2O}$	-	-	10^6	10^6	10^8
L_i^{energy}	-	-	5	5	5
U_i^{energy}	-	-	50	50	100
M_i^{energy}	-	-	10	10	1
$M_i^{H_2O}$	-	-	10^{-4}	15^{-4}	40^{-4}
M_i^{profit}	100	100	2	1.3	5

APPENDIX C

DYNAMIC STEAM CRACKING OPTIMIZATION MODEL FOR ETHYLENE AND PROPYLENE PRODUCTION

C.1 Model Equations for Ethane and Propane Cracking

The tubular steam cracking reactor model considers the mass (Equation C.1), energy (Equation C.3), and momentum (Equation C.7) balances, which are further explained in the relevant subsections. The full notation of the model equations are provided in Section C.1.1.

C.1.1 Notation

ΔH_i	Enthalpy of reaction i, $J/kmol$
ϵ	Pipe roughness, m
μ^p	Pure component viscosity, $Pa \cdot s$
μ_m	Viscosity of gas mixture, $Pa \cdot s$
ν_i	Order of reaction i
$\overline{C}$	Average mass concentration, kg/m^3
$\overline{T}$	Average temperature, K
ρ_c	Density of coke, kg/L
τ	Reactor runtime before decoking cycle begins, hr
A_o	Pre-exponential factor
Cp_j	Heat capacity of species j, $J/kmol \cdot K$
D_t^{new}	Tube diameter after coking, m
D_t	Tube diameter, m

E_a^{coke}	Activation energy of coking reaction, $kcal/mol$
E_a	Activation energy, $J/kmol$
F_j	Molar flowrate of species j, $kmol/s$
Fr	Friction factor, m^{-1}
G_m	Mass flux of gas mixture, $kg/m^2 \cdot s$
i	Reaction index
j, jj	Species index
J_{coke}	Subset of species j identified as coke precursor in a given cracking model
k_{coke}	Reaction rate constant for coking reaction
M_m	Mean molecular weight of gas mixture, $kg/kmol$
MW_j	Molecular weight of species j, $kg/kmol$
obj^j	Objective function for product j, $\$/s$
P	Pressure, Pa
$P^{\$}_{decoking}$	Decoking cost, $\$/cycle$
$P^{\$}_{heat}$	Heating cost, $\$/J$
$P^{\$}_{inv}$	Investment cost, $\$/m \cdot s$
$P^{\$}_j$	Price of species j, $\$/kg$
$Q(z)$	External heat flux, W/m^2
Q^{total}	Area under the heat flux vs. length curve, W/m
R	Gas constant, $J/kmol \cdot K$
Re	Reynolds Number

r_i	Rate of reaction i
Red	Reduction percentage in diameter per hour, hr^{-1}
s_{ij}	Stoichiometric coefficient of species j in reaction i
T	Temperature, K
t_s	Timestep, s
V	Volumetric flowrate, m^3/s
x_j	Mole fraction of species j
z	Spatial coordinate

C.1.2 Mass Balance

The molecular reaction scheme and their respective kinetic parameters for ethane and propane cracking are provided in Tables C.1 and C.2. For both cracking models, the reactions are assumed to follow Arrhenius-type kinetics (Equation C.2) [14]. The steam cracking of ethane and propane is modeled based on the kinetic information provided in [14–17, 165]. For ethane cracking, 11 molecular species and 15 reactions are considered, whereas for the propane cracking 9 molecular species and 13 reactions are considered in the model.

$$\frac{dF_j}{dz} = -(\sum_i s_{ij} r_i)\frac{\pi D_t^2}{4} \qquad \forall j \tag{C.1}$$

$$r_i = A_o \cdot exp(-\frac{E_a}{RT}) \prod_{j \in Rxn} [\frac{F_j}{V}]^{\nu_i} \qquad \forall i \tag{C.2}$$

C.1.3 Energy Balance

The heat balance, shown in Equation C.3, includes the external heat flux term, $Q(z)$, provided to the reactor to enable the endothermic cracking reactions. Previously, Tarafder et al. [166] explored polynomial external heat flux trajectories over the spatial coordinate, z, whereas Onel

Table C.1: Molecular reaction scheme and their respective kinetic parameters for ethane cracking. The reaction mechanisms are adapted from [14–17].

