

Optimization and Control of Spatiotemporal Systems

Chenyu Wang, Ph.D.

University of Connecticut, 2022

Engineered systems exhibit transient spatially dependent phenomena. Advances in computational methods enable applications of quantitative and formal methods for simulation of complex spatiotemporal phenomena. This thesis develops advanced computational methods for modeling, simulation, and optimization of dynamical systems with spatial dependence. In particular, deterministic optimization-based approaches are developed and used to solve parameter estimation problems for rigorous model validation, as well as optimal design and design under uncertainty problems. The modeling and optimization methods developed in this work are demonstrated through application to various spatiotemporal systems from different fields, providing deeper insights into system mechanisms to enable accurate prediction and control of system behavior.

Three specific applications involving spatiotemporal systems motivated the development of the methods detailed in this thesis. Advanced optimization methods were utilized and their advantages are demonstrated through these studies. In the first study, non-ideal heterogeneous mixing models are developed and incorporated within advanced model predictive control strategies to reduce the environmental discharge and the energy consumption of wastewater treatment systems. The new technology platform, consisting of high-resolution sensors, non-ideal heterogeneous mixing modeling, deterministic global dynamic optimization, and model predictive control, offers superior performance over current approaches in water and wastewater treatment processes. The second study is motivated by the need for

rigorous *in silico* optimization methods for decision-making in cancer research. A combination of first-principles mechanistic modeling and artificial neural network surrogate modeling is proposed to establish a digital testbed for solid tumors that is used with global optimization to solve nonlinear programs for optimal dose selection and macromolecule design. The experimentally-validated digital testbed approach enables optimal therapy design and anti-cancer drug size design as well as provides insight into how model-based technologies assist in medical-relevant studies. The last study demonstrates a process design problem that must account for uncertainty: worst-case design of performance /safety-critical process systems. An essential point with respect to worst-case designs, is the need for high-accuracy and low-complexity models. Thus, in this study, hybrid first-principles data-driven models are proposed to dramatically improve model prediction accuracy and simplify model structure, stepping closer to the digital twin concept. A semi-infinite program (SIP), as an alternative bilevel formulation which have better solution methods, is considered and formalized with hybrid modeling approaches to enable their use in worst-case designs under uncertainty. This approach is applied to a dynamical system that sufficiently demonstrates its practicability for treating spatiotemporal systems.

Optimization and Control of Spatiotemporal Systems

Chenyu Wang

B.S., Tianjin University, 2016

A Dissertation

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

in Chemical & Biomolecular Engineering

at the

University of Connecticut

2022

Copyright by
Chenyu Wang

Doctor of Philosophy Dissertation
Optimization and Control of Spatiotemporal Systems

Presented by
Chenyu Wang, B.S.

Approved by:

Major Advisor: Dr. Matthew D. Stuber _____

Associate Advisor: Dr. John D. Martin _____

Associate Advisor: Dr. Ranjan Srivastava _____

Associate Advisor: Dr. Yongku Cho _____

Associate Advisor: Dr. Jennifer Pascal _____

University of Connecticut

2022

ACKNOWLEDGMENTS

First and foremost, I am extremely grateful to my advisor Prof. Matthew Stuber for his continuous support, unwavering guidance, and invaluable advice he provided me during my pursuit of Ph.D. degree. I have been always encouraged by his immense knowledge and plentiful experience in all the time of my academic research and daily life. I feel very fortunate to have him as my advisor and I am especially thankful for the research and communication skills he taught me. His guidance always brought me edification and inspired me to critically think about problems. Every time we had communication, I always felt happier and more optimistic to move forward with my research and make progress. I believe that the knowledge he instilled in me will always help me throughout my life. In addition, I would like to thank him for his patience and kindness with respect to my personal life. In particular, I had an international vacation back to my country in 2019 but unfortunately experienced a two-month administrative check for visa application. I felt very anxious at that time, but he always encouraged me and allowed me to work remotely. Due to his flexibility and understanding, I did not lose progress and successfully survived that hard time. Overall, I am proud to be part of the graduate students in the wonderful PSOR group and could never have imagined having a better advisor and mentor for my Ph.D. study.

I would also like to thank my other thesis committee members including Dr. John Martin, Dr. Ranjan Srivastava, Dr. Yongku Cho, and Dr. Jennifer Pascal for their guidance and advice over the years. I owe special thanks to Dr. John Martin not only for his invaluable experimental data but also for his patient guidance and countless hours spent on providing advice for editing the paper regarding the content in Chapter 4. His generous guidance

and precious advice taught me how to think and write systematically and logistically, and finally helped me refine my ideas into a high-quality manuscript. I also owe thanks to Dr. Baikun Li for providing me with high-fidelity sensing data for model development and valuable guidance and feedback for the publication in Chapter 3. Again, I want to thank all committee members for their precious questions during my proposal, which inspired me to broaden my research from a more comprehensive and multifaceted perspective.

I would also like to thank many of my colleagues and friends at the University of Connecticut who have not only supported and helped my research but have also fulfilled my personal life with enjoyment and pleasure. First, I would like to thank my lab mates Dr. Matthew Wihelm, Dr. Robert Gottlieb, and Pengfei Xu. I really enjoy the time we spent together talking on research and life, attending our group meetings, and hanging out. Thank Dr. Robert Gottlieb for teaching me how to play chess, bringing me a lot of fun. I owe special thanks to Samuel Degnan-Morgenstern for his efforts and work on the tumor project in Chapter 4. And I really enjoyed the summer time we cooperated, hanged out, and had fun together. I also would like to thank Dr. Tianbao Wang for contributing the sensor data and other experimental material to Chapter 3. And I also enjoyed the time we spent in the gym. I would like to mention other friends in the Chemical & Biomolecular Engineering Department: Tuo Gao, Alan Shen, Guanhua Hong, Hao Ding, Xin Qian, Xi Yang, Chengwu Zhang; visiting scholars in other departments: Xin Ma, Heqiang Wang, Lu Jia, Ce Wen and Mengyun Su. You all brought me so much invaluable happiness to my life and I will never forget our memorial time having parties, swimming, playing table tennis, and travelling to a lot of fascinating places in America. I also owe thanks to the amazing Uconn staff, including Leah Winterberger, Susan Soucy, and Katie O’Keefe. I sincerely thank all of you

for reminding me a lot of critical deadlines and providing countless academic information. I cannot imagine how lucky I am to get along with all of you and I will look back on fondly.

On a personal note, I owe too much thanks to my family, especially my parents Yazhen Yu and Hang Wang. Without their unselfish supports and love, I could not accomplish this journey. I cannot find words to describe how much you mean to me, but I must say I really love you. I am so lucky to have such a harmonious family that we have family parties every week. Even though I am abroad, I still had a video call to meet almost all members every week, which made me feel warm and encouraged. I am sorry that we cannot meet personally for a very long time due to the COVID-19 pandemic, especially for my grandparents: Shufen Wang, Zhonghou Wang, Linju Liu, and Baogen Yu. I know you missed me so much and every time we had a video call I feared that it might be the last chance to meet you. I know you feared that you would never meet me in person again. Although it might unfortunately happen, I really need to take this chance to express me: I miss you too, I love you and wish you healthy!

Lastly, I'd like to acknowledge National Science Foundation and University of Connecticut for funding this research. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of those of the sponsor.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Approaches for Solving Optimization Problems of Spatiotemporal Systems	5
1.2.1	NAND for Local Optimization	5
1.2.2	SAND for Local Optimization	6
1.2.3	Global Optimization	6
1.3	Scope of Dissertation	11
2	Bounding Numerical Solutions of Parametric Partial Differential Equations	14
2.1	Mathematical Preliminaries	16
2.1.1	Interval Arithmetic	16
2.1.2	Affine Arithmetic	17
2.1.3	Parametric Partial Differential Equations	20
2.1.4	Finite Difference Method	23
2.1.5	Differential Inequalities	25
2.1.6	Optimization Problem Formulation	29

2.2	Methods	30
2.2.1	Bounding Solutions of Parametric IBVPs	31
2.2.2	IBVPs with Coupled States	43
2.2.3	IBVPs With Higher Spatial Dimensionality	44
2.2.4	Volume of State Bounds for IBVP Systems	47
2.2.5	Deterministic Global Optimization	49
2.3	Examples	54
2.3.1	Transient Plug Flow Reactor	55
2.3.2	Convection-Diffusion System	58
2.3.3	Convection-Reaction System	61
2.3.4	Diffusion-Reaction System	63
2.3.5	Coupled IBVPs	64
2.4	Conclusion	73
3	Precise Control of Water and Wastewater Treatment Systems with Non-Ideal Heterogeneous Mixing Models and High-Fidelity Sensing	74
3.1	Introduction	75
3.2	Materials and Methods	79
3.2.1	Non-Ideal Mixing Profiling Using Milli-Electrode Array (MEA) Sensors	80
3.2.2	Non-Ideal Mixing Model Development	81
3.2.3	Parameter Estimation and Model Validation Using Global Dynamic Optimization	85
3.2.4	Precise Control of A Wastewater Nitrification System	87

3.3	Results and Discussion	93
3.3.1	Optimal Solutions for Conductivity, pH, and Temperature Mixing Models from Parameter Estimation	93
3.3.2	Comparison of Non-Ideal Mixing Models with Pure Data-Driven Models and CFD Models	97
3.3.3	Validation and Calibration of Sensors and Non-Ideal Mixing Models .	100
3.3.4	Non-Ideal Mixing Models for Improved Wastewater Treatment with Precise Control	103
3.4	Conclusions	108
4	Optimal Therapy Design With Tumor Microenvironment Normalization	110
4.1	Introduction	111
4.1.1	Cancer Biology	111
4.1.2	Modeling and Simulation for Cancer	113
4.2	Methods	116
4.2.1	Parameter Estimation and Model Validation by Deterministic Global Optimization	117
4.2.2	Bounding Methods for Tumor Transport Model	120
4.2.3	Machine Learning Model	123
4.2.4	Simplified Parameter Estimation Problem	127
4.2.5	TME-Normalizing Therapy Design for Dose Selection	129
4.2.6	Drug Size Design	131
4.2.7	Settings for Solving Optimization Problems	134

4.3	Results and Discussion	135
4.3.1	Global Optimization Results for Model Validation	135
4.3.2	Dexamethasone Increases Convective Transvascular Flux in Tumors	145
4.3.3	Global Optimization Reveals Dose of Dexamethasone Maximizing Nanocarrier Accumulation	151
4.3.4	Dexamethasone Dose Affects the Transvascular Convective Transport Size-Dependently	153
4.3.5	Global Optimization Determines the Dexamethasone Dose and Nanocarrier Size Maximizing Accumulation	155
4.4	Conclusions	157
5	Semi-Infinite Optimization with Hybrid Models	160
5.1	Introduction	161
5.2	Mathematical Background	166
5.2.1	Multilayer Perceptrons	166
5.2.2	Hybrid Model Architecture	167
5.3	Optimization of Hybrid Models	170
5.4	Semi-Infinite Optimization with Hybrid Models	175
5.5	Global Solution of SIPs	179
5.6	Case Studies	182
5.6.1	Case Study 1: Robust Feasibility of a Nitrification CSTR	182
5.6.2	Case Study 2: Worst-Case Design of Subsea Production Facilities - Mitigation of Domain Violations	190

5.7	Extension to Implicit Forms	202
5.8	Conclusion	203
6	Conclusions	206
A	Supplementary Information	210
A.1	Supplementary Information for Chapter 3	210
A.1.1	Development of pH Mixing Models	210
A.1.2	Global Optimization Formulation and Validation for Temperature Mixing Model	213
A.1.3	Development of A Continuous Flow Conductivity Transport Model for Nitrification CSTR	214
A.1.4	Global Optimization Formulation for Conductivity and pH Mixing Model	219
A.1.5	Calibration Process for Revised Conductivity and pH Mixing Models for Additional Experiments	220
A.1.6	Development of Computational Fluid Dynamics (CFD) Models	221
A.2	Supplementary Information for Chapter 4	226
A.2.1	Tumor Transport Model	226
A.2.2	Simplification of Inequality Constraints	232
B	Related Publications and Presentations	246
B.1	Journal Articles	246
B.2	Conference Proceedings	247

List of Figures

2.1	Transient plug flow reactor	22
2.2	Algorithm for bounding IBVPs	31
2.3	Architecture of the deterministic global optimization algorithm	54
2.4	Bounding results for the PFR system (Ex. 2.3.1)	58
2.5	Bounding results for the convection-diffusion system (Ex. 2.3.2)	60
2.6	Bounding results for the convection-reaction system (Ex. 2.3.3)	62
2.7	Results for diffusion-reaction system (Ex. 2.3.4)	63
2.8	The convergence profiles for the global optimization algorithm using pure IA and mixed IA/AA for Ex. 2.3.5	69
2.9	Concentration profiles for the system of coupled IBVPs (Ex. 2.3.5)	71
3.1	Comparison of the conductivity profiles using local optimization and global optimization	78
3.2	Schematic of the experimental setup and non-ideal heterogeneous mixing mod- els in an online model-predictive controller (MPC) system	79
3.3	Schematics of the conductivity and pH transport models	82
3.4	Optimal conductivity profiles from the global optimization results	94

3.5	Comparisons of the simulation results of the non-ideal heterogeneous mixing model with the fitting results	99
3.6	The predictive simulation results	102
3.7	The results for the uncontrolled, PI control, and MPC simulations	104
3.8	The overall discharge and total compressed air consumption over the simulation horizon for different control strategies	106
4.1	a systematical framework for optimal therapy design within the context of tumor microenvironment normalization	116
4.2	Numerical solutions and bounding results for the tumor transport model	124
4.3	A fully connected feed-forward multilayer perceptron artificial neural network surrogate model	126
4.4	Experimental data and and corresponding regression models	130
4.5	Radial dose-dependent interstitial fluid pressure and velocity profiles	139
4.6	Radial and temporal dose-dependent solute concentration profiles	141
4.7	Dose-dependant transvascular convective and diffusive flux profiles	145
4.8	Cross sections of the spherical tumor are illustrated in this schematic for the control and 3 mg/kg DEX treatment cases	150
4.9	Interstitial concentrations of different sizes nanocarriers	153
5.1	A multilayer perceptron as a directed acyclic graph	168
5.2	Flow diagrams of typical hybrid model architectures	169
5.3	The SIPres algorithm that presented by Stuber and Barton [165]	181
5.4	The dynamic nitrification CSTR system	183

5.5	The hybrid model architecture used for the nitrification CSTR case study . . .	186
5.6	The process flow diagram for the subsea separator (adapted from [14])	192
A.1	The optimal temperature profile from the global optimization results versus the measured temperature sensor data	225
A.2	A diagram of the tumor microenvironment illustrating fluid and solute trans- port from the blood vessels to the interstitium with high transvascular per- meability [6].	227
A.3	The radial interstitial concentration profiles at the time corresponding to the highest spatially-averaged concentrations	236
A.4	The percentages of the spatially-averaged concentrations	237
A.5	The contributions from convective and diffusive flux to spatially-averaged con- centrations versus time	238
A.6	The transvascular flux profiles with 70 kDa dextran	239

List of Tables

2.1	CPU time for constructing state bounds of the PFR, convection-diffusion, convection-reaction, diffusion-reaction, and ozone reaction models	70
2.2	The volumes of state bounds for the solutions of PFR, convection-diffusion, convection-reaction and diffusion-reaction models	72
2.3	The volumes of state bounds for the solutions of each species in the ozone reaction model solutions	72
3.1	Input and output variables for the nitrification wastewater system with different control strategies	89
3.2	The control performance indices quantified by a response to a step disturbance change	90
4.1	The physical bounds on the superficial (peripheral) tumor IFP	120
4.2	Volume and time cost for different bounding methods	123
4.3	The coefficients for the constraints on the superficial (peripheral) IFP	128
4.4	Global optima for parameter estimation problems using the mechanistic model and the ANN model	136

4.5	The computational time costs for the parameter estimation problems using the mechanistic model and the ANN model	137
4.6	Optimal solutions and time costs of drug size design problems	156
5.1	The model parameters used for the nitrification CSTR case study	186
5.2	The state variables for the subsea separator case study	195
A.1	The global optimal parameter values obtained from (a) the conductivity parameter estimation problems, and (b) the optimal parameter values obtained from the pH parameter estimation problems.	222
A.2	Time cost for solving conductivity and pH global optimization problems for independent high-zone, middle-zone and low-zone injection experiments, are reported in this table.	223
A.3	Electrical conductivity (mS/cm) for NaCl, KCl, and MgSO ₄ at different mass percentages [241].	223
A.4	This table contains the data (in percentages) for environmental discharge and energy usage from the no control, open-loop control, PI control, MPC, EMPC1, and EMPC2 simulations.	223
A.5	The parameters for continuous flow conductivity mixing model are summarized in this table.	224
A.6	The lower and upper bounds for uncertain parameters in nonideal heterogeneous mixing models (a: conductivity, b: pH)	224
A.7	The physical properties of water at 20°C are presented below.	225
A.8	Hydrodynamic Coefficients for Cylindrical Pore Model [243]	240

A.9 The bounds on input variables, L_p and K , used for the surrogate model construction	241
A.10 The benchmark metrics for development time and performance of artificial neural network surrogate models	242
A.11 Data for diffusion coefficients and blood half-life circulation time with respect to nanocarrier sizes	243
A.12 Benchmark metrics of time and performance for ANN surrogate model development	244
A.13 Physiological parameter values for the tumor transport model	245

Chapter 1

Introduction

1.1 Motivation

Modeling and simulation are critical in engineering as they enable formal approaches to the design and optimization of systems to enhance their performance, reduce costs, and improve safety, among other benefits. Furthermore, modeling and simulation-based approaches offer the advantage of exploring and verifying system behavior with respect to performance and safety specifications *in silico*. Often, engineered systems of most interest exhibit phenomena that are dependent on both spatial and temporal domains. In particular, ordinary differential equations (ODEs) and transient partial differential equations (PDEs) are ubiquitous in model-based systems engineering for modeling spatiotemporally varying phenomena. Such models are encountered in a wide range of applications from traditional chemical engineering fields, including reacting flow systems [1, 2, 3] and thermofluid systems [4, 5] to non-traditional applications of drug delivery in solid tumors [6]. The development of modeling and simulation methods that lend themselves to mathematical programming techniques en-

ables accurate prediction within optimization-based approaches for systems design, decision-making, and advanced control architectures. The application of these methods provides an immense opportunity for high-impact innovations and discovery in spatiotemporal systems.

Optimization plays a significant role in engineering applications for model validation, optimal design, and optimal control of spatiotemporal systems, among many other applications. In particular, optimizing large-scale models, as the focus of this dissertation on PDE-constrained optimization problems, remains a popular and challenging research topic [7]. In recent years, there has been development of useful methods for solving optimization problems involving spatiotemporal systems.

Clason and Kaltenbacher [8] proposed a method for solving optimal control problems for nonlinear PDE models involving singular solution-dependent terms. However, for general problems, their approach could only provide suboptimal local solutions. Kolvenbach et al. [9] developed a robust optimization framework utilizing quadratic approximations of a worst-case objective with an assumption that the parameter is restricted to an ellipsoidal set. Special convexity conditions are required to guarantee a global solution by means of their proposed method, which utilizes the SQP algorithm. Sharma et al. [10] proposed a new heuristic for solving PDE-constrained mixed-integer nonlinear programs that combines a problem-specific rounding scheme with an improvement heuristic. Wu and Zhou [11] proposed a diagonalization-based approach for solving parabolic PDE-constrained optimization problems that can dramatically reduce time cost. Pearson et al. [12] introduced fast and efficient preconditioners that can improve the tractability of PDE-constrained optimization problems with the application of an interior-point scheme. However, all these methods either rely on specific convexity assumptions, or are not concerned with guaranteed

global optimality.

Developing and utilizing valid and efficient deterministic global optimization methods for solving PDE-constrained nonlinear programs remains a challenging topic of current research. Deterministic global optimization relies on the branch-and-bound (B&B) algorithm, which iteratively partitions the domain of decision variables into successively smaller subdomains and solves a sequence of upper-bounding and lower-bounding problems to bound a global optimum. The algorithm is guaranteed to converge to an epsilon-optimal global solution in finitely-many iterations. In this context, rigorous and tight bounds on solutions are critical for solving this difficult class of optimization problems. When applied within the B&B algorithm, the efficient calculation of tight bounds can greatly reduce the total computational cost for solving these problems by potentially avoiding excessive subdivision of the decision space. The incorporation of PDE models dramatically complicates global dynamic optimization problems, since numerical techniques required for PDE discretization are often computationally intensive and result in large-scale stiff systems. Thus, providing rigorous bounds on the numerical solutions of PDEs with less conservatism and low computational cost should be a research focus.

Another challenging problem with respect to spatiotemporal phenomena is the worst-case design of engineering systems that are deemed “safety critical” and require rigorous guarantees of system safety and performance before deployment and during operation. In these cases, it is necessary to identify the worst-case performance with respect to uncertainty of such safety-critical systems to mitigate the impacts of uncertainty at the design stage. For example, in energy systems, engineers must guarantee that a safety-critical state, such as the pressure in a steam boiler, is never higher than a maximum safe level to avoid an

operational failure that may incur a severe cost, such as substantial system damage or loss of life, while ensuring that a minimum required performance specification is met. From a model-based perspective, robust design is a method capable of providing rigorous guarantees of performance/safety of process systems under uncertainty [13, 14] to avoid operational failures. These problems are typically formulated as bilevel programs, and deterministic global optimization methods are required to guarantee worst-case realizations of uncertainty. Bilevel optimization problems are extremely difficult to solve, which poses a tremendous challenge to the class of systems with complicated spatiotemporal dynamics.

The objective of this work is to investigate valid and efficient methods to solve deterministic global optimization problems constrained by spatiotemporal systems models. Further, their applications are explored in different engineering fields to demonstrate the practicability. The contributions of this work can be summarized as follows:

1. Develop valid and efficient methods to construct bounds on the states of parametric ODE/PDE models for solving global optimization problems involving spatiotemporal systems;
2. Explore solution methods for solving challenging problems involving a variety of spatiotemporal systems, including system designs of process control in traditional chemical engineering field and drug delivery in cancer research;
3. Investigate efficient model-based methods to solve worst-case design problems of performance/ safety-critical systems under uncertainty.

1.2 Approaches for Solving Optimization Problems of Spatiotemporal Systems

The main objective of this dissertation is to develop and validate system models of interest from both wastewater treatment and cancer research fields, which are governed by systems of ODEs/PDEs. In addition, validated models will be used to study the optimal design and operation of such systems. Advanced optimization algorithms are utilized in these studies, and their superiority in solving this challenging class of problems is demonstrated. Classically, there are two local optimization strategies to solve these problems locally: Nested Analysis and Design (NAND) and Simultaneous Analysis and Design (SAND) [15], which are introduced below.

1.2.1 NAND for Local Optimization

In NAND, state variables are considered to be an implicit function of optimization variables governed by the ODEs/PDEs, and the ODE/PDE solver is called as a nested routine for the evaluation of the objective and constraint functions. This approach offers the benefit that the dimensionality of the optimization variables corresponds with the low dimensionality of the original decision space (i.e., the original problem dimensionality is retained). However, traditional NAND approaches are “black-box” in that the ODE/PDE solver is a separate routine with an “input-output” relationship that is unable to provide exact gradient/sensitivity information along with numerical solutions, and such information is usually provided using discrete function evaluations and finite differencing. This approach requires repeatedly solving the system, which can be extremely time-consuming.

1.2.2 SAND for Local Optimization

In SAND, the ODE/PDE constraints are treated explicitly as a system of discretized equations specified by a numerical method (e.g., collocation) and all the discretized state variables are treated as decision variables in addition to the optimization variables of the original problem. SAND approaches formulate and solve (approximate) KKT systems to identify suboptimal local solutions for nonconvex problems. One of the drawbacks of this approach is that the problem dimensionality will be extremely large, especially when solving time-dependent PDE-constrained optimization problems [16].

1.2.3 Global Optimization

Approaches for solving ODE/PDE-constrained optimization problems to global optimality rely on the spatial branch-and-bound (B&B) algorithm. In the B&B algorithm, lower and upper bounds on all variables and functions are required. An upper bound can be simply calculated as an objective function evaluation at any feasible point. However, the lower bounds require rigorous and accurate calculations of global bounds for all state variables and system functions, which poses significant challenges. When applied within the B&B algorithm for deterministic global optimization, efficient calculation of tight global bounds can greatly reduce the total computational cost to solve these problems by potentially avoiding excessive subdivision of the decision space. In particular, spatiotemporal systems involving PDEs dramatically complicates global dynamic optimization problems since numerical techniques required for PDE discretization are often computationally intensive and result in large-scale stiff systems. Thus, efficient bounding methods that can provide rigorous bounds

with less conservatism and low computational cost are required. Classically, these methods are classified into two categories: the relax-then-discretize methods and the discretize-then-relax methods. Both categories of methods are based on set-valued mapping approaches for calculating rigorous bounds.

Relax-Then-Discretize Method

In relax-then-discretize methods, relaxations are derived for the continuous-time system prior to discretization using numerical methods. This method constructs an auxiliary system of parametric ODEs that describes convex/concave bounding functions of the parametric solutions pointwise in time. There are two early contributions of the relax-then-discretize method using different approaches. Papamichail and Adjiman [17] introduced a rigorous method that utilized the α BB convexification theory [18]. In their method, a convex relaxation of the parametric ODE solutions is calculated at a given time point by adding a large-magnitude quadratic term. Then, an enclosure of the interval Hessian matrix is computed by bounding the second-order sensitivities of the ODEs to determine weights for the quadratic terms. Concurrently, Singer and Barton [19] proposed to construct an auxiliary initial value problem (IVP) in ODEs that describes convex/concave bounds on the ODE solutions based on McCormick's relaxation technique [20]. Many further developed approaches [21, 22, 23, 24, 25, 26] based on the relax-then-discretize method pursue reducing the conservatism of the bounds and enhancing computational efficiency. The limitation inherent to relax-then-discretize methods is that the truncation error cannot be accounted for due to the numerical solution techniques (which neglect truncation error through calculation).

Discretize-Then-Relax Method

The discretize-then-relax method constructs bounds based on an explicit integration scheme. The bounding procedure utilizes interval-based methods for ODEs and propagates convex/concave bounds at each integration step by using McCormick relaxations [27, 28, 29, 30]. One advantage of this method is that the truncation errors can be quantified and accounted for by applying a two-step procedure at each integration step. First, *a priori* bounds enclosing the ODE solutions over the integration step are calculated. Then, pointwise-in-time convex/concave bounds are obtained that enclose the ODE solutions at the end of the integration step. Eventually, valid bounds of entire numerical solutions of the parametric ODE-IVP are obtained.

Both of the introduced methods have problems in handling overestimation of bounds caused by the dependency problem of interval arithmetic (IA). The dependency problem is overestimation caused by multiple occurrences of the same variables in a function. Since each occurrence is taken independently in IA calculations, this leads to an expansion of the resulting bounds of a natural interval extension. The dependency problem may even lead to rapid divergence of the bounds when implemented within ODE-IVP integration methods. This motivates researchers to develop methods to reduce conservatism of the bounds for parametric nonlinear dynamical systems. Taylor series and differential inequalities are two common methods among these developments.

Taylor Series

Taylor series methods make use of a Taylor series expansion of parameter-dependent state variables to propagate lower and upper bounds through discrete time steps [30, 31, 32]. The resulting rigorous interval bounds enclose the truncation error. Modern approaches can achieve high accuracy by representing the interval bounds as a Taylor model consisting of the Taylor polynomial and an interval remainder bound enclosing errors associated with the floating point representation of the Taylor polynomial [33]. The polynomial parts are computed using real arithmetic, while the remainder terms are computed using IA. Advanced developments following the Taylor model theory have achieved better accuracy than classical Taylor model methods by using ellipsoidal and other non-interval remainder bounds. As stated above, one advantage of the Taylor series method is that the truncation and round-off errors are also enclosed by the interval operations and explicitly expressed in the results [34, 35]. Furthermore, Taylor series methods can reduce the dependency problem inherent in IA with high-order polynomial approximations [36, 37]. However, higher accuracy requires higher-order Taylor models, which significantly increases the computational cost [33, 38] of this approach. In particular, the Taylor series methods have more difficulty in handling discretized PDE system because it requires second or higher order Taylor models to combat the dependency problem of large-scale discretized models. The considerably higher computational cost limits applications of the Taylor series methods, especially for their use in deterministic global optimization.

Differential Inequalities

Differential inequalities (DI) is a computationally inexpensive bounding approach that utilizes IA to construct bounds for ODEs [39]. This method uses rules of IA to construct auxiliary ODEs that produce rigorous bounds on the states over the parameter domain at each time node through integration. This approach can make use of a continuous-time form that can be implemented using any state-of-the-art numerical integration rules, making it an efficient method for global optimization. Although the resulting bounds do not explicitly account for the numerical truncation errors, the resulting enclosures are valid within an acceptable margin of error because modern numerical integration algorithms perform very well for stable systems controlling error with acceptable accuracy. Another problem is that the bounds obtained by DI can be conservative when dealing with ODE-IVP systems that are not quasimonotone [40, 41]. Several methods have recently been developed to reduce the conservatism of the resulting bounds. [42] developed an algorithm utilizing the physical information from the system quantified as a crude set alongside an *interval refinement* operation to calculate greatly enhanced bounds at the expense of additional computational cost. [24, 25] utilized interval bounding information to construct McCormick-based convex and concave bounding functions that are tighter than the interval bounds themselves and address the *wrapping effect* associated with IA (i.e., the overestimation associated with an interval box bounding a non-interval set). [26] found that including redundant forms of modeling equations within the bounding calculations could result in refined bounds with less conservatism.

1.3 Scope of Dissertation

The goal of this dissertation is to develop valid and efficient methods for modeling and simulation of spatiotemporal systems specifically for their application in deterministic global optimization for model validation and decision-making for system design. In the following chapters, the theoretical developments of these methods and their applications in different fields, from traditional chemical engineering systems to medical-relevant studies, are discussed sequentially.

In Chapter 2, an algorithm is presented to construct bounds that enclose the sets of numerical solutions of parametric nonlinear PDEs. Compared with ODE systems, the additional complexity from the spatial derivative terms poses a challenge to current bounding procedures to provide accurate bounds. The proposed algorithm addresses this by exploiting affine arithmetic to bound finite-difference approximations of the spatial derivatives. This approach reduces the conservatism of state bounds caused by the interval dependency problem introduced by finite difference approximations of the spatial derivative terms. The algorithm is demonstrated on several PDE models relevant to process systems engineering, including the ϵ -optimal global solution of PDE-constrained dynamic optimization problems.

In Chapter 3, an application of spatiotemporal systems is presented for the precise control of water and wastewater treatment process systems. Non-ideal heterogeneous mixing models are developed based on high-fidelity sensing. Deterministic global dynamic optimization is used for training and recalibration of the non-ideal heterogeneous mixing models to guarantee the best-possible fits to the sensor data. The models are then deployed within standard model predictive control and two economic model predictive control strategies to demonstrate

model-based decision making for disturbance rejection and optimal operation of aeration in a continuous-flow nitrification system utilizing high-resolution sensor data from several spatial positions.

Chapter 4 addresses another application of spatiotemporal system in cancer research. This study uses mechanistic and machine learning modeling approaches to establish and validate a model for mass transport in a tumor. The validated model not only contributes to a better understanding of the fundamental transport mechanism, but also enables the exploration of optimal treatment strategies for the best individual patient outcomes. In particular, the new findings concerning the relative contributions of convection and diffusion to transvascular transport elucidates the impacts of tumor microenvironment normalization on drug delivery. In addition, the proposed optimal therapy design strategies demonstrate how *in silico* modeling approaches aid in predicting dose response in preclinical studies.

Chapter 5 addresses a method to solve worst-case design problems of performance/safety-critical process systems that involve hybrid models. Worst-case design problems are typically formulated as bilevel programs, which are a class of optimization problems whose feasible set is characterized by another optimization problem. These problems are extraordinarily difficult to solve, especially when involving complex spatiotemporal models. Hybrid first-principles data-driven models can be a promising approach because of their ability to provide accurate simulations with reduced computational complexity. Within this context, worst-case engineering design feasibility and reliability problems are formulated as a class of semi-infinite programs governed by hybrid models. Two challenging case studies are addressed to demonstrate the practicability of the proposed method: a nitrification reactor for a wastewater treatment system to address worst-case feasibility verification of dynamical

systems; and a three-phase separation system plagued by numerical domain violations to demonstrate how they can be overcome using a nonsmooth SIP formulation with hybrid models and validity constraints.

Lastly, Chapter 6 concludes the major contributions of this dissertation. A summary of the essential methods and results for each previous chapter are discussed, and recommendations for future research are proposed.

Chapter 2

Bounding Numerical Solutions of Parametric Partial Differential Equations

As introduced in Section 1.2, multiple methods have been developed to bound solutions of nonlinear dynamical systems. However, a crucial problem is that current state-of-the-art algorithms are only targeted at ODE-IVP systems and lack systematic approaches to be applied to PDE systems. In this chapter, a method is proposed to efficiently calculate valid state bounds with reduced conservatism on solutions of initial boundary value problems (IBVPs) on the domain of the parameters. The central idea is to combine numerical methods for IBVPs, set-valued arithmetics, and DI to construct tight enclosures on the solutions of PDE systems with improved efficiency. Specifically, we use the finite difference (FD) method for the discretization of spatial derivatives and reformulate the parametric PDE as a large

coupled system of ODE-IVPs using the method of lines. We addressed how set-valued arithmetics are employed for calculating bounds on the states that are involved in spatial and temporal derivatives. In this procedure, apart from applying traditional IA, an approach utilizing affine arithmetic (AA) is developed to reduce the overestimation introduced by the dependency problem of IA. Note that the method involving AA is empirically valid; however, it requires further research and formal proof to ensure its validity rigorously. Then, a modified DI method is implemented to calculate the state bounds of the large coupled ODE-IVP system, effectively bounding the states over the spatial domain and parameter/uncertainty domain of the original IBVP. We demonstrate this approach on several examples, including common transient convection-diffusion-reaction models, a plug flow reactor (PFR) system, and a multi-species model for atmospheric ozone reactions.

The performance of our approach is measured using two metrics: the computational cost and the volume of the calculated bounds. As such, we formalize a computational procedure to measure the conservatism of the state bounds of IBVPs. The results of the numerical experiments indicate that the approach exhibits desirable properties of computational efficiency, improved reduction in overestimation of bounds, and overall effectiveness for broad applications. In addition to these numerical experiments, a case study is conducted whereby the new bounding algorithm is used within a deterministic global optimization problem for optimizing the operation of an unsteady PFR to demonstrate its applicability within the context of PDE-constrained global dynamic optimization.

2.1 Mathematical Preliminaries

In this section, the mathematical notation, definitions, and key foundational results used in this chapter, are introduced.

2.1.1 Interval Arithmetic

In this chapter, an n -dimensional interval $A = [\mathbf{a}^L, \mathbf{a}^U]$ is defined in the standard way as a nonempty compact set $A \equiv \{\mathbf{a} \in \mathbb{R}^n : \mathbf{a}^L \leq \mathbf{a} \leq \mathbf{a}^U\}$ with $\mathbf{a}^L \in \mathbb{R}^n$ and $\mathbf{a}^U \in \mathbb{R}^n$ as the lower and upper bounds of A , respectively [43]. In addition, let $\mathbb{I}\mathbb{R}^n$ be the set of all nonempty interval subsets of \mathbb{R}^n . Thus, $\mathbb{I}A$ denotes the set of all nonempty interval subsets of $A \subset \mathbb{R}^n$. An interval vector $A \in \mathbb{I}\mathbb{R}^n$ is an n -dimensional vector with interval-valued components denoted as A_1, \dots, A_n .

IA is an arithmetic performed on interval numbers according to primitive interval computation rules [43]. For $A \subset \mathbb{R}^m$, an interval-valued function given by $F : \mathbb{I}A \rightarrow \mathbb{I}\mathbb{R}^m$ is called an *interval extension* of the real-valued function $\mathbf{f} : A \rightarrow \mathbb{R}^m$ on A , if

$$\mathbf{f}(\mathbf{a}) = \mathbf{y} = [\mathbf{y}, \mathbf{y}] = F([\mathbf{a}, \mathbf{a}]), \forall \mathbf{a} \in A.$$

Further, we call F *inclusion monotonic* on A if for every $X, Z \in \mathbb{I}A$, we have

$$X \subset Z \Rightarrow F(X) \subset F(Z).$$

Lastly, F is called an *inclusion function* of \mathbf{f} on A if the following relationship holds:

$$\mathbf{f}(B) = \{\mathbf{f}(\mathbf{b}) : \mathbf{b} \in B\} \subset F(B), \forall B \in \mathbb{IA}.$$

In other words, $F(B)$ bounds the image of B under the mapping $\mathbf{f}(\cdot)$. $F(B)$ may be calculated as an inclusion monotonic interval extension of \mathbf{f} on $B \in \mathbb{IA}$ by the *Fundamental Theorem of Interval Analysis* (see Moore [44, p. 21]). If \mathbf{f} consists of finitely many compositions of elementary arithmetic operations and transcendental functions, and $F(B)$ is calculated by applying the primitive interval computation rules to the expression \mathbf{f} with interval operand B , it is called a *natural interval extension* of \mathbf{f} on B . Natural interval extensions are inclusion monotonic.

IA suffers from the *dependency problem* as the interval operands in an equation are treated as entirely independent variables. When some of the interval operands depend on each other (e.g., a variable occurs several times in an equation), the combinations of IA operations of F may significantly overestimate the image set. This conservatism becomes amplified as the function \mathbf{f} becomes more complex. A simple example is $f(a) = a - a$. In this case, let $a \in A = [a^L, a^U]$, then $f(A) = [0, 0]$ for any given A , while $F(A) = [a^L - a^U, a^U - a^L]$ does not equal zero for nondegenerate intervals, and therefore can significantly overestimate the image set.

2.1.2 Affine Arithmetic

Affine Arithmetic (AA) [45] aims to overcome the overestimation induced by the dependency problem of traditional IA. AA keeps track of the dependency between the interval operands

throughout the calculations, resulting in better interval approximations in most cases [46, 47].

In addition, the associated properties for the joint range of the interval variables can be represented as a geometry by AA that reduces overestimation.

Definition 2.1.1 (Affine Form [45]). *In AA, a number $a \in \mathbb{R}$ is represented by an affine form \hat{a} defined as a first-degree polynomial:*

$$\hat{a} = a_0 + \sum_{i=1}^m a_i \varepsilon_i,$$

where a_0 is called the central value of \hat{a} , and a_1, \dots, a_m are called the partial deviations corresponding to the noise symbols $\varepsilon_1, \dots, \varepsilon_m$ of \hat{a} . Each noise symbol ε_i is a symbolic unknown real variable lying in the interval $U = [-1, 1]$ that represents an independent component of uncertainty for the quantity a .

Lemma 1 (Affine Forms and Interval Bounds). *Let $\varepsilon \in U^m$ and let the function $\hat{a} : U^m \rightarrow \mathbb{R}$ represent an affine form of $a \in \mathbb{R}$ as a function of the noise symbols ε . Then, interval bounds $\hat{A} \in \mathbb{IR}$ on a quantity a can be calculated from an associated affine form $\hat{a}(\cdot)$. Furthermore, an affine form $\hat{a}(\cdot)$ of a quantity a can be constructed from its interval bounds A .*

Proof. An interval \hat{A} can be calculated by taking a natural interval extension of \hat{a} on U^m , which corresponds with $\hat{A} = [\hat{a}^L, \hat{a}^U] = [a_0 - r_a, a_0 + r_a]$, where $r_a = \sum_{i=1}^m |a_i|$ is the total deviation of the affine form \hat{a} . As a result, $a \in \hat{A}$. Alternatively, a quantity $a \in A = [a^L, a^U] \in \mathbb{IR}$ can be transformed into an affine form \hat{a} , defined in Definition 2.1.1, by setting $a_0 = (a^L + a^U)/2$, $a_1 = (a^U - a^L)/2$, and $a_i = 0, \forall i > 1$. □

Note that \hat{A} specifically denotes the interval bounds from taking an interval extension of

the affine form \hat{a} , distinguished from a general interval A . AA libraries can utilize intervals both as inputs and outputs by this conversion.

Definition 2.1.2 (Affine Computation [47]). *Affine computations in AA are defined as:*

$$\eta_1 \hat{a} + \eta_2 \hat{b} + \eta_3 = (\eta_1 a_0 + \eta_2 b_0 + \eta_3) + \sum_{i=1}^n (\eta_1 a_i + \eta_2 b_i) \varepsilon_i,$$

where η_1, η_2 , and η_3 are scalar coefficients.

Note that AA can account for the dependency on each affine form through shared noise symbols. For example, supposed that $f(a) = a - a$ and $a \in A = [a^L, a^U]$ is nondegenerate. If the standard subtraction rule of IA is used and a natural interval extension of f on A is calculated, we obtain $F(A) = A - A = [a^L, a^U] - [a^L, a^U] = [a^L - a^U, a^U - a^L] \neq [0, 0]$. In contrast, if the quantity a is represented by an affine form $\hat{a}(\varepsilon)$, then $f(\hat{a}(\varepsilon)) = a_0 + \sum_i a_i \varepsilon_i - a_0 - \sum_i a_i \varepsilon_i = 0$, and thus, taking the natural interval extension of the resulting difference results in $[0, 0]$.

The main feature of AA is the same noise symbol ε_i can be shared with multiple affine forms representing the partial dependency between them [46]. The corresponding partial deviation a_i represents the magnitude of dependency for the component ε_i . For example, consider two affine forms \hat{a} and \hat{b} :

$$\begin{aligned} \hat{a} &= a_0 + \sum_{i=1}^m a_i \varepsilon_i, \\ \hat{b} &= b_0 + \sum_{i=1}^m b_i \varepsilon_i. \end{aligned}$$

These two affine forms \hat{a} and \hat{b} provide the corresponding interval forms: $\hat{A} = [a_0 - r_a, a_0 + r_a]$

and $\hat{B} = [b_0 - r_b, b_0 + r_b]$ by Lemma 1. It is indicated by the shared noise symbols ε_i , that the joint range J of the two affine forms \hat{a} and \hat{b} is not only the rectangle $R = A \times B$ calculated by IA. The actual range is $J = \{(\hat{a}, \hat{b}) \in \mathbb{R}^2 : \varepsilon_i \in U, i = 1, \dots, m\} \subset R$ and can be represented by a convex polytope under the affine conversion $\mathbb{R}^m \rightarrow \mathbb{R}^2$:

$$\begin{pmatrix} a_0 \\ b_0 \end{pmatrix} + \begin{pmatrix} a_1 & \dots & a_m \\ b_1 & \dots & b_m \end{pmatrix} \cdot (\varepsilon_1, \dots, \varepsilon_m).$$

This convex polytope is centrally symmetric with respect to the point (a_0, b_0) and has $2m$ sides assuming $a_i, b_i \neq 0, \forall i \geq 1$. In general, n affine forms will yield a joint range as a zonotope in \mathbb{R}^n .

Note that for non-affine operations, such as multiplication, an extra noise term is required to estimate the affine approximations of the non-affine part for each operation. Generally speaking, this results in the elementary operations of AA to be more computationally expensive than standard IA. However, the method proposed in this chapter will only apply AA for affine operations. As such, non-affine operations and the additional complexity that they introduce, are ignored, resulting in no extra time cost over standard IA.

2.1.3 Parametric Partial Differential Equations

Let $t \in I = [t_0, t_f]$, $D \subset \mathbb{R}^{n_x}$ open, $y \in Y = [y_l, y_r] \in \mathbb{I}\mathbb{R}$, $\mathbf{p} \in P \in \mathbb{I}\mathbb{R}^{n_p}$, and define the mappings $\tilde{\mathbf{f}} : Y \times I \times D \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\zeta}_0 : Y \times P \rightarrow D$, $\tilde{\mathbf{f}}_l : D \times \mathbb{R}^{n_x} \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\mathbf{x}}_l : I \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\mathbf{f}}_r : D \times \mathbb{R}^{n_x} \times P \rightarrow \mathbb{R}^{n_x}$, and $\tilde{\mathbf{x}}_r : I \times P \rightarrow \mathbb{R}^{n_x}$, with the lower-case subscripts l and r pertaining to the spatial position y_l and y_r (i.e., respectively the left and

right boundaries of the spatial domain), respectively. Under these definitions, we consider a PDE defined as the parametric IBVP:

$$\begin{aligned}
\partial_t \tilde{\mathbf{x}}(y, t, \mathbf{p}) &= \tilde{\mathbf{f}}(y, t, \tilde{\mathbf{x}}(y, t, \mathbf{p}), \partial_y \tilde{\mathbf{x}}(y, t, \mathbf{p}), \partial_{yy} \tilde{\mathbf{x}}(y, t, \mathbf{p}), \mathbf{p}) & (2.1) \\
\tilde{\mathbf{x}}(y, t_0, \mathbf{p}) &= \tilde{\boldsymbol{\zeta}}_0(y, \mathbf{p}) \\
\tilde{\mathbf{f}}_l(\tilde{\mathbf{x}}(y_l, t, \mathbf{p}), \partial_y \tilde{\mathbf{x}}(y_l, t, \mathbf{p}), \mathbf{p}) &= \tilde{\mathbf{x}}_l(t, \mathbf{p}) \\
\tilde{\mathbf{f}}_r(\tilde{\mathbf{x}}(y_r, t, \mathbf{p}), \partial_y \tilde{\mathbf{x}}(y_r, t, \mathbf{p}), \mathbf{p}) &= \tilde{\mathbf{x}}_r(t, \mathbf{p}),
\end{aligned}$$

where $\partial_t \tilde{\mathbf{x}} : Y \times I \times P \rightarrow \mathbb{R}^{n_x}$ is the first partial derivative of $\tilde{\mathbf{x}}$ with respect to time t , and $\partial_y \tilde{\mathbf{x}} : Y \times I \times P \rightarrow \mathbb{R}^{n_x}$ and $\partial_{yy} \tilde{\mathbf{x}} : Y \times I \times P \rightarrow \mathbb{R}^{n_x}$ are the first and second partial derivatives of $\tilde{\mathbf{x}}$ with respect to y , respectively. The tilde accent signifies that the corresponding function is involved in the continuous-space formulation (i.e., that it is continuous with respect to the spatial variables). A typical plug flow reactor (PFR) system is illustrated in Figure 2.1 and introduced as an example of a PDE system under consideration in this work.

Remark 1. *For ease of introduction, (2.1) is defined for only one spatial dimension and it can be extended to multiple spatial dimensions with the notation introduced in Section 2.2.3. If higher-order time derivatives are involved, auxiliary variables can be introduced to transform the original system into a system of PDEs with first-order time derivatives. Such cases are considered as out of the scope of this chapter and the reader is directed to reference texts on numerical methods for ODE-IVPs and IBVPs, e.g. [48, 49].*

Assumption 1. *The parametric IBVP (2.1) satisfies the following conditions:*

1. $\tilde{\mathbf{f}} : Y \times I \times D \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\boldsymbol{\zeta}}_0 : Y \times P \rightarrow D$, $\tilde{\mathbf{f}}_l : D \times \mathbb{R}^{n_x} \times P \rightarrow \mathbb{R}^{n_x}$,

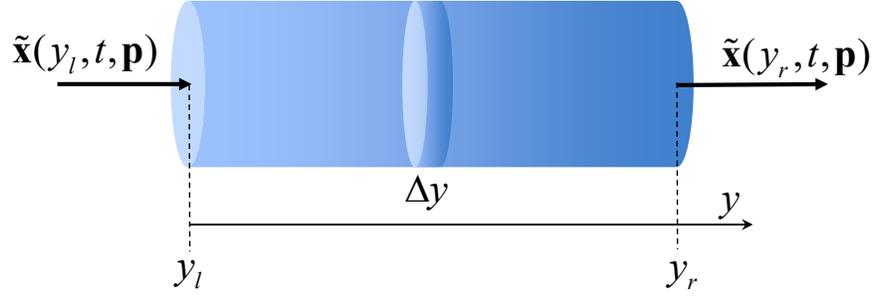


Figure 2.1: A transient plug flow reactor (PFR) is illustrated as a typical system modeled using PDEs. $\tilde{\mathbf{x}}(y, t, \mathbf{p})$ is a vector of system states, e.g. concentration of chemical species. y is the horizontal spatial coordinate for this PFR, and y_l and y_r are the left and right boundaries of the PFR, respectively. \mathbf{p} are uncertain parameters that may affect the system states, e.g. flowrate.

$\tilde{\mathbf{x}}_l : I \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\mathbf{f}}_r : D \times \mathbb{R}^{n_x} \times P \rightarrow \mathbb{R}^{n_x}$ and $\tilde{\mathbf{x}}_r : I \times P \rightarrow \mathbb{R}^{n_x}$ are locally Lipschitz continuous.

2. There exists a unique solution over the domain $Y \times I$ for every $\mathbf{p} \in P$.

3. A solution of (2.1) is denoted $\tilde{\mathbf{x}} : Y \times I \times P \rightarrow D$, and is twice continuously differentiable for all $y \in Y$.

Definition 2.1.3 (Reachable Set). The reachable set of (2.1) at $(y, t) \in Y \times I$ is defined as:

$$\mathcal{R}(y, t) \equiv \{\tilde{\mathbf{x}}(y, t, \mathbf{p}) : \mathbf{p} \in P\}. \quad (2.2)$$

Definition 2.1.4 (State Bounds). Functions $\tilde{\mathbf{x}}^L, \tilde{\mathbf{x}}^U : Y \times I \rightarrow \mathbb{R}^{n_x}$ are referred to as state bounds of (2.1) if they satisfy:

$$\tilde{\mathbf{x}}^L(y, t) \leq \tilde{\mathbf{x}}(y, t, \mathbf{p}) \leq \tilde{\mathbf{x}}^U(y, t), \quad \forall \mathbf{p} \in P,$$

where $\tilde{\mathbf{x}}(y, t, \mathbf{p})$ are the state variables of (2.1).

Under Definitions 2.1.3 and 2.1.4, we have $\mathcal{R}(y, t) \subset [\tilde{\mathbf{x}}^L(y, t), \tilde{\mathbf{x}}^U(y, t)]$. The primary objective of this work is to calculate valid state bounds enclosing the reachable set $\mathcal{R}(y, t)$ without significant conservatism/overestimation.

2.1.4 Finite Difference Method

FD methods are a class of common numerical approaches to solving differential equations by approximating derivative terms in differential equations using finite difference quotients [50]. In this section, a convenient nomenclature is developed for the common finite difference terms that enable an intuitive presentation of the bounding theory developed in this work.

Definition 2.1.5 (Finite Difference Approximations). *Consider a twice-differentiable function $\tilde{\psi} : Y \times I \times P \rightarrow \mathbb{R}$ where $y \in Y$ denotes the one-dimensional space variable. Let the spatial domain $Y = [y_l, y_r]$ be discretized into N nodes y_1, \dots, y_N , where $y_1 = y_l$, $y_N = y_r$, and the discretization step be defined for each node (left node of the step) as $\Delta y_i = y_{i+1} - y_i$, $\forall i = 1, \dots, N - 1$. Define the function $\psi_i : I \times P \rightarrow \mathbb{R}$ such that $\psi_i(\cdot, \cdot) \equiv \tilde{\psi}(y_i, \cdot, \cdot)$ for $i = 1, \dots, N$ and define the vector-valued function $\boldsymbol{\psi}(\cdot, \cdot) = (\psi_1(\cdot, \cdot), \dots, \psi_N(\cdot, \cdot)) \in \mathbb{R}^N$ whose components are indexed corresponding to the discrete space nodes. Then, the forward, backward, and centered FD approximations for the first-order spatial derivatives, $\psi_i^f, \psi_i^b, \psi_i^c :$*

$I \times P \rightarrow \mathbb{R}$, respectively, are defined as:

$$\begin{aligned}\psi_i^f(t, \mathbf{p}) &\equiv \frac{\psi_{i+1}(t, \mathbf{p}) - \psi_i(t, \mathbf{p})}{\Delta y_i} \approx \partial_y \tilde{\psi}(y_i, t, \mathbf{p}), \quad \forall i = 1, \dots, N-1, \\ \psi_i^b(t, \mathbf{p}) &\equiv \frac{\psi_i(t, \mathbf{p}) - \psi_{i-1}(t, \mathbf{p})}{\Delta y_{i-1}} \approx \partial_y \tilde{\psi}(y_i, t, \mathbf{p}), \quad \forall i = 2, \dots, N, \\ \psi_i^c(t, \mathbf{p}) &\equiv \frac{\Delta y_{i-1}^2 \psi_{i+1}(t, \mathbf{p}) + (\Delta y_i^2 - \Delta y_{i-1}^2) \psi_i(t, \mathbf{p}) - \Delta y_i^2 \psi_{i-1}(t, \mathbf{p})}{\Delta y_{i-1} \Delta y_i (\Delta y_{i-1} + \Delta y_i)} \\ &\approx \partial_y \tilde{\psi}(y_i, t, \mathbf{p}), \quad \forall i = 2, \dots, N-1,\end{aligned}$$

for every $(t, \mathbf{p}) \in I \times P$. Further, the centered FD approximation of the second-order spatial derivative is defined as:

$$\begin{aligned}\psi_i^{c2}(t, \mathbf{p}) &\equiv \frac{2(\Delta y_{i-1} \psi_{i+1}(t, \mathbf{p}) - (\Delta y_{i-1} + \Delta y_i) \psi_i(t, \mathbf{p}) + \Delta y_i \psi_{i-1}(t, \mathbf{p}))}{\Delta y_{i-1} \Delta y_i (\Delta y_{i-1} + \Delta y_i)} \\ &\approx \partial_{yy} \tilde{\psi}(y_i, t, \mathbf{p}), \quad \forall i = 2, \dots, N-1,\end{aligned}$$

for every $(t, \mathbf{p}) \in I \times P$.

Remark 2. IA can be applied to the FD approximations with $\Psi_i^j(t, P)$ representing an interval extension of $\psi_i^j(t, \cdot)$ on P at time $t \in I$ with $j \in \{f, b, c, c2\}$. For example, $\Psi_i^f = (\Psi_{i+1}(t, P) - \Psi_i(t, P)) / \Delta y_i$. Interval bounds are denoted by $\Psi_i^j(t, P) = [\psi_i^{j,L}(t), \psi_i^{j,U}(t)]$. We denote the interval vector as $\Psi^j(t, P) = (\Psi_1^j(t, P), \dots, \Psi_N^j(t, P)) \in \mathbb{IR}^N$.

Remark 3. If equidistant step sizes are considered for spatial discretization, they are given by $\Delta y = (y_r - y_l) / (N - 1)$. Then, the centered FD approximations can be simplified to $\psi_i^c = (\psi_{i+1} - \psi_{i-1}) / (2\Delta y)$ and $\psi_i^{c2} = (\psi_{i+1} - 2\psi_i + \psi_{i-1}) / \Delta y^2$ for first and second derivatives, respectively.

Remark 4. *The truncation errors (i.e., the error introduced by truncating the Taylor series representations of discrete-space states) for the FD approximations are not considered explicitly in this chapter. Specifically, when using the forward FD approximation of the first derivative, the truncation error is $\mathcal{O}(\Delta y_i)$; when using the backward FD approximation of the first derivative, the truncation error is $\mathcal{O}(\Delta y_{i-1})$; when using the centered FD approximation of the first derivative and second derivative, the truncation error is $\mathcal{O}(\max\{\Delta y_i^2, \Delta y_{i-1}^2\})$. However, in general, truncation errors cannot be fully quantified for spatial FD methods for solving BVPs and IBVPs as closed-form analytical solutions, and their higher-order derivatives, are unavailable. The reader is directed to numerical PDE texts for discussion on this point, e.g. [50] and [48].*

2.1.5 Differential Inequalities

[39] introduced the application of DI methods with IA for constructing the componentwise lower and upper bounds on the reachable set of a system of ODE-IVPs. The DI methods can be implemented relatively fast with any numerical integrator, but the numerical errors are not accounted for in the calculation procedures. The DI methods can be categorized into two types: continuous-time DI and discrete-time DI. In continuous-time DI [40], an auxiliary system of ODE-IVPs is formulated and directly sent to a numerical integrator for constructing the bounds. In contrast, the discrete-time DI introduced by [51] reformulates the system of ODE-IVPs into a discrete-time form. Then the bounding rules are applied at each discrete time point.

Continuous-Time Differential Inequalities

Consider a continuous-time system of ODE-IVPs:

$$\dot{\mathbf{x}}(t, \mathbf{p}) = \mathbf{f}(t, \mathbf{x}(t, \mathbf{p}), \mathbf{p}), \quad \mathbf{x}(t_0, \mathbf{p}) = \boldsymbol{\zeta}_0(\mathbf{p}), \quad t \in I. \quad (2.3)$$

Here, $I = [t_0, t_f]$, $D \subset \mathbb{R}^{n_x}$ is open, $P \subset \mathbb{R}^{n_p}$ is compact, $\mathbf{f} : I \times D \times P \rightarrow \mathbb{R}^{n_x}$ and $\boldsymbol{\zeta}_0 : P \rightarrow D$ are locally Lipschitz continuous, $\mathbf{x}(t, \mathbf{p}) \in D$ is the vector of state variables, $\mathbf{p} \in P$ is the vector of parameters in the system of ODE-IVPs, and $\boldsymbol{\zeta}_0 : P \rightarrow D$ is the initial condition at $t = t_0$. It is assumed that there is a unique solution in I for every $\mathbf{p} \in P$.

The purpose is to calculate the lower and upper bounds of the state variables $\mathbf{x}(t, \mathbf{p})$, defined as $\mathbf{x}^L, \mathbf{x}^U : I \rightarrow \mathbb{R}^{n_x}$, enclosing the reachable set $\mathcal{R}(t) = \{\mathbf{x}(t, \mathbf{p}) : \mathbf{p} \in P\} \subset [\mathbf{x}^L(t), \mathbf{x}^U(t)]$. For each $i = 1, \dots, n_x$, an interval extension of f_i and $\zeta_{0,i}$ will be denoted $F_i([t, t], D, P) = [f_i^L([t, t], D, P), f_i^U([t, t], D, P)]$ and $Z_{0,i}(P) = [\zeta_{0,i}^L(P), \zeta_{0,i}^U(P)]$, respectively.

The interval form of (2.3) is given by an interval extension of systems of ODE-IVPs:

$$\dot{x}_i^L = f_i^L([t, t], [\mathbf{x}^L(t), \mathbf{x}^U(t)], P), \quad (2.4)$$

$$\dot{x}_i^U = f_i^U([t, t], [\mathbf{x}^L(t), \mathbf{x}^U(t)], P),$$

$$[x_i^L(t_0), x_i^U(t_0)] = Z_{0,i}(P)$$

Due to the dependency problems induced by IA, the bounds derived by (2.4) are often extremely weak. Especially for complicated nonlinear systems, calculated bounds will be more likely to rapidly diverge to $\pm\infty$. To mitigate this expansiveness, standard DI exploits the following flattening operators.

Definition 2.1.6 (Flattening Operator [39]). *Flattening operators* $\beta_i^L, \beta_i^U : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_x}$ are defined as:

$$\begin{aligned}\beta_i^L([\mathbf{x}^L, \mathbf{x}^U]) &= \{\mathbf{x} \in [\mathbf{x}^L, \mathbf{x}^U] : x_i = x_i^L\}, \\ \beta_i^U([\mathbf{x}^L, \mathbf{x}^U]) &= \{\mathbf{x} \in [\mathbf{x}^L, \mathbf{x}^U] : x_i = x_i^U\}.\end{aligned}$$

State bounds of the system (2.3) can be obtained by solving the following auxiliary system of ODE-IVPs:

$$\begin{aligned}\dot{x}_i^L &= f_i^L([t, t], \beta_i^L([\mathbf{x}^L(t), \mathbf{x}^U(t)]), P), \\ \dot{x}_i^U &= f_i^U([t, t], \beta_i^U([\mathbf{x}^L(t), \mathbf{x}^U(t)]), P), \\ [x_i^L(t_0), x_i^U(t_0)] &= Z_{0,i}(P)\end{aligned}\tag{2.5}$$

It is guaranteed by the initial condition that $Z_{0,i}(P)$ encloses all solutions of (2.3) at $t = t_0$. Furthermore, (2.5) ensures propagation of this inclusion property for $t \in (t_0, t_f]$. This can be visualized, for example, with the lower bound as follows. The “flattening” operator β_i^L guarantees that f_i^L computes the lower bound of f_i on the i^{th} lower face of $[\mathbf{x}^L(t), \mathbf{x}^U(t)]$ where $x_i^L(t)$ decreases faster than all trajectories of (2.3) because it has already been set incident on the i^{th} lower bound ($x_i(t, \mathbf{p}) = x_i^L(t)$). Propagation of much narrower intervals refined by β_i^L, β_i^U will help overcome the dependency problem and dramatically enhance the effectiveness of the calculated bounds.

For systems in which additional bounding information X_s is known *a priori*, say by some physical information, and we have $X_s \subset \mathcal{R}(t)$, an interval refinement operator can be

applied to the standard DI for further reducing overestimation of the bounding results [42]. Furthermore, redundant model equations could also be utilized to reduce the conservatism of the bounds, as detailed in the recent work by Shen and Scott [26].

Discrete-Time Differential Inequalities

In discrete-time DI, the system of ODE-IVPs (2.3) is reformulated using the explicit Euler method by discretizing the time domain I into $K + 1$ nodes:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + hf(t_k, \mathbf{x}_k, \mathbf{p}), \quad \mathbf{x}_0 = \zeta_0(\mathbf{p}), \quad k = 0, \dots, K, \quad (2.6)$$

where $k \in \{0, \dots, K\}$ is the time node after explicit Euler discretization, and $h = (t_f - t_0)/K$ is the step size. Analogous to solving the interval-based system (2.4), the explicit Euler algorithm can be extended for bounding the system by implementation of IA calculations numerically:

$$\begin{aligned} x_{k+1,i}^L &= x_{k,i}^L + hf_i^L([t_k, t_k], [\mathbf{x}_k^L, \mathbf{x}_k^U], P), \quad x_{0,i}^L = \zeta_{0,i}^L(P), \\ x_{k+1,i}^U &= x_{k,i}^U + hf_i^U([t_k, t_k], [\mathbf{x}_k^L, \mathbf{x}_k^U], P), \quad x_{0,i}^U = \zeta_{0,i}^U(P). \end{aligned} \quad (2.7)$$

The key idea of discrete-time DI is applying β_i^L, β_i^U defined by Definition 2.1.6 to the discretized system of ODEs (2.6). Thus, the standard discrete-time DI algorithms using the

explicit Euler method can be defined as:

$$\begin{aligned} x_{k+1,i}^L &= x_{k,i}^L + hf_i^L([t_k, t_k], \beta_i^L([\mathbf{x}_k^L, \mathbf{x}_k^U]), P), \quad x_{0,i}^L = \zeta_{0,i}^L(P), \\ x_{k+1,i}^U &= x_{k,i}^U + hf_i^U([t_k, t_k], \beta_i^U([\mathbf{x}_k^L, \mathbf{x}_k^U]), P), \quad x_{0,i}^U = \zeta_{0,i}^U(P). \end{aligned} \quad (2.8)$$

In the continuous-time system, the trajectories of \mathbf{x} cannot leave the interval enclosure X without crossing its bounds. Therefore, in the discrete-time system, only ensuring $x_{k,i}^L$ and $x_{k,i}^U$ decrease or increase, respectively, faster than all trajectories $x_i(t)$, as $k \rightarrow K$ for the i^{th} lower and upper face of X will guarantee that all trajectories of the continuous-time solutions $x_i(t)$ lie within the calculated bounds. The discrete-time DI method can also be improved by introducing the interval refinement operator when extra physical information can be extracted from the physical or logical properties of the original system and quantified as physical bounds X_s [51].

2.1.6 Optimization Problem Formulation

The development of bounding strategies is motivated by the need to solve optimization problems constrained by spatiotemporal PDEs to guaranteed global optimality. A general

formulation of this problem is:

$$\begin{aligned}
\phi^* = & \min_{\mathbf{p} \in PC\mathbb{R}^{n_p}} \phi(\tilde{\mathbf{x}}(y_i, t_f, \mathbf{p}), \mathbf{p}) \\
\text{s.t. } & \partial_t \tilde{\mathbf{x}}(y, t, \mathbf{p}) = \tilde{\mathbf{f}}(y, t, \tilde{\mathbf{x}}(y, t, \mathbf{p}), \partial_y \tilde{\mathbf{x}}(y, t, \mathbf{p}), \partial_{yy} \tilde{\mathbf{x}}(y, t, \mathbf{p}), \mathbf{p}), \quad \forall t \in I = [t_0, t_f] \quad (2.9) \\
& \tilde{\mathbf{x}}(y, t_0, \mathbf{p}) = \tilde{\boldsymbol{\zeta}}_0(y, \mathbf{p}) \\
& \tilde{\mathbf{f}}_l(\tilde{\mathbf{x}}(y_l, t, \mathbf{p}), \partial_y \tilde{\mathbf{x}}(y_l, t, \mathbf{p}), \mathbf{p}) = \tilde{\mathbf{x}}_l(t, \mathbf{p}) \\
& \tilde{\mathbf{f}}_r(\tilde{\mathbf{x}}(y_r, t, \mathbf{p}), \partial_y \tilde{\mathbf{x}}(y_r, t, \mathbf{p}), \mathbf{p}) = \tilde{\mathbf{x}}_r(t, \mathbf{p}) \\
& \mathbf{g}(\tilde{\mathbf{x}}(y_i, t_f, \mathbf{p}), \mathbf{p}) \leq \mathbf{0},
\end{aligned}$$

where $\phi : D \times P \rightarrow \mathbb{R}$ is the objective function and $\mathbf{g} : D \times P \rightarrow \mathbb{R}^{n_g}$ are the inequality constraints. These functions are shown to have explicit dependence on the final time node t_f and spatial node y_i ; however, these functions could explicitly depend on multiple discrete time and spatial nodes (e.g., for parameter estimation problems). The major complicating detail in this optimization formulation (2.9), is that it is constrained by an IBVP. Thus, simply verifying feasibility of a point requires the solution of the IBVP. In addition, (2.9) is nonconvex, in general, and therefore requires deterministic global optimization to guarantee optimality. Thus, providing rigorous and efficient global bounds are extremely important for enabling the solution of this problem.

2.2 Methods

In this section, a new algorithm is developed for bounding solutions of parametric PDEs. In addition, a method for quantifying the *tightness* or conservatism of calculated bounds

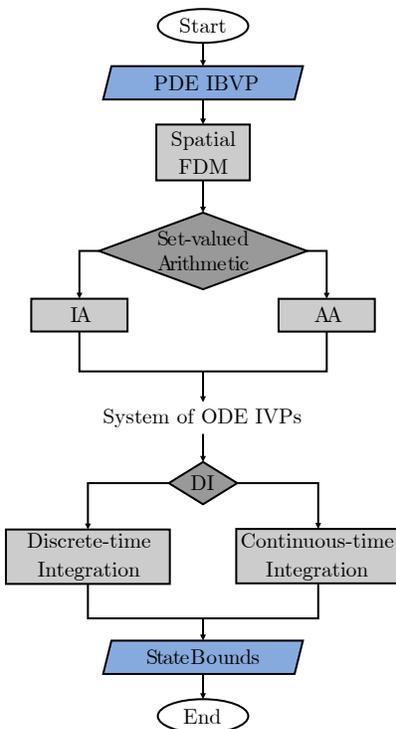


Figure 2.2: A flowchart of the high-level structure of our proposed algorithm for bounding IBVPs is illustrated. The IBVP is first reformulated into a system of ODE-IVPs by applying the spatial finite difference (FD) method. The user can select the set-valued arithmetic (conventional IA or the new IA/AA method) and type of differential inequalities (DI) approach (discrete- or continuous-time) to calculate state bounds.

is developed to enable quantitative comparisons of different approaches. Moreover, an approach for applying the new bounding methods to deterministic global optimization problems constrained by spatiotemporal PDEs, is introduced.

2.2.1 Bounding Solutions of Parametric IBVPs

The algorithm flowchart for bounding a PDE system is illustrated in Figure 2.2. The basic approach is to use the method of lines with spatial FD approximations, and subsequently calculate state bounds of the resulting large-scale ODE-IVP system. For ease of introduction,

we consider a simplified case of (2.1) where the state $\tilde{x}(y, t, \mathbf{p})$ is a scalar:

$$\partial_t \tilde{x}(y, t, \mathbf{p}) = \tilde{f}(y, t, \tilde{x}(y, t, \mathbf{p}), \partial_y \tilde{x}(y, t, \mathbf{p}), \partial_{yy} \tilde{x}(y, t, \mathbf{p}), \mathbf{p}) \quad (2.10)$$

$$\tilde{x}(y, t_0, \mathbf{p}) = \tilde{\zeta}_0(y, \mathbf{p})$$

$$\tilde{f}_l(\tilde{x}(y_l, t, \mathbf{p}), \partial_y \tilde{x}(y_l, t, \mathbf{p}), \mathbf{p}) = \tilde{x}_l(t, \mathbf{p})$$

$$\tilde{f}_r(\tilde{x}(y_r, t, \mathbf{p}), \partial_y \tilde{x}(y_r, t, \mathbf{p}), \mathbf{p}) = \tilde{x}_r(t, \mathbf{p}),$$

with $y \in Y = [y_l, y_r] \in \mathbb{I}\mathbb{R}$ and $\mathbf{p} \in P \in \mathbb{R}^{n_p}$. First, the FD method is applied to discretize the spatial domain of the IBVP (2.10). By Definition 2.1.5, the continuous state variable $\tilde{x}(\cdot, \cdot, \cdot)$ is converted into an unknown discrete space state variable vector $\mathbf{x}(\cdot, \cdot) = (x_1(\cdot, \cdot), \dots, x_N(\cdot, \cdot)) \in \mathbb{R}^N$. Then, the spatial derivatives $\partial_y \tilde{x}(\cdot, \cdot, \cdot)$ and $\partial_{yy} \tilde{x}(\cdot, \cdot, \cdot)$ are approximated using the FD approximations. In general, the FD approximations for the convection term $\partial_y \tilde{x}(\cdot, \cdot, \cdot)$ should respect the direction of information propagation for numerical stability. In this study, we consider all forward, backward, and centered FD approximations separately to fully demonstrate the results of the bounding procedure for each form. The FD approximations for the spatial derivatives are given by Definition 2.1.5 and represented here by $\mathbf{x}^f(t, \mathbf{p})$, $\mathbf{x}^b(t, \mathbf{p})$, and $\mathbf{x}^c(t, \mathbf{p})$, for the forward, backward, and centered FD approximations of the first derivatives, respectively, and $\mathbf{x}^{c2}(t, \mathbf{p})$ for the second derivatives, with $(t, \mathbf{p}) \in I \times P$. Based on this notation, interval extensions of these FD approximations on P at time t are represented by $X^j(t, P) = [\tilde{\mathbf{x}}(t)^{j,L}, \tilde{\mathbf{x}}(t)^{j,U}]$ with $j \in \{f, b, c, c2\}$. Consequently, the original IBVP (2.10) can be converted into an N -dimensional system of ODE-IVPs and the equations for the interior nodes ($i = 2, \dots, N-1$)

are given by:

$$\frac{d}{dt}x_i(t, \mathbf{p}) = f_i(t, x_i(t, \mathbf{p}), x_i^j(t, \mathbf{p}), x_i^{c2}(t, \mathbf{p}), \mathbf{p}) \quad (2.11)$$

where $f_i : I \times D \times \mathbb{R} \times \mathbb{R} \times P \rightarrow \mathbb{R}$ is defined as $f_i(\cdot, \cdot, \cdot, \cdot, \cdot) = \tilde{f}(y_i, \cdot, \cdot, \cdot, \cdot)$, for $i = 2, \dots, N-1$, and $j \in \{f, b, c\}$ represent forward, backward, and centered FD approximations, respectively. The initial conditions for the interior nodes can be set as $x_i(t_0, \mathbf{p}) = \tilde{\zeta}_0(y_i, \mathbf{p})$.

Boundary Conditions

The theory of boundary conditions are summarized by [48]. For consistency and completeness, we introduce in this section the notation and formatting for each type of boundary condition ($i \in \{1, N\}$) considered herein for discrete-space systems discretized from (2.10).

A Dirichlet boundary condition takes the form

$$\tilde{x}(y_l, t, \mathbf{p}) = \tilde{x}_l(t, \mathbf{p}), \text{ or}$$

$$\tilde{x}(y_r, t, \mathbf{p}) = \tilde{x}_r(t, \mathbf{p}).$$

When a Dirichlet boundary condition is applied, we simply set the value/form of the corresponding node and take the time derivative:

$$\begin{aligned} \frac{d}{dt}x_1(t, \mathbf{p}) &= \frac{d}{dt}\tilde{x}_l(t, \mathbf{p}), \\ \frac{d}{dt}x_N(t, \mathbf{p}) &= \frac{d}{dt}\tilde{x}_r(t, \mathbf{p}), \end{aligned}$$

and the initial value of the corresponding variable is set to the specified value (e.g., $x_1(t_0, \mathbf{p}) = \tilde{x}(y_l, t_0, \mathbf{p})$).

A Neumann boundary condition takes the form

$$\partial_y \tilde{x}(y_l, t, \mathbf{p}) = \tilde{x}_l(t, \mathbf{p}), \text{ or}$$

$$\partial_y \tilde{x}(y_r, t, \mathbf{p}) = \tilde{x}_r(t, \mathbf{p}).$$

When a Neumann boundary condition is applied, different formats of the corresponding ODE-IVP are defined based on the type of FD approximations. If the discretized right-hand side function only contains forward or backward FD approximations, a Neumann boundary condition may be directly applied at the left node as

$$\begin{aligned} x_1^f(t, \mathbf{p}) &= \tilde{x}_l(t, \mathbf{p}) \\ \Rightarrow \frac{d}{dt} x_1(t, \mathbf{p}) &= f_1(t, x_1(t, \mathbf{p}), \tilde{x}_l(t, \mathbf{p}), \mathbf{p}), \end{aligned}$$

and directly applied at the right node as

$$\begin{aligned} x_N^b(t, \mathbf{p}) &= \tilde{x}_r(t, \mathbf{p}) \\ \Rightarrow \frac{d}{dt} x_N(t, \mathbf{p}) &= f_N(t, x_N(t, \mathbf{p}), \tilde{x}_r(t, \mathbf{p}), \mathbf{p}), \end{aligned}$$

If centered FD approximations are used for the interior nodes, to retain the same level of accuracy, fictitious spatial nodes should be introduced. On the left-hand boundary y_l , a fictitious node y_* should be introduced with a corresponding value of $\tilde{x}(y_*, t, \mathbf{p}) = x_*(t, \mathbf{p})$. Then, the centered FD approximation can be used at the left boundary condition: $(x_2(t, \mathbf{p}) - x_*(t, \mathbf{p})) / (2\Delta y_1) = \tilde{x}_l(t, \mathbf{p})$. As a result, the value of the fictitious node can be solved for

as $x_*(t, \mathbf{p}) = x_2(t, \mathbf{p}) - 2\tilde{x}_l(t, \mathbf{p})\Delta y_1$ and this result is substituted into the right-hand side equation for the interior nodes evaluated at node $i = 1$ as

$$\frac{d}{dt}x_1(t, \mathbf{p}) = f_1(t, x_1(t, \mathbf{p}), x_1^c(t, \mathbf{p}), x_1^{c2}(t, \mathbf{p}), \mathbf{p}).$$

The initial condition for this boundary node can be set the same as the rest of the interior nodes as $x_1(t_0, \mathbf{p}) = \tilde{\zeta}_0(y_1, \mathbf{p})$. The same procedures can be applied for the Neumann boundary condition at the right boundary node ($i = N$) with the fictitious node as $x_* = x_{N-1}(t, \mathbf{p}) + 2\tilde{x}_r(t, \mathbf{p})\Delta y_{N-1}$.

A Robin boundary condition takes the form

$$\begin{aligned}\alpha_1\partial_y\tilde{x}(y_l, t, \mathbf{p}) + \alpha_2\tilde{x}(y_l, t, \mathbf{p}) &= \tilde{x}_l(t, \mathbf{p}), \quad \text{or} \\ \alpha_1\partial_y\tilde{x}(y_r, t, \mathbf{p}) + \alpha_2\tilde{x}(y_r, t, \mathbf{p}) &= \tilde{x}_r(t, \mathbf{p}).\end{aligned}$$

When a Robin boundary condition is applied, a combined strategy with that of the Dirichlet and Neumann conditions is applied. For example, with a Robin condition on the right boundary, if centered FD approximations are used, a fictitious spatial node y_* is introduced with a corresponding state $x_*(t, \mathbf{p})$. Then we set $\alpha_1(x_*(t, \mathbf{p}) - x_{N-1}(t, \mathbf{p})) / (2\Delta y_{N-1}) + \alpha_2x_N(t, \mathbf{p}) = \tilde{x}_r(t, \mathbf{p})$ and obtain the following expression for the fictitious node:

$$x_*(t, \mathbf{p}) = \frac{2\Delta y_{N-1}}{\alpha_1}(\tilde{x}_r(t, \mathbf{p}) - \alpha_2x_N(t, \mathbf{p})) + x_{N-1}(t, \mathbf{p}).$$

Then, this expression can be substituted into the discretized right-hand side function for

interior nodes evaluated at node N as

$$\frac{d}{dt}x_N(t, \mathbf{p}) = f_N(t, x_N(t, \mathbf{p}), x_N^c(t, \mathbf{p}), x_N^{c2}(t, \mathbf{p}), \mathbf{p}).$$

The initial condition at this node set as $x_N(t_0, \mathbf{p}) = \tilde{\zeta}_0(y_N, \mathbf{p})$. The same procedures can be applied for the Robin boundary condition at the left boundary node ($i = 1$) with the fictitious node as

$$x_*(t, \mathbf{p}) = \frac{2\Delta y_1}{\alpha_1}(\alpha_2 x_1(t, \mathbf{p}) - \tilde{x}_l(t, \mathbf{p})) + x_2(t, \mathbf{p}).$$

State Bounds of Discrete-Space ODE-IVPs

The vector form of the discrete-space system with interior equations defined as (2.11) and boundary equations given by Section 2.2.1 can be expressed as

$$\dot{\mathbf{x}}(t, \mathbf{p}) = \mathbf{f}(t, \mathbf{x}(t, \mathbf{p}), \mathbf{x}^j(t, \mathbf{p}), \mathbf{x}^{c2}(t, \mathbf{p}), \mathbf{p}), \quad \mathbf{x}(t_0, \mathbf{p}) = \boldsymbol{\zeta}_0(\mathbf{p}), \quad \forall t \in I, \quad (2.12)$$

where $\boldsymbol{\zeta}_0(\cdot) = (\zeta_{0,1}(\cdot), \dots, \zeta_{0,N}(\cdot)) \in \mathbb{R}^N$, and $\zeta_{0,i}(\cdot) = \tilde{\zeta}_0(y_i, \cdot)$ for $i = 2, \dots, N - 1$. As for boundary nodes ($i = 1$ or N), the value of $\zeta_{0,i}(\cdot)$ depends on the type of boundary conditions, and details are introduced in Section 2.2.1.

The objective is to compute state bounds for the system (2.12) on $I \times P$. As indicated by Definition 2.1.3, the reachable set for the continuous PDE (2.10) is given by $\mathcal{R}(y, t) = \{\tilde{x}(y, t, \mathbf{p}) : \mathbf{p} \in P\}$. Similarly, the reachable set of (2.12) is defined for every $i \in \{1, \dots, N\}$ by:

$$\mathcal{R}_i(t) = \{x_i(t, \mathbf{p}) : \mathbf{p} \in P\}, \quad (2.13)$$

and $x_i^L(t), x_i^U(t)$ are state bounds for $x_i(t, \mathbf{p})$ if $\mathcal{R}_i(t) \subset [x_i^L, x_i^U]$, with $t \in I$. It is assumed that an interval extension and inclusion function $F : [t, t] \times \mathbb{I}D \times \mathbb{I}P \rightarrow \mathbb{I}\mathbb{R}^N$ is available for \mathbf{f} . The i -th component of F is denoted $F_i([t, t], D, P) = [f_i^L([t, t], D, P), f_i^U([t, t], D, P)]$. For the initial condition, $\zeta_0(\mathbf{p}) \in Z_0(P) = [\zeta_0^L, \zeta_0^U]$. With these definitions, the bounds for each component x_i from (2.12) on P can be derived as:

$$\begin{aligned} \dot{x}_i^L(t) &= f_i^L([t, t], [\mathbf{x}^L(t), \mathbf{x}^U(t)], [\mathbf{x}^{j,L}(t), \mathbf{x}^{j,U}(t)], [\mathbf{x}^{c2,L}(t), \mathbf{x}^{c2,U}(t)], P), \\ \dot{x}_i^U(t) &= f_i^U([t, t], [\mathbf{x}^L(t), \mathbf{x}^U(t)], [\mathbf{x}^{j,L}(t), \mathbf{x}^{j,U}(t)], [\mathbf{x}^{c2,L}(t), \mathbf{x}^{c2,U}(t)], P), \\ [x_i^L(t_0), x_i^U(t_0)] &= Z_{0,i}(P), \end{aligned} \quad (2.14)$$

where $j \in \{\mathbf{f}, \mathbf{b}, \mathbf{c}\}$. The bounds for FD approximations $[\mathbf{x}^{j,L}(t), \mathbf{x}^{j,U}(t)]$ can be calculated by taking natural interval extensions of the FD approximations on the interval $[\mathbf{x}^L(t), \mathbf{x}^U(t)]$ based on Definition 2.1.5. Then, DI methods should be applied to the interval-valued ODE-IVPs (2.14) to further reduce conservatism of the state bounds. In particular, flattening operations on FD approximations are introduced in the following definition.

Definition 2.2.1 (Flattening Operator on Finite Difference Approximations). *Let $\mathcal{F}^j : \mathbb{I}\mathbb{R}^N \rightarrow \mathbb{I}\mathbb{R}^N$ be such that $\mathcal{F}^j(X) \equiv X^j$ with $j \in \{\mathbf{f}, \mathbf{b}, \mathbf{c}, \mathbf{c2}\}$, that represents the FD approximation function based on the corresponding FD scheme (Definition 2.1.5) and boundary conditions (Section 2.2.1). Then, the flattening operators $\mathcal{B}_i^L, \mathcal{B}_i^U : \mathbb{I}\mathbb{R}^N \rightarrow \mathbb{I}\mathbb{R}^N$ are defined*

for FD approximations as:

$$\mathcal{B}_i^L([\mathbf{x}^{j,L}, \mathbf{x}^{j,U}]) = \mathcal{F}^j(\beta_i^L([\mathbf{x}^L, \mathbf{x}^U])),$$

$$\mathcal{B}_i^U([\mathbf{x}^{j,L}, \mathbf{x}^{j,U}]) = \mathcal{F}^j(\beta_i^U([\mathbf{x}^L, \mathbf{x}^U])),$$

where $j \in \{\text{f, b, c, c2}\}$. In other words, $\mathcal{B}_i^L, \mathcal{B}_i^U$ are applied to interval extensions of FD approximations as polynomial expressions of $\beta_i^L([\mathbf{x}^L, \mathbf{x}^U])$ and $\beta_i^U([\mathbf{x}^L, \mathbf{x}^U])$.

While applying DI methods, the flattening operators in Definitions 2.1.6 and 2.2.1 are utilized to create state bounds as the solutions of the following auxiliary ODE-IVPs:

$$\dot{x}_i^L(t) = f_i^L([t, t], \beta_i^L([\mathbf{x}^L(t), \mathbf{x}^U(t)]), \mathcal{B}_i^L([\mathbf{x}^{j,L}(t), \mathbf{x}^{j,U}(t)]), \mathcal{B}_i^L([\mathbf{x}^{c2,L}(t), \mathbf{x}^{c2,U}(t)]), P), \quad (2.15)$$

$$\dot{x}_i^U(t) = f_i^U([t, t], \beta_i^U([\mathbf{x}^L(t), \mathbf{x}^U(t)]), \mathcal{B}_i^U([\mathbf{x}^{j,L}(t), \mathbf{x}^{j,U}(t)]), \mathcal{B}_i^U([\mathbf{x}^{c2,L}(t), \mathbf{x}^{c2,U}(t)]), P),$$

$$[x_i^L(t_0), x_i^U(t_0)] = Z_{0,i}(P),$$

where $j \in \{\text{f, b, c}\}$. Both discrete-time and continuous-time integration approaches can be used here. When applying discrete-time DI, we have:

$$x_{k+1,i}^L = x_{k,i}^L + hf_i^L([t_k, t_k], \beta_i^L([\mathbf{x}_k^L, \mathbf{x}_k^U]), \mathcal{B}_i^L([\mathbf{x}_k^{j,L}, \mathbf{x}_k^{j,U}]), \mathcal{B}_i^L([\mathbf{x}_k^{c2,L}, \mathbf{x}_k^{c2,U}]), P), \quad (2.16)$$

$$x_{k+1,i}^U = x_{k,i}^U + hf_i^U([t_k, t_k], \beta_i^U([\mathbf{x}_k^L, \mathbf{x}_k^U]), \mathcal{B}_i^U([\mathbf{x}_k^{j,L}, \mathbf{x}_k^{j,U}]), \mathcal{B}_i^U([\mathbf{x}_k^{c2,L}, \mathbf{x}_k^{c2,U}]), P),$$

$$[x_{0,i}^L, x_{0,i}^U] = Z_{0,i}(P).$$

Note that the validity of (2.16) can only be established according to the hypotheses of Theorem 1 of [51].

If prescribed bounding information $X_s \subset \mathbb{R}^N$ is known *a priori* by physical or mathematical arguments, an interval refinement operator can also be applied to (2.15). However, it was observed through several proof-of-concept experiments, that this approach is typically much more computationally expensive due to the dimensionality of the ODE-IVP system formed by discretizing the IBVP. It was observed that the proposed DI algorithm will take around 50 to 100 times longer with the interval refinement operators than without. The significant disparity in time cost overshadows any improvements gained in the reduction of conservatism in the resulting bounds. This result is only compounded for global optimization. [42] reached an opposite conclusion (i.e., the additional time cost was “far outweighed by the quality of the resulting enclosures”). We suspect that the much larger problem dimensionality is a major factor here.

The method of incorporating AA is also proposed as an alternative to purely using IA for calculating interval extensions and inclusion functions of the FD approximations for the spatial derivatives x_i^j with $j \in \{\text{f, b, c, c2}\}$.