Reaction Scheme, i	A_o (s^{-1} or 1 $mol^{-1}s^{-1}$)	E_a (kcal/mol)
$C_2H_6 \xrightarrow{k_1} C_2H_4 + H_2$	$4.652 \cdot 10^{13}$	65.20
$C_2H_4 + H_2 \xrightarrow{k_2} C_2H_6$	$8.490 \cdot 10^{8}$	32.62
$2\,C_2H_6 \xrightarrow{k_3} C_3H_8 + CH_4$	$3.850 \cdot 10^{11}$	65.25
$C_3H_6 \xrightarrow{k_4} C_2H_2 + CH_4$	$3.794 \cdot 10^{11}$	59.39
$C_2H_2 + CH_4 \xrightarrow{k_5} C_3H_6$	$1.990 \cdot 10^{7}$	29.23
$C_2H_2 + C_2H_4 \xrightarrow{k_6} C_4H_6$	$1.026 \cdot 10^{12}$	41.26
$C_2H_4 + C_2H_6 \xrightarrow{k_7} C_3H_6 + CH_4$	$7.083 \cdot 10^{13}$	60.43
$C_2H_4 + C_4H_6 \xrightarrow{k_8} C_6H_6 + 2\,H_2$	$8.385 \cdot 10^{9}$	34.56
$C_3H_8 \xrightarrow{k_9} C_3H_6 + H_2$	$5.888 \cdot 10^{10}$	51.29
$C_3H_6 + H_2 \xrightarrow{k_{10}} C_3H_8$	$9.030 \cdot 10^{5}$	22.78
$C_3H_8 + C_2H_4 \xrightarrow{k_{11}} C_2H_6 + C_3H_6$	$2.536 \cdot 10^{13}$	59.06
$2\,C_3H_6 \xrightarrow{k_{12}} 3\,C_2H_4$	$1.514 \cdot 10^{11}$	55.80
$nC_4H_{10} \xrightarrow{k_{13}} C_4H_8 + H_2$	$1.637 \cdot 10^{12}$	62.36
$C_4H_8 + H_2 \xrightarrow{k_{14}} nC_4H_{10}$	$1.780 \cdot 10^{7}$	32.30
$C_3H_6 + C_2H_6 \xrightarrow{k_{15}} C_4H_8 + CH_4$	$5.553 \cdot 10^{14}$	60.01

Table C.2: Molecular reaction scheme and their respective kinetic parameters for propane cracking. The reaction mechanisms are adapted from [14, 15, 17].

Reaction Scheme, i	A_o (s^{-1} or 1 $mol^{-1}s^{-1}$)	E_a (kcal/mol)
$C_2H_6 \xrightarrow{k_1} C_2H_4 + H_2$	$4.652 \cdot 10^{13}$	65.20
$C_2H_4 + H_2 \xrightarrow{k_2} C_2H_6$	$8.490 \cdot 10^{8}$	32.62
$C_3H_6 \xrightarrow{k_3} C_2H_2 + CH_4$	$3.794 \cdot 10^{11}$	59.39
$C_2H_2 + CH_4 \xrightarrow{k_4} C_3H_6$	$1.990 \cdot 10^{7}$	29.23
$C_2H_2 + C_2H_4 \xrightarrow{k_5} C_4H_6$	$1.026 \cdot 10^{12}$	41.26
$C_2H_4 + C_2H_6 \xrightarrow{k_6} C_3H_6 + CH_4$	$7.083 \cdot 10^{13}$	60.43
$C_3H_8 \xrightarrow{k_7} C_3H_6 + H_2$	$5.888 \cdot 10^{10}$	51.29
$C_3H_6 + H_2 \xrightarrow{k_8} C_3H_8$	$9.030 \cdot 10^{5}$	22.78
$C_3H_8 + C_2H_4 \xrightarrow{k_9} C_2H_6 + C_3H_6$	$2.536 \cdot 10^{13}$	59.06
$2\,C_3H_6 \xrightarrow{k_{10}} 3\,C_2H_4$	$1.514 \cdot 10^{11}$	55.80
$C_3H_6 + C_2H_6 \xrightarrow{k_{11}} C_4H_8 + CH_4$	$5.553 \cdot 10^{14}$	60.01
$C_3H_8 \xrightarrow{k_{12}} C_2H_4 + CH_4$	$4.692 \cdot 10^{10}$	50.60
$2\,C_3H_6 \xrightarrow{k_{13}} 0.5\,C_6 + 3\,CH_4$	$1.423 \cdot 10^{9}$	45.50

[17] implemented a piece-wise constant external heat flux over a fixed reactor length. In this work, a similar piece-wise constant external heat flux trajectory is implemented, but over a continuously varying reactor length and with a wider range of operating conditions. In this approach, the reactor is assumed to be divided into 5 distinctive regions, where each region can be supplied with a different constant heat flux.

$$\frac{dT}{dz} = \frac{Q(z)\pi D_t + \pi \frac{D_i^2}{4} \sum_i r_i(-\Delta H_i)}{\sum_j F_j Cp_j} \tag{C.3}$$

The energy balance expression considers the enthalpy of reaction, ΔH_i:

$$\Delta H_i = \Delta H_i^o + \int_T^{298} \sum_{j \in Reactants} Cp_j \cdot |s_{i,j}| dT + \int_{298}^{T} \sum_{j \in Products} Cp_j \cdot s_{i,j} dT \qquad \forall i \tag{C.4}$$

Table C.3: Formation enthalpy of species considered in thermal cracking of ethane and propane [18].

Molecular Specie, j	ΔH^o_{fj} (kJ/mol)
Methane, CH_4	-74.87
Acetylene, C_2H_2	226.73
Ethylene, C_2H_4	52.47
Ethane, C_2H_6	-84
Propylene, C_3H_6	20.41
Propane, C_3H_8	-104.7
Butadiene, C_4H_6	108.8
Butene, C_4H_8	-0.63
Butane, C_4H_{10}	-125.6
Benzene*, C_6H_6	82.9
Hydrogen, H_2	0
Water, H_2O	-241.826

*Taken for C_6

Hess' law is used for calculating the standard enthalpy of reaction in Equation C.4:

$$\Delta H^o_i = \sum_j s_{i,j} \cdot \Delta H^o_{fj} \qquad \forall i \tag{C.5}$$

In Equation C.5, the formation enthalpy, ΔH^o_{fj}, of molecular species is retrieved from the gas phase thermochemistry data provided at NIST [18]. Furthermore, the heat capacity is calculated using the ideal gas state heat capacity polynomial (Equation C.6) provided in the textbook by Smith et al. [19]. The values of the polynomial coefficients and the formation enthalpy of all molecular species are listed in Tables C.4 and C.3, respectively.

$$\frac{Cp_j}{R} = A_j + B_j \cdot T + C_j \cdot T^2 + \frac{D_j}{T^2} \qquad \forall j \tag{C.6}$$

Table C.4: Coefficients of the polynomial $Cp_j/R = A_j + B_j \cdot T + C_j \cdot T^2 + D_j \cdot T^{-2}$ for the calculation of the heat capacity in ideal gas state [19].