Definition 2.2.2 (Finite Difference Approximations in Affine Form). *Let $\hat{x}_i(t, \boldsymbol{\varepsilon}) = \chi_{0,i} + \sum_{l=1}^m \chi_{l,i} \varepsilon_l$ denote the affine form of the real quantity $x_i(t, \mathbf{p}) \in [x_i^L(t), x_i^U(t)]$, $\forall (t, \mathbf{p}) \in I \times P$, where $\chi_{0,i} = (x_i^L(t) + x_i^U(t))/2$, $\chi_{1,i} = (x_i^U(t) - x_i^L(t))/2$, and $\chi_{l,i} = 0$, $\forall l > 1$. Written compactly in vector form, we have $\hat{\mathbf{x}}(t, \boldsymbol{\varepsilon}) = (\hat{x}_1(t, \boldsymbol{\varepsilon}), \dots, \hat{x}_N(t, \boldsymbol{\varepsilon}))$. The FD approximations*

for the spatial derivatives in affine form are defined as:

$$\hat{x}_i^f(t, \varepsilon) = \frac{\hat{x}_{i+1}(t, \varepsilon) - \hat{x}_i(t, \varepsilon)}{\Delta y_i}, \quad \forall i = 1, \dots, N-1$$

$$\hat{x}_i^b(t, \varepsilon) = \frac{\hat{x}_i(t, \varepsilon) - \hat{x}_{i-1}(t, \varepsilon)}{\Delta y_{i-1}}, \quad \forall i = 2, \dots, N$$

$$\hat{x}_i^c(t, \varepsilon) = \frac{\Delta y_{i-1}^2 \hat{x}_{i+1}(t, \varepsilon) + (\Delta y_i^2 - \Delta y_{i-1}^2) \hat{x}_i(t, \varepsilon) - \Delta y_i^2 \hat{x}_{i-1}(t, \varepsilon)}{\Delta y_{i-1} \Delta y_i (\Delta y_{i-1} + \Delta y_i)}, \quad \forall i = 2, \dots, N-1$$

$$\hat{x}_i^{c2}(t, \varepsilon) = \frac{2(\Delta y_{i-1} \hat{x}_{i+1}(t, \varepsilon) - (\Delta y_{i-1} + \Delta y_i) \hat{x}_i(t, \varepsilon) + \Delta y_i \hat{x}_{i-1}(t, \varepsilon))}{\Delta y_{i-1} \Delta y_i (\Delta y_{i-1} + \Delta y_i)}, \quad \forall i = 2, \dots, N-1$$

Remark 5. Interval bounds for these FD approximations in affine form are denoted $\hat{X}^j(t) = [\hat{\mathbf{x}}^{j,L}(t), \hat{\mathbf{x}}^{j,U}(t)]$ with $j \in \{f, b, c, c2\}$. The upper and lower bounds can be computed based on Lemma 1.

Remark 6. When using equidistant step sizes $\Delta y = (y_r - y_l)/(N-1)$, FD approximations in affine form can be simplified to $\hat{x}_i^f = (\hat{x}_{i+1} - \hat{x}_i)/\Delta y$, $\hat{x}_i^b = (\hat{x}_i - \hat{x}_{i-1})/\Delta y$ and $\hat{x}_i^c = (\hat{x}_{i+1} - \hat{x}_{i-1})/(2\Delta y)$ for first derivatives and $\hat{x}_i^{c2} = (\hat{x}_{i+1} - 2\hat{x}_i + \hat{x}_{i-1})/\Delta y^2$ for second derivatives.

If we use AA for calculating FD approximations and apply DI with flattening operators defined in Definition 2.1.6 and Definition 2.2.1, state bounds are calculated by integrating:

$$\dot{x}_i^L(t) = f_i^L([t, t], \beta_i^L([\mathbf{x}^L(t), \mathbf{x}^U(t)]), \mathcal{B}_i^L([\hat{\mathbf{x}}^{j,L}(t), \hat{\mathbf{x}}^{j,U}(t)]), \mathcal{B}_i^L([\hat{\mathbf{x}}^{c2,L}(t), \hat{\mathbf{x}}^{c2,U}(t)]), P), \quad (2.17)$$

$$\dot{x}_i^U(t) = f_i^U([t, t], \beta_i^U([\mathbf{x}^L(t), \mathbf{x}^U(t)]), \mathcal{B}_i^U([\hat{\mathbf{x}}^{j,L}(t), \hat{\mathbf{x}}^{j,U}(t)]), \mathcal{B}_i^U([\hat{\mathbf{x}}^{c2,L}(t), \hat{\mathbf{x}}^{c2,U}(t)]), P),$$

$$[x_i^L(t_0), x_i^U(t_0)] = Z_{0,i}(P),$$

where $j \in \{f, b, c\}$. The discrete-time analog of (2.17) is established as:

$$x_{k+1,i}^L = x_{k,i}^L + hf_i^L([t_k, t_k], \beta_i^L([\mathbf{x}_k^L, \mathbf{x}_k^U]), \mathcal{B}_i^L([\hat{\mathbf{x}}_k^{j,L}, \hat{\mathbf{x}}_k^{j,U}]), \mathcal{B}_i^L([\hat{\mathbf{x}}_k^{c2,L}, \hat{\mathbf{x}}_k^{c2,U}]), P), \quad (2.18)$$

$$x_{k+1,i}^U = x_{k,i}^U + hf_i^U([t_k, t_k], \beta_i^U([\mathbf{x}_k^L, \mathbf{x}_k^U]), \mathcal{B}_i^U([\hat{\mathbf{x}}_k^{j,L}, \hat{\mathbf{x}}_k^{j,U}]), \mathcal{B}_i^U([\hat{\mathbf{x}}_k^{c2,L}, \hat{\mathbf{x}}_k^{c2,U}]), P),$$

$$[x_{0,i}^L, x_{0,i}^U] = Z_{0,i}(P).$$

To save computational cost for converting between traditional intervals and affine forms (see Lemma 1), we reformulate the subtraction rule of AA for two correlated intervals. This is because we only invoke AA rules for calculating interval bounds of FD approximations and the addition rule is identical between IA and AA. Therefore, reformulating the AA subtraction rule into an interval-valued format avoids conversion and invocation of an AA library to greatly accelerate calculation speeds.

Lemma 2. *Consider two correlated real scalars $a \in A = [a^L, a^U]$ and $b \in B = [b^L, b^U]$. They are converted to their affine forms based on Lemma 1 as $\hat{a} = a_0 + a_1\varepsilon_1$ and $\hat{b} = b_0 + b_1\varepsilon_1$. Define $\hat{c} = \hat{a} - \hat{b}$ and an interval extension and inclusion function of \hat{c} as \hat{C} . Then,*

$$\hat{C} = \begin{cases} [a^L - b^L, a^U - b^U] & \forall a_1 \geq b_1 \\ [a^U - b^U, a^L - b^L] & \text{else.} \end{cases} \quad (2.19)$$

Proof. By Lemma 1, $a_0 = (a^L + a^U)/2$, $a_1 = (a^U - a^L)/2$, $b_0 = (b^L + b^U)/2$, $b_1 = (b^U - b^L)/2$.

Then,

$$\begin{aligned}
\hat{c} &= \hat{a} - \hat{b} = (a_0 + a_1\varepsilon_1) - (b_0 + b_1\varepsilon_1) \\
&= (a_0 - b_0) + (a_1 - b_1)\varepsilon_1 \\
&= \left(\frac{a^L + a^U}{2} - \frac{b^L + b^U}{2} \right) + \left(\frac{a^U - a^L}{2} - \frac{b^U - b^L}{2} \right) \varepsilon_1
\end{aligned}$$

In interval form, we can write $\hat{C} = [c_0 - r_c, c_0 + r_c]$ with $c_0 = \left(\frac{a^L + a^U}{2} - \frac{b^L + b^U}{2} \right)$ and $r_c = \left| \left(\frac{a^U - a^L}{2} - \frac{b^U - b^L}{2} \right) \right|$. Suppose $a_1 \geq b_1$. Then, $a^U - a^L \geq b^U - b^L$, and we get $r_c = \left(\frac{a^U - a^L}{2} - \frac{b^U - b^L}{2} \right)$.

In interval form we get

$$\begin{aligned}
\hat{C} &= [c_0 - r_c, c_0 + r_c] \\
&= \left[\frac{a^L + a^U + a^L - a^U}{2} - \frac{b^L + b^U + b^L - b^U}{2}, \frac{a^L + a^U - a^L + a^U}{2} - \frac{b^L + b^U - b^L + b^U}{2} \right] \\
&= [a^L - b^L, a^U - b^U]
\end{aligned}$$

Now, suppose $a_1 < b_1$. Then, we have the opposite case ($a^U - a^L < b^U - b^L$), and we get $r_c = -\left(\frac{a^U - a^L}{2} - \frac{b^U - b^L}{2} \right)$. Therefore, in interval form we get

$$\begin{aligned}
\hat{C} &= [c_0 - r_c, c_0 + r_c] \\
&= \left[\frac{a^L + a^U + a^U - a^L}{2} - \frac{b^L + b^U - b^L + b^U}{2}, \frac{a^L + a^U + a^L - a^U}{2} - \frac{b^L + b^U + b^L - b^U}{2} \right] \\
&= [a^U - b^U, a^L - b^L]
\end{aligned}$$

□

In the proposed method, AA is only applied for calculations of bounds for FD approximations, and the affine forms $\hat{x}_1, \dots, \hat{x}_N$ are regarded as correlated. This is because the discrete-space states $x_i(t, \mathbf{p})$, $\forall i$ are all discretized forms of the same continuous-space states $\tilde{x}(y_i, t, \mathbf{p})$. Note that this approach should follow the assumption that affine forms of the states share the same noise term ε_1 . The inclusion property of this approach requires further research and proof. However, several numerical examples have been implemented to verify empirically the validity of the approach.

As will be demonstrated through several numerical experiments in Section 2.3, using AA instead of conventional IA for FD approximations does not sacrifice computational efficiency. In addition, the resulting bounds constructed by means of this mixed IA/AA approach are at least as tight as the IA approach with similar computational cost. In particular, while using the centered FD approximation for discretization of the first-order spatial derivative, performing mixed IA/AA method can provide evidently better bounds without violating the inclusion properties. To deal with more complicated PDE systems, the superiority of using the IA/AA method instead of the pure IA method becomes more apparent with equivalent time cost.

2.2.2 IBVPs with Coupled States

In many chemical engineering applications, models capture conservation of several species and energy, simultaneously. Consequently, IBVPs modeling these systems have multiple spatiotemporal states that are coupled. To consider these cases, in this section a generalization of the discrete-space form (2.12) to n_x states, originally defined in (2.1) as

$\tilde{\mathbf{x}}(y, t, \mathbf{p}) = (\tilde{x}_1(y, t, \mathbf{p}), \dots, \tilde{x}_{n_x}(y, t, \mathbf{p})) \in D \subset \mathbb{R}^{n_x}$, is presented.

First, the spatial domain is again discretized into N nodes y_1, \dots, y_N by Definition 2.1.5. Thus, at each node y_i , there are corresponding values of each state $\tilde{x}_1(y_i, t, \mathbf{p}), \dots, \tilde{x}_{n_x}(y_i, t, \mathbf{p})$ to be determined. In other words, there are $n_x \times N$ unknown dependent state variables:

$$x_{m,i}(t, \mathbf{p}) \equiv \tilde{x}_m(y_i, t, \mathbf{p}), \quad m = 1, \dots, n_x, \quad i = 1, \dots, N.$$

Next, a global state variable vector is defined that interlaces the individual state variables pertaining to each spatial node i as:

$$\mathbf{z}(t, \mathbf{p}) = (\mathbf{x}_1(t, \mathbf{p}), \mathbf{x}_2(t, \mathbf{p}), \dots, \mathbf{x}_N(t, \mathbf{p})) \in \mathbb{R}^{n_x N}$$

with $\mathbf{x}_i = (x_{1,i}, \dots, x_{n_x,i}) \in \mathbb{R}^{n_x}$ for $i = 1, \dots, N$.

In terms of the global index $l = n_x \cdot i - (n_x - m)$ with $m = 1, \dots, n_x$ and $i = 1, \dots, N$, we have

$$z_l = x_{m,i}, \quad l = 1, \dots, n_x N. \quad (2.20)$$

Therefore, all of the equations in (2.1) can be rewritten in terms of \mathbf{z} and the system of IBVPs is converted into an $n_x \times N$ -dimensional system of ODE-IVPs. Then, the same bounding methods can be applied to compute the state bounds for this system.

2.2.3 IBVPs With Higher Spatial Dimensionality

In this section, IBVPs with more than one spatial dimension are considered. Specifically, notation for systems with two spatial dimensions ($\mathbf{y} = (y_1, y_2)$) is presented to demonstrate

the the higher-dimensionality generalization approach. Let $t \in I = [t_0, t_f]$, $D \subset \mathbb{R}$ open, $Y \in \mathbb{I}\mathbb{R}^2$, $P \in \mathbb{I}\mathbb{R}^{n_p}$, $\tilde{\mathbf{f}} : Y \times I \times D \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\boldsymbol{\zeta}}_0 : Y \times P \rightarrow D$, $\tilde{\mathbf{f}}_l : D \times \mathbb{R}^{n_x} \times I \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\mathbf{x}}_l : I \times P \rightarrow \mathbb{R}^{n_x}$ at the left boundary, $\tilde{\mathbf{f}}_r : D \times \mathbb{R}^{n_x} \times I \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\mathbf{x}}_r : I \times P \rightarrow \mathbb{R}^{n_x}$ at the right boundary, $\tilde{\mathbf{f}}_t : D \times \mathbb{R}^{n_x} \times I \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\mathbf{x}}_t : I \times P \rightarrow \mathbb{R}^{n_x}$ at the top boundary, and $\tilde{\mathbf{f}}_b : D \times \mathbb{R}^{n_x} \times I \times P \rightarrow \mathbb{R}^{n_x}$, $\tilde{\mathbf{x}}_b : I \times P \rightarrow \mathbb{R}^{n_x}$ at the bottom boundary, then an IBVP with two spatial dimensions is defined as:

$$\partial_t \tilde{\mathbf{x}}(\mathbf{y}, t, \mathbf{p}) = \tilde{\mathbf{f}}(\mathbf{y}, t, \tilde{\mathbf{x}}(\mathbf{y}, t, \mathbf{p}), \partial_{y_1} \tilde{\mathbf{x}}(\mathbf{y}, t, \mathbf{p}), \partial_{y_2} \tilde{\mathbf{x}}(\mathbf{y}, t, \mathbf{p}), \partial_{y_1 y_1} \tilde{\mathbf{x}}(\mathbf{y}, t, \mathbf{p}), \partial_{y_2 y_2} \tilde{\mathbf{x}}(\mathbf{y}, t, \mathbf{p}), \mathbf{p}) \quad (2.21)$$

$$\tilde{\mathbf{x}}(\mathbf{y}, t_0, \mathbf{p}) = \tilde{\boldsymbol{\zeta}}_0(\mathbf{y}, \mathbf{p}),$$

$$\tilde{\mathbf{f}}_l(\tilde{\mathbf{x}}((y_l, y_2), t, \mathbf{p}), \partial_{y_1} \tilde{\mathbf{x}}((y_l, y_2), t, \mathbf{p}), \mathbf{p}) = \tilde{\mathbf{x}}_l(t, \mathbf{p}),$$

$$\tilde{\mathbf{f}}_r(\tilde{\mathbf{x}}((y_r, y_2), t, \mathbf{p}), \partial_{y_1} \tilde{\mathbf{x}}((y_r, y_2), t, \mathbf{p}), \mathbf{p}) = \tilde{\mathbf{x}}_r(t, \mathbf{p}),$$

$$\tilde{\mathbf{f}}_t(\tilde{\mathbf{x}}((y_1, y_t), t, \mathbf{p}), \partial_{y_2} \tilde{\mathbf{x}}((y_1, y_t), t, \mathbf{p}), \mathbf{p}) = \tilde{\mathbf{x}}_t(t, \mathbf{p}),$$

$$\tilde{\mathbf{f}}_b(\tilde{\mathbf{x}}((y_1, y_b), t, \mathbf{p}), \partial_{y_2} \tilde{\mathbf{x}}((y_1, y_b), t, \mathbf{p}), \mathbf{p}) = \tilde{\mathbf{x}}_b(t, \mathbf{p}),$$

which is a generalization of (2.1) in two spatial dimensions.

The multivariate state variable is defined as $\tilde{\mathbf{x}}(y, t, \mathbf{p}) = (\tilde{x}_1(\mathbf{y}, t, \mathbf{p}), \dots, \tilde{x}_{n_x}(\mathbf{y}, t, \mathbf{p}))$. The spatial domain in this case is a plane that is discretized with discrete points $y_{1,1}, \dots, y_{1,N_1}$ over the y_1 dimension and $y_{2,1}, \dots, y_{2,N_2}$ over the y_2 dimension. The discretization step sizes are then $\Delta y_{1,i} = y_{1,i+1} - y_{1,i}$ and $\Delta y_{2,i} = y_{2,i+1} - y_{2,i}$. For problems that lack complicated geometries and/or highly nonlinear phenomena, equidistant step sizes can be used to give $\Delta y_1 = (y_r - y_l)/(N_1 - 1)$ and $\Delta y_2 = (y_t - y_b)/(N_2 - 1)$. Non-uniform discretization is

useful when dealing with systems with strong nonlinearities or when spatial dimensions are not normalized. For spatial discretization, a grid of $N_1 \times N_2$ spatial nodes can be created as $\mathbf{y}_{i,j} = (y_{1,i}, y_{2,j})$ for $1 \leq i, j \leq N_1 N_2$. Thus, there are $N_1 \times N_2$ unknown states $\tilde{\mathbf{x}}((y_{1,i}, y_{2,j}), t, \mathbf{p})$ required to be determined over the integration time horizon:

$$x_{m,i,j}(t, \mathbf{p}) \equiv \tilde{x}_m(\mathbf{y}_{i,j}, t, \mathbf{p}), \quad m = 1, \dots, n_x, \quad i = 1, \dots, N_1, \quad j = 1, \dots, N_2.$$

For the spatial node (i, j) , a global index is introduced as $q = i + (j - 1)N_1$ with $i = 1, \dots, N_1$ and $j = 1, \dots, N_2$. Thus, we have

$$x_{m,q} = x_{m,i,j}, \quad m = 1, \dots, n_x, \quad q = 1, \dots, N_1 N_2.$$

Using the method introduced in Section 2.2.2 to interlace the individual state variables pertaining to each spatial node q , a single vector of unknown dependent variables is defined:

$$\mathbf{z}(t, \mathbf{p}) = (\mathbf{x}_1(t, \mathbf{p}), \dots, \mathbf{x}_{N_1 N_2}(t, \mathbf{p})) \in \mathbb{R}^{n_x N_1 N_2},$$

with $\mathbf{x}_q = (x_{1,q}, \dots, x_{n_x,q}) \in \mathbb{R}^{n_x}$ for $q = 1, \dots, N_1 N_2$.

Similarly, a global index for multiple coupled states is introduced as $l = n_x \cdot q - (n_x - m)$ with $m = 1, \dots, n_x$ and $q = 1, \dots, N_1 N_2$. Thus, the components of \mathbf{z} are defined as:

$$z_l = x_{m,q}, \quad l = 1, \dots, n_x N_1 N_2.$$

For the interior nodes ($2 \leq i \leq N_1 - 1$ and $2 \leq j \leq N_2 - 1$), the spatial derivatives can

be approximated using the same FD approximations by Definition 2.1.5 corresponding with the 2D indices. For example, $\partial_{y_1 y_1} \tilde{x}_m((y_{1,i}, y_{2,j}), t, \mathbf{p}) \approx (x_{m,i+1,j} - 2x_{m,i,j} + x_{m,i-1,j})/\Delta y_1^2$.

For the boundaries of the domain, the equations for the edges (excluding corner nodes) should be determined by the type of boundary conditions using the same methods from Section 2.2.1 applied based on the specific conditions. There may be discontinuities at the corners when two adjacent edges have different boundary conditions. Thus, we should use the average value of the adjacent variables for these nodes. In the case where one side of the corner has a Dirichlet boundary condition and the other side has a Neumann or Robin boundary condition, the Dirichlet condition takes precedence [48].

As a result, a system of $n_x \times N_1 \times N_2$ -dimensional ODE-IVPs is established and the proposed bounding method can be implemented with this system. Spatial dimensionalities greater than 2 (i.e., $\mathbf{y} \in \mathbb{R}^n$, $n > 2$) may also be considered using the same approach, but care must be taken to establish the appropriate global indexing schemes for the unknown dependent variable vector \mathbf{z} .

2.2.4 Volume of State Bounds for IBVP Systems

To compare the effectiveness of different bounding procedures, a method is required to quantify the tightness of the state bounds produced by each method. In the PDE context, the volume V_α of the enclosure produced by method α is defined as the volume between the upper and lower state bounds on $\tilde{x}(y, t, \mathbf{p})$. For systems with a single spatial dimension, this is a three-dimensional region calculated as the distance between the upper and lower state bounds integrated over the spatial dimension y and time dimension t . We propose

that the volume enclosed by the state bounds can be regarded as a criterion to evaluate the performance of different algorithms. The smaller the volume of the computed enclosure for the PDE system indicates less conservatism of the bounds. The 2D trapezoidal rule is used to integrate the volume numerically. From a continuous-space perspective, we want to evaluate

$$\mathcal{I}_Y(\mathcal{I}_I(\tilde{x}_\alpha^q(y, t))) = \int_{y_l}^{y_r} \int_{t_0}^{t_f} \tilde{x}_\alpha^q(y, t) dt dy \quad (2.22)$$

for bounding method α for each bound $q \in \{L, U\}$, and take the difference. The inner integral can be defined as $\mathcal{I}_I^q(y) = \int_{t_0}^{t_f} \tilde{x}_\alpha^q(y, t) dt$, and then the integral (2.22) can be rewritten as $\mathcal{I}_Y(\mathcal{I}_I^q(y)) = \int_{y_l}^{y_r} \mathcal{I}_I^q(y) dy$. For the discrete-space system, the spatial variable y is discretized into N nodes y_1, \dots, y_N , and thus, the outer integral of $\mathcal{I}_Y(\mathcal{I}_I^q(y))$ can be approximated using the one-dimension (1D) trapezoidal rule:

$$\mathcal{I}_Y(\mathcal{I}_I^q(y)) \approx \Delta y \left(\frac{1}{2} \mathcal{I}_I^q(y_1) + \mathcal{I}_I^q(y_2) + \dots + \mathcal{I}_I^q(y_{N-1}) + \frac{1}{2} \mathcal{I}_I^q(y_N) \right).$$

Next, the time variable t is discretized into $K + 1$ nodes: t_0, \dots, t_K . The inner integral for each $\mathcal{I}_I^q(y_i)$ is approximated by the 1D trapezoidal rule as:

$$\mathcal{I}_I^q(y_i) \approx h \left(\frac{1}{2} \tilde{x}_\alpha^q(y_i, t_0) + \tilde{x}_\alpha^q(y_i, t_1) + \dots + \tilde{x}_\alpha^q(y_i, t_{K-1}) + \frac{1}{2} \tilde{x}_\alpha^q(y_i, t_K) \right),$$

where $t_k = t_0 + kh, \forall k = 0, \dots, K$, and the step size is $h = (t_f - t_0)/K$. Since the FD method is used to discretize the spatial domain, we have $x_{i,\alpha}^q(t_k) \approx \tilde{x}_\alpha^q(y_i, t_k)$. Then the total integral

can be approximated numerically as:

$$\mathcal{I}_Y(\mathcal{I}_I(\tilde{x}_\alpha^q(y, t))) \approx \sum_{i=1}^N \sum_{k=0}^K \omega_{i,k} x_{i,\alpha}^q(t_k) h \Delta y,$$

with

$$\omega_{i,k} = \begin{cases} 1/4 & \text{at corner nodes} \\ 1/2 & \text{at edge nodes} \\ 1 & \text{at interior nodes.} \end{cases}$$

The truncation error for this approach is $\mathcal{O}(\Delta y^2 + h^2)$.

The volume V_α of the enclosure calculated by method α can then be approximated as:

$$\begin{aligned} V_\alpha &= \mathcal{I}_Y(\mathcal{I}_I(\tilde{x}_\alpha^U(y, t))) - \mathcal{I}_Y(\mathcal{I}_I(\tilde{x}_\alpha^L(y, t))) \\ &\approx \sum_{i=1}^N \sum_{k=0}^K \omega_{i,k} (x_{i,\alpha}^U(t_k) - x_{i,\alpha}^L(t_k)) h \Delta y. \end{aligned} \quad (2.23)$$

2.2.5 Deterministic Global Optimization

In this section, we formalize the application of the bounding procedure for enabling global dynamic optimization of spatially varying dynamical models. First, we formulate the dynamic optimization problem within the context of the spatially discretized system in this section. An IBVP as (2.10) was discretized into a system of ODE-IVPs as (2.12) for numer-

ical solution. In this context, (2.9) is rewritten as the dynamic optimization problem:

$$\begin{aligned}
\phi^* &= \min_{\mathbf{p} \in PC\mathbb{R}^{n_p}} \phi(\mathbf{x}(t_f, \mathbf{p}), \mathbf{p}) \\
\text{s.t. } \dot{\mathbf{x}}(t, \mathbf{p}) &= \mathbf{f}(t, \mathbf{x}(t, \mathbf{p}), \mathbf{x}^j(t, \mathbf{p}), \mathbf{x}^{c2}(t, \mathbf{p}), \mathbf{p}), \quad j \in \{\text{f, b, c}\}, \quad \forall t \in I = [t_0, t_f] \quad (2.24) \\
\mathbf{x}(t_0, \mathbf{p}) &= \zeta_0(\mathbf{p}) \\
\mathbf{g}(\mathbf{x}(t_f, \mathbf{p}), \mathbf{p}) &\leq \mathbf{0}
\end{aligned}$$

where ϕ and \mathbf{g} are continuously differentiable on their domains. This generalized formulation represents an objective function and inequality constraints dependent on the final time t_f , which accounts for typical process control applications where a decision is determined based on a desired final-time state. However, we do not restrict the optimization formulation to this and can also define an objective and constraints that depend on several discrete time points. We denote the global optimal solution value as ϕ^* and a global optimal solution as \mathbf{p}^* .

When applying a discrete-time scheme, such as explicit Euler, for integrating the equality constraints, the system of N ODE-IVPs is reformulated into a system of $N \times (K+1)$ algebraic equations. Thus, (2.24) is reformulated as:

$$\begin{aligned}
\phi^* &= \min_{\mathbf{p} \in PC\mathbb{R}^{n_p}} \phi(\mathbf{x}_K(\mathbf{p}), \mathbf{p}) \\
\text{s.t. } \mathbf{x}_{k+1}(\mathbf{p}) &= \mathbf{x}_k(\mathbf{p}) + h\mathbf{f}(t_k, \mathbf{x}_k(\mathbf{p}), \mathbf{x}_k^j(\mathbf{p}), \mathbf{x}_k^{c2}(\mathbf{p}), \mathbf{p}), \quad j \in \{\text{f, b, c}\}, \quad \forall k = 0, \dots, K \\
\mathbf{g}(\mathbf{x}_K(\mathbf{p}), \mathbf{p}) &\leq \mathbf{0}
\end{aligned} \tag{2.25}$$

where $\mathbf{x}_0, \dots, \mathbf{x}_K$ are the vector of state variables at each time node t_k , $k = 0, \dots, K$.

The architecture of the B&B algorithm for deterministic global optimization is illustrated in Figure 2.3. At a high level, the B&B algorithm iteratively partitions the original domain P (i.e., branches) into successively smaller subdomains P^l (i.e., nodes), such that $P = \cup P^l$. The algorithm then solves a sequence of upper- and lower-bounding problems on each subdomain. By comparing the obtained bounds across nodes, and by finding infeasible regions, the algorithm converges to an ϵ -optimal global solution in finitely-many iterations. In this work, the flexible B&B algorithm of [52], is used.

Valid upper bounds on (2.24) and (2.25) are provided in Definitions 2.2.3 and 2.2.4, respectively. It is typical to employ a local solver, such as IPOPT, to determine a feasible local optimal solution. Calculating valid lower bounds is the greatest challenge in solving (2.24) and (2.25). The corresponding lower-bounding problems are given in Definitions 2.2.5 and 2.2.6, respectively, and rely on the ability to calculate rigorous bounds on the ranges of all equations and convex outer-approximations of the feasible sets. As the algorithm proceeds, the best-found bounds are stored for comparison.

If the lower bound on ϕ in a subdomain is greater than the current best upper bound, then that subdomain cannot contain a global optimal solution, and is fathomed (deleted). The remaining subdomains are kept on a stack for further processing. The algorithm converges when the upper and lower bounds are within some tolerance ϵ . Since thousands of subdomains may be required for bounding and fathoming, a fast bounding procedure that is able to provide tight lower bounds is a significant advantage to accelerate overall convergence of the algorithm, provided that the procedure itself is not too computationally expensive. The bounding routines that are used by the B&B algorithm in this work, are defined in the

following.

Definition 2.2.3 (Continuous-Time System Upper Bound). *An upper bound on the optimal solution value of (2.24) is given by*

$$\phi_i^{UBD} = \phi(\mathbf{x}(t_f, \bar{\mathbf{p}}), \bar{\mathbf{p}}),$$

where $\bar{\mathbf{p}} \in P^l$ is any feasible point in the interval $P^l \subset P$.

Definition 2.2.4 (Discrete-Time System Upper Bound). *An upper bound on the optimal solution value of (2.25) is given by*

$$\phi_i^{UBD} = \phi(\mathbf{x}_K(\bar{\mathbf{p}}), \bar{\mathbf{p}}),$$

where $\bar{\mathbf{p}} \in P^l$ is any feasible point in the interval $P^l \subset P$.

Assumption 2. *Let $X = [\mathbf{x}^L(t_f), \mathbf{x}^U(t_f)]$ or $X = [\mathbf{x}_K^L, \mathbf{x}_K^U]$, depending on whether (2.17) or (2.18), respectively, is applied on P , and define $Z = X$. There exist continuous functions $\Phi : \mathbb{I}Z \times \mathbb{I}P \rightarrow \mathbb{I}\mathbb{R}$ and $G : \mathbb{I}Z \times \mathbb{I}P \rightarrow \mathbb{I}\mathbb{R}^{n_g}$ such that Φ is an interval extension and inclusion function of ϕ on $Z \times P$ and G is an interval extension and inclusion function of \mathbf{g} on $Z \times P$.*

Remark 7. *For ϕ and \mathbf{g} continuous on open sets containing $Z \times P$ that are composed of finitely-many arithmetic operations and compositions with transcendental functions, functions Φ and G satisfying 2 can be calculated by taking natural interval extensions [44].*

Definition 2.2.5 (Continuous-Time System Lower Bound). *Consider an interval $P^l \subset P$ and let $G(Z^l, P^l)$ and $\Phi(Z^l, P^l)$ adhere to Assumption 2 with $Z^l = X^l = [\mathbf{x}_i^L(t_f), \mathbf{x}_i^U(t_f)]$*

calculated from integrating the auxiliary ODE-IVPs (2.17) on P^l with $j \in \{f, b, c\}$. If $\mathbf{g}^L(Z^l, P^l) \leq \mathbf{0}$, then a lower bound on the global optimal solution value of (2.24) is given by:

$$\phi_l^{LBD} = \phi^L(Z^l, P^l). \quad (2.26)$$

A point at which ϕ_l^{LBD} is attained can be set as any $\mathbf{p} \in P^l$. If $\mathbf{g}^L(Z^l, P^l) > \mathbf{0}$, (2.24) is infeasible on P^l and $\phi_l^{LBD} := +\infty$.

Definition 2.2.6 (Discrete-Time System Lower Bound). Consider an interval $P^l \subset P$ and let $G(Z^l, P^l)$ and $\Phi(Z^l, P^l)$ adhere to Assumption 2 with $Z^l = X^l = [x_{l,K}^L, x_{l,K}^U]$ calculated from applying (2.18) on P^l with $j \in \{f, b, c\}$. If $\mathbf{g}^L(Z^l, P^l) \leq \mathbf{0}$, then a lower bound on the global optimal solution value of (2.25) is given by

$$\phi_l^{LBD} = \phi^L(Z^l, P^l).$$

A point at which ϕ_l^{LBD} is attained can be set as any $\mathbf{p} \in P^l$. If $\mathbf{g}^L(Z^l, P^l) > \mathbf{0}$, (2.25) is infeasible on P^l , and $\phi_l^{LBD} := +\infty$.

The spatial B&B algorithm [52, Alg. 3.1] finitely converges to ϵ -optimality using the appropriate bounding definitions (Definitions 2.2.3 through 2.2.6).

Deterministic Global Optimization Algorithm

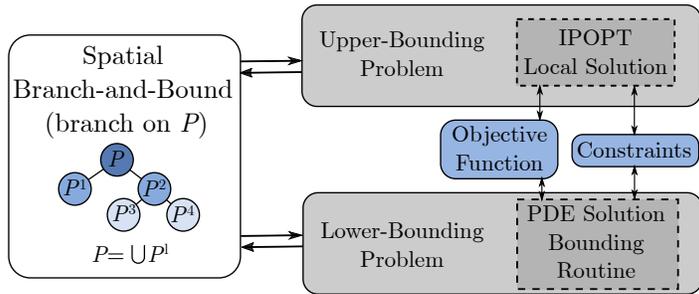


Figure 2.3: A block diagram illustrating the architecture of the deterministic global optimization algorithm is presented. The novel contributions of this work lie in the PDE Solution Bounding Routine. The implementation of this algorithm utilizes the flexible branch-and-bound framework of the EAGO solver [52]

2.3 Examples

In this section, we demonstrate the performance of our novel bounding algorithm (Section 2.2.1) on five relevant examples including: a transient plug-flow reactor (PFR) system; generic convection-diffusion, convection-reaction, and diffusion-reaction models; and a multi-species atmospheric ozone reaction system modeled by coupled multivariate IBVPs. For comparison, in each example we apply both a *pure* IA (2.15) and the new *mixed* IA/AA (2.17) methods, and employ both continuous-time and discrete-time DI for bounding the discrete space system of ODE-IVPs. Generally, when the centered FD scheme is considered for discretization of the first derivatives, we can construct tighter bounds using the mixed IA/AA method. When forward or backward FD schemes are considered, the DI method may make the bounds the same tightness between IA and IA/AA methods. These cases are illustrated and discussed in the PFR example (Ex. 2.3.1). Additionally, a PDE-constrained optimization problem is formulated for the PFR model example and solved to global optimality to further demonstrate the validity and practicability of the proposed PDE bounding

methods for use with global dynamic optimization. We implemented the algorithm and all examples in the Julia programming language version 1.3.1 [53]. For the continuous-time DI methods, we used the CVODES numerical integrator with default settings from the Sundials v3.8.0 solver via the Sundials.jl package [54]. Each example was run on a personal workstation with an Intel Xeon E3-1270v5 4-core/8-thread CPU at 3.60GHz/4.00GHz (base/turbo) frequency and 32GB ECC RAM running Windows 10 Version 2004. For each example, the wall clock times for implementations of different algorithm routines were recorded in Table 2.1 and the volume of the corresponding state bounds are calculated and listed in Tables 2.2-2.3.

2.3.1 Transient Plug Flow Reactor

A single-species degradation reaction in an air-sparged PFR is introduced in this section. It is assumed that the degradation reaction is first-order under isothermal conditions. In addition, the diffusion of species inside the PFR is negligible. The PFR reaction model can be expressed as the following dimensionless PDE:

$$\frac{\partial \tilde{x}}{\partial t} = -\frac{\partial \tilde{x}}{\partial y} - Da\tilde{x}, \quad t \in [0, 1], \quad y \in [0, 1], \quad (2.27)$$

where \tilde{x} is the dimensionless species concentration, $Da = k\tau$ is the Damköhler number, $\tau = 10\text{h}$ is the mean residence time, and $k [\text{h}^{-1}]$ is the first-order reaction rate constant with uncertainty in the bounds $[0.1, 0.4]$. We assume the zero initial condition $\tilde{x}(y, 0) = 0$ for the model and the inlet concentration is fixed to $\tilde{x}(0, t) = 1$. We discretize the system using the

backward FD approximations which yields the following discrete system of ODEs:

$$\frac{\partial \mathbf{x}}{\partial t} = -\mathbf{x}^b - D\mathbf{a}\mathbf{x}, \quad (2.28)$$

where $\mathbf{x} \in \mathbb{R}^N$, $N = 20$ is the discrete state variables assigned at the discrete spatial grid points y_1, \dots, y_N , and $x_i^b = (x_i - x_{i-1})/\Delta y$ is the backward FD approximation. The proposed methods are applied to create the state bounds of x for (2.28). The step size of $h = 0.005$ is used when applying discrete-time DI. The bounding results are plotted in Figure 2.4. As expected, the trajectories of the species concentration with parametric uncertainty lie within the bounds over the space and time domains. It is observed that applying IA and AA will result in the same state bounds (the volumes are calculated and listed in Table 2.2). This is because the flattening operator β in Definition 2.1.6 effectively eliminates the differences in the calculated bounds for the discretized system by FD approximations calculated using IA and AA. As indicated in Figure 2.4, if DI is not applied and IA is directly applied to the discrete-time explicit Euler form, the resulting state bounds exhibit extreme overestimation. The state bounds calculated by directly applying the mixed IA/AA method without DI are much tighter than applying IA alone. However, when the DI algorithm is applied, pure IA and mixed IA/AA methods will eventually result in the same tight bounds. This indicates that the DI algorithm greatly reduces conservatism of the resulting state bounds and effectively reconciles the observed differences between IA and IA/AA approaches. The time costs for using discrete-time and continuous-time DI using IA and AA constructing state bounds are summarized in Table 2.1. Since the step size for discrete-time DI can be chosen manually, it is superior for this problem in terms of speed, without sacrificing accuracy

(the resulting volumes are very close between continuous-time DI and discrete-time DI). As expected, the time cost between execution of the PDE bounding algorithm using IA and IA/AA for spatial discretization are similar.

To explore the performance of the new bounding procedure for global dynamic optimization of PDEs, we consider the case of minimizing the compressed air required to meet a desired effluent quality. This problem is formulated as the following optimization problem:

$$\begin{aligned} \min_{p \in P} \quad & p \\ \text{s.t.} \quad & z_{K,exit} - \lambda \leq 0, \end{aligned} \tag{2.29}$$

where $p \in P = [0, 1]$ is a parameter related to the flowrate of air into the system and $z_{K,exit}$ is the effluent concentration. It is assumed that the reaction is limited by mass transport and can be manipulated by the compressed air flowrate through the Damköhler number as $Da = k\tau = (0.1 + 0.3p)\tau$. For this study, it is required that the effluent concentration $z_{K,exit}$ should be below $\lambda = 0.08$ as a critical performance or safety requirement.

Just as before, we set $N = 20$ for spatial discretization using the backward FD approximation with $h = 0.005$ ($K = 200$) using explicit Euler. The problem is solved using the EAGO v0.4.1 solver (EAGO.jl) [52] in the Julia programming language via the JuMP v0.20.1 modeling language (JuMP.jl). We set the absolute and relative convergence tolerances to be 10^{-3} . The optimization problem is defined in a discrete-time form based on (2.25) and custom bounding routines are utilized for the B&B algorithm. Specifically, we construct the bounds of $\mathbf{x}_k(p)$ by applying IA/AA for spatial FD approximations and discrete-time DI for integration. The upper-bounding and lower-bounding problems are established by Defini-

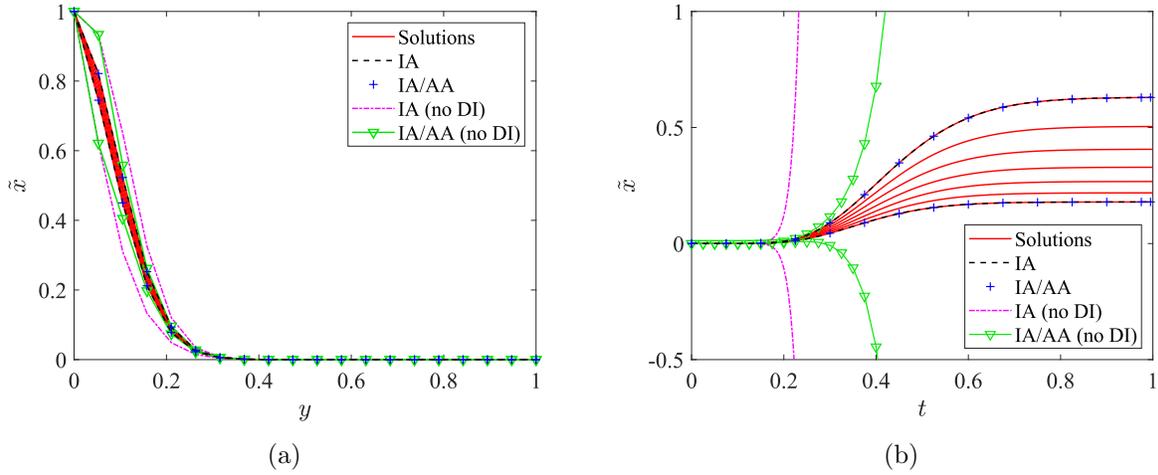


Figure 2.4: Results for the PFR system (Ex. 2.3.1) are plotted. (a) The spatial profiles of the species concentration $\tilde{x}(y, t, p)$ in the PFR at $t = 0.1$ are plotted for several values of p along with the state bounds derived from pure IA and mixed IA/AA. (b) The trajectories of the species concentration $\tilde{x}(y, t, p)$ in the PFR at the position $y = 0.5$ are plotted for several values of p along with the state bounds derived from pure IA and mixed IA/AA. $\tilde{x}(y, t, p)$ is approximated by corresponding numerical solutions of discrete-space states calculated by the explicit Euler method and state bounds are calculated by the discrete-time DI method.

tion 2.2.4 and Definition 2.2.6. It takes 38.7 s to achieve global optimality and the optimal solution is given by $p^* = 0.4523$. This represents a significant advantage over the implicit approach of [55], who solved this problem in 382 s using a second-order implicit integration method.

2.3.2 Convection-Diffusion System

Consider a scalar convection-diffusion system as an IBVP with two parameters:

$$\frac{\partial \tilde{x}}{\partial t} = p_1 \frac{\partial^2 \tilde{x}}{\partial y^2} - p_2 \frac{\partial \tilde{x}}{\partial y}, \quad t \in [0, 1], \quad y \in [0, 1], \quad (2.30)$$

subject to the initial condition $\tilde{x}(y, 0) = 2$, as well as the boundary conditions $\tilde{x}(0, t) = 0$, $\tilde{x}(1, t) = 1$. Parametric uncertainty is specified as: $\mathbf{p} \in P = [0.015, 0.045] \times [0.02, 0.06]$.

The spatial domain of this PDE is discretized into $N = 100$ nodes. Specifically, we applied centered, forward, and backward FD approximations for spatial discretization of the convective term to fully demonstrate the bounding results under different discretization approaches. Both pure IA and mixed IA/AA methods are used for calculating state bounds for comparison. Then, both continuous-time DI and discrete-time DI (with $h = 0.001$) are applied to integrate the model and construct bounds on the solutions of this system with parametric uncertainty. The calculated state bounds are illustrated in Figure 2.5 and the corresponding volumes are indicated in Table 2.2. It is indicated that the bounds obtained by the new mixed IA/AA method are much tighter than the pure IA approach when centered FD approximations are used (see Figure 2.5(b)). Moreover, it is observed that the state bounds derived from using forward and backward FD approximations with pure IA and mixed IA/AA are identical. Using DI in the bounding procedure will reduce the differences between applying IA and IA/AA while using forward and backward FD approximations for the convection term. Furthermore, it is observed that the backward FD approximations provide the tightest bounds for this model. As summarized in Table 2.1, the costs for calculating bounds using IA and IA/AA are similar. Thus, it is recommended to utilize the mixed IA/AA approach to achieve at least equivalent state bounds, and possibly better bounds, without increasing time cost.

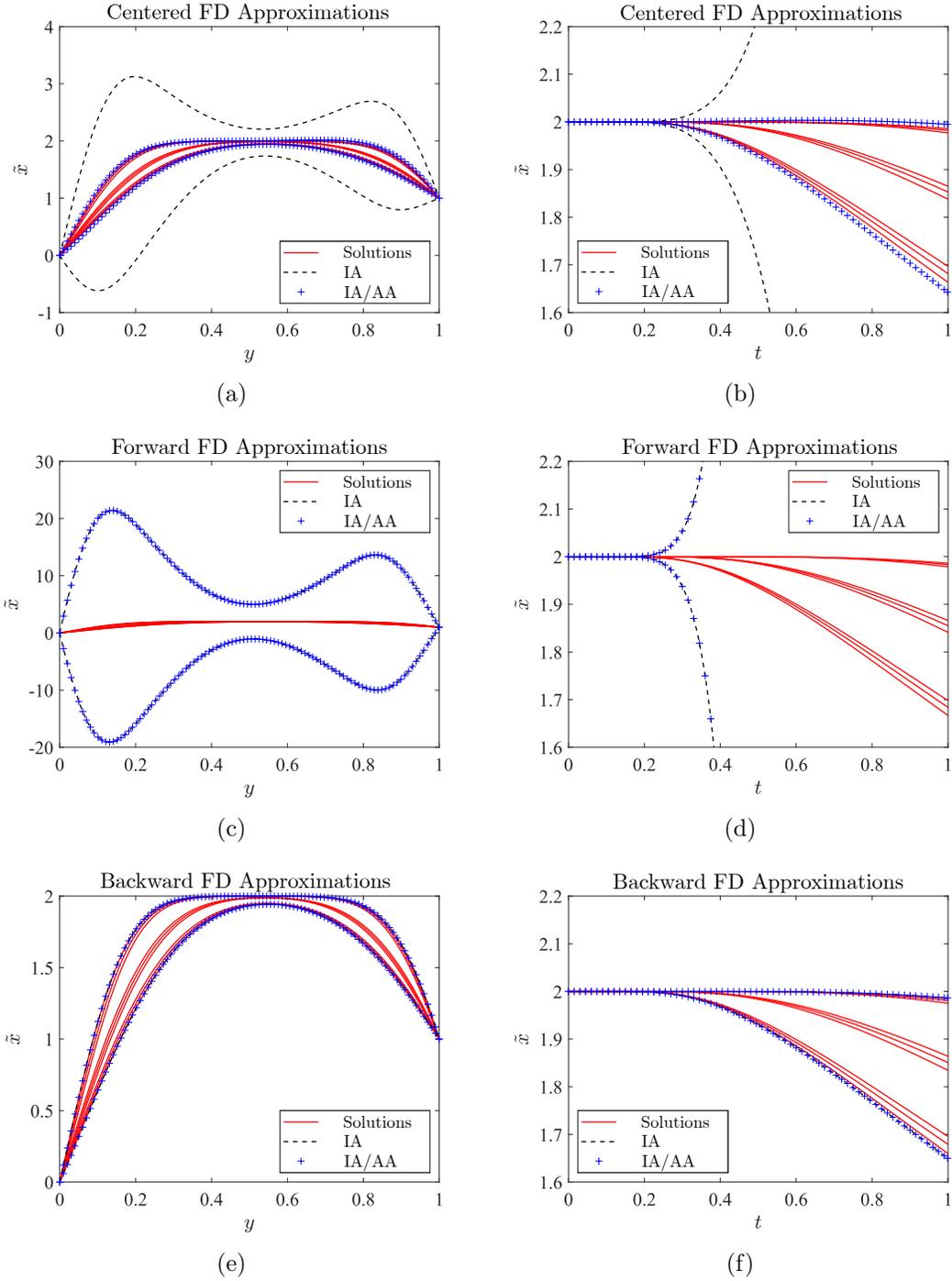


Figure 2.5: Results for the convection-diffusion system (Ex. 2.3.2) are plotted. (a,c,e) The spatial profiles of numerical approximations of $\tilde{x}(y, t, \mathbf{p})$ for several values of \mathbf{p} are plotted at $t = 0.5$, along with their bounds obtained using pure IA and mixed IA/AA methods. (b,d,f) The trajectories of numerical approximations of $\tilde{x}(y, t, \mathbf{p})$ for several values of \mathbf{p} are plotted at $y = 0.5$, along with their bounds obtained using pure IA and mixed IA/AA methods. (a,b) Centered FD approximations were used for spatial discretization of the convection term. (c,d) Forward FD approximations were used for spatial discretization of the convective term. (e,f) Backward FD approximations were used for spatial discretization of the convective term.

2.3.3 Convection-Reaction System

Consider a convection-reaction system with two parameters $\mathbf{p} = (p_1, p_2)$ representing convection and reaction coefficients, respectively:

$$\frac{\partial \tilde{x}}{\partial t} = -p_1 \frac{\partial \tilde{x}}{\partial y} - p_2 \tilde{x}, \quad t \in [0, 1], \quad y \in [0, 1], \quad (2.31)$$

where the initial condition is $\tilde{x}(y, 0) = 1$, and the boundary conditions are defined as: $\tilde{x}(0, t) = 0$, $\frac{d\tilde{x}}{dy}|_{y=1} = 0$. The uncertain parameters are $\mathbf{p} \in P = [0.1, 0.3] \times [0.2, 0.6]$. The proposed methods are implemented to calculate state bounds on the solutions of this system. Specifically, the space domain of (2.31) is discretized into $N = 20$ nodes. The convective term is approximated using centered FD and backward FD approximations for comparison. The forward FD approximation was not considered as the resulting discretized system is unstable. The step size was chosen to be $h = 0.01$ when applying discrete-time DI. The trajectories of the solutions and the state bounds are plotted over the space domain and time domain as shown in Figure 2.6. The volumes of state bounds are also reported in Table 2.2. The bounds obtained by the IA/AA approach are tighter than by the pure IA approach in the case of using centered FD approximations. The time costs for constructing the bounds using each algorithm are reported in Table 2.1.

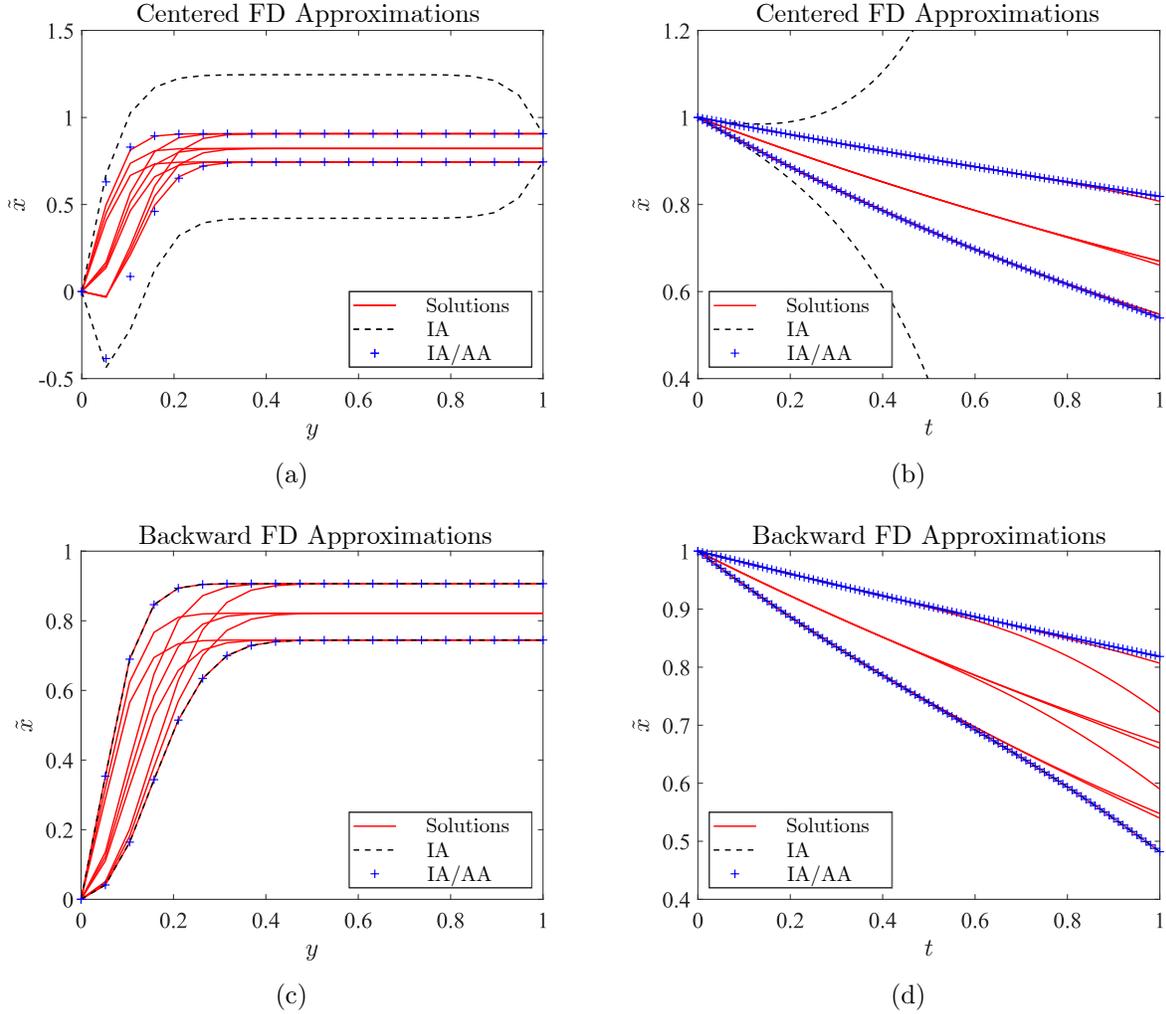


Figure 2.6: Results for the convection-reaction system (Ex. 2.3.3) are plotted: (a,c) The spatial profiles of numerical approximations of $\tilde{x}(y, t, \mathbf{p})$ for several values of \mathbf{p} are plotted at $t = 0.5$, along with their bounds obtained using pure IA and mixed IA/AA methods. (b,d) The trajectories of numerical approximations of $\tilde{x}(y, t, \mathbf{p})$ for several values of \mathbf{p} are plotted at $y = 0.5$, along with their bounds obtained using pure IA and mixed IA/AA methods. (a,b) Centered FD approximations were used for spatial discretization of the convection term. (c,d) Backward FD approximations were used for spatial discretization of the convective term. $\tilde{x}(y, t, p)$ is approximated by corresponding numerical solutions of discrete-space states calculated by the explicit Euler method and state bounds are calculated by the discrete-time DI method.

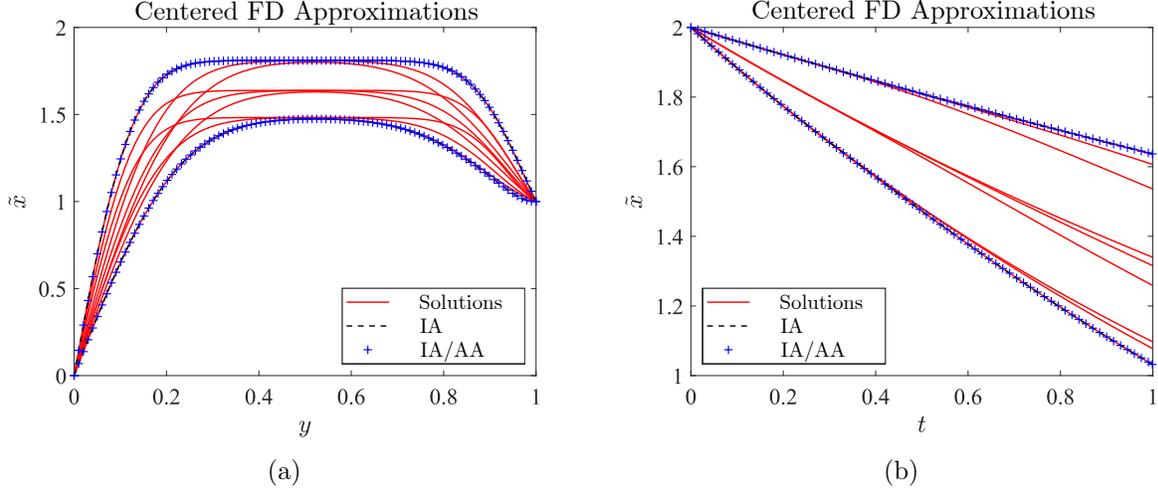


Figure 2.7: Results for diffusion-reaction system (Ex. 2.3.4) are plotted: (a) The spatial profiles of numerical approximations of $\tilde{x}(y, t, \mathbf{p})$ for several values of \mathbf{p} are plotted at $t = 0.5$, along with their bounds obtained using pure IA and mixed IA/AA methods. (b) The profiles of numerical approximations of $\tilde{x}(y, t, \mathbf{p})$ for several values of \mathbf{p} are plotted at $y = 0.5$, along with their bounds obtained using pure IA and mixed IA/AA methods. Centered FD approximations were used for spatial discretization. $\tilde{x}(y, t, p)$ is approximated by corresponding numerical solutions of discrete-space states calculated by the explicit Euler method and state bounds are calculated by the discrete-time DI method.

2.3.4 Diffusion-Reaction System

Consider a diffusion-reaction system with two parameters $\mathbf{p} = (p_1, p_2)$ representing diffusion and reaction coefficients, respectively:

$$\frac{\partial \tilde{x}}{\partial t} = p_1 \frac{\partial^2 \tilde{x}}{\partial y^2} - p_2 x, \quad t \in [0, 1], \quad y \in [0, 1]. \quad (2.32)$$

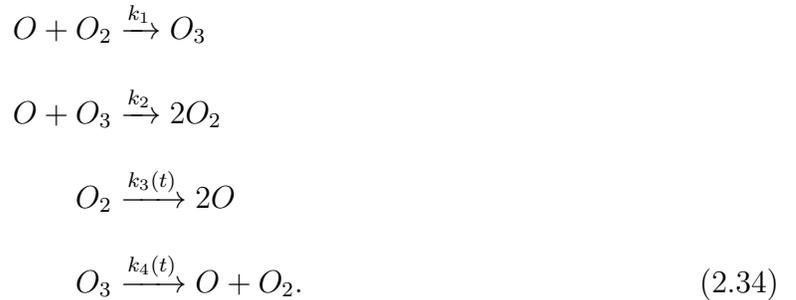
The initial condition is $\tilde{x}(y, 0) = 2$, and boundary conditions are given by $\tilde{x}(0, t) = 0$, $\tilde{x}(1, t) = 1$. Uncertain parameters are defined as $\mathbf{p} \in P = [0.01, 0.03] \times [0.2, 0.6]$. We approximate the diffusive term in (2.32) using centered FD approximation and integrate over the time horizon using discrete-time and continuous-time DI approaches. The resulting state bounds are illustrated in the Figure 2.7 and their volumes are given in Table 2.2.

2.3.5 Coupled IBVPs

In this section, we study a model for atmospheric ozone reactions introduced by [56]. It is assumed that the atmosphere of the earth is a closed system at a constant temperature and volume. Three species: free oxygen O , molecular oxygen O_2 , and ozone O_3 are involved in the atmosphere system. In addition, we assume that the concentration of these three species can vary with the altitude $y \in [0, 1]$ and the movement of species is caused by wind. Furthermore, the species diffusion interactions are also considered in the system. The transport phenomena in this ozone atmosphere system can be represented by the following coupled convection-diffusion-reaction IBVPs:

$$\begin{aligned}
 \frac{\partial c_O}{\partial t} + u_1 \frac{\partial c_O}{\partial y} &= D_1 \frac{\partial^2 c_O}{\partial y^2} + r_{c_O}, \\
 \frac{\partial c_{O_2}}{\partial t} + u_2 \frac{\partial c_{O_2}}{\partial y} &= D_2 \frac{\partial^2 c_{O_2}}{\partial y^2} + r_{c_{O_2}}, \\
 \frac{\partial c_{O_3}}{\partial t} + u_3 \frac{\partial c_{O_3}}{\partial y} &= D_3 \frac{\partial^2 c_{O_3}}{\partial y^2} + r_{c_{O_3}},
 \end{aligned} \tag{2.33}$$

where c_O , c_{O_2} , and c_{O_3} denote the corresponding species concentration, u_1 , u_2 , and u_3 are the corresponding convection coefficients, D_1 , D_2 , and D_3 are the corresponding diffusion coefficients. Moreover, the reaction mechanism of ozone in the atmosphere is simplified as:



In this reaction mechanism, k_1 , k_2 , $k_3(t)$, and $k_4(t)$ are the rate constants for each reaction. Since the last two reactions describe the photodissociation process of O_2 and O_3 associated with the effect of sunlight, $k_3(t)$ and $k_4(t)$ will change periodically with time. Based on the above reaction mechanism (2.34), we can derive the reaction rates of each species:

$$\begin{aligned} r_{cO} &= -k_1 c_O c_{O_2} - k_2 c_O c_{O_3} + 2k_3(t) c_{O_2} + k_4(t) c_{O_3}, \\ r_{c_{O_2}} &= -k_1 c_O c_{O_2} + 2k_2 c_O c_{O_3} - k_3(t) c_{O_2} + k_4(t) c_{O_3}, \\ r_{c_{O_3}} &= k_1 c_O c_{O_2} - k_2 c_O c_{O_3} - k_4(t) c_{O_3}. \end{aligned}$$

Here, the rate constants k_1 and k_2 are constants: $k_1 = 1.63 \times 10^{-16}$, $k_2 = 4.66 \times 10^{-16}$. The other two rate constants $k_3(t)$ and $k_4(t)$ follow a two-day periodical cycle as:

$$k_i(t) = \begin{cases} \exp(-c_i / \sin(\omega t)) & \text{if } \sin(\omega t) \geq 0 \\ 0 & \text{if } \sin(\omega t) < 0, \quad i = 3, 4, \end{cases}$$

where $c_3 = 22.62$, $c_4 = 7.601$, and $\omega = \pi/43200 \text{ s}^{-1}$. The rate constants k_3 and k_4 will increase quickly at the beginning ($t = 0$), rise to a peak at noon ($t = 6 \times 3600 \text{ s}$), then drop

to zero at sunset ($t = 12 \times 3600$ s). The initial conditions of this ozone model are given by

$$c_O(y, 0) = \begin{cases} 10^6 & \text{if } 0.3 \leq y \leq 0.6 \\ 0 & \text{otherwise,} \end{cases}$$

$$c_{O_2}(y, 0) = \begin{cases} 3.7 \times 10^{16} & \text{if } 0.3 \leq y \leq 0.6 \\ 0 & \text{otherwise,} \end{cases}$$

$$c_{O_3}(y, 0) = \begin{cases} 10^{12} & \text{if } 0.3 \leq y \leq 0.6 \\ 0 & \text{otherwise.} \end{cases}$$

At the bottom layer of the atmosphere, the boundary conditions are given by $c_O(0, t) = 10^6$, $c_{O_2}(0, t) = 3.7 \times 10^{16}$, $c_{O_3}(0, t) = 10^6$.

In this model, we make the assumption that the convection and diffusion coefficients of the species will remain the same due to the uniformity of the atmosphere: $u = u_1 = u_2 = u_3$ and $D = D_1 = D_2 = D_3$. These two coefficients are considered uncertain with bounds $u \in [4e-3, 6e-3]$ and $D \in [4e-3, 6e-3]$. To solve the model, we first discretize the space domain into $N = 50$ nodes y_1, \dots, y_N . The corresponding numerical values of $c_O(y, t, \mathbf{p})$, $c_{O_2}(y, t, \mathbf{p})$, and $c_{O_3}(y, t, \mathbf{p})$ are assigned at each node y_i ($i = 1, \dots, N$) and are represented by $x_{1,i}(t, \mathbf{p})$, $x_{2,i}(t, \mathbf{p})$, and $x_{3,i}(t, \mathbf{p})$ to be determined. We interlace these discrete variables to obtain $\mathbf{z}(t, \mathbf{p}) = (x_{1,1}(t, \mathbf{p}), x_{2,1}(t, \mathbf{p}), x_{3,1}(t, \mathbf{p}), \dots, x_{1,N}(t, \mathbf{p}), x_{2,N}(t, \mathbf{p}), x_{3,N}(t, \mathbf{p}))$. In other words, we have $x_{1,i} = z_{3i-2}$, $x_{2,i} = z_{3i-1}$ and $x_{3,i} = z_{3i}$ for $i = 1, \dots, N$. In terms of the global index $l = 3i - (3 - m)$ with $m = 1, 2, 3$ and $i = 1, \dots, N$, we have

$$z_l = x_{m,i}, \quad l = 1, \dots, 3N.$$

Next, centered FD approximations are used to approximate the corresponding spatial derivative terms and the discrete model is rewritten in terms of $z_l(t, \mathbf{p})$, that can be represented by the vector form:

$$\dot{\mathbf{z}}(t, \mathbf{p}) = \mathbf{f}(t, \mathbf{z}(t, \mathbf{p}), \mathbf{z}^c(t, \mathbf{p}), \mathbf{z}^{c2}(t, \mathbf{p}), \mathbf{p}), \quad \mathbf{z}(t_0, \mathbf{p}) = \mathbf{z}_0(\mathbf{p}), \quad \forall t \in I.$$

The system of auxiliary ODE-IVPs is integrated using discrete-time DI ($h = 0.016$) and continuous-time DI over the time domain $I = [0, 3.2]$. Both pure IA and mixed IA/AA are applied to calculate state bounds. As shown in the Figure 2.9, all trajectories for species concentration lie within the bounds, as expected. Furthermore, it is obvious that the state bounds for each species constructed by using IA/AA are tighter than using IA alone, which is also confirmed by the volume calculations reported in Table 2.3. In addition, as indicated by the time costs for each method, listed in Table 2.1, applying IA/AA versus IA results in nearly identical computational expense.

To further illustrate the performance and practicability of the proposed bounding approach, a parameter estimation problem is formulated for this model and solved to guaranteed global optimality. A dataset of ozone concentration is given at specific spatial positions in the atmosphere and we seek to determine the optimal values of the uncertain parameters to achieve the best-possible fit of the model to the data. We use the discrete-time scheme for integration ($K = 200$), and the objective function is formulated as the sum of squared error between the data and model at specific spatial nodes ($i = 5j$ with $j = 1, \dots, 10$) with

$N = 50$, over the entire time horizon:

$$\begin{aligned} & \min_{\mathbf{p} \in P} \sum_{j=1}^{10} \sum_{k=1}^K (x_{3,5j,k}(\mathbf{p}) - c_{O_3,i,k}^{\text{data}})^2 \\ & \text{s.t. } \mathbf{z}_0(\mathbf{p}) = \mathbf{x}(0, \mathbf{p}) \\ & \mathbf{z}_{k+1}(\mathbf{p}) = \mathbf{z}_k(\mathbf{p}) + h\mathbf{f}(t_k, \mathbf{z}_k(\mathbf{p}), \mathbf{z}_k^c(\mathbf{p}), \mathbf{z}_k^{c2}(\mathbf{p}), \mathbf{p}), \quad \forall k = 0, \dots, K \end{aligned} \tag{2.35}$$

where $\mathbf{p} = (u, D) \in P = [4\text{e-}3, 6\text{e-}3] \times [4\text{e-}3, 6\text{e-}3]$ are the uncertain parameters, and $c_{O_3,i,k}^{\text{data}}$ is the ozone concentration data measured at spatial node i and time node k . For the purposes of this study, the data was generated by a random number generator with Gaussian noise.

For the purpose of comparison, both pure IA and mixed IA/AA algorithms are used for bounding routines. We solve this problem using the EAGO v0.4.1 solver (EAGO.jl) [52] via the JuMP v0.20.1 modeling language (JuMP.jl). The relative global convergence tolerance is set as 5×10^{-2} . An optimal solution is found at $\mathbf{p}^* = (5.12\text{e-}3, 4.48\text{e-}3)$. It takes 16.4 h and 30 h to reach global optimality using mixed IA/AA and pure IA, respectively. The performance of the algorithms is illustrated as a convergence plot in Figure 2.8. As illustrated in Figure 2.8 with the orange profiles of mixed IA/AA always higher than the blue profiles of pure IA, the tighter bounds constructed by using the mixed IA/AA reduces the total time cost for achieving convergence of the global optimization algorithm.

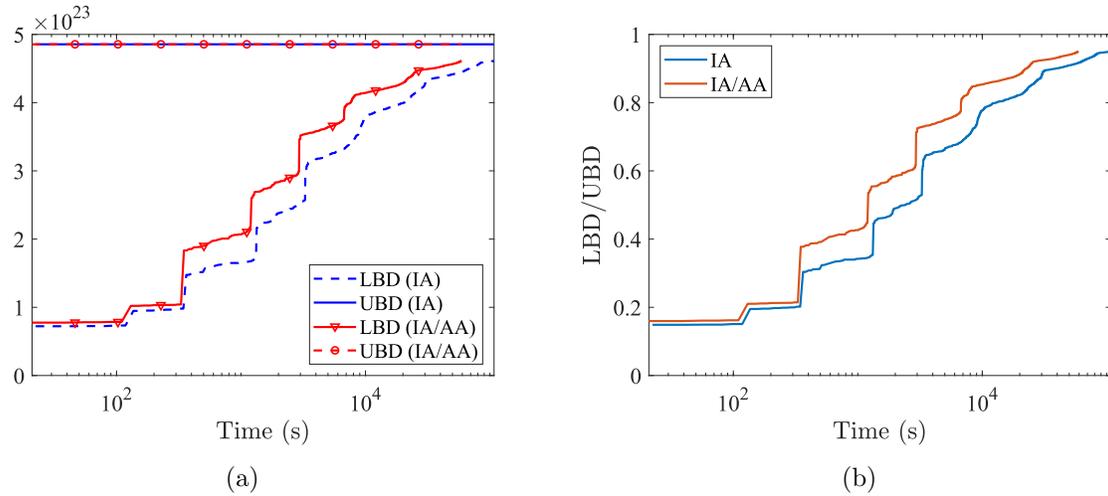


Figure 2.8: The convergence profiles are plotted for the global optimization algorithm using pure IA and mixed IA/AA for Ex. 2.3.5. (a) The LBD and UBD are plotted separately to resolve the improvements in each bound as the algorithm iterates. The UBD values are relatively constant, meaning that the upper-bounding routine located a very good solution early on. (b) The ratio of LBD/UBD is plotted to illustrate relative convergence of the bounds as the algorithm iterates. As can be seen, the mixed IA/AA performs better than the pure IA bounding method by providing better lower bounds, resulting in much faster convergence of the B&B algorithm.

Table 2.1: CPU time for constructing state bounds of the PFR, convection-diffusion, convection-reaction, diffusion-reaction, and ozone reaction models are reported in this table. The forward, backward, and centered FD approximations refer to the schemes applied to the convection term in these models.

PFR Model			
Time (ms)		Discrete-Time DI	Continuous-Time DI
Centered FD approximations	IA	59.332	158.823
	IA/AA	62.024	162.567
Convection-Diffusion Model			
Time (s)		Discrete-Time DI	Continuous-Time DI
Centered FD approximations	IA	17.883	72.112
	IA/AA	17.592	80.821
Forward FD approximations	IA	17.870	70.490
	IA/AA	18.458	68.100
Backward FD approximations	IA	18.171	82.415
	IA/AA	18.476	79.773
Convection-Reaction Model			
Time (ms)		Discrete-Time DI	Continuous-Time DI
Centered FD approximations	IA	9.593	58.849
	IA/AA	9.252	40.116
Backward FD approximations	IA	9.228	22.920
	IA/AA	9.276	23.042
Diffusion-Reaction Model			
Time (ms)		Discrete-Time DI	Continuous-Time DI
Centered FD approximations	IA	59.332	158.823
	IA/AA	62.024	162.567
Ozone Reaction Model			
Time (s)		Discrete-Time DI	Continuous-Time DI
Centered FD approximations	IA	16.324	317.329
	IA/AA	15.992	322.567

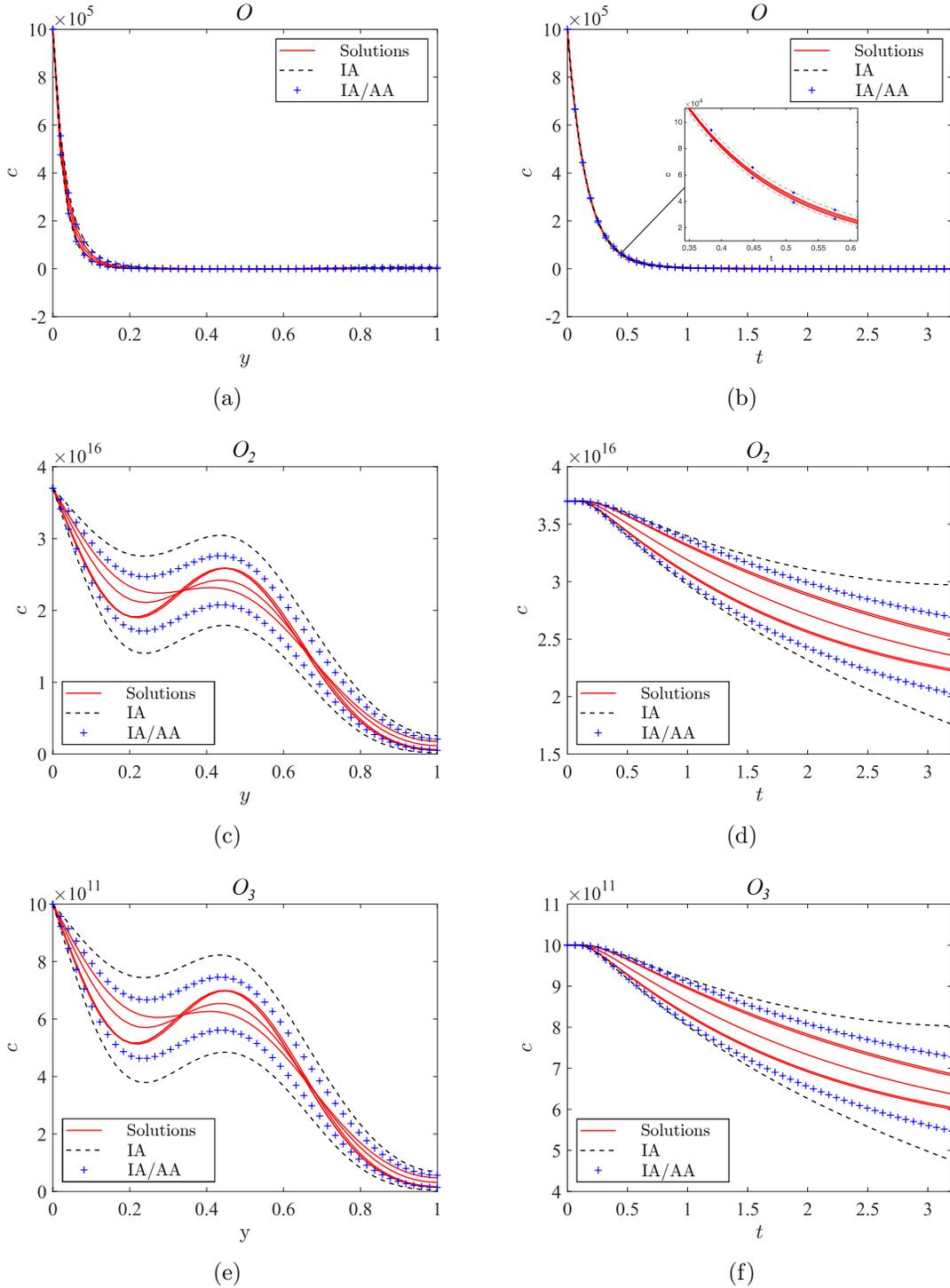


Figure 2.9: Concentration profiles for the system of coupled IBVPs (Ex. 2.3.5) are plotted. (a,c,e) Concentration profiles for several values of (u, D) over the entire altitude of the modeled atmosphere for species O (a), O_2 (c), O_3 (e) are plotted at $t = 3.2s$, along with their bounds obtained using pure IA and mixed IA/AA. (b,d,f) Concentration profiles for several values of (u, D) for species O (b), O_2 (d), O_3 (f) are plotted at $y = 0.5$ through time horizon, along with their bounds obtained using pure IA and mixed IA/AA.

Table 2.2: The volumes of state bounds for the solutions of PFR, convection-diffusion, convection-reaction and diffusion-reaction models are presented in this table. The forward, backward and centered FD approximations especially refer to the schemes applied to the convection term in these models.

PFR Model			
Volume		Discrete-Time DI	Continuous-Time DI
Backward FD approximations	IA	0.1740	0.1726
	IA/AA	0.1740	0.1726
Convection-Diffusion Model			
Volume		Discrete-Time DI	Continuous-Time DI
Centered FD approximations	IA	4.6177	4.6538
	IA/AA	0.2513	0.2542
Forward FD approximations	IA	516.5	541.8
	IA/AA	516.5	543.1
Backward FD approximations	IA	0.2319	0.2317
	IA/AA	0.2319	0.2347
Convection-Reaction Model			
Volume		Discrete-Time DI	Continuous-Time DI
Centered FD approximations	IA	2.1868	2.3318
	IA/AA	0.2753	0.2724
Backward FD approximations	IA	0.2090	0.2067
	IA/AA	0.2090	0.2067
Diffusion-Reaction Model			
Volume		Discrete-Time DI	Continuous-Time DI
Centered FD approximations	IA	0.3686	0.3840
	IA/AA	0.3686	0.3840

Table 2.3: The volumes of state bounds for the solutions of each species in the ozone reaction model solutions are presented in this table. Centered FD approximations are used for the convection terms in the model.

Volume	Discrete-Time DI			Continuous-Time DI		
	Species	O	O_2	O_3	O	O_2
IA	20692	4.9558E15	1.3394E11	20701	4.9389E15	1.3348E11
IA/AA	18073	3.6062E15	9.7464E10	18067	3.5919E15	9.7079E10

2.4 Conclusion

We presented a novel algorithm for constructing rigorous bounds enclosing the solutions of unsteady PDEs was presented. This approach has broad applicability across process systems engineering applications for uncertainty quantification, robust simulation, and deterministic global dynamic optimization. In this algorithm, a mixed IA/AA set-valued mapping approach is used to reduce the overestimation introduced by the dependency problem of standard IA, when applied to FD approximations for spatial discretization. A modified DI method is employed with numerical integration to propagate bounds through time. In addition, a discrete-time DI approach was also extended to this class of problems. The methods were demonstrated on five examples, including variations of common convection-diffusion-reaction systems and coupled IBVPs with multiple state variables, to demonstrate the performance of the generated state bounds and practicability for applications in deterministic global optimization with unsteady PDE constraints. The bounds constructed by this method are less conservative and do not add additional computational cost over conventional IA approaches. Therefore, this approach is expected to enable the solution of a broader class of global dynamic optimization problems.

In chapter 3 - 5, the methodologies for modeling spatiotemporal systems and implementing optimization-based system designs are elucidated with respect to different real-world applications. The proposed bounding PDE algorithm in this chapter is utilized in chapter 4 for solving deterministic global optimization problems.

Chapter 3

Precise Control of Water and Wastewater Treatment Systems with Non-Ideal Heterogeneous Mixing Models and High-Fidelity Sensing

In this chapter, a traditional engineering application for controlling water and wastewater treatment systems is presented. The transport systems are modeled using a small number of ODE-IVPs to capture transport phenomena at different spatial positions. Furthermore, the developed non-ideal heterogeneous mixing models are incorporated within advanced closed-loop control strategies utilizing high-resolution sensing to maximize the resiliency and minimize the energy consumption of water treatment processes. The proposed non-ideal heterogeneous mixing models capture continuity (heat and mass conservation), yet are

extremely simple with few parameters, so they lend themselves to fast online prediction (with extrapolation capabilities) and regular recalibration. The models are deployed within standard model predictive control and two economic model predictive control strategies to demonstrate model-based decision-making for disturbance rejection and optimal operation of aeration in a continuous-flow nitrification system utilizing high-resolution sensor data from several spatial positions.

3.1 Introduction

The urgent demand for enhanced water quality, high resilience, high treatment efficiency, low costs, and environmentally-friendly operations have promoted the development of model-based decision-making and control strategies in water and wastewater treatment plants (WTPs and WWTPs) [57, 58, 59, 60]. The prerequisite to an effective control system is the ability to measure important water quality information (e.g., conductivity, pH, temperature, etc.) by high-fidelity sensing technologies, that can then be used to build reliable predictive models as well as for real-time state measurement. Though classical, purely physics-based unit operations models (e.g., activated sludge model (ASM) [61, 62, 63, 64], hydraulic model [65], sedimentation tank model [57, 66]) have been used extensively in the past, they have their limitations. For example, building models is time-consuming and some necessary parameters (e.g., cell growth rate, biomass yield) in these models are unmeasurable in WTP/WWTPs in real time. Such variables are normally determined by offline analyses in the laboratory, causing serious time delays [57, 67, 68] in operator response. As a consequence, transients and spatial heterogeneity in process units go undetected, hampering

efficiency improvement efforts due to an incomplete understanding of the transport processes of water quality information [69, 70, 71].

In situ data-driven approaches could overcome the existing modeling drawbacks by enabling identification of the “black box” systems (e.g., aeration tanks) whose underlying mechanisms are otherwise unknown to operators, using data from the practical process under real operating conditions [72, 73]. However, traditional data-driven methods in WTPs/WWTPs have two obstacles. First, some data-driven models are developed based on markedly large datasets collected and processed over long periods of time (e.g., greater than ten years in WTP/WWTPs). Given that only a small amount of data points have critical influence on the system, using excess data can dilute important signals, making it unsuitable for identifying transient situations (e.g., fast chemical transport processes) [69]. The second limitation is that data-driven approaches are based purely on regression without considering physical principles (e.g., dynamic neural network prediction of flow rate [74]) in WTP/WWTPs, and are therefore limited to interpolated prediction. These data-driven identification techniques may perform well for complicated systems since empirical correlations and first-principles may not accurately capture the behavior of the system due to limited information [74, 75]. As a result, these models are unable to capture important fundamental mechanisms, and thus are generally less functional for further model-based control applications where extrapolative prediction is needed. Fortunately, well-understood phenomena, like heat and mass transport, can be readily and accurately modeled using first-principles.

This study explores an innovative approach to obtain high-resolution spatiotemporal data from real-time in-situ water quality sensors, build and validate accurate models for the measured water quality parameters, and deploy those models for precise model-based

control of WTP/WWTPs. Simple non-ideal heterogeneous mixing models are developed to simulate and predict heterogeneous mass transport in WTP/WWTPs. In order to achieve visualization of the whole system, the transport characteristics of three important attributes of water quality (conductivity [76], pH [77], and temperature [78]) are elucidated with only a small volume of data collected within short periods (e.g., 2–10 min). These models are then utilized within a technology platform for the precise control of WTP/WWTPs using several different control strategies and architectures, including conventional and economic modelpredictive control for improving energy and chemical-use efficiency in WTP/WWTPs.

One novel contribution of this study is that deterministic global (dynamic) optimization [17, 79, 80, 81, 82] is employed for better understanding the heterogeneous mixing phenomena via rigorous parameter estimation. Finding a global optimum is far more difficult than finding an arbitrary local solution; yet, a mismatch between the model and the data cannot be declared unless the best-possible fit is verified. Therefore, this approach provides additional benefits for preventing erroneously invalidating proposed mechanisms in cases where local algorithms return poor, suboptimal fits. A comparison between local and global optimal solutions within this context is shown in Figure 3.1, illustrating the conductivity profile in different positions of a tank. As shown, the suboptimal solution obtained by a local optimization algorithm [83, 84] differs significantly from a global solution. The advantages of the developed non-ideal heterogeneous mixing models are demonstrated through their straightforward, interpretable mathematical expressions able to achieve a better fit as compared to both pure data-driven symbolic regression machine learning approaches using Eureqa® (Version 1.24.0 (build 9367), DataRobot) [85] and a pure computational fluid dynamics (CFD) approach. Furthermore, the predictive capabilities are also validated by

additional experimental datasets of chemical species (e.g., KCl, MgSO₄, NaOH), verifying that provided models could work for model predictive control (MPC) in a broad spectrum of operating scenarios.

This chapter is organized as follows. In Section 2, the Materials and Methods used in this study are presented. Summarily, the experimental methods are discussed as well as the model development, rigorous parameter estimation, and the control architectures for precise control of WWTPs. Section 3 contains the Results and Discussion, whereby the performance of the non-ideal heterogeneous mixing models is demonstrated, as are the control architectures for precise control of WWTPs. Conclusions follow in Section 4.

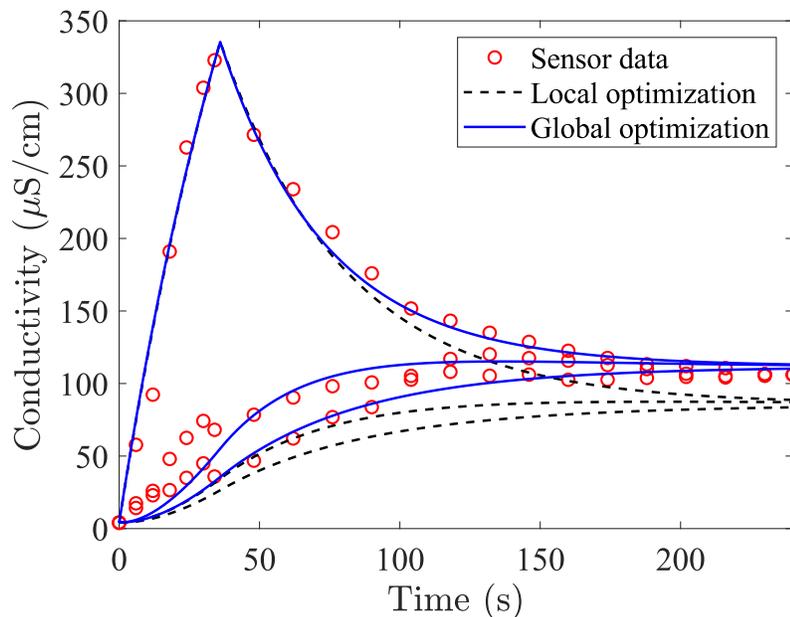


Figure 3.1: Comparison of the conductivity profiles using local optimization (black dashed lines) and global optimization (blue solid lines) are presented in this figure. The conductivity model is a dynamic system under a pulse response. The three trajectories represent profiles in three different positions of the tank (high, middle, and low zones).

3.2 Materials and Methods

Figure 3.2 illustrates at a high level, the methods used in this study. In Figure 3.2(a), we illustrate the experimental methods for data acquisition as discussed in Section 3.2.1. Figure 3.2(b) illustrates the overall schematics of calibration (detailed in Section 3.2.3) and advanced control strategies of WTP/WWTPs developed in this work (detailed in Section 3.2.4). Figure 3.2(c) illustrates a continuous flow nitrification reactor representative of a commercial WWTP operation (detailed in Section 2.4).