Molecular Specie, j	A	B	C	D
Methane, CH_4	1.702	$9.081 \cdot 10^{-3}$	$-2.164 \cdot 10^{-6}$	0
Acetylene, C_2H_2	6.132	$1.952 \cdot 10^{-3}$	0	$-1.299 \cdot 10^{5}$
Ethylene, C_2H_4	1.424	$14.394 \cdot 10^{-3}$	$-4.392 \cdot 10^{-6}$	0
Ethane, C_2H_6	1.131	$19.225 \cdot 10^{-3}$	$-5.561 \cdot 10^{-6}$	0
Propylene, C_3H_6	1.637	$22.706 \cdot 10^{-3}$	$-6.915 \cdot 10^{-6}$	0
Propane, C_3H_8	1.213	$28.785 \cdot 10^{-3}$	$-8.824 \cdot 10^{-6}$	0
Butadiene, C_4H_6	2.734	$26.786 \cdot 10^{-3}$	$-8.882 \cdot 10^{-6}$	0
Butene, C_4H_8	1.967	$31.630 \cdot 10^{-3}$	$-9.873 \cdot 10^{-6}$	0
Butane, C_4H_{10}	1.935	$36.915 \cdot 10^{-3}$	$-11.402 \cdot 10^{-6}$	0
Benzene*, C_6H_6	-0.206	$39.064 \cdot 10^{-3}$	$-13.301 \cdot 10^{-6}$	0
Hydrogen, H_2	3.249	$0.422 \cdot 10^{-3}$	0	$0.083 \cdot 10^{5}$
Water, H_2O	3.470	$1.450 \cdot 10^{-3}$	0	$0.121 \cdot 10^{5}$

*Taken for C_6

C.1.4 Momentum Balance

The momentum balance equation (Equation C.7) considers the mean molecular weight (M_m), the tube pressure (P), the friction factor (Fr) and the total mass flux (G_m).

$$\frac{dP}{dz} = \frac{\frac{d}{dz}(\frac{1}{M_m}) + \frac{1}{M_m}(\frac{1}{T}\frac{dT}{dz} + Fr)}{\frac{1}{M_m P} - \frac{P}{G_m^2 RT}} \tag{C.7}$$

Total mass flux and mean molecular weight of the gas mixture are expressed respectively as:

$$G_m = \frac{\sum_j F_j \cdot MW_j}{\pi \cdot D_t^2/4} \tag{C.8}$$

$$M_m = \frac{\sum_j F_j \cdot MW_j}{\sum_j F_j} \tag{C.9}$$

The friction factor is calculated using Equation C.10, where f is calculated using the average of the Swamee-Jain and Halaand approximations for the Colebrook equation, where $\epsilon = 4.572 \cdot 10^{-5}$

m for steel pipe.

$$Fr = \frac{2f}{D_t} \tag{C.10}$$

$$f = 0.125 \cdot \left[0.25 \cdot \left(\log \left[\frac{\epsilon/D_t}{3.7} + \frac{5.74}{Re^{0.9}} \right] \right)^{-2} + \left(-1.8 \cdot \log \left[\left(\frac{\epsilon/D_t}{3.7} \right)^{1.11} + \frac{6.9}{Re} \right] \right)^{-2} \right] \tag{C.11}$$

The Reynolds number of the mixture is calculated using the following equation:

$$Re = \frac{\sum_j F_j \cdot MW_j}{\pi D_t/4 \cdot \mu_m} \tag{C.12}$$

The viscosity of the gas mixture, μ_m, is computed using the Wilke's method (Equation C.13) [168], which also contains an expression for the temperature dependent pure component viscosity, μ^p. The DIPPR 102 vapor viscosity model (Equation C.15) is used for retrieving the temperature dependent pure viscosity of molecular species [20]. The coefficients of the DIPPR model for each molecular specie is reported in Table C.5, along with their respective temperature range of validity. For temperature values outside the DIPPR model, the viscosity of pure molecular species is calculated using linear extrapolation.

$$\mu_m = \sum_j \frac{\mu_j^p}{1 + \frac{1}{x_j} \cdot \sum_{jj \neq j} \phi_{j,jj} \cdot x_{jj}} \tag{C.13}$$

$$\phi_{j,jj} = \frac{\left[1 + (\mu_j^p/\mu_{jj}^p)^{0.5} \cdot (MW_{jj}/MW_j)^{0.25}\right]^2}{(4/\sqrt{2})\left[1 + (MW_j/MW_{jj})\right]^{0.5}} \tag{C.14}$$

$$\mu_j^p = \frac{A_j \cdot T^{B_j}}{1 + \frac{C_j}{T} + \frac{D_j}{T^2}} \tag{C.15}$$

C.1.5 Coking Effects

Coke formation in thermal cracking of natural gas liquids is an important consideration as this phenomena can lead to an increase in pressure drop, an interruption in the steady-state operation and an increase in tube wall temperature due to heat transfer limitations [17]. Hence, it is essential

Table C.5: Coefficients of the DIPPR model $\mu_j^p = A_j \cdot T^{B_j} / [1 + C_j \cdot T^{-1} + D_j \cdot T^{-2}]$ for the calculation of pure component gas phase viscosity in Pa·s and their respective valid temperature range (T^{min} - T^{max}) in K [20].