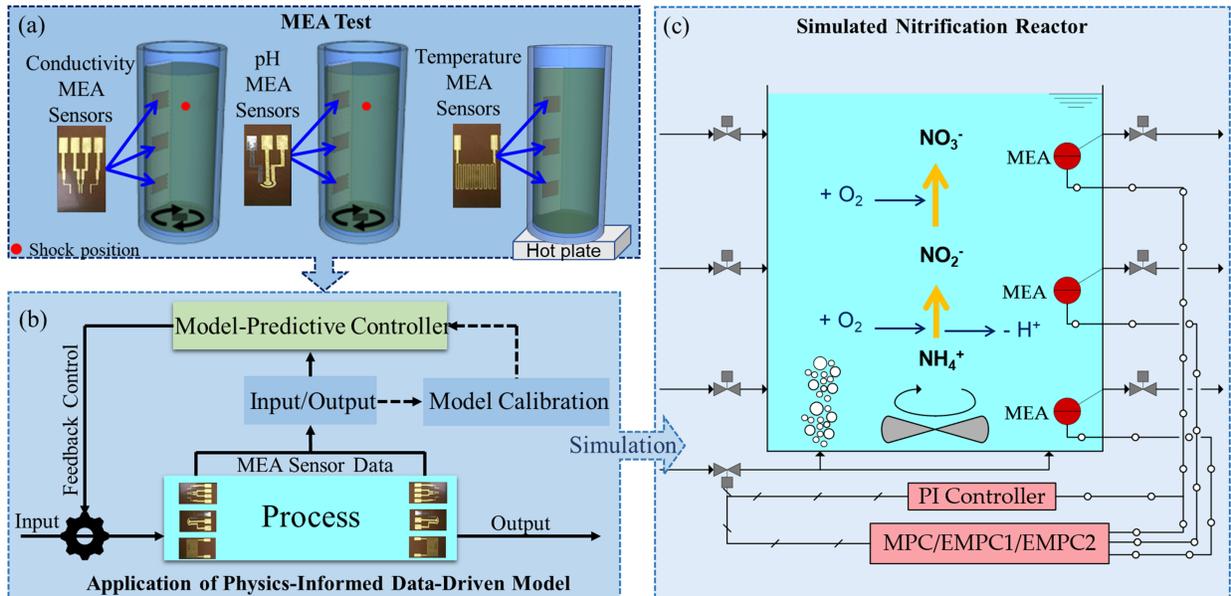


Figure 3.2: (a) The experimental setup is illustrated for profiling tests along the batch reactor depth under the conductivity shock, pH shock, and temperature shock measured by the MEA sensors. (b) A schematic of the application of non-ideal heterogeneous mixing models in an online model predictive control (MPC) system is illustrated. (c) The simulated continuous stirred tank nitrification reactor is illustrated with inflows and outflows located at high, middle, and low zones, and a PI controller or MPC for controlling aeration.

3.2.1 Non-Ideal Mixing Profiling Using Milli-Electrode Array (MEA)

Sensors

All three types of MEA sensors (each size: 2 cm \times 0.5 cm) targeting three water quality parameters (conductivity, pH, and temperature) were precisely printed on Kapton polyimide film (FPC, thickness: 127 μ m, American Durafilm) by a Dimatix Materials Printer (ModelDMP-2800, FUJIFILM Dimatix, Inc.) as previously reported [86, 87]. Three assemblies of MEA sensors of each were deployed at three locations (high position: 40 mm below the water surface; middle position: 85 mm below the water surface; low position: 130 mm below the water surface) of a batch stirred reactor (diameter: 62 mm, height: 180 mm) (Figure 3.2(a)) to accurately profile the whole reactor.

The sensor readings were recorded by a multi-channel potentiostat (1040C 8-channel potentiostat, CH Instruments, Inc.) every 2 s. It took 10 min to reach steady-state operation with continuous stirring with a rotation rate of 50 RPM. Then, different species (chemicals purchased from Fisher Science, Co.) were individually injected into the reactor to simulate transient shocks. Specifically, for conductivity shocks, 200 μ L (100 g/L) sodium chloride was injected into the reactor on the three locations (high, middle, and low) respectively. For pH shock, 200 μ L (1 M) potassium hydroxide solution mixed in 2 mg/L sodium chloride solution (pH = 14) was injected into the reactor (initial pH: 7.22) on the three locations (high, middle, and low) in turn. For temperature shock, the water solution in the reactor (initial temperature: 18.5 ± 0.12 °C, room temperature) was placed on a heating plate that was heated to 200 °C within 30 s and then shut off. All shock tests were conducted with three repeated experiments and the average values were calculated to compensate for the

uncertainty in experimental procedure. Validation tests were carried out under the same conditions, except the shock substance was changed to 200 μL KCl solution (100 g/L) and 200 μL MgSO_4 solution (100 g/L) for conductivity model validation, and to 200 μL NaOH solution (1 M) for pH shock validation. These shock substances were only injected into the high zone of the reactor.

3.2.2 Non-Ideal Mixing Model Development

Non-ideal heterogeneous mixing models were developed to simulate conductivity, pH, and heat transport processes inside the reactor. Specifically, to capture the spatiotemporal heterogeneity of the conductivity and pH profiles, the tank reactor was modeled using multiple regions with interchange [88]. The models were established based on heat and mass conservation that characterize the observed physical phenomena. These models were then used within a continuous flow nitrification reactor model for precise control.

Conductivity Mixing Model

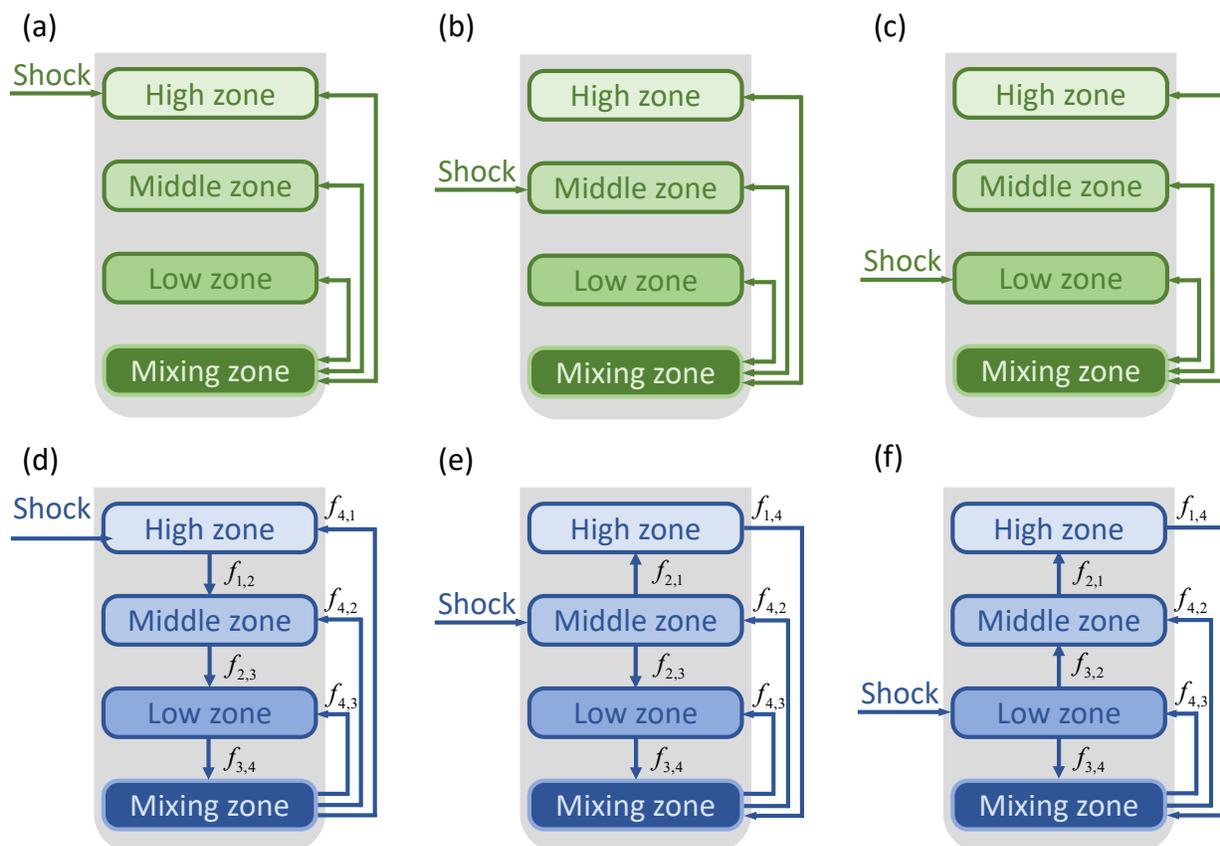


Figure 3.3: Schematics of the NaCl electrolyte conductivity transport model are illustrated for shocks injected at (a) high zone, (b) middle zone, and (c) low zone, and the hydroxide ion transport model with shocks injected at (d) high zone, (e) middle zone, and (f) low zone.

In this section, non-ideal mixing models are developed to capture the conductivity heterogeneity with respect to sensor measurements at different positions. The tank was partitioned into different nominal zones based on the positions of three MEA conductivity sensors (Figure 3.3(a)-(c)). The high, middle, and low zones represent the regions corresponding with the sensor assigned within that location. The mixing zone represents the region where the stirrer resides on the bottom of the reactor. The dominant mechanism for mixing is forced convection, which is significantly greater than the diffusive mixing between each zone [89]. Thus, it was assumed that the stirring power dominates the mass transfer in the reactor

under fast stirring, while the diffusion between adjacent zones could be neglected. The Reynolds number for the conductivity experiments is $Re = 1198.76$, which coincides with the laminar/turbulent transition region in an agitated cylindrical tank. The ion transport between the mixing zone and each sensor zone is defined as:

$$\begin{aligned}\frac{dC_i}{dt} &= \frac{1}{4V}k_i(C_0 + C_4 - C_i), \quad i = 1, 2, 3 \\ \frac{dC_4}{dt} &= \frac{1}{4V}k_4(C_1 + C_2 + C_3 - 3C_4).\end{aligned}\tag{3.1}$$

Here, C_i represents the electrolytic conductivity ($\mu\text{S}/\text{cm}$) of the solutions in each zone $i = 1, 2, 3, 4$, V is the volume of the reactor (0.38 L), k_i is a volumetric mass transfer coefficient (L/s) that represents the rate of forced convective mass transfer between the mixing zone and zone i . C_0 represents the inlet conductivity in the corresponding zone, which is equal to C_v (shock conductivity, $\mu\text{S}/\text{cm}$) in the zone with the injected shock during the injection period (i.e., the time duration for which the conductivity in the injection zone rises to a peak) or zero for other cases. Three series of experiments were performed, where a high-concentration shock solution was injected into each high, middle, and low zones. The corresponding model was then developed based on the injection position.

pH Mixing Model

In this section, non-ideal mixing models are established to capture pH heterogeneity within the high, middle, and low zones of the reactor. As the solute, hydroxide ions exhibit anomalously high apparent mobilities in aqueous solutions [90, 91]. Previous studies found that this anomalous transport behavior at the molecular level [92] was attributed to continu-

ous interconversion between a hydration complex of hydroxide ions and water molecules. Thereby, apart from forced convection and diffusive transport, the hydroxide ions undergo electrochemical interactions with water molecules, grabbing protons from adjacent water molecules to generate hydroxide ion clusters at adjacent new sites. The superficial transfer coefficients are introduced to represent these multifactorial interactions between each zone (Figure 3.3(d)-(f)). Three independent experiments were conducted with shocks injected at high, middle, and low injection positions. Different models are developed based on the corresponding transfer mechanism, as illustrated in Figure 3.3(d)-(f). For the KOH shock high-zone injection case, the model is established as:

$$\begin{aligned}
\frac{dH_1}{dt} &= \frac{1}{v_1V}(H_0 + \kappa_1H_4 - \kappa_1H_1), \\
\frac{dH_2}{dt} &= \frac{1}{v_2V}(\kappa_1H_1 + \kappa_2H_4 - (\kappa_1 + \kappa_2)H_2), \\
\frac{dH_3}{dt} &= \frac{1}{v_3V}((\kappa_1 + \kappa_2)H_2 + \kappa_3H_4 - (\kappa_1 + \kappa_2 + \kappa_3)H_3), \\
\frac{dH_4}{dt} &= \frac{1}{(1 - v_1 - v_2 - v_3)V}((\kappa_1 + \kappa_2 + \kappa_3)H_3 - (\kappa_1 + \kappa_2 + \kappa_3)H_4).
\end{aligned} \tag{3.2}$$

In this model, H_i represents the pH corresponding to each zone ($i = 1, 2, 3, 4$) in the reactor, v_i is the volume fraction of zone i , and κ_i is the superficial transport coefficient, indicating the “flow rate” of OH^- between adjacent zones that accounts for both reaction and convection transport (L/s). H_0 represents the input in the high zone which is equal to H_v (shock pH) during the injection period (0–8 s) and is equal to zero for the remaining process. The details and the development of pH models for middle and low zone injection cases are summarized in Appendix A.1.1.

Temperature Profiling Model

Heat transport was assumed to be significantly slower than fluid transport [89], indicating that the transport process of heat in each sensor zone should be the same (as shown in Fig A.1). Thus, only a single equation is required to accurately model the bulk fluid temperature based on the energy balance for the batch system [93]:

$$\frac{dT}{dt} = -\frac{UA_c}{V\rho C_p}(T - T_c). \quad (3.3)$$

Here, T is the temperature of the water solution ($^{\circ}\text{C}$) and A_c is the crosssectional area of the cylindrical container (m^2) across which heat transfer is occurring. ρ and C_p are respectively the density (kg/m^3) and heat capacity ($\text{kJ}/(\text{kg}\cdot^{\circ}\text{C})$) of the water ($\rho = 998.19$, $C_p = 4.18$). T_c is the temperature of the inner face of the bottom of the reactor and U is the overall heat transfer coefficient of the system ($\text{kW}/(\text{m}^2\cdot^{\circ}\text{C})$). The full details for the development of the temperature model are presented in Appendix A.1.2.

3.2.3 Parameter Estimation and Model Validation Using Global Dynamic Optimization

Rigorous deterministic global optimization was used to determine the uncertain parameters for validation of optimal mixing models to capture the mixing dynamics for the three targeted properties (e.g., conductivity, pH, and temperature). The general form of the global dynamic

optimization problem is defined as:

$$\begin{aligned}
& \min_{\mathbf{p} \in \Pi \subset \mathbb{R}^{n_p}} \phi(\mathbf{x}(\mathbf{p}, t_1), \dots, \mathbf{x}(\mathbf{p}, t_{N_t}), \mathbf{p}) \\
& \text{s.t. } \dot{\mathbf{x}}(\mathbf{p}, t) = \mathbf{f}(\mathbf{x}(\mathbf{p}, t), \mathbf{p}, t), \forall t \in I = [t_0, t_f] \\
& \mathbf{x}(\mathbf{p}, t_0) = \mathbf{x}_0(\mathbf{p})
\end{aligned} \tag{3.4}$$

In this formula, ϕ is the objective function formulated as the sum of squared error (SSE) between the model and the experimental data at specific discrete time points t_1, \dots, t_{N_t} corresponding with the experimental data, for each zone in the tank. \mathbf{x} is the generic state variable vector which represents $\mathbf{C} = (C_1, C_2, C_3, C_4)$ for the conductivity model, $\mathbf{H} = (H_1, H_2, H_3, H_4)$ for the pH model, and T for heat transfer model. \mathbf{p} is the uncertain parameter vector requiring estimation by optimization, which belongs to the parameter set $\Pi \subset \mathbb{R}^{n_p}$. \mathbf{x}_0 is the initial value vector for \mathbf{x} at $t = t_0$. The optimization problem is nonconvex and constrained by a system of ordinary differential equation (ODE) initial value problems (IVPs). The specific optimization formulations for conductivity and pH mixing models are summarized in Appendix A.1.3.

There has been active development of novel deterministic methods for solving eq(2) to guaranteed global optimality [17, 21, 25, 27, 55]. In this study, for conductivity and pH models, the nonlinearity comes from the bilinear terms of the system of ODEs. To solve the parameter estimation problems for these models, the models were reformulated into a system of nonlinear algebraic equations using an explicit Euler discretization and accounted for as equality constraints. As a result, the bilinear terms become recursively multiplied, resulting in the feasible set being nonconvex. The ANTIGONE v1.0 solver [94] in GAMS

v24.7.4 [95] was used to solve these parameter estimation problems to guarantee global optimality (absolute stopping tolerance is set to 0; relative stopping tolerance is set to 0.1; absolute feasibility tolerance is set to 1E-6). All global optimization results were obtained within 1.5 h, which is important for applications in real-time MPC of WTPs/WWTPs with a much longer residence time. The analytical expression for the temperature model was derived and applied within the global optimization formulation (Appendix A.1.4). The global optimization problem for the temperature model was solved using the EAGO v0.2.1 solver (EAGO.jl) [52] in the Julia programming language [53] via the JuMP v0.18 modeling language (JuMP.jl) [96]. The global results for the temperature model could be obtained within 2 min, which is appropriate for prompt temperature control in WTPs/WWTPs. The wall clock times were reported for GAMS and JuMP implementations run on a personal workstation with an Intel Xeon E3-1270v5 4-core/8-thread processor at 3.60 GHz/4.00 GHz (base/turbo) frequency running Windows 10 with 32 GB of ECC memory.

3.2.4 Precise Control of A Wastewater Nitrification System

The non-ideal heterogeneous mixing models for conductivity were expanded from a batch system to an unsteady pilot-scale continuous flow nitrification CSTR (1000L) to demonstrate their applicability to real WWTPs, with a controller being implemented to showcase disturbance rejection and energy saving operations (Figure 3.2 (c)). Three inlet streams continuously flow into the tank corresponding to the high, middle, and low zones. Similarly, three outlet streams continuously flow out of the tank at the corresponding zones. In addition, there is an air diffuser at the tank bottom continuously aerating for nitrification to

oxidize NH₄Cl [97]. A conductivity sensor is deployed in each zone to measure the corresponding conductivity at 10 s intervals. An impeller is in the mixing zone to continuously stir the liquid inside the tank. For traditional proportional-integral (PI) control, feedback is provided from the high-zone MEA conductivity sensor and a control signal is sent to the valve on the air stream at the bottom of the vessel. For MPC and its variants, sensors in each zone are utilized for feedback.

A modified conductivity mixing model that accounts for continuous operations in this tank is established to simulate the nitrification step:

$$\begin{aligned}\frac{dC_i}{dt} &= \frac{1}{4V}(k_i(C_4 - C_1) + \dot{m}_{in,i}C_{in,i} - \dot{m}_{out,i}C_i) + R_{NH_4^+}, \quad i = 1, 2, 3, \\ \frac{dC_4}{dt} &= \frac{1}{4V}k_4(C_1 + C_2 + C_3 - 3C_4) + R_{NH_4^+}, \\ \frac{dc_O}{dt} &= r_O + k_{la}(c_O^* - c_O),\end{aligned}\tag{3.5}$$

where $\dot{m}_{in,i}$ and $\dot{m}_{out,i}$ are continuous inlet and outlet flow rate at zone i (L/s), respectively, $C_{in,i}$ represents the conductivity of the inlet stream at zone i ($\mu\text{S}/\text{cm}$), and $R_{NH_4^+}$ is the reaction rate law for NH_4^+ consumption measured as conductivity ($\mu\text{S}/\text{cm}/\text{s}$), c_O is the oxygen concentration (mg/L), and r_O is the oxygen consumption rate described by a reaction rate law (mg/L/s). The aeration process is modeled by the rate of mass transfer of oxygen into the reactor liquid from air bubbles $k_{la}(c_O^* - c_O)$, where k_{la} is the volumetric mass transfer coefficient (s^{-1}) [98], and c_O^* is the saturated dissolved oxygen concentration (9.1 mg/L at 20 °C) [99]. The standard oxygen transfer rate (SOTR, mg/s) is defined as $SOTR = k_{la}c_O^*V$, and represents the amount of oxygen transferred per second at 20 °C. The standard oxygen transfer efficiency (SOTE, %) refers to the ratio of oxygen in the inlet air stream dissolved in

the liquid at 20 °C, given by $SOTE = SOTR/W_O$, with W_O the mass flow of oxygen in the air stream (mg/s). W_O can be calculated by an empirical formula: $W_O = 0.2967Q$, where Q is the airflow rate adjusted by the controller. The mass transfer coefficients were adjusted to construct a modified model that can account for all situations with single or multiple shocks at high, middle, and low zones. The model development process, the detailed kinetics for the nitrification reaction, and the mechanism for airflow and transfer into the liquid are introduced in Appendix A.1.3.

A case study of removing excess NH_4Cl in a nitrification CSTR of a WWTP is simulated as shown in Figure 3.2 (c). The input and output variables for the control system are given in Table 3.1. According to the standard of moderate municipal wastewater, the concentration of ammonium ions in the effluent should not exceed 30 mg N- NH_4^+ /L [100]. Thus, the operating setpoint (SP) is set at 280 $\mu S/cm$ corresponding to the standard concentration. Independent numerical experiments were conducted to assess the behavior of the system under four different influent shock conditions (i-iv) and six different operating scenarios to compare the system performance with various control approaches. From $t = 2100$ s to $t = 2250$ s, influent shocks (as step disturbances) in NH_4Cl concentration were introduced in each case as: (i) $C_{in,1} = 320 \mu S/cm$, (ii) $C_{in,2} = 320 \mu S/cm$, (iii) $C_{in,3} = 320 \mu S/cm$, (iv) $C_{in,1} = 300\mu S/cm, C_{in,2} = 350 \mu S/cm$ and $C_{in,3} = 270 \mu S/cm$.

Table 3.1: Input and output variables for the nitrification wastewater system with different control strategies are listed in this table.

Control system	PI control		MPC,EMPC1,EMPC2	
Input variables	Q	Airflow rate	Q	Airflow rate
			C_1	High zone conductivity
Output variables	C_1	High zone conductivity	C_2	Middle zone conductivity
			C_3	Low zone conductivity

Proportional-Integral (PI) Control

A traditional closed-loop PI-controller was modeled and tuned for rejecting influent conductivity shocks. The PI controller only makes decisions based on feedback signals from measurements in the high zone and adjusts the mass flow rate of air entering the system. The Internal Model Control (IMC) correlations are used to tune the PI parameters at first. Then, the parameters are further adjusted manually through simulations until the closed-loop system performs as desired. The details of the PI tuning process are introduced in Appendix A.1.3 . An experiment is performed on the system with a unit step disturbance to evaluate the control performance, the integral time-weighted absolute error (ITAE), integral time-weighted squared error (ITSE), integral absolute error (IAE) and integral squared error (ISE) are quantified with a settling time set as 1000 s, as listed in Table 2. The economic performance of the PI controller was assessed based on energy consumption and concentration disturbance rejection under the four influent shock scenarios.

Table 3.2: The control performance indices quantified by a response to a step disturbance change are presented in this table. The settling time is set as 1000 s.

Performance index	ITAE	ITSE	IAE	ISE
Value	2.817E4	1.606E3	5.414E1	3.185

Model Predictive Control (MPC)

MPC is an advanced control technique widely used in the process industries. MPC has been proposed for applications in WTPs/WWTPs to deal with the complexities from disturbances in the influent and physical and chemical phenomena [101, 102, 103]. MPC allows for tunable closed-loop response with its primary advantage being its intuition of process dynamics and capability to naturally handle multi-input/multi-output systems. In addition, compared

with conventional proportional-integral-derivative (PID) control and interval model control (IMC), MPC can handle more complicated systems (e.g., time delay, nonlinearity, open-loop instability), and provide a better response with less settling time. Therefore, a multi-input MPC was designed to improve disturbance (step function) rejection in WTP/WWTPs. The core concept of MPC is to solve an optimization problem at predetermined time points k to determine a control action that best drives the system towards the SP. In this study, the control action step size δ is set as 10 s (i.e., same as for PI control). An objective function is formulated as the sum of squares of the predicted errors (differences between the SP and the model predicted outputs) over a prediction horizon of P control action steps [104]:

$$\min_{u_k, \dots, u_{k+M-1}} \sum_{i=1}^3 \sum_{j=1}^P (\text{SP} - \hat{C}_{i,k+j})^2. \quad (3.6)$$

Here, u_m ($m = k, \dots, k + M - 1$) is the control variable which is equal to the airflow rate Q in this study (kg/s), SP is the setpoint ($\mu\text{S/cm}$), \hat{C}_i is the model predicted output ($\mu\text{S/cm}$) in zone i , and the subscripts indicate the sample time (k is the current sample time). P is the number of control actions in the prediction horizon ($P = 20$), and M is the number of control actions in the control horizon ($M = 3$). M control variables u_k, \dots, u_{k+M-1} are optimized at control action step k , but only the first control action u_k is implemented. Then, similarly, a new optimization problem is solved with respect to M control variables over a prediction horizon of P at step $k + 1$. The dynamic matrix control (DMC) method was used to evaluate model predicted process outputs \hat{C}_i [104]. The performance of the MPC was assessed based on energy consumption and concentration disturbance rejection under the four influent shock scenarios.

Economic Model Predictive Control (EMPC)

Economic MPC (EMPC) is a method for accounting for real-time process operations with respect to economic performance [105]. As such, EMPC can directly account for process economics in the determination of appropriate control response, and therefore is ideal for the development of next-generation WTP/WWTPs, such as real-time energy management and market-driven production [106, 107]. In this study, we formulate and implement EMPC with two different objectives: one is targeted at reducing environmental discharge (EMPC1); and the other is targeted at saving energy (EMPC2). The optimization problem for EMPC1 is defined as

$$\min_{u_k, \dots, u_{k+M-1}} \sum_{i=1}^3 \sum_{j=1}^P D_{i,k+j}^2 + w \left(\sum_{i=0}^{M-1} u_{k+i} (P - M + 1) u_{k+M-1} \right), \quad (3.7)$$

where D_i is the discharge from zone i ($\mu\text{S}/\text{cm}$), that can be expressed as

$$D_{i,k+j} = \begin{cases} \hat{C}_{i,k+j} - \text{SP} & \text{if } (\hat{C}_{i,k+j} - r) > 0 \\ 0 & \text{if } (\hat{C}_{i,k+j} - r) \leq 0 \end{cases}$$

The objective is to minimize the overall discharge above the SP over the prediction horizon of P . $w(\sum_{i=0}^{M-1} u_{k+i} (P - M + 1) u_{k+M-1})$ in (3.7) is a penalty function with respect to the control variables that guarantees the lowest oxygen consumption when the discharge is already below the SP (the penalty coefficient w is set as 0.03) and penalizes oxygen consumption when making control decisions.

EMPC2 is formulated as the constrained optimization problem:

$$\begin{aligned} \min_{u_k, \dots, u_{k+M-1}} & \left(\sum_{i=0}^{M-1} u_{k+i} (P - M + 1) u_{k+M-1} \right) \\ \text{s.t. } & \hat{C}_{i,k+j} - \text{SP} \leq 0, \quad \forall i = 1, 2, 3, \quad j = 1, \dots, M, P \end{aligned} \quad (3.8)$$

We seek to minimize the overall oxygen consumption over the prediction horizon to reflect energy management during operations. The inequality constraints ensure that the conductivity profiles over the control horizon, and at the end of the prediction horizon, will be at or below the SP. The performance of each EMPC was assessed based on energy consumption and concentration disturbance rejection under the four influent shock scenarios.

3.3 Results and Discussion

3.3.1 Optimal Solutions for Conductivity, pH, and Temperature

Mixing Models from Parameter Estimation

The global optimal solutions for parameters of conductivity and pH models are listed in Table A.1. The time costs for solving these global optimization problems are reported in Table A.2. The vast disparity in solution times for solving pH problems is due to the “curse of dimensionality” of deterministic global optimization [108] as the high zone injection problem has nearly double the optimization variables of the low- and middle-zone injection problems.

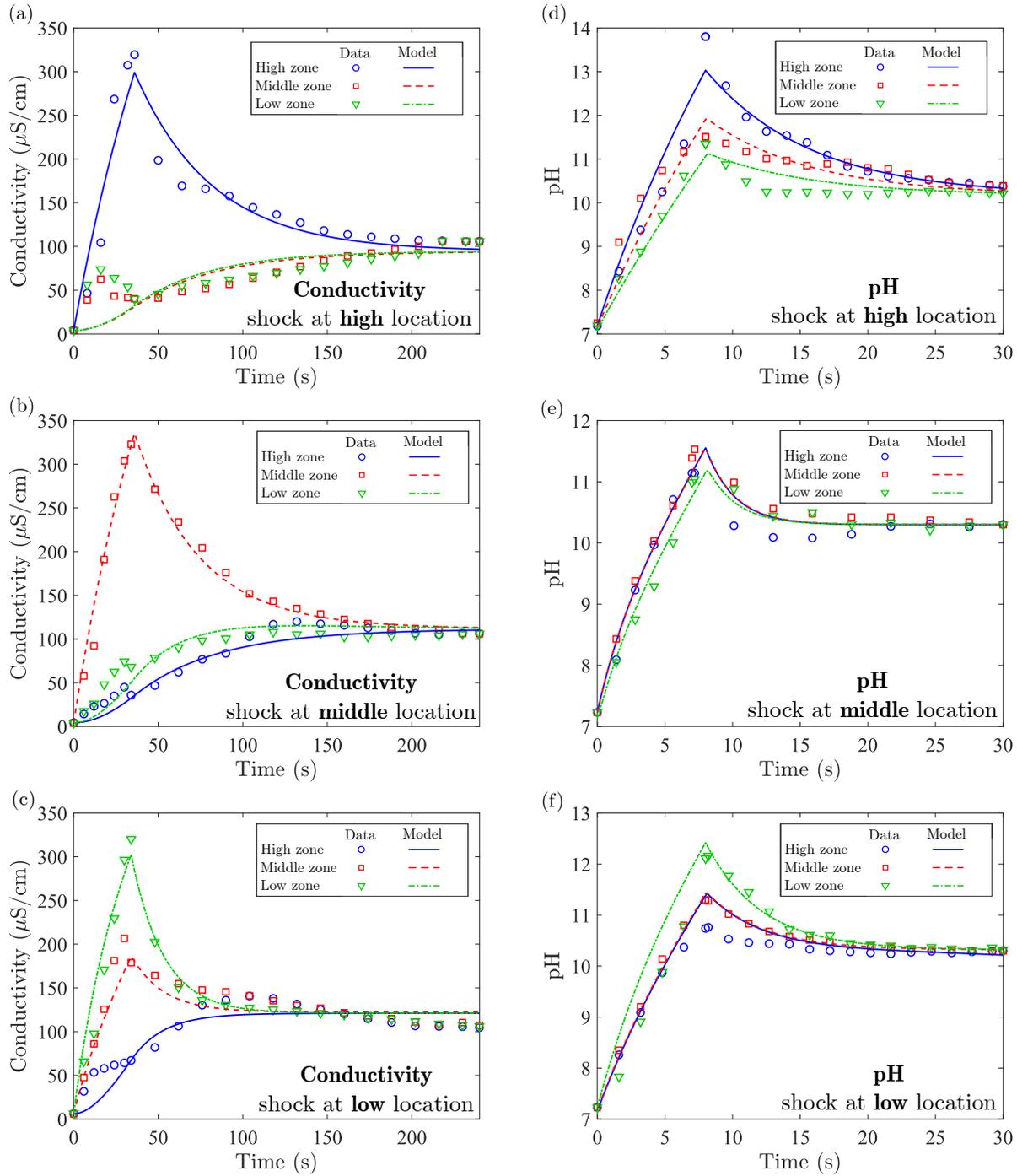


Figure 3.4: Optimal conductivity profiles from the global optimization results are plotted against a subset of data for (a) the high-zone injection model, (b) middle-zone injection model, and (c) low-zone injection model. Optimal pH profiles from the global optimization results are plotted against a subset of data for (d) the high-zone injection model, (e) middle-zone injection model, and (d) low-zone injection pH model.

The optimal conductivity profiles for different injection positions are presented in Figure

3.4(a)-(c). Overall, the conductivity profiles fit the experimental data well, and are able to capture the transient peaks caused by shock injections. This detection of transient maximum conductivity is of great importance in applications such as preventing the damage to bacterial cells in WWTPs [109], since both nitrogen removal and phosphorus removal processes in WWTPs exhibit significant changes in the conductivity of wastewater [76]. For the high-zone injection case (Figure 3.4(a)), it is observed that the optimal profiles in the middle and low zones do not exhibit small peaks like the data, indicating that the lower zones of the physical system receive ionic solutes from the upper zones in small amounts, which is not accounted for in the proposed model. Since the relative mismatch is quite small, no change was deemed necessary for the model. The low-zone injection profile (Figure 3.4(c)) exhibits two conductivity peaks in the low zone and middle zone, which may be attributed to the shock injection position (low zone) being very close to the mixing zone, thus transport to the middle zone occurs rapidly. In addition, the optimal parameter C_v^* for the low-zone injection case is much smaller than the high-zone and middle-zone cases (Table A.1), indicating that the mixing force quickly dilutes the shock in the low zone due to the closest proximity to the mixing zone.

The optimal pH profiles are shown in Figure 3.4(d)-(f). The models fit well for high-zone and middle-zone injections, while a small deviation can be observed in the high-zone pH profile in the low-zone injection model. The pH profiles for the middle-zone injection are more uniform, due to the equal probability for apparent OH^- transport towards the high and low sensor zones, supporting the hypothesis that the dominant driving force for apparent OH^- transport is the electrochemical reaction instead of forced convection. As compared with the conductivity transport model, proton (charge) transfers much faster (around 30 s

to achieve equilibrium).

The optimal parameter values for the temperature model are $U^* = 1.9183$ and $T_c^* = 26.40$. It took 65.97 s to solve the parameter estimation problem to global optimality. Since the entire system was considered as a single stirred batch reactor (conforming to the well-mixed assumption), the optimal profile exhibits no spatial variations between each sensor zone and fits the data (Figure A.1) almost exactly. Furthermore, the convective heat transfer coefficient was also estimated, using fundamental heat transfer principles and the Nusselt number (Nu), which is the ratio of convective to conductive heat transfer across a boundary. The Nusselt number is defined as $Nu = h_l D_c / \lambda$ [89], where h_l is the convection heat transfer coefficient of the flow equivalent to the overall heat transfer coefficient U for this heat transfer model, λ is the thermal conductivity of water (W/(m·°C)) listed in Table A.7, and D_c is the characteristic length that is equal to the surface area A_c divided by the perimeter P_c of the bottom inner surface ($D_c = A_c / P_c = 0.01375$). In general, the Nusselt number can be calculated as a function of the Reynolds number (Re) and the Prandtl number (Pr). In this experiment, a cubic stirring bar ($d = 38$ mm) was used at a rotation speed (ω) of 50 RPM ($5/6$ s⁻¹). The Reynolds number is then calculated as $Re = \rho d^2 \omega / \mu = 1198.76$, indicating that it is within the transitional region for flow in a cylindrical tank ($1000 \leq Re \leq 10000$), where μ is the viscosity of water (Pa·s) listed in Table A.7. The Nusselt number (Nu) for this system can then be calculated by $Nu = 0.664 Re^{0.5} Pr^{1/3} = 44.405$ [110], where Pr is the Prandtl number of water listed in Table A.7. Finally, the heat transfer coefficient can be estimated as $h_l = \lambda Nu / D_c = 1.9289$ kW/(m²·°C), which is very close (0.55% deviation) to the optimal solution U^* . Additionally, the optimal surface temperature T_c^* was higher than the observed solution temperature, which is consistent with the observed heat transfer

(raising solution temperature) over the entire time horizon. The observed temperature profile shows a significant reduction in heat transfer rate as the solution temperature approaches T_c^* as the rate of temperature increase (i.e., heat transfer) slows down over this period.

3.3.2 Comparison of Non-Ideal Mixing Models with Pure Data-Driven Models and CFD Models

CFD models account for complex physical phenomena [111] and therefore are extremely computationally expensive. Typical CFD simulations of the batch reactor took 1.5 h on computers similar to the one reported previously and required excessive memory storage. Note that this computational cost does not account for the substantial time investment needed for the model setup and testing. Furthermore, CFD models once developed can rarely be adapted to new situations with new parameter values. More details about the CFD model used in this study are recorded in Section A.1.6 in SI. The simulation of a CFD model (dashed line) under the high-conductivity high-zone shock fits well with the MEA sensor profiles (blue points) (Figure 3.5(a)), while there was a large discrepancy between the CFD simulated result and the MEA sensor data points under the middle and low shock (Figure 3.5(b) and (c)). This might be attributed to the simplification in CFD simulation for ion transport processes as it only considers ideal conditions and neglects some side effects such as the difference in surface smoothness and difference in mixing ability in each compartment (high, middle, and low) of the batch reactor. For instance, the middle zone was assumed to have the weakest mixing ability leading to the lowest mass transfer effectiveness while the mass transfer effectiveness should be highest when the shock came from the low position

closest to the mixing bar. In contrast, using the non-ideal mixing model, the SSE was reduced by 92.23% and 80.45% (Figure 3.5). It should be noted that a CFD model that compartmentalizes the reactor in a similar manner to our simple non-ideal mixing model, is expected to perform much better. However, the development and computational costs for such a model are considered to be prohibitive for any practical, real-time implementation and use for precise control systems.

The observed mixing trends could not be represented well by the CFD simulation under conductivity shocks (Figure 3.5), let alone the fast-transient scenarios of pH with multifactorial reactions. As an example, an axisymmetric model of a pH-sensitive electrochemical field effect sensor comprising 13650 elements was deployed to simulate a geometrical domain of 0.09 mm², demonstrating that it is intractable to apply CFD models in the batch reactors used in this study (volume: 380 mL) as the grid would have to be refined by a factor of 150.

The non-ideal mixing model was also compared with a pure datadriven model from the Eureqa modeling engine (DataRobot), that generates differential equations trained on the same data set. All data points were equally weighted for training and were integrated as black lines in Figure 3.5(d)-(f). The mean SSE (full name) values between the original sensor data and machine learning results were 144760, 110959, and 63,068 for each shock (high, middle, and low locations), respectively. In contrast, the SSE value between the original sensor data and the non-ideal mixing model's simulation results were markedly lowered by 68–83%, respectively. The poor fit of the Eureqa regression models is attributed to lacking the conservation principles as a basis, so that the regression models deviate from the main trends of mass and heat transfer.

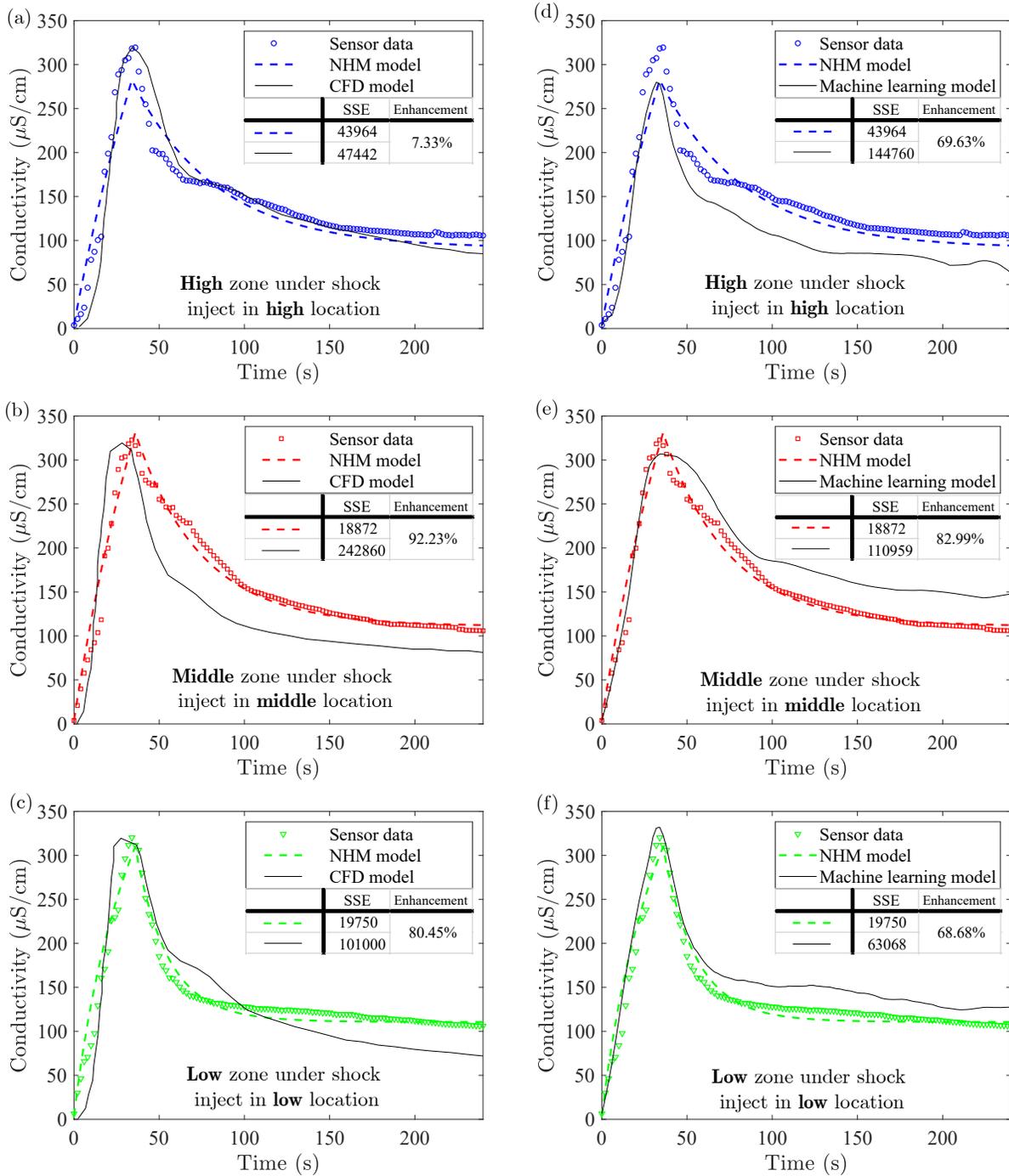


Figure 3.5: Comparisons of the simulation results of the non-ideal heterogeneous mixing model (dashed line) with the fitting results of (a-c) the CFD model (solid line) and with the fitting results of (d-f) the machine learning model (solid line), are plotted for each zone corresponding with the conductivity shock injection locations. (a) and (d): plots of the high zone under shock injected in the high zone; (b) and (e): plots of the middle zone under shock injected in the middle zone; (c) and (f): plots of the low zone under shock injected in the low zone). (Note: NHM = Non-ideal heterogeneous mixing; The sensor data profile (dots) and CFD model simulation are extracted from the previous study [112])

3.3.3 Validation and Calibration of Sensors and Non-Ideal Mixing Models

The non-ideal mixing models can be easily modified to simulate conductivity or pH profiles of other solute species by calibrating the parameters. In terms of conductivity, the main transport mechanism of ions without chemical reaction should be the same. However, the uniqueness of each ion is associated with distinct conductivity values, posing the requirement for calibrating the as-developed models to sustain accuracy under varying scenarios. To further validate the applicability of the developed non-ideal heterogeneous mixing models, additional experiments were conducted by injecting different soluble compounds (KCl, MgSO₄ and NaOH). The conductivity and pH profiles were simulated versus the experimental sensor data using the corresponding models with the parameters calibrated based on the actual experimental conditions.

For validation of the conductivity model, the original optimal parameter values (as listed in Table A.1) were used to predict the KCl and MgSO₄ conductivity profiles. The results showed that the simulated profiles using the original optimal parameter values qualitatively follow the same trends as the new data (Figure 3.6(a) and (c)). The reason for the significant quantitative mismatch is due to the differences in injection periods between the new and the original experiments. Since all shock injections are manual operations, the injection speeds cannot be regarded as a controllable experimental condition. The high-zone data reaches a peak much faster, indicating that the injection speeds for the KCl and MgSO₄ experiments are faster than the NaCl experiment. Apart from the injection speeds, the absolute injection quantity of different ions is another attribute leading to a mismatch. For example,

though the mass concentrations of KCl and MgSO₄ (200 μL, 100 g/L) injected are the same compared with the original NaCl conductivity experiment, different ions lead to different conductivities in solution and different shock conductivity C_v . The electrical conductivities of the ionic solutions based on mass percent are listed in Table A.3. Since aqueous NaCl and KCl solutions exhibit nearly the same conductivities, the predicted KCl profile using the original NaCl injection model exhibits nearly the same steady-state conductivity versus the new experimental data. In contrast, the steady-state conductivity of the predicted MgSO₄ is around three times higher than the new experimental data (Figure 3.6(c)). This difference is expected since MgSO₄ exhibits roughly one third of the electrical conductivity of NaCl across the mass percentage range (Table A.3). To improve the simulation results (i.e., model prediction accuracy), a simple calibration procedure (Appendix A.1.5) was conducted without modifying the model structure, so that the underlying physical phenomena captured by the original model could be preserved. The revised conductivity profiles for KCl and MgSO₄ (Figure 3.6(b) and (d)) exhibit substantially improved fits.