Molecular Specie, j	A	B	C	D	T^{min}	T^{max}
Methane, CH_4	$5.2546 \cdot 10^{-7}$	0.59006	105.67	0	90.69	1000
Acetylene, C_2H_2	$1.2025 \cdot 10^{-6}$	0.4952	291.4	0	192.40	600
Ethylene, C_2H_4	$2.0789 \cdot 10^{-6}$	0.4163	352.7	0	169.41	1000
Ethane, C_2H_6	$2.5906 \cdot 10^{-7}$	0.67988	98.902	0	90.35	1000
Propylene, C_3H_6	$7.3919 \cdot 10^{-7}$	0.5423	263.73	0	87.89	1000
Propane, C_3H_8	$4.9054 \cdot 10^{-8}$	0.90125	0	0	85.47	1000
Butadiene, C_4H_6	$2.696 \cdot 10^{-7}$	0.6715	134.7	0	164.25	1000
Butene, C_4H_8	$6.9744 \cdot 10^{-7}$	0.5462	305.25	0	87.80	1000
Butane, C_4H_{10}	$3.4387 \cdot 10^{-8}$	0.94604	0	0	134.86	1000
Benzene*, C_6H_6	$3.134 \cdot 10^{-8}$	0.9676	7.9	0	278.68	1000
Hydrogen, H_2	$1.797 \cdot 10^{-7}$	0.685	-0.59	140	13.95	3000
Water, H_2O	$1.7096 \cdot 10^{-8}$	1.1146	0	0	273.16	1073.15

*Taken for C_6

to consider the coking effects in the steam cracking model as the reactor coking and its regeneration will affect the profitability of operation. For ethane cracking, C_4^+ species are identified as the coke precursors and its kinetics are implemented based on Sundaram et al. [167]. On the other hand, for propane cracking, C_3H_6 is determined as the main coke precursor and the reaction mechanism and kinetic data are adapted from [15].

For natural gas liquid feeds, the mechanisms and kinetic data provided in [167] indicates that coking reaction rate is slow compared to that of the cracking reactions. As a result, the coke buildup and reduction in the tube diameter can be calculated under the quasi-steady state assumption as suggested by Onel [17]. Under this assumption, the reduction in the tube diameter due to coking is calculated after solving the cracking model at steady-state. Once the cracking model is solved, the coke accumulation on the reactor wall is calculated for a timestep t_s, and the diameter is updated respectively [17]. When the reactor reaches a 25% reduction in its diameter, the decoking cycle is initiated.

$$D_t^{new} = D_t - 2 \cdot k_{coke} \cdot exp(-E_a^{coke}/R\overline{T}) \cdot \frac{\overline{C} \cdot t_s}{\rho_c \cdot 10^6} \tag{C.16}$$

Calculating reduction percentage per hour:

$$Red = \frac{(D_t - D_t^{new})}{D_t} \cdot 100 \cdot t_s \cdot 3600 \tag{C.17}$$

Calculating the reactor runtime before decoking cycle begins:

$$\tau = 25/Red \tag{C.18}$$

C.1.6 Model Parameters, Process Constraints, Decision Variables and the Objective Function

The output temperature is constrained to be less than 1300 K (Equation C.19) due the limitations in the metallurgy of the reactor. Also, the output pressure of the reactor is limited to be at least 1 atm (Equation C.20). In these case studies, T^{out} and P^{out} are considered as grey-box constraints, as their value can only be obtained after solving and simulating the entire system.

$$T^{out} \leq 1300 \tag{C.19}$$

$$P^{out} \geq 101325 \tag{C.20}$$

In ethylene/propylene production, the objective function is to maximize the profitability of operation by selecting the optimal values for the reactor length, inlet temperature, inlet pressure, ethane/propane flowrate, steam flowrate, and the piece-wise constant external heat flux profile along the reactor length. For ethane production, the objective function considers the production of

ethylene as the sole product:

$$obj^{\mathrm{C_2H_4}} = \Bigg(P^{\$}_{\mathrm{C_2H_4}} \cdot F_{\mathrm{C_2H_4}} \cdot MW_{\mathrm{C_2H_4}} - P^{\$}_{\mathrm{C_2H_6}} \cdot F^{o}_{\mathrm{C_2H_6}} \cdot MW_{\mathrm{C_2H_6}} - P^{\$}_{\mathrm{H_2O}} \cdot F^{o}_{\mathrm{H_2O}} \cdot MW_{\mathrm{H_2O}} \\ - P^{\$}_{inv} \cdot L - P^{\$}_{heat} \cdot Q^{total} \cdot \pi \cdot D_t \Bigg) \cdot \frac{\tau}{\tau + 48} - \frac{P^{\$}_{decoking}}{(\tau + 48) \cdot 3600} \tag{C.21}$$

For propylene production, the reaction mechanism allows a single-feed-multi-product system, where propylene and ethylene are the two main products. Hence the objective function considers the profit made from these two:

$$obj^{\mathrm{C_3H_6}} = \Bigg(P^{\$}_{\mathrm{C_3H_6}} \cdot F_{\mathrm{C_3H_6}} \cdot MW_{\mathrm{C_3H_6}} + P^{\$}_{\mathrm{C_2H_4}} \cdot F_{\mathrm{C_2H_4}} \cdot MW_{\mathrm{C_2H_4}} - P^{\$}_{\mathrm{C_3H_8}} \cdot F^{o}_{\mathrm{C_3H_8}} \cdot MW_{\mathrm{C_3H_8}} \\ - P^{\$}_{\mathrm{H_2O}} \cdot F^{o}_{\mathrm{H_2O}} \cdot MW_{\mathrm{H_2O}} - P^{\$}_{inv} \cdot L - P^{\$}_{heat} \cdot Q^{total} \cdot \pi \cdot D_t \Bigg) \cdot \frac{\tau}{\tau + 48} \\ - \frac{P^{\$}_{decoking}}{(\tau + 48) \cdot 3600} \tag{C.22}$$

If any combination of input variables lead to a simulation result with no coking, then the cost for decoking the reactor is neglected and the profit calculation is solely based on the operating (heating, production and consumption of the output and input materials) and fixed cost of the reactor. The model parameters for the cracking case studies are summarized in Table C.6 and bounds on the decision variables are provided in Table C.7. In addition to the variables bounds, a known constraint (Equation C.23 for ethane cracking model, Equation C.24 for propane cracking model) is also included in the formulation, where the total inlet flowrate to the reactor constrained to be less than or equal to 0.05 kmol/s. This constraint avoids simulating cases that will create a large pressure drop in the reactor, and essentially won't yield a feasible solution. As this relationship is available in closed form and valid for any candidate sampling point for simulation, only the samples that satisfy this known relationship are used to construct the SVM-feasibility constraint.

$$F^{o}_{\mathrm{C_2H_6}} + F^{o}_{\mathrm{H_2O}} \leq 0.05 \tag{C.23}$$

$$F^{o}_{\mathrm{C_3H_8}} + F^{o}_{\mathrm{H_2O}} \leq 0.05 \tag{C.24}$$

Table C.6: Parameters considered in modeling thermal cracking of ethane and propane.

Parameter	Value
Coking Parameters	
Activation Energy for Coking (Ethane Cracking), E_a^{coke}	28.25 kcal/mol
Activation Energy for Coking (Propane Cracking), E_a^{coke}	73.58 kcal/mol
Decoking Cycle Downtime	48 hr
Density of Coke, ρ_c	1.6 kg/L
Reaction Rate Constant (Ethane Cracking)	$8.55 \cdot 10^5$ g coke/m^2·s/(kg j/m^3)
Reaction Rate Constant (Propane Cracking)	$5.82 \cdot 10^{14}$ g coke/(m^2·s)/(mol j/L)
Timestep (Ethane Cracking), t_s	72,000 s
Timestep (Propane Cracking), t_s	144,000 s
Cost Parameters	
Decoking Cost, $P^{\$}_{decoking}$	\$66,000/cycle
Heating Cost, $P^{\$}_{heat}$	$\$1.26 \cdot 10^{-8}$/J
Investment Cost, $P^{\$}_{inv}$	$\$2.725 \cdot 10^{-4}$/m·s
Price of Ethane, $P^{\$}_{C_2H_6}$	\$0.3/kg
Price of Ethylene, $P^{\$}_{C_2H_4}$	\$1.382/kg
Price of Propane, $P^{\$}_{C_3H_8}$	\$0.55/kg
Price of Propylene, $P^{\$}_{C_3H_6}$	\$1.340/kg
Price of Steam, $P^{\$}_{H_2O}$	\$0.0129/kg
Reactor Parameters	
Diameter, D_t	0.108 m

Table C.7: Decision variables for the grey-box optimization problem.

Decision Variables	Lower Bound	Upper Bound
External Heat Flux (Region 1, kW/m^2), Q_1^o	10	1000
External Heat Flux (Region 2, kW/m^2), Q_2^o	10	1000
External Heat Flux (Region 3, kW/m^2), Q_3^o	10	1000
External Heat Flux (Region 4, kW/m^2), Q_4^o	10	1000
External Heat Flux (Region 5, kW/m^2), Q_5^o	10	1000
Feed Flowrate of Ethane (kmol/s), $F^o_{C_2H_6}$	0.003	0.05
Feed Flowrate of Propane (kmol/s), $F^o_{C_3H_8}$	0.003	0.05
Feed Flowrate of Steam (kmol/s), $F^o_{H_2O}$	0.003	0.05
Inlet Temperature (K), T^o	700	1100
Inlet Pressure (kPa), P^o	290	500
Reactor Length (m), L	5	100

C.2 Offline Phase SVM Model Performance Validation Results

Table C.8: Ethane cracking SVM model performance for the second session of runs with ARGONAUT.