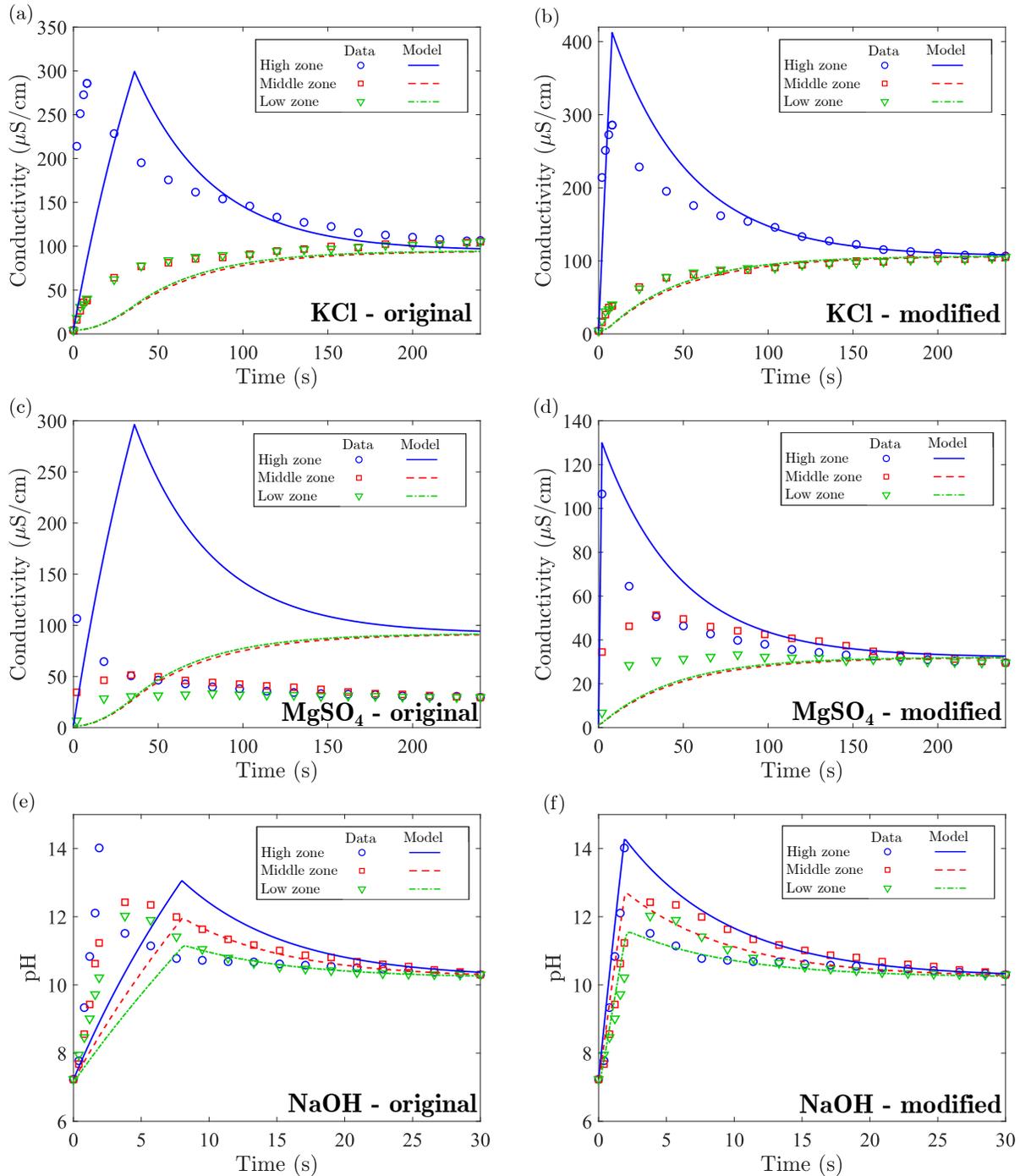


Figure 3.6: The predictive simulation results are plotted for KCl conductivity profiles versus experimental data using (a) original optimal parameters and (b) modified parameters. The predictive simulation results are plotted for MgSO₄ conductivity profile versus experimental data using (c) original optimal parameters and (d) modified parameters. The predictive simulation results are plotted for NaOH pH profiles versus experimental data using (e) original optimal parameters and (f) modified parameters.

To validate the pH model for the NaOH experiment, the pH profile was simulated using the original parameters for the KOH injection case. The simulated pH profile exhibits the same qualitative behavior as the data (Figure 3.6(e)). Again, the major mismatch is caused by the observed difference between the injection periods of the NaOH and KOH data. The time duration for the pH of the NaOH experiment to reach the peak is much shorter. Thus, the corresponding shock parameter Hv should be calibrated to mitigate this difference. The calibration process is summarized in Appendix A.1.5. The revised pH profile (Figure 3.6(f)) exhibits a far better fit than the original simulation, where the peak pH from the model also matches the data. The profile of the high-zone pH in the short time horizon after the peak does not accurately fit the data. The probable reason is that the quasi dynamics of OH^- transport after the shock is not accurately captured by the model or the experimental errors.

3.3.4 Non-Ideal Mixing Models for Improved Wastewater Treatment with Precise Control

Simulation Results for The Wastewater Nitrification System with Closed-Loop Controls

The simulation results for each independent study are illustrated in Figure 3.7. For the high-zone shock case, the PI controller begins to adjust the air valve to accelerate the airflow rate for excess ammonium removal once the disturbance occurring in the high-zone inlet flow is detected. The high-zone conductivity quickly drops below the SP, then the valve on the air stream is closed and the conductivity gradually rises towards the SP. As for middle-zone and low-zone shocks, the conductivities can still be controlled at the SP despite only

using feedback readings from the high-zone sensor. The reason is that the conductivity becomes quickly mixed at around $t = 2400$ s resulting in the overall conductivity of the tank approaching the SP under control.

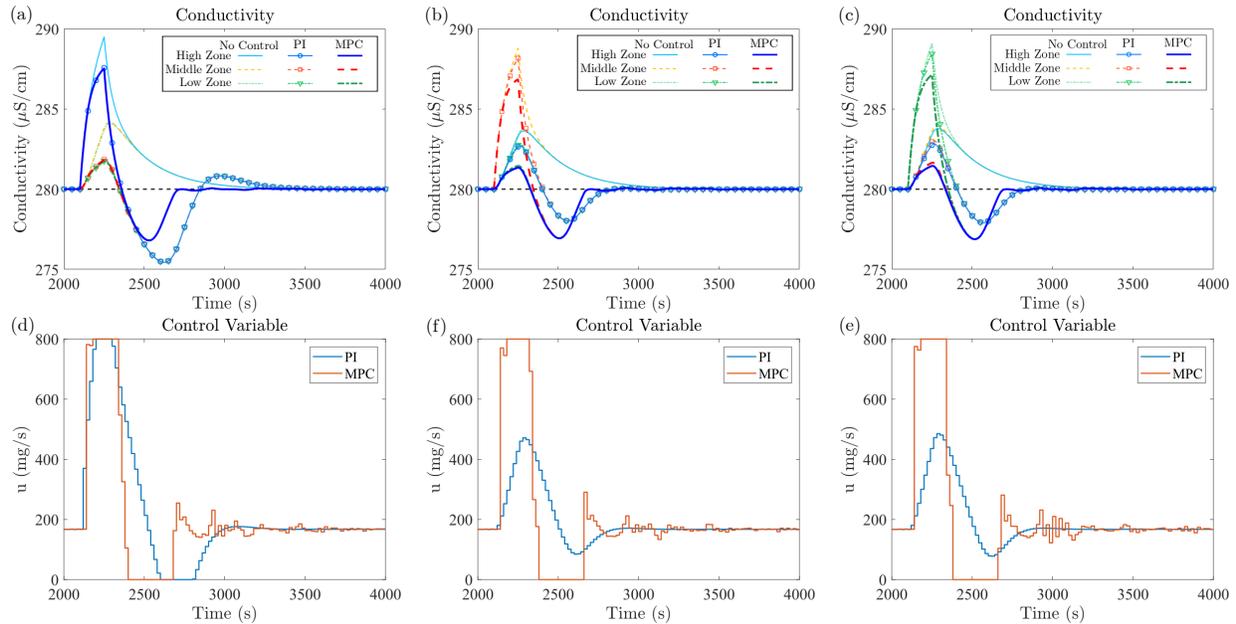


Figure 3.7: The conductivity profiles are plotted for the uncontrolled, PI control, and MPC simulations for independent studies with NH_4Cl conductivity shock ($320 \mu\text{S}/\text{cm}$) continuously injected from 2100 s to 2250 s, respectively, in the (a) high, (b) middle, and (c) low zones of a continuous-flow nitrification system. The PI (blue) and MPC (orange) control actions for the (d) high-zone shock case, (e) middle-zone shock case, and (f) low-zone shock case, are presented below their corresponding conductivity profiles.

For comparison, the MPC simulation results are also illustrated in Figure 3.7. For the high-zone shock simulation, the conductivities can be directed to the steady state at SP much faster than PI control. As indicated by the control variable, the consumption of oxygen is reduced significantly by MPC, implying substantial energy savings. Furthermore, the middle-zone and low-zone shock simulations show that with MPC, much less ammonium is discharged to the environment than with PI control. This is because multi-input MPC can account for feedback signals from all sensors, make accurate predictions of process transients

using the non-ideal heterogeneous mixing models, and take appropriate action versus the PI controller that only considers feedback signals from the high-zone sensor.

Evaluation of Treatment Performance and Energy Savings

An open-loop controller was also simulated for each study to represent conventional and conservative operations as a reference for comparisons. Once the shock from the influent is observed ($t = 2100$ s), the operator will open the aeration valve by an amount estimated from the difference between the shock value and SP for full oxidation (u is set as 561.44 mg/s). After the system's fixed settling time (1000 s), the operator will adjust the valve again. In contrast, for the uncontrolled simulations, the control variable is always set at the initial value ($u_0 = 168$ mg/s) which results in the steady-state effluent conductivity meeting the SP under steady influent conditions.

To evaluate the system's performance, the excess ammonium discharge was quantified as the area under the conductivity profiles as they go above the SP over the simulation horizon (2000 s). In addition, the energy consumption was quantified as the overall amount of air used for ammonium oxidization over the simulation horizon. The comparisons between uncontrolled, open-loop control, PI control, MPC, EMPC1, and EMPC2 cases are illustrated in Figure 3.8 with the data values for these plots listed in Table A.5. The discharge quantification for each case is calculated as the percentage of the uncontrolled simulation, whereas the energy quantification is represented by the percentage of the open-loop control simulation. It is apparent from the discharge plot (Figure 3.8(a)), that all the control strategies can greatly reduce the discharge compared with uncontrolled simulations. MPC and both EMPC strategies perform much better than PI control as less ammonium is discharged into

the environment. This is especially clear for the middle and low-zone shock studies, where MPC and EMPC account for multiple input measurements simultaneously, while PI control can only account for the high-zone measurement. Specifically, EMPC1 has the best performance for reducing discharge, coinciding with its underlying design objective. On the other hand, energy usage (Figure 3.8(b)) under PI control, MPC, and both EMPC strategies, is reduced versus open-loop control. As for the high-zone shock injection study, MPC has the greatest advantage due to the least energy usage among other control strategies while its discharge is only slightly higher than EMPC1 and EMPC2. As for the middle- and low-zone shock studies, PI control saves the most energy, but also has the greatest discharge compared with the other strategies.

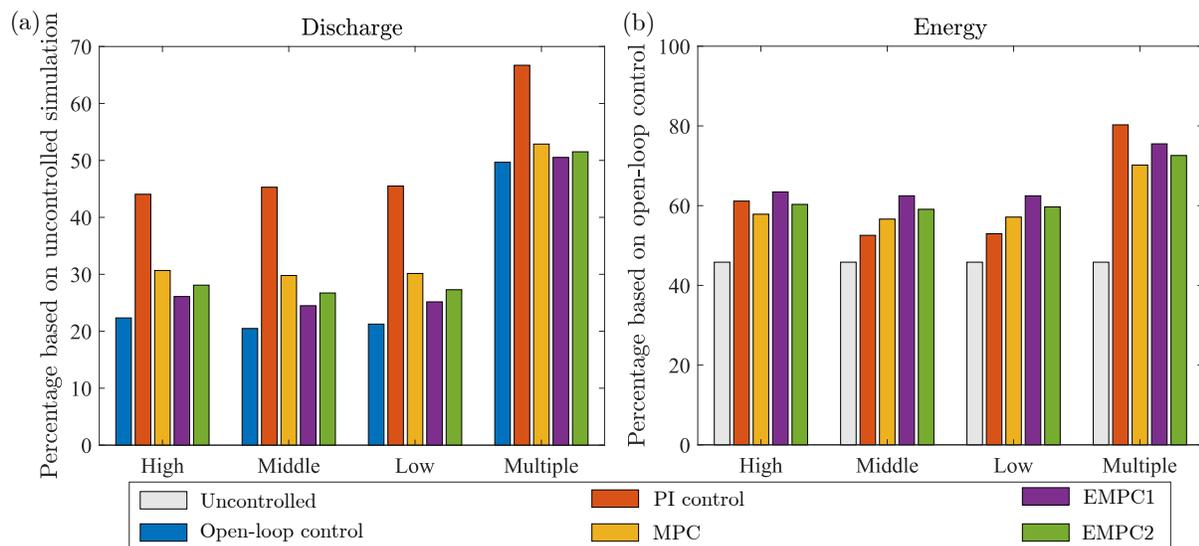


Figure 3.8: (a) The overall discharge of NH_4Cl above the setpoint (SP) over the simulation horizon, is plotted for each control scenario as a percentage normalized against the uncontrolled case (the uncontrolled case is 100%). (b) The total compressed air consumption (equating to energy usage) over the simulation horizon is plotted for each control scenario as a percentage normalized against the open-loop control simulation (the open-loop control case is 100%). The labels “High”, “Middle”, “Low”, and “Multiple” represent high-zone, middle-zone, low-zone, and multiple-zone shocks, respectively. In both cases, lower percentages equate to better performance.

It is observed that MPC has lower energy usage compared with EMPC1 and EMPC2. EMPC1 is formulated to minimize environmental discharge and it is apparent that more energy is consumed to achieve this objective. EMPC2 is formulated to minimize air consumption with constraints on discharge, but as indicated, it still consumes slightly more energy than MPC. The reason for this behavior is that EMPC2 seeks a control setting that has the lowest energy consumption with a prerequisite to strictly satisfy the discharge constraints while MPC only minimizes the errors without any specifications on discharge.

For the multiple shocks study, barring open-loop control, PI control is apparently the worst control strategy using the most energy and resulting in the most ammonium discharge. The energy usage and discharge for MPC, EMPC1, and EMPC2 are relatively similar, and any lower discharge observed must be paid for with greater energy consumption. It is suggested that for real-world operation, control strategies should be determined based on the specific conditions of the WTP/WWTP. This is aligned with the notion of “smart plant operations,” where process control, plant-wide management, and corporate office systems communicate in real-time through networks to satisfy targeted economic, environmental, and safety performance objectives [113]. As a result, the developed conductivity model, modified for continuous flow is valid for MPC and EMPC, and promising for real-time decision-making over the network for better management, energy savings, and handling of market/demand changes in WTPs/WWTPs.

3.4 Conclusions

WTP/WWTPs have been well-known for their large amounts of data generated with low efficiency of data utilization, operational uncertainty, and fluctuations in water quality/quantity [74, 75]. These fluctuations require frequent parameter adjustment and model recalibration during operation for effective MPC, but traditional physics-based mechanistic models are incapable of adapting to these changes in a timely manner [114]. The non-ideal heterogeneous mixing models proposed in this study are simple with few fitting parameters and take much less time for simulation than traditional CFD models. The collected high resolution sensor data can be instantly transmitted to the model calibration process, ensuring the calibration of non-ideal heterogeneous mixing models in a real-time *in situ* mode during on-going operation. This unique feature mitigates the severe time-delay problems of traditional pure physics-based models and enables a prompt modification for higher accuracy system identification based on authentic representations of the system. Exploiting this technology within closed-loop control, such as MPC, enables a novel precise control system for WTP/WWTPs.

Moreover, the heterogeneity profiling models can be applied for pattern recognition so as to better understand the internal mechanisms of complex processes (e.g., transport mechanisms of different ions), with or without involving algorithms and redundant equation deduction. Such generalized methodology can serve as a platform for simulating state variables for other chemical species with similar physical principles.

In the next chapter, a medical-relevant application of the spatiotemporal system that involves in PDEs is demonstrated through a cancer research study. The method proposed in Chapter 2 is utilized to construct state bounds for solving challenging PDE-constrained

nonlinear programs to global optimality.

Chapter 4

Optimal Therapy Design With Tumor Microenvironment Normalization

In this chapter, an application of formal methods to optimal therapy design in cancer research is discussed. The vasculature in the tumor microenvironment (TME) are poorly-organized and leaky. Normalizing the TME can improve therapy efficacy by restoring the vasculature to increase anticancer nanocarrier delivery. A well-established digital testbed for solid tumors can provide a better understanding of transport mechanisms of TME biophysics to enable optimal therapy design. Deterministic global dynamic optimization with the novel bounding routine introduced Chapter 2 is used to validate a mechanistic model against *in vivo* data. We find that TME normalization with dexamethasone increases the maximum transvascular convection rate of nanocarriers by 48-fold, the tumor volume fraction exhibiting convection by 61%, and the total amount of convection by 360%. However, 22% of the tumor still lacks convection. These findings underscore both the effectiveness and limits of TME normalization. As a consequence, this digital testbed quantifies transport and can be

used to perform optimal therapy design.

4.1 Introduction

Solid tumors feature pathophysiological abnormalities that are biophysical barriers to the transport of anticancer drugs. These barriers impede the effectiveness of such therapies by limiting their accumulation and spatial distribution [115]. Ameliorating the pathophysiology such that tumor microenvironment (TME) components have a more “normalized” phenotype increases small-molecule and nanocarrier-based therapies’ delivery and efficacy in cancer patients [116, 117, 118]. However, TME normalization combined with anticancer therapies has yet to lead to cures throughout a cancer patient population. Thus, a deeper understanding of how TME normalization affects the transport of therapies within tumors is necessary to fully bypass these spatially and temporally heterogeneous biophysical barriers. Mathematical modeling can be used to construct a robust framework for studying how the normalized TME modulates biophysical barriers to transport phenomena in tumors, thereby enabling the discovery of deeper insights into effective TME normalization. In turn, such a framework may serve as the foundation for establishing a technology platform for effective therapy design to improving therapeutic efficacy.

4.1.1 Cancer Biology

Nano-sized anticancer therapies on the order of dozens of nanometers, including macromolecules such as polymeric micelles and antibodies, benefit from: longer systemic circulation owing to slower clearance; selective accumulation in tumors owing to leaky tumor

blood vessels; and long retention in tumor tissue owing to dense fibrosis and non-functional lymphatics in the TME [115]. In fact, nano-sized therapies are currently in use today with cancer patients [117]. Nonetheless, leaky blood vessels, dense fibrosis, and nonfunctional lymphatics collaborate to construct biophysical barriers that reduce the effectiveness of cancer treatments [115, 117, 118, 119]. Nano-sized therapies are affected in a size-dependent manner [120, 121]. In tumors, plasma from circulation excessively extravasates from leaky blood vessels to the interstitial (i.e. extravascular) space, yet moves slowly because dense fibrosis limits fluid movement [122]. Ultimately, fluid cannot be cleared because tumor lymphatics are non-functional [123]. Thus, one distinguishing feature of tumors is an elevated interstitial fluid pressure (IFP), that eliminates transvascular convective transport of drugs in tumors by reducing the transvascular pressure gradient to zero [115, 122].

Vascular normalization involves fortifying leaky tumor blood vessels by blocking angiogenesis [6, 116]. ECM normalization involves reversing dense fibrosis by reprogramming cancer-associated fibroblasts to a quiescent phenotype so that the fibroblasts stop producing and maintaining excessive levels of extracellular matrix (ECM) [6, 116, 124]. As a result, the dense fibrosis, which slows interstitial fluid movement and compresses intratumor lymphatic tumor vessels such that they are nonfunctional [125], is diminished. Already, vascular normalization is used with nanomedicine in patients [126], while ECM normalization recently succeeded in a clinical trial with small-molecule chemotherapy [127].

We recently discovered that dexamethasone, which is a glucocorticoid steroid often used to manage chemotherapy-related toxicities, can induce vascular and ECM normalization simultaneously if used at an appropriate dose and schedule [6]. Yet, how dexamethasone affects blood vessel leakiness, fibrosis, and lymphatic vessel function towards alleviating

IFP and restoring a transvascular pressure gradient is multi-factored. Each factor depends on the dose of dexamethasone differently. Furthermore, how the size of nanocarrier-based anticancer drugs interacts with these factors is unclear. Therefore, enhancing the delivery of nanocarriers is a multi-faceted engineering problem, so a model-based systems engineering approach is required to better understand the underlying physical phenomena and complex relationships of the biological system. Throughout this work, we will use the term *nanocarrier* to include nano-sized therapies, in general.

4.1.2 Modeling and Simulation for Cancer

Transport of nanocarriers from the systemic circulation to cancer cells includes three steps: flow through blood vessels to different regions of the tumor, transvascular transport, and transport through the interstitial space of tumor. Specifically, the capillary vasculature is a highly dynamic region for transvascular transport of medicine, nutrients, and waste materials being exchanged between the blood vessels and the interstitial space. There are two key transvascular transport mechanisms: diffusion and convection. Generally, smaller nanocarriers benefit from diffusion using concentration gradients as an additional driving force for transvascular transport, whereas larger nanocarriers must rely on convective transport using pressure gradients due to steric hindrances that make diffusion very slow [115, 128]. Previous studies have indicated that diffusion is the main mechanism of mass transport across the vessel wall in tumors, because of the lack of transvascular pressure gradients [115, 123]. However, dexamethasone affects blood vessel leakiness, fibrosis, and lymphatic function, so it could restore transvascular pressure gradients. How diffusion and convection are affected for

differently sized nanocarriers is unclear. To investigate, a first-principles-based modeling approach is required to quantify the important physiological parameters that govern transport in tumors.

The vascular and interstitial transport phenomena in tumors have been extensively modeled. Baxter and Jain [129, 130, 131, 132] developed a one-dimensional spherical tumor model that describes fluid and nanocarrier transport. Baish et al. [133] developed a two-dimensional fluid transport model that considered coupling between the vessels and the interstitial space of tumors in a unified theoretical framework. Afterwards, Chauhan et al. [121] applied this coupled transport system to a percolation-based tumor vasculature network and subsequently established the solute transport model on this network. Sweeney et al. [134] further developed a three-dimensional model to simulate vascular blood flow and interstitial fluid transport. Their model integrated the complex vascular structure to provide a visualization of spatial heterogeneity, which can predict the response of fluid dynamics following vascular normalization therapy. Through simulation of vascular normalization, the authors concluded that therapies should “seek to develop an IFP gradient,” which is consistent with the overall simulation results and conclusions of previous computational studies of tumor vascular normalization.

In addition to first-principles mechanistic models, artificial intelligence (AI) has been gradually becoming a popular model-based approach in pharmacokinetics/ pharmacodynamics (PKPD) studies [135, 136, 137]. An efficient machine learning model simplifies computationally intensive simulations by creating mathematically simple regression models that capture input-output relationships with high accuracy [138]. Specifically, artificial neural networks (ANNs) are powerful computational models that are capable of approximating and

predicting the behavior of such complicated systems with high accuracy and efficiency[139].

In this work, we establish a systematic *in silico* model-based framework using deterministic global optimization for optimal therapy design within the context of TME-normalization processes. First, we propose using deterministic global optimization to solve the parameter estimation problems and provide a rigorous quantitative foundation for *in silico* model discrimination. Using this foundation, we quantify the relative contributions of convection and diffusion to solute transport across the vessel walls. Moreover, we develop an optimal TME-normalizing therapy design approach for dose selection that demonstrates the relationship between dexamethasone dose and the interstitial concentration of anticancer drugs in the pharmacokinetic system. Finally, we use this tumor transport model to determine an optimal nanomedicine size for the greatest accumulation in the tumor interstitial space. We also propose an ANN surrogate modeling approach to reduce the computational cost of solving challenging deterministic global optimization problems for model validation, dexamethasone dose selection, and anticancer nanocarrier size selection. The details of establishing and using such machine learning models within optimization-based decision-making frameworks are presented in this work.

Our work seeks to enhance the practicability and predictive capabilities of tumor transport models using mechanistic and data-driven model validation approaches and rigorous methods in global optimization for stronger model-based systems engineering approaches for optimal therapy design in cancer research. The information obtained through this approach aides in the development of better models and provides deeper insight into the physical behavior of molecular transport during TME normalization to guide drug development and delivery.

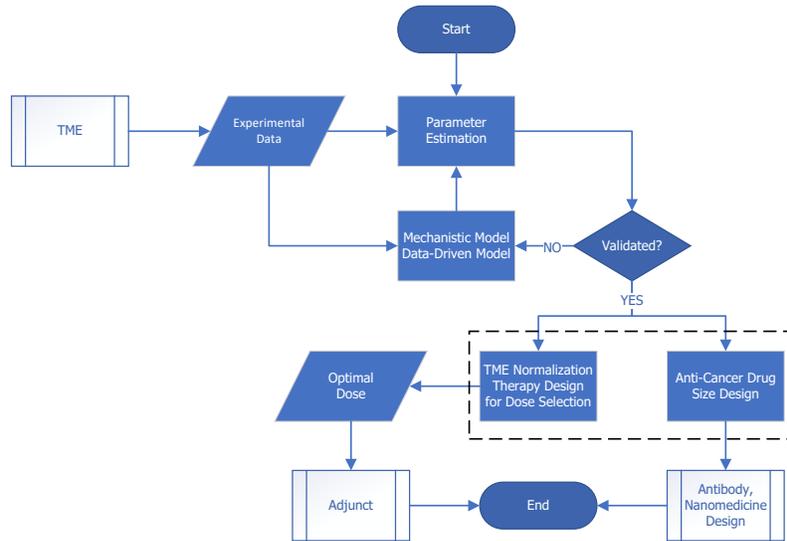


Figure 4.1: A flowchart is illustrated demonstrating a systematical framework for optimal therapy design within the context of tumor microenvironment (TME) normalization. Based on the experimental data, parameter estimation is utilized to validate/invalidate a proposed mechanistic model or data-driven model of the tumor. Validated models are then applied to tumor TME normalization therapy design for dose selection and anticancer drug size design. Note that the TME normalization therapy design and drug size design in the dashed line box can be implemented separately, sequentially or simultaneously.

4.2 Methods

Figure 4.1 illustrates the overall systematical framework proposed for model-based TME-normalizing therapy and drug size design. To enhance the predictive capabilities of the models and provide confidence in their utility for the model-based approach for drug and therapy development, we propose to use formal methods to estimate and quantify the critical parameters for model validation. This approach requires solving a nonconvex nonlinear program (NLP) constrained by the mechanistic tumor transport model as an unsteady partial differential equation (PDE). A simulation-based feasible path approach is proposed and the PDE-constrained optimization problem is reformulated as a box-constrained NLP. In

addition, ANN machine learning methods are proposed to construct surrogate models for reducing the time costs of solving global optimization problems. Moreover, the well-established mechanistic and ANN models are also used in TME-normalizing therapy design for optimal neoadjuvant dose selection as well as drug size design for anticancer nanocarriers.

4.2.1 Parameter Estimation and Model Validation by Deterministic Global Optimization

The glucocorticoid steroid DEX, an agent mainly used for alleviating chemotherapy side effects, has been identified as a pre-treatment adjunct agent for normalizing metastatic tumor vessels and ECM for enhanced efficacy of drug delivery [6]. To validate the effects of DEX on nanocarrier delivery through vascular and ECM normalization processes, we propose to verify the optimal solutions of the parameter estimation problems introduced in Martin et al. [6] by deterministic global optimization. This approach is significant because only global optimal solutions can guarantee the most accurate fit to the obtained experimental data. The mechanistic tumor transport model used in this work is introduced in Appendix A.2.

Martin et al. [6] conducted a series of experiments *in vivo* to investigate the efficacy of DEX. In these experiments, immunocompetent mice bearing orthotopic 4T1 breast cancer were treated with 3 and 30 mg/kg DEX daily for 4 days. After which, two types of fluorescent dyes (70 kDa rhodamine-bound dextran and 500 kDa FITC-bound dextran) were injected as tracers [6]. *In vivo* confocal laser scanning microscopy was employed to characterize the spatiotemporal distribution of dextrans in mouse tumors treated with different

doses of DEX [120]. Based on the intravital microscopy images, the effective permeability P_{eff} was quantified as the rate of nanoparticle fluorescent signal passing through the vessel walls normalized to the vessel surface area and the transvascular concentration difference [121]. Note that the effective permeability includes both convective and diffusive components; however, it significantly overestimates the diffusive part and may not be consistent with actual transcapillary transport [129]. Then, the spatial average concentration of the interstitial space was calculated from the conservation equation[121]:

$$\frac{dc_{avg}^{data}}{dt} = P_{eff} \frac{S}{V} (c_v - c_{avg}^{data}),$$

where c_v is the solute concentration in the vessels of a tumor (g/mL) and $\frac{S}{V}$ is the vascular surface area per unit volume (cm^{-1}). This serves as an experimental concentration profile for subsequent parameter estimation problems used for elucidating the physiological effects of DEX treatment.

In this work, a similar approach is taken whereby the dimensionless spatially-averaged concentration of solute \hat{c}_{avg}^{data} (determined from the overall conservation equation) serves as an experimental concentration profile for each P_{eff} measured experimentally, and used for parameter estimation of the mechanistic model of interest. Deterministic global optimization methods are used to validate the mechanistic model and verify the TME-normalization process. The objective function is formulated as the sum-of-squared errors (SSE) between the average concentration profile predicted by the model and the measured data (from the overall conservation expression with the experimentally measured P_{eff}) at discrete time points over the entire time horizon of the experiment. Inequality constraints are formulated

for the IFP profiles based on experimentally determined values. The parameter estimation problem is formulated as:

$$\begin{aligned} & \min_{\boldsymbol{\pi} \in \Pi} \sum_{i=1}^n (\hat{c}_{\text{avg}}(t_i, \boldsymbol{\pi}, d_m) - \hat{c}_{\text{avg}}^{\text{data}}(t_i))^2 & (4.1) \\ \text{s.t. } & \hat{p}_{\text{peri}}(\boldsymbol{\pi}) \leq \hat{p}_{\text{peri},\text{max}} \\ & \hat{p}_{\text{peri}}(\boldsymbol{\pi}) \geq \hat{p}_{\text{peri},\text{min}}, \end{aligned}$$

where the dimensionless spatially-averaged concentration of solute \hat{c}_{avg} is calculated by averaging the dimensionless concentration \hat{c} for all spatial nodes (discretization details are introduced in Section 4.2.7) from the mechanistic solute transport model (details are introduced in Appendix A.2.1) and taken as the parametric model output for the parameter estimation problem. The decision variables $\boldsymbol{\pi} = (L_p, K) \in \Pi \subset \mathbb{R}^{n_\pi}$ is the vector of physiological parameters of the model to be estimated, with L_p the hydraulic conductivity of the microvascular wall (cm/mm Hg-sec) and K the hydraulic conductivity of tumor interstitium (cm²/mm Hg-sec). The parameter d_m is the diameter of the nanocarrier (nm) used in the corresponding experiment. The SSE objective fits the model-predicted profile to the experimental profile at each time node t_i selected within the time horizon (5 min), with $i \in \{1, \dots, n\}$. For the inequality constraints, we introduce \hat{p}_{peri} as the dimensionless superficial (peripheral) IFP which is calculated by the dimensionless IFP \hat{p} in the superficial region (introduced in Section 4.2.7), and $\hat{p}_{\text{peri},\text{max}}$ and $\hat{p}_{\text{peri},\text{min}}$ as the physical bounds of \hat{p}_{peri} , with values taken from Martin et al. [6] and listed in Table 4.1.

Table 4.1: The physical bounds on the superficial (peripheral) tumor IFP for the control, 3 mg/kg, and 30 mg/kg DEX treatment case are listed in this table as determined by Martin et al. [6].

Dose	Control	3 mg/kg	30 mg/kg
$\hat{p}_{peri,min}$ (mmHg)	4.87	3.02	1.95
$\hat{p}_{peri,max}$ (mmHg)	5.67	3.62	2.45

4.2.2 Bounding Methods for Tumor Transport Model

Deterministic global optimization can prevent erroneously invalidating mechanistic models in cases where suboptimal solutions obtained by local optimization algorithms result in poor fits. Methods for solving global optimization problems in this work rely on the branch-and-bound (B&B) framework [140] for deterministic search. Specifically, we employ the flexible and open-source B&B-based solver EAGO [52, 141]. The B&B algorithm iteratively partitions the search space into successively smaller subdomains and solves a sequence of lower- and upper-bounding subproblems on each subdomain. The algorithm converges in finitely-many iterations to an ϵ -optimal global solution or terminates with a certificate of infeasibility by comparing the obtained bounds across nodes. The upper-bounding problems typically determine a feasible local solution (if one exists) on each subdomain. The lower-bounding problems rely on the ability to calculate rigorous global bounds on all variables and functions involved in the optimization formulation. Calculating valid lower bounds for a global optimization problem is the most challenging procedure. This is especially true for PDE systems encountered in this work, as this task amounts to constructing rigorous bounds on the spatiotemporal state solutions over the entire domain of optimization variables (i.e., the reachable set).

In this section, we present a method for constructing global bounds enclosing the reach-

able sets of the tumor transport model. Several different bounding methods are presented and analyzed in this work to determine the most effective method for use with the tumor transport model. The fundamental approach is to use the method of lines with finite differences for spatial discretization and then differential inequalities (DI) [39, 40] to construct state bounds of the discretized large-scale ODE-IVP system. Note that apart from implementing interval arithmetic (IA) for constructing bounds, a mixed interval arithmetic/affine arithmetic (IA/AA) approach was also implemented [142]. In addition to standard DI, a modified DI approach with interval refinement operators[42] was also implemented for problems with prescribed bounding information known *a priori*.

Significant nonlinearity of the models poses a major challenge to efficiently constructing tight bounds. In the tumor transport model, a problematic term that requires special consideration is the solute source term that describes the transvascular mass transport of nanocarriers:

$$\phi_s = L_p \frac{S}{V} (p_v - p)(1 - \sigma)c_v + P \frac{S}{V} (c_v - c) \frac{Pe}{e^{Pe} - 1}. \quad (4.2)$$

Here, p_v is the vascular pressure (mm Hg), p is the interstitial fluid pressure (IFP) (mm Hg), σ is the solute reflection coefficient, P is the vascular permeability of the solute through the vascular wall (cm/sec), c is the solute concentration in the interstitial space of the tumor (g/mL), $Pe = L_p(p_v - p)(1 - \sigma)/P$ is the Péclet number representing the ratio of the rates of convection to diffusion across the vascular wall.

The solute source term suffers from the *dependency problem* of IA (i.e., the overestimation of interval operations due to the same variables being treated independently). The

nonlinearity caused by the exponential terms significantly magnifies this overestimation. We overcome the dependency problem using the following strategy. Since Pe appears in both the numerator and denominator of the $\frac{Pe}{e^{Pe}-1}$ term in (4.2), without special consideration, the dependency problem will lead to an appreciable overestimation of the bounds that will be detrimental to the B&B procedure. To avoid this, we consider the function $z(x) = \frac{x}{e^x-1}$, where we seek a real interval $Z = [z^L, z^U]$ such that $z(x) \in Z$ for every $x \in [Pe^L, Pe^U]$, for known values Pe^L and Pe^U . It is easy to prove that z is a monotonically decreasing function of x , and therefore, the exact bounds on the range of z on the domain $[Pe^L, Pe^U]$, can be derived as:

$$z^L = \frac{Pe^U}{e^{Pe^U} - 1},$$

$$z^U = \frac{Pe^L}{e^{Pe^L} - 1}.$$

The definitions of these exact bounds are used throughout this work.

Bounds on the state variables of the tumor transport model were constructed based on four approaches. The spatial domain was discretized into $N = 100$ nodes and the discrete-time DI scheme [51] was used to construct the bounds through the simulation time (5 min) with 21 time steps. The two physiological parameters are considered as decision variables and bounded by an interval domain $\boldsymbol{\pi} = (L_p, K) \in \Pi = [7.5 \times 10^{-7}, 7.6 \times 10^{-7}] \times [1.15 \times 10^{-6}, 1.2 \times 10^{-6}]$. The numerical solutions and bounding results are illustrated in Figure 4.2 for the four bounding methods considered: IA and DI, IA and DI with interval refinement, IA/AA and DI, and IA/AA and DI with interval refinement.

To compare the effectiveness of different bounding procedures, the volumes between the

upper and lower bounds on the dimensionless concentration over the entire spatial and time domain are quantified in Table 4.2 with time costs summarized in Table 4.2. It is observed that the time costs for pure IA and mixed IA/AA methods are almost the same, but the mixed IA/AA method can provide much tighter bounds. If taking the prescribed physical bounds $\hat{c} \in G = [0, 5]$ into account with the modified DI method, both pure IA and mixed IA/AA methods can enhance the bounding results. However, the increased computational costs are nearly 2 orders-of-magnitude more than standard DI due to the curse of the dimensionality of the discretized systems. The dramatic burden in time cost using the modified DI method overshadows any improvement of the bounding results in this case. As indicated by the volumes in Table 4.2, the bounds constructed by mixed IA/AA and standard DI methods are already relatively efficient (91.3 % tighter than the IA method, 62.4 % tighter than the IA (DI with G) method, and only 37.6 % larger than the IA/AA (DI with G) method), and the modified DI will not contribute much to reducing the conservatism. Therefore, in this study, we propose to use the mixed IA/AA and standard DI method as the bounding routine for solving all global optimization problems.

Table 4.2: Volume and time cost for different bounding methods are reported in this table

Bounding Methods	IA	IA/AA	IA (DI with G)	IA/AA (DI with G)
Volume	0.6438	0.0560	0.1489	0.0407
Time (s)	0.5342	0.5312	47.21	47.67

4.2.3 Machine Learning Model

Machine learning regression is proposed to establish a computationally efficient artificial neural network (ANN) as a surrogate for the mechanistic tumor transport model. The es-

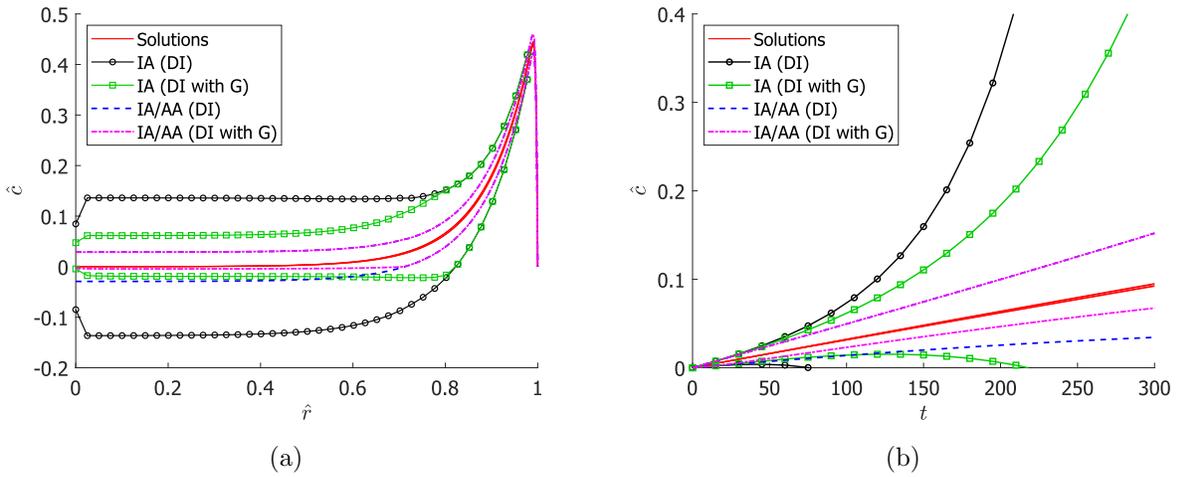


Figure 4.2: Numerical solutions and bounding results for the tumor transport model are plotted. (a) The spatial profiles of the dimensionless concentration \hat{c} in the tumor at $t = 150$ s are plotted for several values of π along with the state bounds derived from pure IA, IA with modified DI, mixed IA/AA, and mixed IA/AA with modified DI. (b) The trajectories of the solute concentration \hat{c} in the tumor at the position $\hat{r} = 0.5$ are plotted for several values of π along with the state bounds derived from pure IA, IA with modified DI, mixed IA/AA, and mixed IA/AA with modified DI. \hat{c} is approximated by corresponding numerical solutions calculated by the explicit Euler method and state bounds are calculated by the discrete-time DI method.

established ANN models will then be used to solve the model validation parameter estimation problems as formulated in (4.1). This approach is proposed to analyze the relative performance and accuracy of ANN models to assess their applicability within the proposed framework for drug and therapy design, as well as broader contexts of scientific machine learning in cancer research and therapy. The work in this subsection was implemented in Julia[53] 1.6.1 running on an Intel Xeon W-2195 (18-core/32-thread) 2.3 GHz/4.3 GHz (base/turbo) CPU with 64GB RAM running Windows 10 Pro. The inputs for the ANNs considered are the two physiological parameters L_p and K , discussed previously. Since DEX treatment normalizes the TME and, in turn, affects the transport phenomena in tumors, different ANN surrogate models were constructed to represent the control and DEX treated tumors for greater accuracy. Furthermore, since the experimental data varied slightly across the 70 kDa nanocarrier and 500 kDa nanocarrier experimental groups, separate ANN surrogate models were also constructed for greater accuracy within these mouse groups. Thus, four distinct ANN surrogates are considered: 70 kDa nanocarrier control case, 70 kDa nanocarrier 3 mg/kg and 30 mg/kg DEX treatment cases, 500 kDa nanocarrier control case, and 500 kDa nanocarrier 3mg/kg and 30 mg/kg DEX treatment cases. For each case, the tumor transport model was parameterized by L_p and K . The discretized fluid and solute transport models were solved using the method of lines via the stiff QNDF solver in DifferentialEquations.jl [143] for data acquisition. Then, the spatially-averaged concentrations over a discrete time horizon of 5 minutes were taken as outputs.

A Sobol[144] sequence sampling protocol in Surrogates.jl [145] was used to generate a data set of 10^6 points within the bounds described in Table A.9. Then, the data was scaled using min-max normalization and randomly divided into a training set (70%) and test set

(30%). The ANN models were trained and constructed using Flux.jl [146, 147]. Architectures of 2-4 hidden layers, 16-32 nodes per hidden layer, and several different activation functions (sigmoid, tanh, gelu, and swish) were considered. Through tuning and comparison, a two-hidden-layer model with 24 neurons each with the swish activation function was chosen for use in this work. This ANN model is depicted in Figure 4.3.