Run ID	Accuracy (%)	Precision (%)	Recall (%)	AUC (%)	F_1 score (%)
1	98.492	98.895	99.444	99.825	99.169
2	98.995	98.901	100	99.942	99.448
3	98.492	98.788	99.390	99.843	99.088
4	97.487	98.276	98.844	99.800	98.559
5	97.487	98	98.658	99.691	98.328
6	100	100	100	100	100
7	98.492	98.582	99.286	99.964	98.932
8	97.487	98.225	98.810	99.789	98.516
9	99.000	99.425	99.425	99.889	99.425
10	99.497	99.296	100	99.963	99.647
11	98.492	98.571	99.281	99.820	98.925
12	98.995	99.301	99.301	99.988	99.301
13	98.492	99.242	98.496	99.875	98.868
14	98.995	98.765	98.765	99.937	98.765
15	99.497	100.000	99.265	99.965	99.631
16	98.492	98.701	99.346	99.943	99.023
17	99.497	100	99.315	99.922	99.656
18	98.995	98.844	100	99.645	99.419
19	97.487	98.193	98.788	99.643	98.489
20	100	100	100	100	100

Table C.9: Propane cracking SVM model performance for the second session of runs with ARGONAUT.

Run ID	Accuracy (%)	Precision (%)	Recall (%)	AUC (%)	F_1 score (%)
1	95.477	97.541	95.200	99.351	96.356
2	97.487	97.647	96.512	99.784	97.076
3	97.990	97.321	99.091	99.908	98.198
4	95.980	96.190	96.190	99.473	96.190
5	98.995	100	98.795	99.945	99.394
6	99	98.561	100	99.734	99.275
7	96.482	94.815	100	99.813	97.338
8	96.985	96.460	98.198	99.314	97.321
9	98.492	96.774	100	99.959	98.361
10	97.990	97.872	99.281	99.772	98.571
11	97.487	98.630	97.959	99.843	98.294
12	97.487	95.890	97.222	99.661	96.552
13	96.482	96.970	96.000	99.818	96.482
14	100	100	100	100	100
15	97.487	97.810	98.529	99.895	98.169
16	97.990	97.727	99.231	99.933	98.473
17	100	100	100	100	100
18	97.990	98.000	99.324	99.907	98.658
19	97.990	99.275	97.857	99.891	98.561
20	98.492	100	97.321	99.949	98.643

APPENDIX D

CLUSTERING ANALYSIS AND SIMILARITY ASSESSMENT FOR ENVIRONMENTAL DATASETS

Dendrograms for the geospatial location-based and the pollutant concentration-based clustering are provided along with the Fowlkes-Mallows (FM) index profiles in Figures D.1-D.5. The similarity between these two dendrograms is quantified with the FM index where "k" represents the number of groups in the clustering analysis.

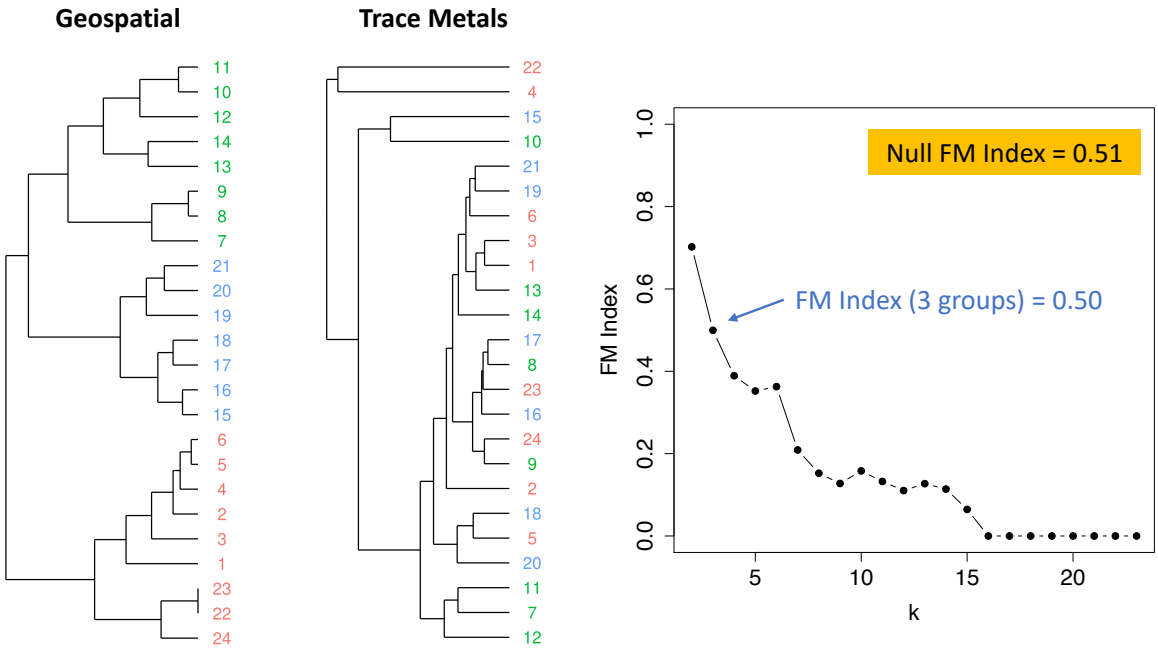

Figure D.1: Dendrograms for the geospatial-based and concentrations of trace metals-based grouping in soil sediments.