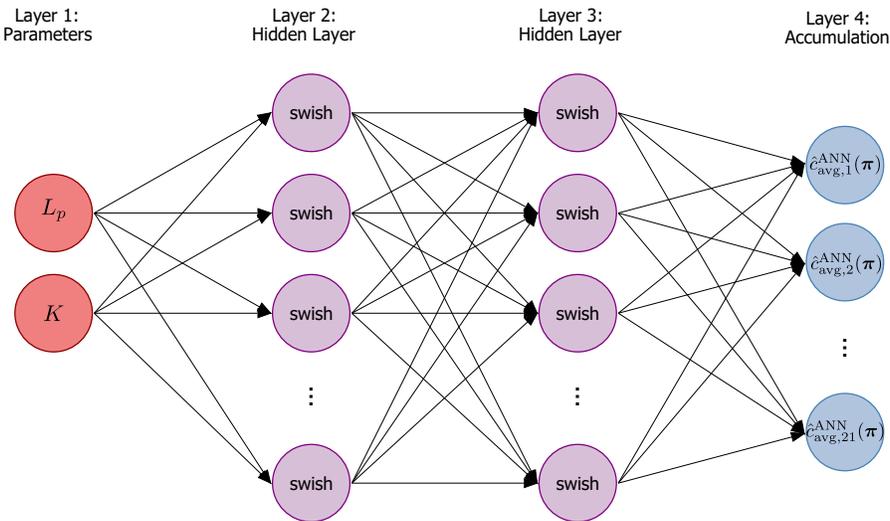


Figure 4.3: A fully connected feed-forward multilayer perceptron artificial neural network surrogate model is illustrated and represents the model architecture used for the simplified parameter estimation problems considered in this. The two node input layer (Layer 1) takes as input the physiological parameters L_p and K . These inputs feed to the two hidden layers using 24 nodes and the swish activation function. The outputs of the second hidden layer (Layer 3) are then passed to the output layer (Layer 4) consisting of 21 nodes, representing the temporally discretized accumulation profile.

Due to the relatively small size of the ANNs, the models were trained using a combination of batch and mini-batch gradient descent with a mini-batch size of 10% of the training data set. The Adam optimizer was used for training with the standard mean-squared-error (MSE) loss function. The model was trained for 50 epochs using an early stopping criteria, with a MSE tolerance of 10^{-7} . The learning rate was kept constant at 10^{-3} . Following training, the MSE and mean relative percent error were evaluated on the test set. This training protocol

was found to be effective as indicated by the time and performance metrics listed in Table A.10.

In previous studies, recurrent neural network model architectures are utilized as a typical method to simulate dynamical systems by directly approximating the numerical integration function as opposed to the entire numerical integration procedure [148]. This method was not employed in this study as it would necessitate an iterative loop in the objective function (due to the feedback of information of earlier-time states) to create the concentration profile for each function evaluation. Such a process would introduce additional complexity that would negatively impact the solution times when included in deterministic global optimization routines used in this work.

4.2.4 Simplified Parameter Estimation Problem

In this section, a simplified parameter estimation problem is proposed using ANN surrogate models introduced in Section 4.2.3. Similar to (4.1), we seek to minimize the SSE between the average concentration predicted by the ANN surrogate model and experimental data over the entire time horizon, with inequality constraints on superficial IFP:

$$\begin{aligned} \min_{\boldsymbol{\pi} \in \Pi} \quad & \sum_{i=1}^n (\hat{c}_{\text{avg},i}^{\text{ANN}}(\boldsymbol{\pi}) - \hat{c}_{\text{avg}}^{\text{data}}(t_i))^2 \\ \text{s.t.} \quad & \hat{p}_{\text{peri}}(\boldsymbol{\pi}) \leq \hat{p}_{\text{peri},\text{max}} \\ & \hat{p}_{\text{peri}}(\boldsymbol{\pi}) \geq \hat{p}_{\text{peri},\text{min}}, \end{aligned} \tag{4.3}$$

where $\hat{c}_{\text{avg},i}^{\text{ANN}}$ represents the dimensionless spatial average nanocarrier concentration at discrete time node i calculated from the ANN model. The inequality constraints on the superficial IFP may be simplified and reformulated as equivalent inequalities that are linear in the optimization variables (model parameters) L_p and K utilizing the closed-form analytical solution for the IFP profile from Baxter and Jain [129]. This simplifies the problem significantly and, in turn, reduces the computational complexity of solving (4.3). The details of how this is done can be found in Appendix A.2.2.

The optimization formulation (4.3) can then be reformulated as:

$$\begin{aligned} \min_{\boldsymbol{\pi} \in \Pi} \sum_{i=1}^n (\hat{c}_{\text{avg},i}^{\text{ANN}}(\boldsymbol{\pi}) - \hat{c}_{\text{avg}}^{\text{data}}(t_i))^2 & \quad (4.4) \\ \text{s.t. } \pi_2 \leq \zeta_{\text{max}} \pi_1 & \\ \pi_2 \geq \zeta_{\text{min}} \pi_1, & \end{aligned}$$

where ζ_{max} and ζ_{min} are listed in Table 4.3 and are calculated based on the physical bounds on the superficial IFP listed in Table 4.1. The calculation procedure is described in Appendix A.2.2.

Table 4.3: The coefficients for the constraints on the superficial (peripheral) IFP of control, 3 mg/kg, and 30 mg/kg DEX treatment case in formulation (4.3) are tabulated.

Dose	Control	3 mg/kg	30 mg/kg
ζ_{min} (cm)	0.2855	0.7355	1.5898
ζ_{max} (cm)	0.3967	1.0577	2.4447

4.2.5 TME-Normalizing Therapy Design for Dose Selection

In this section, we propose a method for optimal TME-normalizing therapy design for dose selection with the overall objective of improving transport and accumulation of anticancer drugs within the tumor interstitium. To do so, we investigate the experimental effects of different doses of pretreatment DEX, and utilize empirical correlations for optimal decision-making. Empirical correlations are required to construct a mathematical relationship between DEX dose and two important physiological parameters: vascular hydraulic conductivity L_p and interstitial hydraulic conductivity K . The purpose of this study is to propose a systematical mathematical methodology for TME-normalizing therapy design.

Based on the preclinical data obtained from Martin et al. [6], we utilize nonlinear regression with a rational model to establish the following relationships:

$$f_{L_p}^r(x) = \frac{-7.519 \times 10^{-8}x^2 + 3.355 \times 10^{-6}x + 6.944 \times 10^{-7}}{x + 0.6175}, \quad (4.5)$$

$$f_K^r(x) = \frac{-2.458 \times 10^{-8}x^2 + 2.524 \times 10^{-6}x + 2.916 \times 10^{-7}}{x + 0.7816}, \quad (4.6)$$

where x denotes pretreatment DEX dose (mg/kg), and the functions $f_{L_p}^r$ and f_K^r represent the values of L_p and K , respectively, following treatment with DEX, as predicted by the rational regression model.

Since the data obtained from Martin et al. [6] are limited to the three pretreatment DEX doses, we also wish to explore different dose-dependent relationships that could exist with other data sets. The purpose of this is to demonstrate the applicability of our proposed method with fictitious experimental data exhibiting complicated dose-dependent treatment

relationships for pretreatment DEX doses between the actual data of 3 mg/kg and 30 mg/kg with dosages set at 10 mg/kg, 15 mg/kg, 20 mg/kg and 25 mg/kg.

The original data, fictitious data, and corresponding polynomial regression models are plotted in Figure 4.4. The regression equations are given by:

$$f_{L_p}^p(x) = -6.23 \times 10^{-13}x^5 - 5.96 \times 10^{-11}x^4 + 5.61 \times 10^{-9}x^3 - 1.272 \times 10^{-7}x^2 + 8.797 \times 10^{-7}x + 1.131 \times 10^{-6}, \quad (4.7)$$

$$f_K^p(x) = 1.139 \times 10^{-11}x^5 - 8.389 \times 10^{-10}x^4 + 2.183 \times 10^{-8}x^3 - 2.421 \times 10^{-7}x^2 + 1.093 \times 10^{-6}x + 3.798 \times 10^{-7}, \quad (4.8)$$

where x is the DEX dose as before, and $f_{L_p}^p$ and f_K^p represent the values of L_p and K , respectively, following treatment with DEX, as predicted by the polynomial regression model.

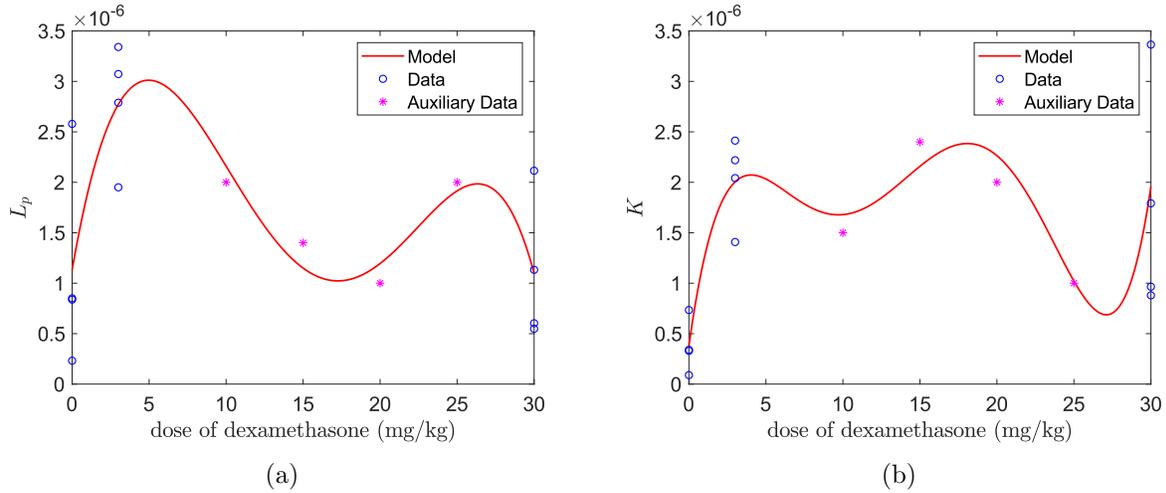


Figure 4.4: Experimental data and corresponding regression models for (4.7) and (4.8) are respectively plotted in (a) L_p versus dose; (b) K versus dose of dexamethasone.

The TME-normalizing therapy design problem is formulated as the following NLP:

$$\max_{x \in X} \hat{c}_{\text{avg}} \left(t_f, \left(f_{L_p}^j(x), f_K^j(x) \right), d_m \right), \quad (4.9)$$

with $j \in \{r, p\}$. The objective is to seek an optimal dose that maximizes the spatial average nanocarrier concentration \hat{c}_{avg} in the tumor interstitium at $t_f = 5$ min. The function \hat{c}_{avg} is evaluated by the numerical solution of the solute transport model, and the correlations between hydraulic conductivities and DEX doses are established as (4.5) and (4.6) for the existing data, and (4.7) and (4.8) for the fictitious data.

4.2.6 Drug Size Design

In this section, the practicability of the tumor transport model for drug size design problems is addressed. After the optimal dose of pretreatment DEX is determined and a patient's response to that treatment is quantified, an anticancer nanocarrier is designed that results in an optimal delivery to the tumor interstitium. For example, a nanoparticle size can be *tuned* for a patient-specific tumor pathophysiology.

There are two physiological parameters directly related to nanocarrier size d_m : diffusion coefficient D and half-life circulation time k_d . The previous experimental results for their correlations are listed in Table A.11. Nonlinear regression models are established (power

model for D versus d_m ; Gaussian model for k_d versus d_m) for these quantities as:

$$f_D(d_m) = 1.981 \times 10^{-6} \cdot d_m^{-1.157} + 2.221 \times 10^{-8},$$

$$f_{k_d}(d_m) = 1081 \exp\left(-\left(\frac{d_m + 16.63}{84.82}\right)^2\right) + 517.4 \exp\left(-\left(\frac{d_m - 65.61}{996.6}\right)^2\right),$$

where f_D and f_{k_d} represent the values of D and k_d , respectively.

After DEX pretreatment, it is desirable to determine an optimal nanocarrier size that can maximize the drug concentration in the interstitial space of the tumor. Alteration in pharmacokinetics, such as distribution and excretion, can have a substantial influence on achieving the desired therapeutic concentration of a particular nanocarrier. A very high concentration may result in side effects or toxicity. A very low concentration will be ineffective. In this situation, an optimal therapy requires a strict guarantee of some safety/performance specifications. The drug size design problem is formulated as a PDE-constrained NLP to account for these potential requirements:

$$\begin{aligned} & \max_{d_m \in Z} \hat{c}_{\text{avg}}(t_f, \boldsymbol{\pi}, d_m) & (4.10) \\ \text{s.t. } & \hat{c}_{\text{peri}}(t_f, \boldsymbol{\pi}, d_m) \leq \lambda_1 \\ & \hat{c}_{\text{peri}}(t_f, \boldsymbol{\pi}, d_m) \geq \lambda_2, \end{aligned}$$

where t_f is the final time ($t_f = 5$ min), \hat{c}_{peri} is the dimensionless nanocarrier concentration in the superficial area of tumor (g/mL). For this work, we use $\lambda_1 = 4.5$ as the threshold for the safety constraint, which is double the periphery nanocarrier concentration for the 3 mg/kg DEX treatment case. Further, we use $\lambda_2 = 3.6$ as the performance constraint, which

is chosen based on periphery nanocarrier concentration for the control case. Note that these thresholds are merely chosen for demonstrating the drug size design approach and how to deal with the situation that a design is implemented under potential performance/safety requirements.

Furthermore, we propose a therapy design strategy that simultaneously seeks an optimal dose of DEX and an optimal nanocarrier size that maximizes the nanocarrier concentration accumulation inside the tumor interstitial space:

$$\begin{aligned}
& \max_{x \in X, d_m \in Z} \hat{c}_{\text{avg}} \left(t_f, \left(f_{L_p}^j(x), f_K^j(x) \right), d_m \right) & (4.11) \\
\text{s.t. } & \hat{c}_{\text{peri}} \left(t_f, \left(f_{L_p}^j(x), f_K^j(x) \right), d_m \right) \leq \lambda_1 \\
& \hat{c}_{\text{peri}} \left(t_f, \left(f_{L_p}^j(x), f_K^j(x) \right), d_m \right) \geq \lambda_2,
\end{aligned}$$

with $j \in \{r, p\}$. This formulation provides an alternative methodology for neoadjuvant therapy that could identify a possible therapy and nanocarrier size combination that leads to improved transport and accumulation over the individual results determined by the sequential design approach.

ANN surrogate models are proposed for the simultaneous design problem (4.11) to reduce the computational burden over the PDE-constrained problem. To accomplish this, two ANNs are established each with L_p , K , and d_m as inputs. The respective ANNs each have a single output $\hat{c}_{\text{avg}}^{\text{ANN}}$ and $\hat{c}_{\text{peri}}^{\text{ANN}}$. To train the ANNs, a Sobol[144] sequence sampling method was again used to create a 10^6 point data set on the domain $(L_p, K, d_m) \in [5 \times 10^{-7}, 5 \times 10^{-6}] \times [5 \times 10^{-7}, 5 \times 10^{-6}] \times [10, 60]$. Consistent with the parameter estimation of ANN models, the data set was scaled using a min-max normalization and divided randomly into training

(70%) and validation (30%) sets. Training was performed using Flux.jl with the Adam optimizer with a learning rate of 10^{-4} . Each ANN model for \hat{c}_{avg}^{ANN} and \hat{c}_{peri}^{ANN} have two hidden layers with 12 neurons using the tanh activation function. The models were trained using an equivalent protocol to that described in Section 4.2.3. The benchmarks for data generation, training time, and performance are shown in Table A.12.

The formulation with ANN models for the simultaneous design approach can be expressed as:

$$\begin{aligned} & \max_{x \in X, d_m \in Z} \hat{c}_{avg}^{ANN} \left(\left(f_{L_p}^j(x), f_K^j(x) \right), d_m \right) & (4.12) \\ \text{s.t. } & \hat{c}_{peri}^{ANN} \left(\left(f_{L_p}^j(x), f_K^j(x) \right), d_m \right) \leq \lambda_1 \\ & \hat{c}_{peri}^{ANN} \left(\left(f_{L_p}^j(x), f_K^j(x) \right), d_m \right) \geq \lambda_2. \end{aligned}$$

4.2.7 Settings for Solving Optimization Problems

The settings used in this study for the numerical methods and software packages are discussed in this section. For the parameter estimation, TME-normalizing therapy design, and drug size design problems, the spatial domain for both fluid transport and solute transport models are discretized into $N = 100$ nodes. The simulation time horizon contains 21 time nodes (5 min). Based on the superficial region (around 0.07 mm from surface) [149] and the average tumor diameter (0.6 - 1.1 cm) in the DEX treatment research [6], we choose $n = 99$ to account for the superficial region of the tumor. The physiological parameters used in the tumor transport model are provided in Table A.13. The parameter estimation, drug size design, and TME-normalizing therapy design problems are all solved to global optimality

using the EAGO[52] v0.6.1 solver via JuMP v0.21.4 [96] in the Julia programming language [53]. Custom bounding routines with the mixed IA/AA method and standard DI are utilized in the B&B algorithm for solving the parameter estimation and drug size design problems. For the parameter estimation problems, the absolute global convergence tolerance is set as 10^{-6} and the relative global convergence tolerance is set as 10^{-1} for each case. For the drug size design and TME-normalizing therapy design problems, the absolute convergence tolerance is set as 10^{-6} , and the relative convergence tolerance set as 10^{-2} . Each problem was run on a personal workstation with an Intel Xeon E3-1270v5 4-core/8-thread CPU operating at 3.60GHz/4.00GHz (base/turbo) frequency and 32GB ECC RAM running Windows 10 Version 2004.

4.3 Results and Discussion

4.3.1 Global Optimization Results for Model Validation

In this section, the results for model validation using global optimization that verifies TME-normalization process by parameter estimation problems are discussed. Though, global optimization method is far more difficult than finding a local solution, this approach provides the strongest guarantee for preventing erroneous models that invalidates proposed mechanisms in cases where poor suboptimal fits are obtained by local optimization algorithms. The global optimal solutions obtained from the parameter estimation problems for different doses of DEX treatment cases are listed in Table 4.4 for each formulation with the original mechanistic tumor transport model (4.1) as well as the ANN surrogate model (4.4). Note

that the dose selection for the experiments was based on previous work, which confirmed 3 mg/kg DEX as the lowest dose to reduce IFP. Additionally, this dose is similar to that used in the clinical trials of CDDP/m (NCT02043288) [150]. The global solutions found for both the mechanistic model and ANN model are very close to one another, with the relative error for each case being within 2.5%. This certifies the accuracy of the ANN surrogate models and the validity of the inequality constraints simplifications. In Martin et al. [6], we obtained local optima for the parameter estimation problems. In that work, it was found that the estimated L_p value for 3 mg/kg DEX treatment case with 500 kDa nanocarrier injection exhibited a decreasing trend from the control case, whereas in this study we found an increasing trend. This doesn't represent a contradiction as the parameter estimation problems differ significantly in that they consider differing simulation time horizons. Additionally, in the case of Martin et al. [6], no inequality constraints on the IFP were considered.

Table 4.4: Global optima for parameter estimation problems using the mechanistic model (4.1) and the ANN model (4.4) are tabulated here. It is noted that the solutions obtained for the ANN surrogate model are very close to those obtained for the mechanistic model. This is to be expected since a high-degree of accuracy of the ANN was obtained when training. The units for L_p^* is cm/mm Hg-sec and for K^* is cm²/mm Hg-sec.

Dextran molecular weight		70 kDa		
Dose	Control	3 mg/kg	30 mg/kg	
P_{eff} (cm/sec)	9.60×10^{-7}	4.61×10^{-6}	2.80×10^{-6}	
L_p^* - mechanistic model	8.51×10^{-7}	2.80×10^{-6}	1.12×10^{-6}	
L_p^* - ANN model	8.39×10^{-7}	2.77×10^{-6}	1.12×10^{-6}	
K^* - mechanistic model	3.35×10^{-7}	2.03×10^{-6}	1.80×10^{-6}	
K^* - ANN model	3.32×10^{-7}	2.04×10^{-6}	1.78×10^{-6}	
Dextran molecular weight		500 kDa		
Dose	Control	3 mg/kg	30 mg/kg	
P_{eff} (cm/sec)	8.18×10^{-7}	4.30×10^{-6}	1.62×10^{-6}	
L_p^* - mechanistic model	8.62×10^{-7}	2.22×10^{-6}	7.50×10^{-7}	
L_p^* - ANN model	8.43×10^{-7}	2.22×10^{-6}	7.50×10^{-7}	
K^* - mechanistic model	3.34×10^{-7}	2.34×10^{-6}	1.21×10^{-6}	
K^* - ANN model	3.29×10^{-7}	2.36×10^{-6}	1.20×10^{-6}	

The time costs for each model are reported in Table 4.5. For DEX treatment cases, the parameter estimation problems with the mechanistic model and the proposed customized bounding routines are extremely computationally expensive. Despite using the most efficient global bounding method considered, these problems still required hours or even days to solve. In contrast, the parameter estimation problems for DEX treatment cases using the ANN surrogate models can be solved within one minute. Even accounting for the time costs of generating data and training, the ANN surrogate models significantly reduce the burden of solving the parameter estimation problems to global optimality. Interestingly, it takes about an order-of-magnitude longer to solve the parameter estimation problems for the control cases with ANN models versus the mechanistic models. In these cases, it is observed that the lower-bounding problems solved for the ANN problems furnish weaker bounds than for the mechanistic modeling case resulting in slower convergence of the B&B algorithm.

Table 4.5: The computational time costs are tabulated for the parameter estimation problems using the mechanistic model (4.1) and the ANN model (4.4). Barring the control case, solving the PDE-constrained optimization problem (4.1) requires significantly more time than the the problem with the ANN (4.4), which does not account for the ANN training time.

Dextran molecular weight	70 kDa			500 kDa		
Dose	Control	3 mg/kg	30 mg/kg	Control	3 mg/kg	30 mg/kg
Mechanistic model (s)	8.5	169558.3	238732.2	8.1	398792.2	50368.1
ANN model (s)	97.9	7.3	25.8	23.2	17.6	18.8

Interstitial Fluid Pressure and Velocity Profiles

Previous studies showed that an important barrier to drug delivery in the TME is the elevated IFP resulting in reduced pressure gradients across the vessel wall [128]. This is due to the interstitial hypertension phenomenon [123] caused by leaky blood vessels and the lack of functional lymphatic vessels, which drain excess fluid from tumor tissue. TME-normalizing

therapy can repair the abnormal vasculature and reduce IFP, resulting in a higher pressure gradient for higher transvascular and interstitial fluid flow. Thus, we quantified the IFP with different doses of DEX treatment to characterize the TME normalization process. The dimensionless IFP profiles as functions of dimensionless radial position with respect the optimal solutions (i.e., from Section 4.3.1) are illustrated in Figure 4.5(a). The IFP profiles tend to reach a steady-state pressure p_{ss} at the tumor center where IFP equals the vascular pressure p_v . However, in the periphery, the IFP rapidly decreases with increasing distance from the tumor center. This finding is consistent with previous mathematical models [129] and experimental findings [128]. Thus, the IFP profiles indicate that the extravasation of fluid from blood vessels is extremely slow near the center, whereas it is highest at the periphery due to lower IFP leading to an increased transvascular pressure gradient. In addition, the model confirms that DEX reduces the spatially-averaged IFP and therefore establishes a more advantageous transvascular pressure gradient that contributes to enhanced transvascular fluid flow [6], that will further affect the interstitial fluid transport.

The interstitial fluid velocity (IFV) is generated from the interior IFP gradient by Darcy's law (introduced in Appendix A.2.1). To investigate the effects of TME normalization on interstitial fluid transport, we quantified the normalized IFV ($\hat{u} = uR/(K(p_{ss} - p_\infty))$) profiles for different doses of DEX. A positive value of IFV indicates that the interstitial fluid flow is from the center to the periphery of the tumor. As illustrated in Figure 4.5(b), we observed that the normalized IFV is very low around the center and increases towards the periphery where there is the highest flow rate. The dimensionless parameter $\alpha = R\sqrt{L_p S/KV}$ (introduced in Appendix A.2.1), which is a measure of the ratio of interstitial to vascular resistances of fluid flow, represents the gradient of increase of normalized IFV. Summarily, a

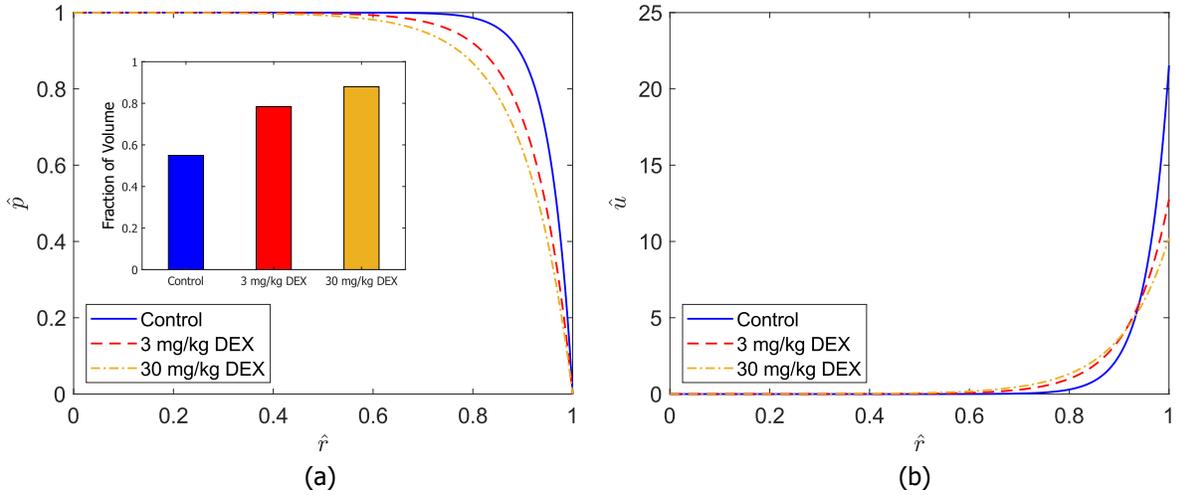


Figure 4.5: Radial dose-dependent interstitial fluid pressure and velocity profiles. (a) Mathematical model-generated profiles of dimensionless interstitial fluid pressure (IFP) $\hat{p} = (p - p_\infty)/(p_{ss} - p_\infty)$ versus the dimensionless tumor radial position \hat{r} from vessel permeability data collected using fluorescently-labelled 500 kDa dextran in control tumors and tumors in mice treated with 3 mg/kg and 30 mg/kg dexamethasone (DEX) daily for four days are presented. Spatially-averaged IFP is reduced with DEX treatment. The interior bar graph illustrates the fraction of tumor volume that has a favorable transvascular pressure gradient (i.e. $\hat{p} \leq 0.9933$). This IFP threshold is determined by the region with $\hat{r} \geq 0.6$ for the 3 mg/kg DEX treatment case, which is taken as the volume with favorable transvascular pressure gradient. (b) Normalized interstitial fluid velocities (IFV) $\hat{u} = uR/(K(p_{ss} - p_\infty))$ are plotted versus dimensionless tumor radial position \hat{r} . Greater IFVs are achieved deeper in the tumor interstitium following DEX treatment with a reduction in velocity nearest the tumor periphery. This results in increased interstitial transport of nanocarriers.

larger value for α indicates a steeper increase in the normalized IFV profile with increasing distance from tumor center. The model-predicted α values for the control, 3 mg/kg DEX treatment, and 30 mg/kg DEX treatment cases from the 500 kDa dextran experimental data are 22.521, 13.756 and 11.219, respectively. Thus, compared to the control, the treated cases have smaller values of α that indicate a gradual increase in normalized IFV from the center tumor over a larger fraction of tumor volume. Note that the normalized IFV neglects the influence of interstitial hydraulic conductivity K . However, K is larger by an order of magnitude for DEX treated cases than the control case (Table 4.4). Thus, the actual IFV for

DEX treated cases is always higher than the control case. Although, we reported in Martin et al. [6] that DEX treatment increases the perfused vascular density, we assumed in the current study that the tumor radius R and vascular density S/V do not vary significantly between each case. Thus, a reduction in the ratio of the vascular hydraulic conductivity to the interstitial hydraulic conductivity (i.e., L_p/K) is the major reason for a reduction in α . A smaller value of L_p/K indicates a larger proportion of interstitial fluid transport. Therefore, a less steep normalized IFV profile resulting from a smaller α caused by a reduction of L_p/K implies enhanced interstitial fluid transport by vascular and ECM normalization.

Solute Concentration Profiles

We next sought to determine the drug distribution within tumors by obtaining solute concentration profiles from the IFP and IFV profiles. The IFP gradient induces transvascular convective transport, the IFV profiles reflect interstitial convective transport, and the solute concentration gradient induces interstitial diffusive transport. Figure 4.6 illustrates the model-predicted solute concentration profiles with respect to dimensionless tumor radial position \hat{r} for the 500 kDa dextran experimental data, with the vascular concentration following an exponential decay post-administration.

As illustrated in Figure 4.6(a), the interstitial concentration at 1 h post-administration of the dextran is equal to the normal tissue concentration (equals 0 in dimensionless form) at the periphery and quickly increases to a peak in the peripheral region where there is a higher transvascular pressure gradient, which significantly enhances transcapillary convective solute transfer. The fraction of tumor volume that has a higher transvascular pressure gradient is graphed for each treatment group in the inset of Figure 4.5(a). Simultaneously, the higher

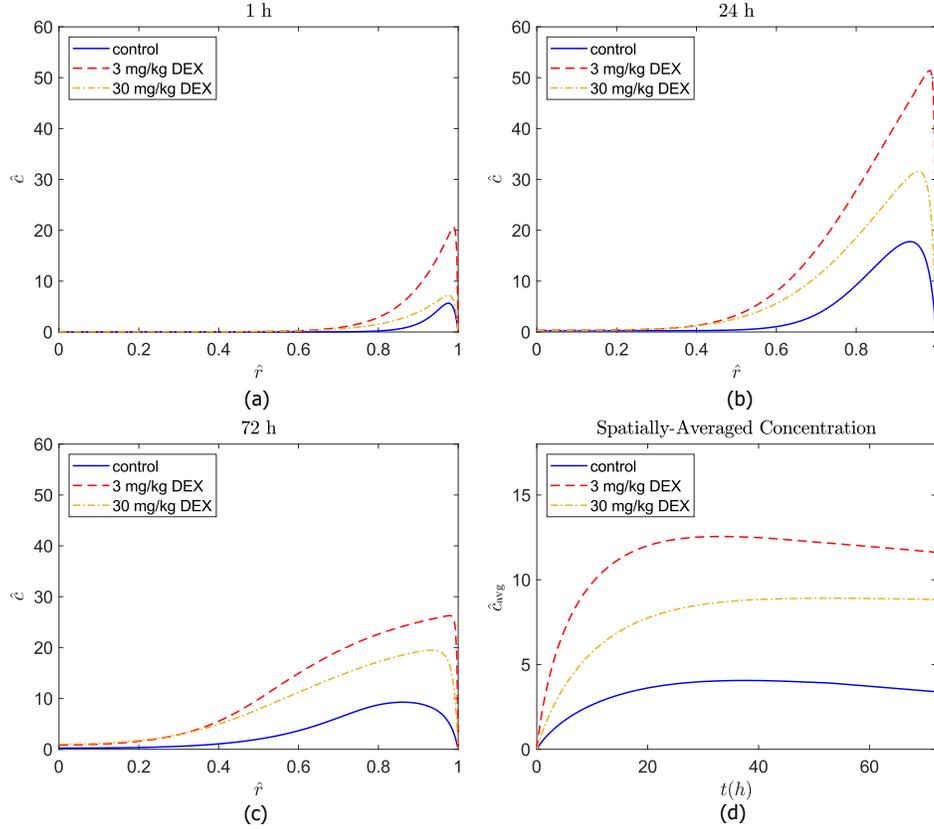


Figure 4.6: Radial and temporal dose-dependent solute concentration profiles. Interstitial concentrations \hat{c} of 500 kDa dextran are plotted versus dimensionless tumor radial position \hat{r} for a vascular concentration with a half-life of around 21 h for the control, 3 mg/kg DEX treatment, and 30 mg/kg DEX treatment cases at (a) 1 h; (b) 24 h; (c) 72 h post-administration. (d) Spatially-averaged interstitial concentrations \hat{c}_{avg} are plotted versus time. After DEX treatment, the overall solute concentration accumulation is increased inside the tumor. The 3 mg/kg DEX treatment case results in the highest overall concentration accumulation.

IFV in the peripheral region causes a higher interstitial fluid flux that carries the solute outwards to the periphery. As a result, the solutes accumulate and reach peak concentration near the periphery, then decrease to zero at around $\hat{r} = 0.8$ for the control case and $\hat{r} = 0.6$ for the 3 mg/kg and 30 mg/kg DEX treatment cases. Indeed, the region with favorable transvascular pressure gradient for DEX treated cases is larger than the control case (Figure 4.5(a)). This pressure gradient leads to an enhanced convective transvascular transport that carries solutes into the interstitial space of a larger proportion of the tumor. In other words,

the region with higher solute accumulation occurs over a longer fraction of tumor radius for the DEX treatment cases compared to control.

As presented in Figure 4.6(b), the interstitial concentration profiles for all treatment cases have higher peaks at 24 h than 1 h. The concentration peaks for all cases at 72 h (Figure 4.6(c)) are lower than 24 h but higher than 1 h. This is because the vascular concentration decays at 72 h compared with 24 h so that there are fewer nanocarriers to be carried by transvascular flow into the interstitial space. In addition, we found that the interstitial concentration profiles at 72 h become flatter than 24 h with a higher concentration retained towards the middle of the tumors, such as at $\hat{r} = 0.5$. This is caused by the slower interstitial diffusion generated from the concentration gradient that gradually transfers nanocarriers from the concentration peak in the periphery towards the tumor center, where the concentration of nanocarriers is near zero. The transvascular flow is limited at 72 h due to the systemic clearance of circulating nanocarriers, but the diffusion caused by concentration gradient becomes more evident in the flatter concentration profiles.

As illustrated in Figure 4.6(d), the spatially-averaged interstitial concentration rises to a peak and stays steady after that. Although the vascular concentration of nanocarriers decays exponentially, the spatially-averaged interstitial concentrations decrease slowly after reaching the peak. The concentration profiles at the time with respect to the highest spatially-averaged concentration accumulation are illustrated in Figure A.3. We observed that the highest spatially-averaged concentration occurs at 38.8 h, 34.2 h and 53.9 h for the control, 3 mg/kg and 30 mg/kg DEX treatment cases, respectively. In general, the nanocarriers accumulate to a peak concentration in the first dozens of hours and then decrease with a slow rate.

The spatially-averaged concentrations at 72 h are 84%, 92% and 99% of their highest concentrations for the control, 3 mg/kg and 30 mg/kg DEX treatment cases, respectively (illustrated in Figure A.4). We found that the spatially-averaged concentrations of the 500 kDa dextran in control tumors only decrease by 16% in 33.2 h after reaching highest concentration, indicating a retention effect. The 3 mg/kg and 30 mg/kg DEX treatments both enhance this retention effect (92% and 99% are higher than the control case). Though the 3 mg/kg DEX treatment does not result in the highest percentage of retention at 72 h (92% ; 99%), it has the highest spatially-averaged concentration throughout the whole time horizon. In contrast, the control case has the lowest percentage and also the lowest spatially-averaged concentration. Thus, these findings demonstrate that DEX treatment not only increases permeability [6] but also retention towards promoting the EPR effect.

We further investigated the relation between the solute concentration distribution over time and dose of DEX treatment. The concentration profile for the 30 mg/kg DEX treatment case is closer to the control case at 1 h post-administration, whereas it is closer to the 3 mg/kg DEX treatment case at 72 h post-administration. At 1 h post-administration, there are many nanocarriers in perfused vessels and they are carried into the tumor tissue by transvascular flow. A larger vascular hydraulic conductivity L_p indicates higher transvascular flow rate. However, L_p for 30 mg/kg DEX treatment case is closer to the control case (Table 4.4). Although the vessels are normalized after 30 mg/kg DEX treatment, there is too much pericyte coverage that reduces the vessel wall pore size [6] thereby limiting nanocarrier extravasation at 1 h post-administration. In contrast, the extravasation of nanocarriers is trivial at 72 h due to the decay of its concentration in the vasculature and the interstitial concentration profile has already reached a peak and decreased. Thus, the interstitial diffu-

sive transport becomes more prominent. Since both 3 mg/kg and 30 mg/kg DEX treatment similarly reduce hyaluronan levels and tissue stiffness [6], resulting in much larger interstitial hydraulic conductivity K than the control case (Table 4.4), the enhanced interstitial diffusive transport results in the observed profiles. In addition, we found that the 3 mg/kg DEX treatment case results in a much higher overall nanocarrier concentration accumulation in the tumor tissue than that of the control and the 30 mg/kg cases at all time nodes (1 h, 24 h, and 72 h), indicating increased delivery of anticancer nanocarriers leading to improved efficacy as demonstrated in Martin et al. [6]. Given that the 3 mg/kg DEX treatment lead to the highest nanocarrier accumulation, we next investigated the convective and diffusive transvascular fluxes separately to understand how DEX increased accumulation.

4.3.2 Dexamethasone Increases Convective Transvascular Flux in Tumors

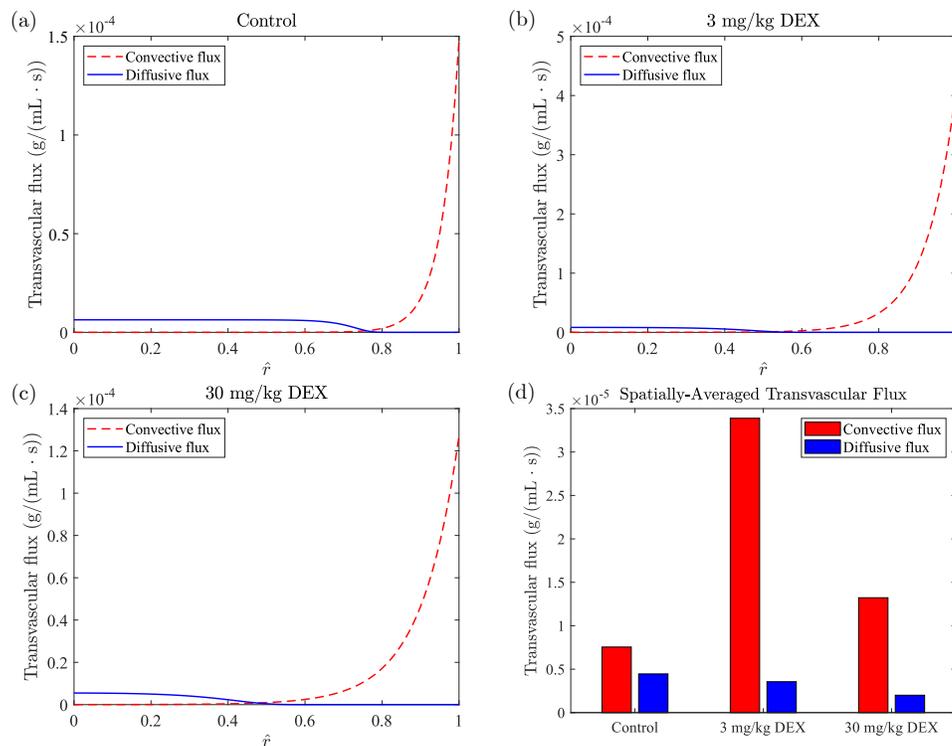


Figure 4.7: Dose-dependant transvascular convective and diffusive flux profiles. The transvascular flux profiles of 500 kDa dextran over the dimensionless tumor radial position \hat{r} one-hour post-administration are plotted for the (a) control; (b) 3 mg/kg dexamethasone (DEX) treatment; and (c) 30 mg/kg DEX treatment cases. (d) The spatially-averaged convective and diffusive fluxes at one-hour post-administration for different doses of DEX are presented in this bar plot. General trends show greatest convective flux at the tumor periphery and greatest diffusive flux deeper at the tumor center. Following DEX treatment, convection accounts for a greater proportion of the spatially-averaged transvascular fluxes, demonstrating how TME normalization induces a larger transvascular pressure gradient that is advantageous for improving nanocarrier delivery in tumors. The 3 mg/kg DEX treatment induces highest convective flux, indicating that a moderate dose of DEX is more advantageous for enhancing convective transport.

After finding the interstitial concentration profiles that dictate enhanced nanocarrier distribution and accumulation with DEX treatment, we hypothesized that the difference in concentrations between the control and DEX treatment cases depends on the relative con-

tributions of convective and diffusive transvascular flux. Previous studies indicated that the main mechanism of transvascular transport is diffusion because elevated IFP in the TME abrogated the transvascular pressure gradient [115]. We reasoned that because DEX reduces IFP it could enhance convective flux, which leads to more rapid transport than diffusive flux. However, the relative contributions from convection and diffusion throughout the entire space of a tumor have never been studied before, because recently developed global optimization techniques have not been applied to the parameter estimation problem to determine the hydraulic conductivity values. We first quantified the model-predicted transvascular convective and diffusive fluxes. As described in (4.2), the convective flux is calculated by $L_p \frac{S}{V} (p_v - p)(1 - \sigma)c_v$, and the diffusive flux is calculated by $P \frac{S}{V} (c_v - c) \frac{Pe}{e^{Pe} - 1}$. The relative contributions from convective and diffusive flux to the spatially-averaged concentration profile with time are illustrated in Figure S4, we found that the convective flux contribution for the DEX treatment case takes a larger proportion throughout the time horizon compared with the control case. This indicates that the normalized TME after DEX treatment is more advantageous for convective transport. To better understand the effects of TME normalization on transvascular transport, we sought to determine the spatial dependence of model-predicted diffusive and convective fluxes in tumors. In Martin et al. [6], we performed continuous intravital microscopy on mice for one hour post-administration and investigated nanocarrier microdistribution. Here, we quantified the spatial convective and diffusive fluxes at one hour post-administration to study their distribution in tumors. As shown in Figure 4.7(a), in the region with $\hat{r} < 0.8$, the convective flux is near zero while the diffusion is the main mode of transport. This is because IFP is close to the microvascular pressure (Figure 4.5(a)), indicating no driving force for convection. Diffusion, although

dominant, is small, so there is not much transvascular flux in the tumor center. In contrast, in the region with $\hat{r} > 0.8$, there is more convective than diffusive flux, with the latter being near-zero. This convective flux at the periphery is 22-fold greater than the diffusive flux at the center. The reason for this is that the IFP in the convection-dominated region is low (Figure 4.5(a)), inducing a high transvascular convective flux driven by a large pressure gradient. Accordingly, the convective transport increases interstitial concentration thereby lessening the concentration gradient and reducing the driving force for diffusion. In addition, the Pe , which represents the ratio of rates of transvascular convection to diffusion, is very large in the periphery, reflecting the extremely small diffusive flux. Thus, as observed in Figure 4.7(a), the diffusive flux in the periphery is near zero. As a result, we found that there is a convection-dominated region and a diffusion-dominated region and the maximum rate of convective flux is order of magnitude greater than the maximum diffusive flux.