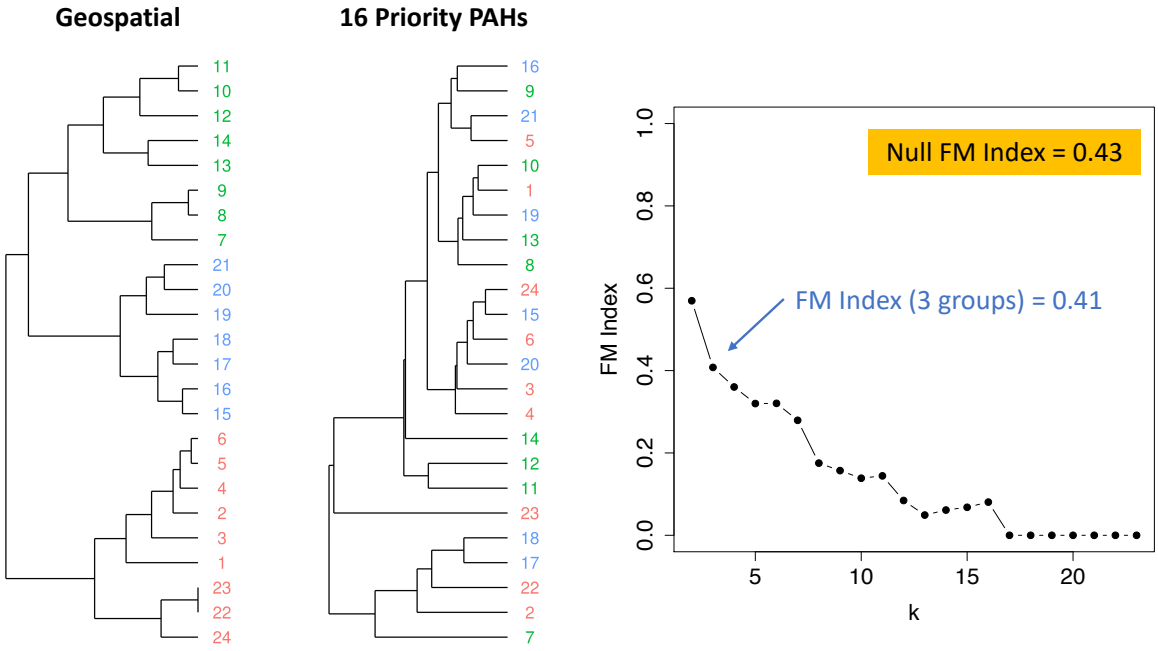

Figure D.2: Dendrograms for the geospatial-based and concentrations of 16 priority pollutant PAHs-based grouping in soil sediments.

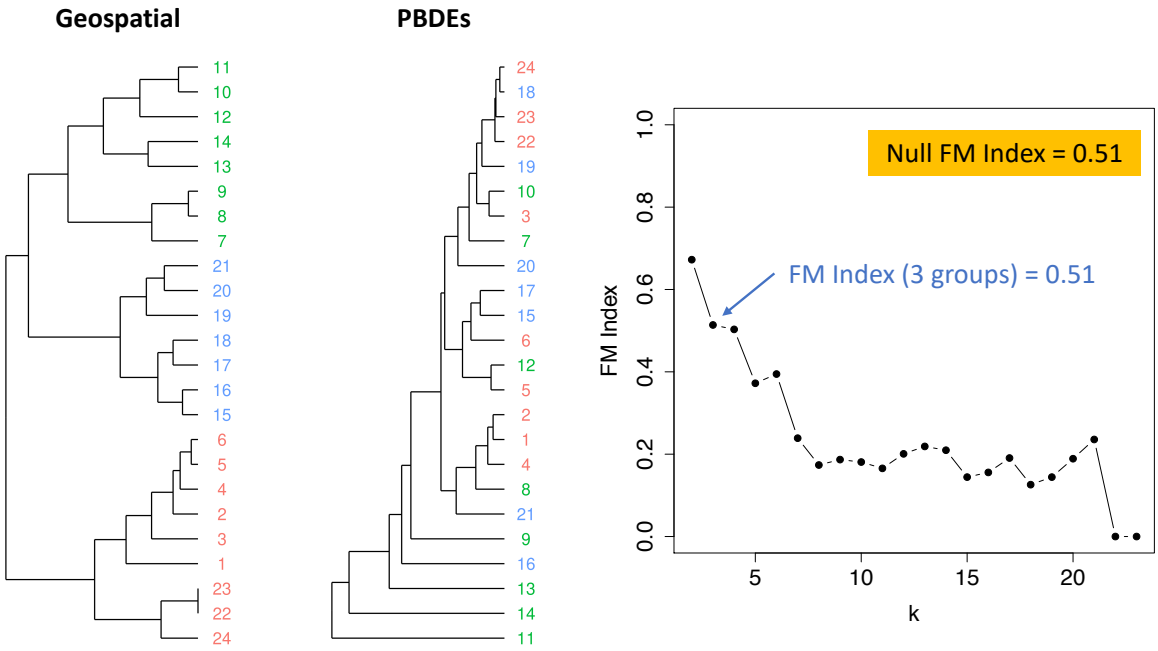

Figure D.3: Dendrograms for the geospatial-based and concentrations of PBDEs-based grouping in soil sediments.

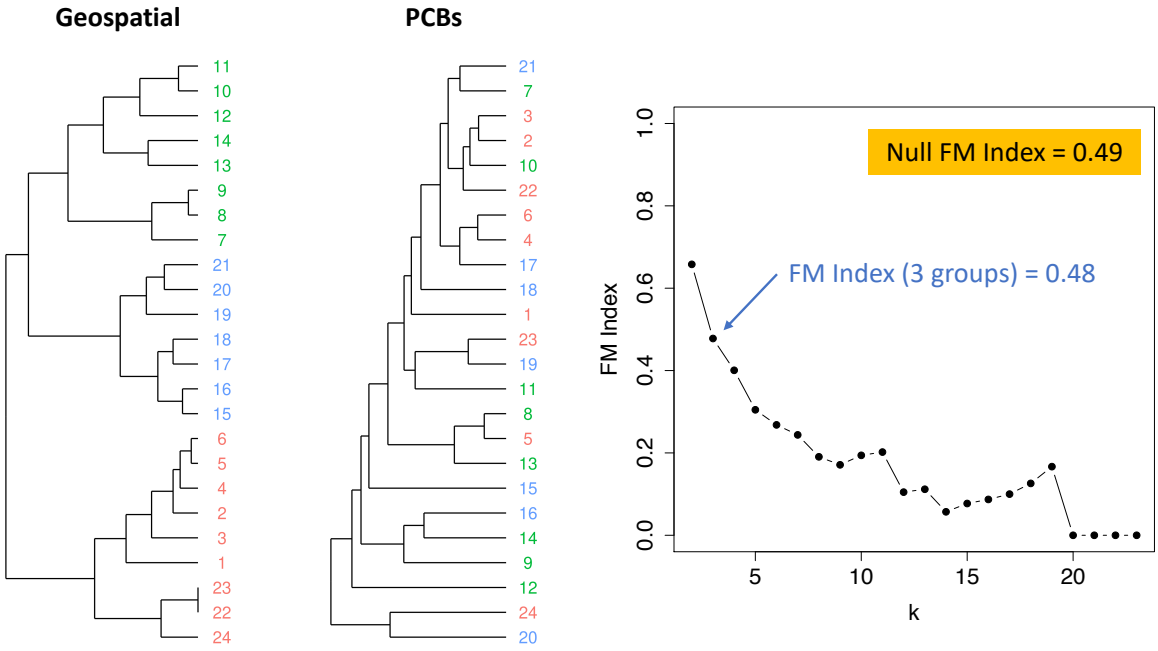

Figure D.4: Dendrograms for the geospatial-based and concentrations of PCBs-based grouping in soil sediments.

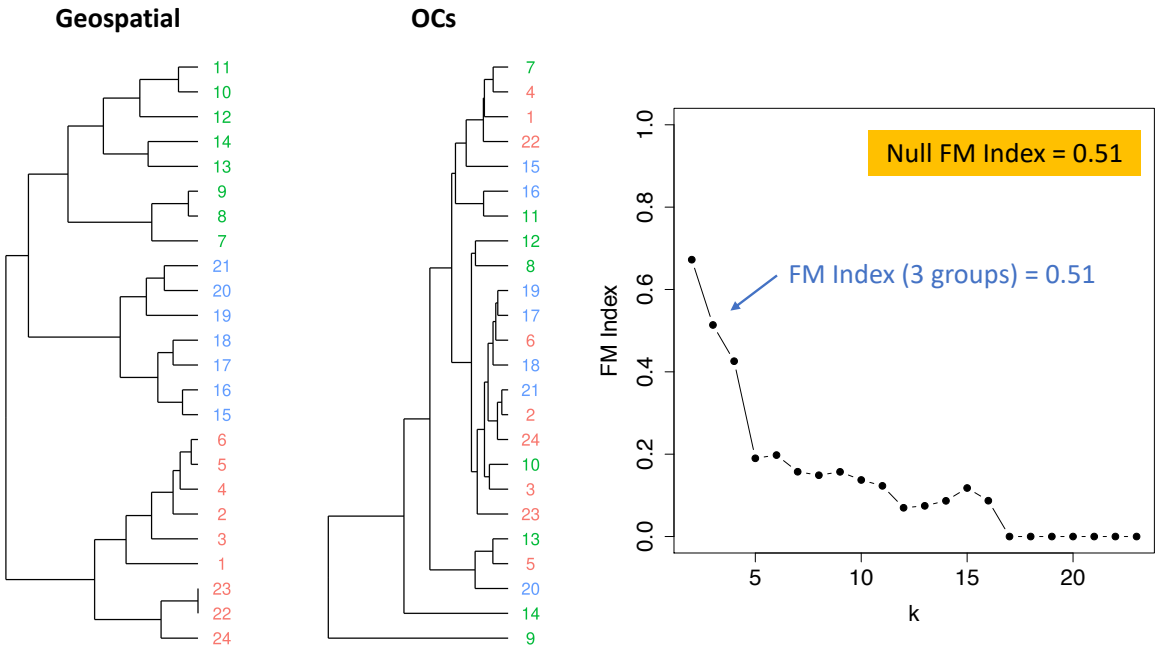

Figure D.5: Dendrograms for the geospatial-based and concentrations of OCs-based grouping in soil sediments.