We next investigated the effect of TME normalization on the spatial distribution of these fluxes. As illustrated in Figure 4.7(b), the maximum convective flux at the periphery for 3 mg/kg DEX is 48-fold greater than the maximum diffusive flux, which occurs in the tumor center. Since the the maximum diffusive flux is close to the control case, this indicates that convection is greatly enhanced and responsible for a larger proportion of total transvascular transport in the normalized TME after treatment with 3 mg/kg DEX. In contrast, in Figure 4.7(c), the maximum convective flux for 30 mg/kg DEX is 22-fold greater than the maximum diffusive flux, which is the comparable to that of the control case. In fact, by comparing the values of maximum convective flux at the periphery, we found that the flux for 30 mg/kg DEX is 14.3% less than the control case. The reason is that the vascular hydraulic conductivity L_p for with 30 mg/kg DEX treatment is 13% less than the control case (Table 4.4). This

is because DEX normalizes the vessels, increases vessel maturity, and thereby reduces vessel leakiness. As a result, the vascular hydraulic conductivity reduces, leading to the lower maximum convective flux at the tumor periphery. However, these findings do not indicate that the convective flux for 30 mg/kg DEX reduces throughout the entire tumor compared to control. This is because we also found that the volume of convection-dominated region is much larger for DEX treatment cases. As illustrated in Figure 4.7(b) and Figure 4.7(c), the convective flux for both 3 and 30 mg/kg DEX treatment cases begins to increase at around $\hat{r} = 0.6$ versus $\hat{r} = 0.8$ for the control case, indicating a larger convection-dominated region. These findings are illustrated in Figure 4.8, which shows a schematic of cross sections of the tumors for the control case and 3 mg/kg DEX treatment case. We found that the tumor volume fraction of convection-dominated region for the control case is only 49%, whereas this jumps to 78% for a tumor treated with DEX. This represents a 61% increase in the volume fraction of the tumor that is dominated by convective transport as a result of TME normalization with DEX treatment. Note that the transvascular fluxes in Figure 4.8 are scaled based on the 70 kDa dextran, and the corresponding convective and diffusive flux profiles are presented in the Figure S5. The findings using the 70 kDa dextran to determine the convection- and diffusion-dominated regions are consistent with those using the 500 kDa dextran, as shown in Figure 4.7. Both doses almost equally increase volume of convection-dominated region, but the 3 mg/kg DEX is superior because it also significantly increases the maximum convective flux.

To quantify the contributions of convection and diffusion throughout the tumor, we assessed the spatially-averaged transvascular convective and diffusive fluxes and presented them in a bar plot as illustrated in Figure 4.7(d). We observed a 360% increase in convective

flux with 3 mg/kg DEX and a 80% increase with 30 mg/kg DEX compared to the control case. This indicates that DEX dose has a significant impact on convective transport. It turns out that a moderate dose of DEX greatly enhances convection. Excess DEX still enhances convective transport, but much less effectively. The reason is that the vascular hydraulic conductivity L_p for 30 mg/kg DEX is much less than 3 mg/kg DEX (Table 4.4). In addition, we found that higher dose of DEX treatment leads to lower spatially-averaged diffusive flux (20% decrease with 3 mg/kg DEX and 65% decrease with 30 mg/kg DEX compared to control). One reason is that the elevated convective flux with DEX treatment results in a much higher interstitial concentration. Thus, the driving force from transvascular concentration gradient decreases, leading to a lower diffusive flux. In addition, we reported in Martin et al. [6] [6] that the vessel wall pore size is smaller with 30 mg/kg DEX treatment because vascular normalization reduces vessel leakiness by shrinking vessel wall pores. Accordingly, the diffusive hindrance (introduced in Appendix A.2.1) is also smaller. Note that a smaller diffusive hindrance represents higher impairment to diffusion[121]. Thus, 30 mg/kg DEX treatment results in a lower diffusive flux. In conclusion, these results demonstrate that DEX increases the accumulation of nanocarriers in tumors by increasing the convective transvascular flux, but the dose of TME normalization treatment should be titrated to avoid reducing vessel wall pore sizes that limit the benefit to enhanced convection.

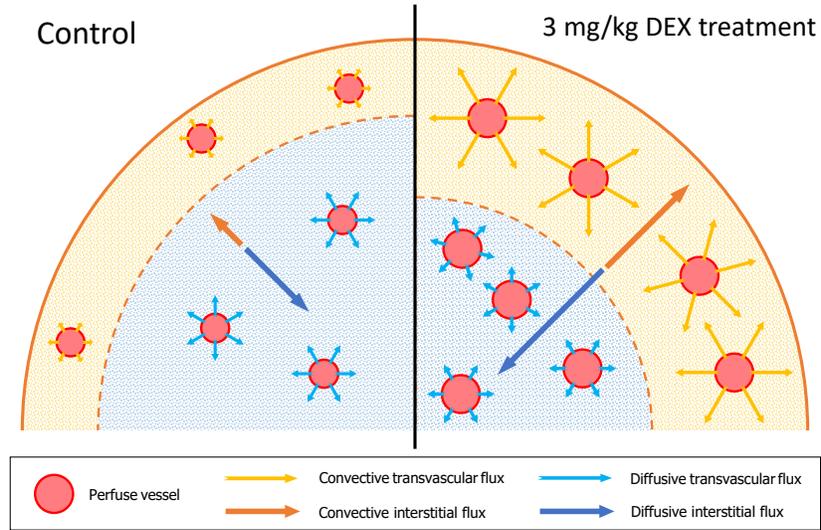


Figure 4.8: Cross sections of the spherical tumor are illustrated in this schematic for the control (left) and 3 mg/kg DEX treatment (right) cases. Perfuse vessels are more abundant and have a larger average diameter following DEX treatment versus the control case; a result of normalizing the tumor microenvironment. The outer orange shaded sections represent the convection-dominated region with significant pressure gradients resulting in predominant convective transvascular flux (yellow arrows). The inner blue shaded sections represent the diffusion-dominated region with almost no pressure gradient (highest interstitial fluid pressure (IFP)) resulting in predominant diffusive transvascular flux (light blue arrows). The blue region is much larger for the control case with the demarcation (orange dashed curves) between regions occurring at $\hat{r} = 0.8$, whereas the demarcation between regions for the DEX treatment case is at $\hat{r} = 0.6$. Convective transvascular flux is significantly enhanced after DEX treatment. The orange arrows pointing radially outward and blue arrows pointing radially inward represent, respectively, the nanocarrier convective and diffusive flux in the tumor interstitium. The direction of interstitial convective transport of nanocarriers is outward towards the periphery, caused by the IFP gradient, while the direction of interstitial diffusive transport of nanocarriers is inward towards the center, caused by the concentration gradient. The overall interstitial fluxes are significantly greater following DEX treatment. The interstitial fluxes and transvascular fluxes are illustrated based on the global optimization results for 13 nm nanocarrier experiments. Note that the interstitial and transvascular flux arrow lengths are each normalized to their own relevant bases for ease of illustration and should not be compared to one another. Also note that since interstitial fluxes are spatially dependent, the arrows represent spatially-averaged fluxes.

4.3.3 Global Optimization Reveals Dose of Dexamethasone Maximizing Nanocarrier Accumulation

Given that a moderate dose of DEX is superior to no DEX and a high dose of DEX for enhancing transvascular transport, we hypothesized that there is an optimal dose of DEX that can maximize nanocarrier or antibody concentration accumulation. As indicated by the previous preclinical study [6], DEX as a drug for TME normalization is both (1) an antiangiogenic agent that can normalize tumor vessels and (2) a cancer-associated fibroblast reprogramming agent that reduces ECM levels leading to decompressed tumor vessels. The functions of (1) and (2) are associated with vascular hydraulic conductivity L_p and interstitial hydraulic conductivity K , respectively. Both L_p and K become more favorable for drug delivery with a moderate dose of DEX treatment, but the relative contributions of (1) and (2) cannot be directly controlled with a drug like DEX that affects both. In addition, as indicated by Table 4.4, an excess dose of DEX decreases L_p thereby limiting transvascular flux for drug delivery. Thus, it is not clear what dose of DEX should be used to maximize the therapeutic effect of a subsequently administered nanocarrier or antibody. The global optimization method and TME-normalizing therapy design formulation (introduced in Section 4.2.5) enable the capability to seek the optimal dose of DEX maximizing the nanocarrier accumulation.

As described in Section 4.2.5, we considered two cases of TME-normalizing therapy design problems: (Case 1) the relationships between DEX dose and L_p and K are established based on the original data from Martin et al. [6], expressed as (4.5) and (4.6); and (Case 2) the relationships between DEX dose and L_p and K are established based on the original data

combined with auxiliary data points, expressed as (4.7) and (4.8). Both TME-normalizing therapy design problems were solved to global optimality. It took 2.5 h to solve Case 1 and 3.6 h to solve Case 2. The more complicated relationship between DEX dose and hydraulic conductivities in Case 2 resulted in higher complexity and a longer solution time to reach global optimality. Nevertheless, the proposed methodology with a mixed IA/AA approach for the bounding routine was able to locate an optimal solution in hours. Thus, this short computation time demonstrates that the proposed TME-normalizing therapy design methodology is practical for real-world clinical studies. An optimal solution for Case 1 is found at $x^* = 5.30$ mg/kg, and for Case 2 is found at $x^* = 4.41$ mg/kg. The optimal dose found in Case 1 results in 3% higher concentration accumulation than 3 mg/kg DEX treatment and 74% higher than 30 mg/kg DEX treatment. As a result, the TME-normalizing therapy design methods in this work demonstrate that global optimization can be used in a reasonable time window to determine the optimal dose of DEX, which is predicted to perform 3% better than the best dose determined by the experiments.

4.3.4 Dexamethasone Dose Affects the Transvascular Convective Transport Size-Dependently

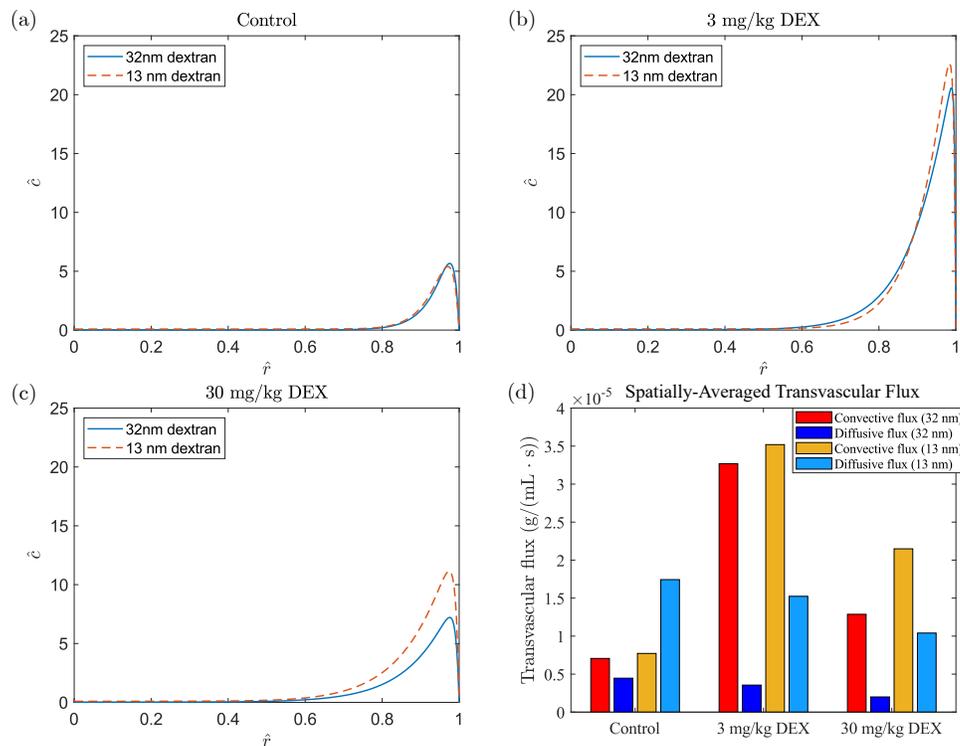


Figure 4.9: Interstitial concentrations \hat{c} of different sizes nanocarriers (500 kDa nanocarrier - 32 nm; 70 kDa nanocarrier - 13nm; Case 1 - 16.40 nm) one-hour post-administration are plotted versus dimensionless tumor radial position \hat{r} for (a) control; (b) 3 mg/kg DEX treatment; and (c) 30 mg/kg DEX treatment cases. (d) The spatially-average transvascular convective and diffusive fluxes are plotted for 32 nm and 13 nm dextrans one-hour post-administration. The interstitial concentration with 30 mg/kg DEX treatment for 32 nm dextran is lower than 13 nm dextran mainly due to its lower convective flux.

After finding that the optimal dose of DEX treatment maximizing concentration accumulation, we hypothesized that size of nanocarriers also affects interstitial concentration. We compared vascular permeability experimental data of two nanocarriers with different hydrodynamic diameters, because previous studies demonstrated that vascular permeability depends on the nanocarrier size [120, 121, 151]. The smaller nanocarrier is 13 nm, which is similar to the size of nanoparticle albumin-bound paclitaxel in circulation [121], and the

larger is similar to the size of NC-6004, which is a clinical-stage polymeric micelle containing cisplatin [6]. Using this experimental data and our mathematical model, we investigated the model-predicted interstitial concentration with respect to tumor radial position for these nanocarriers. As illustrated in Figure 4.9(a), the interstitial concentrations for the control case are almost the same for 32 nm and 13 nm dextrans. And for 3mg/kg DEX treatment case illustrated in Figure 4.9(b), the peak for 13 nm dextran is slightly higher, but the overall concentration distribution is still very close for these dextrans. However, for 30 mg/kg DEX treatment case illustrated in Figure 4.9(c), the concentration profile for 13 nm dextran is higher than 32 nm. A possible reason is that the vessel wall pore size decreases with 30 mg/kg DEX treatment [6]. Thus, the steric hindrance is larger, especially for larger nanocarriers. Consequently, there are fewer larger nanocarriers transporting into the tumor tissue, leading to a lower concentration profile. To better understand this phenomena, we needed to investigate the effects of convective and diffusive transport.

We quantified the spatially-averaged convective and diffusive fluxes for 13 nm and 32 nm dextrans to demonstrate their impact on accumulation. As illustrated in Figure 4.9(d), 3 mg/kg DEX treatment enhances convection to a similar extent for each dextran (360% increase for both 13 and 32 nm dextran compared with the control case). However, 30 mg/kg DEX treatment leads to a 80% increase of convection for 32 nm dextran and a 180% increase for 13 nm dextran. Therefore, the relatively lower convective flux with 30 mg/kg DEX for 32 nm dextran results in less accumulation into the tumor tissue. In addition, we found that DEX reduces diffusion for both nanocarriers. As explained in Section 4.3.2, the higher interstitial concentration and smaller pore size with 30 mg/kg DEX lead to lower diffusive fluxes. Since diffusion is inversely related to hydrodynamic diameter of nanocarriers, reduced

diffusion after DEX is more important for smaller nanocarriers, which rely on diffusion. In conclusion, we found 3 mg/kg DEX enhanced transvascular transport size-independently, which conforms to the findings using the ECM normalizing agent tranilast in Papageorgis et al. [152] [152]. However, given the antiangiogenic properties of DEX, an excess dose of DEX is less effective for enhancing convection especially for larger nanocarriers.

4.3.5 Global Optimization Determines the Dexamethasone Dose and Nanocarrier Size Maximizing Accumulation

We found that DEX enhances convection yet reduces diffusion, so we sought to determine the optimal hydrodynamic diameter of nanocarrier that exploits the balance of these two effects to realize a maximum accumulation with safety/performance specifications. We considered three cases of drug size design problems. These corresponded to 3 mg/kg DEX treatment, the optimal dose of DEX for Case 1 (5.30 mg/kg), and the optimal dose of DEX for Case 2 (4.41 mg/kg), respectively. The 3 mg/kg dose induced the highest transvascular flux in experiments [6], whereas Case 1 and Case 2 were determined from the corresponding TME-normalizing therapy design problems. These drug size design problems formulated as (4.10) were solved to global optimality. The optimal solutions found and time costs for each case are summarized in Table 4.6. Note that the optimal nanocarrier sizes in these designs strictly satisfy the safety/performance requirements to avoid potential side effects and guarantee the effectiveness, which constrain the nanocarrier concentrations in the periphery of tumor normal tissue, as demonstrated in (4.10). Though smaller nanocarriers diffuse and accumulate inside the tumor interstitial space more quickly, it might violate the

safety specifications in these designs. Thus, these optimal solutions account for the drug size design results with requirements. In addition, these problems can be solved in minutes, demonstrating the practicability for real-world applications.

Table 4.6: Optimal solutions and time costs of drug size design problems are listed in this table for the case studies of 3 mg/kg DEX treatment; and Case 1, and Case 2 of the therapy design problem.

Case study	3 mg/kg	Case 1	Case 2
Optimal solution (d_m^* , nm)	19.65	16.55	12.51
Time (s)	355	384	120

The simultaneous therapy design approach with ANN models formulated as (4.12) was also performed with Case 1 and Case 2 studies. An optimal solution for Case 1 was found at $(x^*, d_m^*) = (5.32, 16.40)$ and for Case 2 at $(x^*, d_m^*) = (4.38, 12.41)$. The time costs are 42 s for Case 1 and 350 s for Case 2. However, global optimal solution of (4.11) with the mechanistic model could not be obtained within a reasonable time limit. We expect that continued research on global bounding methods may be able to accelerate convergence and address this issue in the future. Alternatively, we implemented a multi-start local optimization procedure for problems formulated as (4.11) and selected the results with the lowest objective function values: $((x^*, d_m^*) = (5.33, 16.54)$ for Case 1 and $(x^*, d_m^*) = (4.36, 12.58)$ for Case 2). We found that the optimal solutions obtained with the ANN models are very close to the best-found local optimal results obtained via the multi-start procedure. Since the ANN models were very accurate surrogates of the mechanistic model, this provides supporting evidence that the local results obtained are close estimates of the global optima. Therefore, these results provide support for the practicability of the ANN models for use in optimal decision-making in cancer therapeutics.

The therapy design methods in this work provide capability to identify optimal dose and drug size for maximizing the improvement in nanocarrier delivery induced by TME-normalizing therapies. One area of future work is to investigate the effects of half-life circulation time on drug delivery. Smaller nanocarriers, which diffuse faster than larger nanocarriers, benefit from having a higher intravascular concentration resulting from longer circulation time. Thus, testing the impacts of different circulation time on concentration accumulation of different size nanocarriers can be a useful future study. In addition, Martin et al. [6] reported that TME normalization increases the perfused vascular density, which is not incorporated in this work. Higher vascular density indicates additional functional perfused vessels, which is beneficial for drug delivery and accumulation in tumor tissue. Thus, we predicted that results for TME-normalizing therapy will be better if accounting for changing vascular density. Additionally, based on simulations of the model, the variation of vascular density (S/V from 50 to 300) evidently does not affect the nanocarrier concentration profiles. Nevertheless, the impacts of the vascular density require further investigation and integrating this attribute could elucidate more details of the underlying transport mechanisms. While the current work demonstrates the influence of transvascular transport on nanocarrier delivery, investigating how dexamethasone affects interstitial transport could establish a more comprehensive foundation for further enhancing therapy design methods.

4.4 Conclusions

Rigorous methods of model validations and optimal TME-normalizing dose and nanocarrier size therapy designs were developed. This work was motivated by the need for more rigorous

methods for *in silico* model-based decision-making in cancer research. We established and demonstrated the use of a comprehensive theoretical framework for model-based applications in preclinical PKPD research and development . The dynamic optimization problems for this study were formulated as PDE-constrained NLPs and solved to global optimality, providing rigorous solutions for cancer drug delivery studies. An efficient bounding routine using IA/AA and DI approaches and a special bounding rule for the Péclet number in the solute source term were proposed for improving the performance of the global optimization algorithm. In addition, machine learning approaches were utilized to establish a data-driven model via ANNs as surrogate models for the original PDE system. The ANNs were utilized in place of the mechanistic model for solving the parameter estimation problems with a simplified formulation. In particular, based on the global solution values obtained for the hydraulic conductivities, transvascular transport was quantified with respect to convective and diffusive fluxes to elucidate their contributions to the accumulation of anticancer nanocarriers in tumors following TME-normalizing DEX treatment. Moreover, a methodology for optimal TME-normalizing therapy design was proposed to optimize the dose of DEX for enhanced accumulation of anticancer nanocarriers in tumors. The nanocarrier size design method was also proposed to determine an optimal size for patient-specific TMEs with safety/performance specifications. Finally, a simultaneous design formulation was considered to determine an optimal dose of DEX and an optimal nanocarrier size that would lead to maximized accumulation in the tumor interstitium. This work can be extended to robust design problems that account for the impacts of uncertainty that may arise from noisy data or incomplete characterization of a patient's TME.

In the next chapter, the concept of semi-infinite optimization is introduced to demonstrate

worst-case design problems with respect to safety-critical systems. The ANN method utilized in this chapter is also employed to construct hybrid models to represent complicated systems. The approach to incorporate hybrid model into semi-infinite optimization is formalized.

Chapter 5

Semi-Infinite Optimization with Hybrid Models

In this chapter, the solution strategies for solving semi-infinite programs (SIPs) are discussed. Particularly, hybrid first-principles data-driven models are utilized in this study because they provide the potential to dramatically improve model prediction accuracy, stepping closer to the digital twin concept. Within this context, worst-case engineering design feasibility and reliability problems give rise to a class of SIP formulations with hybrid models as coupling equality constraints. Reduced-space deterministic global optimization methods are exploited to solve this class of SIPs to ϵ -global optimality in finitely many iterations. As a consequence, the SIP formulations with hybrid models are formalized and the solution methods are demonstrated through representative cases studies.

5.1 Introduction

Many engineering systems are deemed safety-critical and, as such, require strict guarantees of performance and safety. Uncertainties, such as those introduced by inaccurate data, should be accounted for at the design stage of such systems. Therefore, it is necessary to identify the worst-case performance of these systems to mitigate the impacts of uncertainty on the final design. For example, in many energy-related applications, the costs associated with operational failures are extremely high; often including loss of life, substantial environmental damage, severe economic damage, and major sociopolitical fallout. From a model-based perspective, approaching design problems of this nature amounts to identifying realizations of uncertainty that result in a simulated worst-case violation of performance/safety constraints as governed by a system model. As such, deterministic global optimization methods are required to guarantee worst-case realizations of uncertainty may be identified in the general case.

Worst-case design problems have historically been treated as bilevel or more general multilevel programs. These programs have feasible sets that are characterized by other optimization problems. As such, these programs are extremely challenging or even impossible to solve directly using existing methods. Thus, early studies focused on the simplest cases of worst-case design problems with linearity and convexity conditions [153, 154]. Over the years, relevant studies were extended to more complicated worst-case design problems with nonlinearity [155, 156, 157].

Gümüř and Floudas [158] developed a global optimization algorithm based on relaxations of the feasible region for solving worst-case design problems whose bilevel formulations in-

involve twice-differentiable nonlinear functions. A transformation was proposed to replace the inner problem with its KKT optimality conditions, transforming the inner program into nonlinear algebraic constraints under the linear independence constraint qualification. This approach requires convexity for the KKT conditions to be necessary and sufficient, however general nonconvex functions were considered by exploiting αBB relaxations within a branch-and-bound framework for the solution of the KKT-reformulated NLP. Feasibility and flexibility index problems were considered within this context in a follow-up work[159]. However, this approach cannot provide valid convergent upper bounds for bilevel programs with nonconvex inner programs, in general[160]. Mitsos et al. [160] proposed a bounding algorithm to resolve this problem that can solve nonlinear bilevel programs to global optimality without any convexity assumptions. However, the approach is limited to only considering inequality constraints (see Mitsos et al. [160, Assumption 3]).

As an alternative strategy to solving bilevel programs, multiparametric programming was developed by recasting them into single-level deterministic optimization problems [161, 162]. This strategy is unique in that the parametric solution of the inner program is characterized explicitly and therefore can be utilized in real-time optimization applications. However, the developed methods require the inner programs to be linear or quadratic programs[162]. For general nonconvex objectives and general nonlinear and nonconvex inner programs, multiparametric programming is not applicable. In this work, we investigate the most general worst-case design problems that may involve nonlinear coupling equality constraints, and in doing so we consider the methods that reformulate bilevel programs as equivalent semi-infinite programs (SIPs).

The solution of general nonconvex SIPs has been an active area of research in recent

years, yielding approaches that perform well for solving classes of worst-case design problem formulations. Many of the recent advancements have been based on the discretization-based cutting-plane algorithm developed by Blankenship and Falk [163] [163]. Mitsos [164] developed an algorithm with a new procedure for feasible point generation by setting a restriction condition of right-hand side of the semi-infinite constraints. Stuber and Barton [165] developed a modified version of the SIP algorithm proposed by Mitsos [164] and finally extended the method to the most general nonconvex case accounting for semi-infinite equality constraints without assuming that they admit closed-form parametric solutions.

Djelassi and Mitsos [166] developed a hybrid discretization-based algorithm for the global solution of SIPs without semi-infinite equality constraints. The algorithm proposed by Mitsos [164] is employed for upper-bounding and lower-bounding problems, and an oracle problem adapted from the algorithm proposed by Tsoukalas and Rustem [167] is employed to generate cheap lower bounds and adaptive updates to the restriction of this algorithm. The hybrid algorithm can avoid a dense population of the discretization, and has superior computational performance as a result.

Solution methods for higher-complexity SIP formulations have also been studied. The algorithm of Mitsos [164] was extended to generalized semi-infinite programs (GSIPs) by Mitsos and Tsoukalas [168]. Djelassi et al. [169] then extended this GSIP algorithm [168] and considered the mixed-integer bilevel program to allow the presence of coupling equality constraints. In their method, a subset of the lower-level variables are treated as dependent variables to cope with convergence issues introduced by coupling equality constraints. This algorithm requires an increase in the dimensionality of continuous variables for some sub-problems, but the performance penalty was not observed in their numerical experiments.

Djelassi and Mitsos [170] most recently proposed an algorithm for the global solution of existence-constrained SIPs (ESIPs) that are a generalization of standard SIPs with three levels. This is the first algorithm that can solve ESIPs to global optimality without any convexity assumptions. Some other recent developments in nonconvex SIP applications and algorithms have been reviewed by Djelassi et al. [171].

A key concern pertinent to many applications of SIPs within engineering design, is the need for high-accuracy and low computational complexity models of safety-critical systems (whose performance must satisfy strict requirements) [172]. In many cases, strict performance/safety requirements must be satisfied over a range of potential input disturbances or process noise. While it is possible that such disturbances and noise may be well characterized for some cases, this is not often typical for nascent designs. In addition, many process systems models involve implicit functions as their nonlinearity prohibits explicit closed-form solutions. Even though a method for solving SIPs with implicit functions has been developed by Stuber and Barton [165], the algorithm is computationally expensive and high-complexity models compound the computational cost. Hybrid modeling approaches are attractive here because they can accurately represent complicated process systems that are not fully understood, and may also reduce the mathematical complexities caused by implicit functions, nonlinearity, complicated dynamics, and multivariate uncertainty. Thus, the central goal of our work is to utilize hybrid models within SIP formulations for the optimal design, simulation, and robustness verification of process systems (i.e., process systems that satisfy all predetermined performance/safety requirements [13]) in the face of worst-case uncertainty.

Hybrid models consist of structured combinations of rigorous first-principles models (FPMs) that account for necessary/known system mechanisms and empirical or data-driven

models (DDMs) that describe phenomena that cannot be readily described using FPMs due to a lack of adequate knowledge [173]. Over the past few decades, the use of hybrid modeling approaches, particularly those that exploit machine learning approaches, have found a wide variety of applications in the process systems engineering community. These methods have enhanced process output [172], improved controller performance [174, 175], and enabled integrated system-level designs of highly complex processes [176, 177]. In this paper, we explore applications of hybrid models to worst-case design problems to investigate and verify their applicability in SIPs.

SIPs governed by hybrid models are of particular importance in process systems engineering [177, 178, 179], yet their usage within general (nonconvex) SIP contexts remains absent. This is likely a consequence of the coupling equality constraints introduced by the FPMs that significantly complicate the problem. In this paper, we propose addressing these gaps with the following main novel contributions:

1. We formalize the approach to use hybrid models with SIPs. One application of this approach is to resolve complications due to coupling equality constraints via a *reduced-space* formulation.
2. We illustrate how this SIP formulation that incorporates hybrid models is sufficiently general such that it may be readily applied to exemplary robust design problems incorporating process dynamics.
3. We present a hybrid modeling approach that resolves numerical issues relating to domain violations, ubiquitous in process systems engineering modeling and simulation, through use of a novel nonsmooth SIP formulation that incorporates validity con-

straints.

In this paper, we present new developments on the formulation and solution of SIPs with embedded hybrid models. Particular attention is paid to the models that use artificial neural networks (ANNs) with activation functions that are of interest for deep learning as the DDM. In the following sections, we detail: the mathematical conventions used in the paper (Mathematical Background), formalize optimization problems with hybrid models (Optimization of Hybrid Models); formalize SIPs with hybrid models embedded and present a solution algorithm (Semi-Infinite Optimization with Hybrid Models); present case studies that demonstrate a variety of optimization under uncertainty problems formulated as SIPs with hybrid models embedded (Case Studies); and extend the proposed approach to SIPs with implicit functions embedded (Extension to Implicit Forms). Finally, we suggest future directions for subsequent research.

5.2 Mathematical Background

In this section, the necessary mathematical preliminaries for the framework of SIPs with hybrid models are introduced.

5.2.1 Multilayer Perceptrons

Several DDM methods have been developed and applied to a broad range of process systems, such as support vector machines [180], random forests [181], and ANNs [182, 183]. In this work, ANNs are utilized as a representative DDM approach to demonstrate the formulation of SIPs with hybrid models, and the corresponding notation is formalized in this section

accordingly. The multilayer perceptron (MLP) is one of the most common classes of ANN structures. As illustrated in Figure 5.1, the MLP is composed of a directed acyclic graph (DAG) containing n layers enumerated $k = 1, \dots, n$. The first layer with $k = 1$ represents the inputs of the MLP, whereas the last layer with $k = n$ corresponds to the *output layer*. The $k = 2, \dots, n - 1$ layers are the *hidden layers*. Let $m^{(k)}$ be the number of neurons in layer k , $\mathbf{a}^{(k)} \in \mathbb{R}^{m^{(k)}}$ be the outputs of layer k . As defined, $\mathbf{a}^{(1)}$ is the input vector and $\mathbf{a}^{(n)}$ is the output vector of the MLP. The vector $\mathbf{a}^{(k)}$ for hidden layers $k \in \{2, \dots, n\}$ is defined as

$$\mathbf{a}^{(k)} = f^{(k)} \left(\mathbf{W}^{(k-1)} \mathbf{a}^{(k-1)} + \mathbf{b}^{(k-1)} \right), \quad (5.1)$$

where $f^{(k)} : \mathbb{R} \rightarrow \mathbb{R}$ is an activation function, $\mathbf{W}^{(k-1)} \in \mathbb{R}^{m^{(k)} \times m^{(k-1)}}$ is a *weight matrix*, and $\mathbf{b}^{(k-1)} \in \mathbb{R}^{m^{(k)}}$ is a *bias vector*. For ease of introduction, we define $\mathbf{o} : \mathbb{R}^{m^{(1)}} \rightarrow \mathbb{R}^{m^{(n)}}$ as the representative input-output function for a generic DDM. Thus, as for a MLP, $\mathbf{a}^{(n)} = \mathbf{o}(\mathbf{a}^{(1)})$.

When training MLPs, the weight matrices and bias vectors are regarded as optimization variables whereas the input values of $\mathbf{a}^{(1)}$ are taken as parameters. When using fully-trained MLPs in a hybrid model for simulation or optimization, the weight matrices and bias vectors are fixed to the trained constant parameters. The notation of MLPs in this section is used for a typical class of DDMs that will be used for hybrid model formulations.

5.2.2 Hybrid Model Architecture

In this section, we formalize the notation for first-principles and data-driven sub-models as illustrated in Figure 5.2. A vector of independent input variables is defined as $\mathbf{y} \in Y \subset \mathbb{R}^{n_y}$, a vector $\hat{\mathbf{z}}^{FPM} \in Z^{FPM} \subset \mathbb{R}^{n_{zf}}$ represents state variables governed by the FPMs, $\mathbf{h}^{FPM} :$

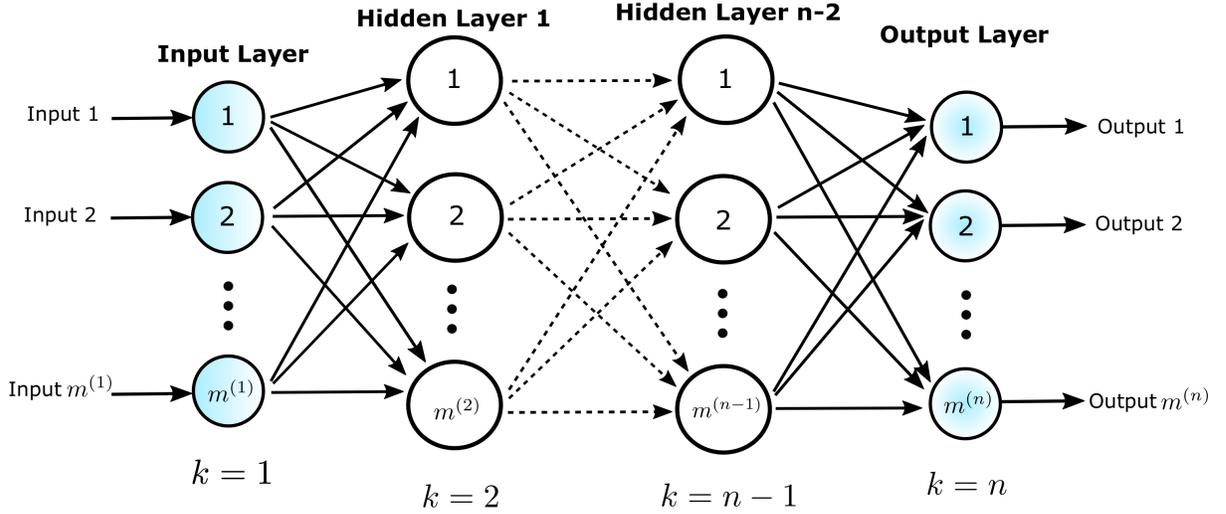


Figure 5.1: A multilayer perceptron with n layers is illustrated as a directed acyclic graph. The input layer corresponds to $k = 1$, the hidden layers correspond to $k = 2, \dots, n - 1$, and the output layer corresponds to $k = n$. The multilayer perceptron has a fully-connected feed-forward network where all neurons in last layer, $k - 1$, are related to all neurons in the subsequent layer, k .

$Z^{FPM} \times Z^{DDM} \times Y \rightarrow \mathbb{R}^{n_{zf}}$ represent FPM equations, $\hat{\mathbf{z}}^{DDM} \in Z^{DDM} \subset \mathbb{R}^{n_{zd}}$ is a vector of output variables of the DDMs. Note that in the scope of this work, we consider MLPs as explicit input-output DDMs that can be represented by $\mathbf{a}^{(n)} = \mathbf{o}(\mathbf{a}^{(1)})$ as previously defined.

In general, the architecture of hybrid models is classified according to the parallel and/or serial arrangement of sub-models (see von Stosch et al. [184] for a thorough review and discussion of hybrid model architectures). The mathematical structure of a parallel hybrid model is illustrated in Figure 5.2(a). In this formulation, the FPM $\mathbf{h}^{FPM}(\hat{\mathbf{z}}^{FPM}, \cdot, \mathbf{y}) = \mathbf{0}$ does not have explicit dependence on $\hat{\mathbf{z}}^{DDM}$ and $\boldsymbol{\mu}$ represents the final outputs of the parallel hybrid model that can be expressed as $\boldsymbol{\mu} = \psi(\hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM})$, where $\psi : \mathbb{R}^{n_{zf}} \times \mathbb{R}^{n_{zd}} \rightarrow \mathbb{R}^{n_{\mu}}$ represents some functional relationship involving the states of the FPMs and outputs of the DDMs. In cases employing a parallel model architecture, the FPMs may not be

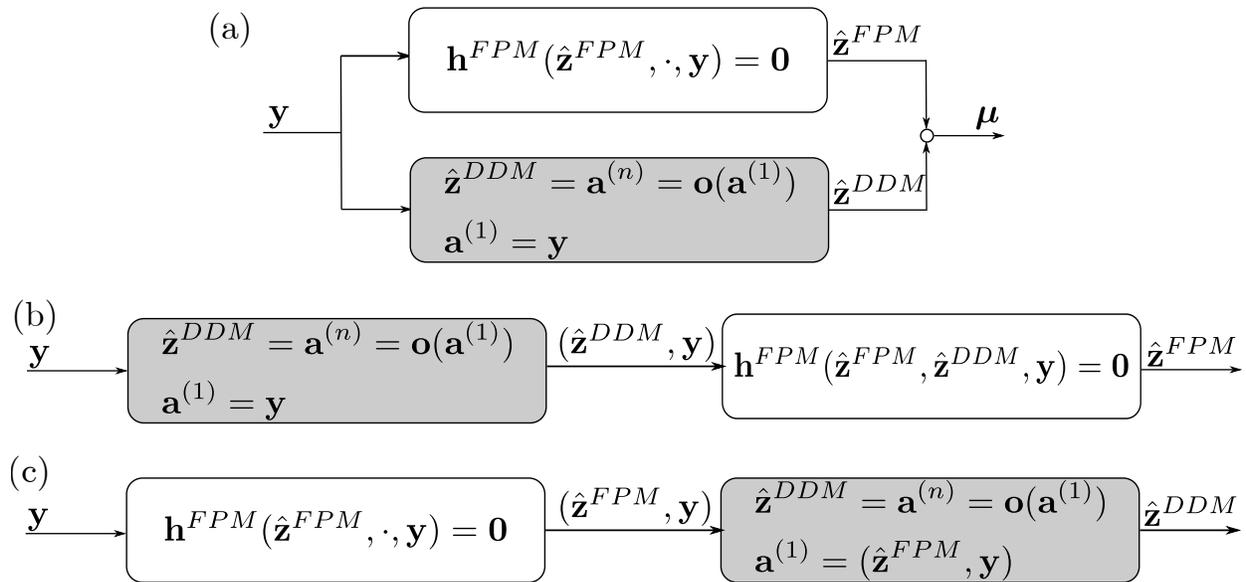


Figure 5.2: Flow diagrams of typical hybrid model architectures are presented in this figure (white blocks represent first-principles sub-models and shaded blocks represent data-driven sub-models). (a) A parallel hybrid model architecture maps inputs to outputs of each model type in parallel. (b) A DDM/FPM serial hybrid model architecture maps inputs of the DDM to outputs that are subsequently used as inputs in the FPM. (c) A FPM/DDM serial model architecture maps inputs to outputs of the FPM that are subsequently used as inputs to the DDM.

able to accurately capture some of the phenomena observed in real systems, resulting in a discrepancy. In this situation, a DDM can be utilized to rectify the mismatch between the prediction of FPM and the observed process data [178].

In the serial architecture, the output of the first sub-model is taken as an intermediate variable that is input to the second sub-model. The DDM/FPM serial architecture is the most common hybrid model architecture [184], illustrated in Figure 5.2(b). In chemical engineering systems, FPMs typically involve conservation laws that may have extremely complicated mathematical expressions and/or source terms that may fail to accurately capture some observed system behavior due to an incomplete understanding of underlying mechanisms. In these situations, DDMs can be used as a surrogate model to represent intractable parameters

and/or subexpressions. Alternatively, the FPM/DDM serial architecture is shown in Figure 5.2(c). This architecture can be used to model a system whose intermediate variables are governed by the first-principles model $\mathbf{h}^{FPM}(\hat{\mathbf{z}}^{FPM}, \cdot, \mathbf{y}) = \mathbf{0}$ (with no explicit dependence on $\hat{\mathbf{z}}^{DDM}$) and the final outputs are some process parameters that are related to the intermediate state variables [185]. In the next section, the notation pertaining to hybrid model architectures in embedded optimization formulations is established.

5.3 Optimization of Hybrid Models

In this section, optimization problems with hybrid ANN models embedded are formalized. We use a general formulation to maintain applicability to a wide variety of surrogate modeling approaches being actively explored by machine learning researchers (e.g., [186, 187, 188, 189]) including neural ordinary differential equations [190] inspired by the success of ResNet [191]. In general, a conventional formulation of an optimization problem involving hybrid models accounts for the modeling equations as explicit equality constraints. Thus, let the optimization formulation with hybrid models be represented by the following nonlinear program

(NLP):

$$\begin{aligned}
& \min_{\mathbf{y}, \hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM}} \phi(\hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM}, \mathbf{y}) & (5.2) \\
& \text{s.t. } \mathbf{h}^{FPM}(\hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM}, \mathbf{y}) = \mathbf{0} \\
& \mathbf{h}^{DDM}(\hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM}, \mathbf{y}) = \mathbf{0} \\
& \mathbf{g}(\hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM}, \mathbf{y}) \leq \mathbf{0} \\
& \mathbf{y} \in Y \in \mathbb{IR}^{n_y} \\
& \hat{\mathbf{z}}^{FPM} \in Z^{FPM} \subset \mathbb{R}^{n_{zf}} \\
& \hat{\mathbf{z}}^{DDM} \in Z^{DDM} \subset \mathbb{R}^{n_{zd}},
\end{aligned}$$

where \mathbb{IR}^n is the set of all n -dimensional real intervals, the decision variables consist of the independent input variables \mathbf{y} (e.g., design variables), the output variables of an ANN $\hat{\mathbf{z}}^{DDM}$ as previously defined, and the system state variables $\hat{\mathbf{z}}^{FPM}$ that are determined by the FPMs. It is assumed that the objective function $\phi : Z^{FPM} \times Z^{DDM} \times Y \rightarrow \mathbb{R}$ and the inequality constraints $\mathbf{g} : Z^{FPM} \times Z^{DDM} \times Y \rightarrow \mathbb{R}^{n_g}$ are continuous. The equality constraints $\mathbf{h}^{DDM} : Z^{FPM} \times Z^{DDM} \times Y \rightarrow \mathbb{R}^{n_{zd}}$ are expressed in standard form and represent the DDM equations (i.e., the ANN equations $\mathbf{h}^{DDM}(\hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM}, \mathbf{y}) = \hat{\mathbf{z}}^{DDM} - \mathbf{a}^{(n)} = \mathbf{0}$). The equality constraints $\mathbf{h}^{FPM} : Z^{FPM} \times Z^{DDM} \times Y \rightarrow \mathbb{R}^{n_{zf}}$ and $\mathbf{h}^{DDM} : Z^{FPM} \times Z^{DDM} \times Y \rightarrow \mathbb{R}^{n_{zd}}$ are also assumed to be continuous. In general, bounds on \mathbf{y} , $\hat{\mathbf{z}}^{DDM}$, and $\hat{\mathbf{z}}^{FPM}$ must be supplied to ensure that the problem is well-posed, although some variables may not require bounds known *a priori*.

The general optimization formulation (5.2) can be reformulated compactly as follows.

Define $\hat{\mathbf{z}} = (\hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM})$, $Z = Z^{FPM} \times Z^{DDM}$, and let $\mathbf{h} : Z \times Y \rightarrow \mathbb{R}^{n_z}$ be the concatenation of \mathbf{h}^{FPM} and \mathbf{h}^{DDM} such that $\mathbf{h}(\hat{\mathbf{z}}, \mathbf{y}) = (\mathbf{h}^{FPM}(\hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM}, \mathbf{y}), \mathbf{h}^{DDM}(\hat{\mathbf{z}}^{FPM}, \hat{\mathbf{z}}^{DDM}, \mathbf{y}))$. Then, (5.2) can be reformulated as the following NLP:

$$\begin{aligned} \min_{\mathbf{y} \in Y, \hat{\mathbf{z}} \in Z} \phi(\hat{\mathbf{z}}, \mathbf{y}) & \quad (5.3) \\ \text{s.t. } \mathbf{h}(\hat{\mathbf{z}}, \mathbf{y}) &= \mathbf{0} \\ \mathbf{g}(\hat{\mathbf{z}}, \mathbf{y}) &\leq \mathbf{0}. \end{aligned}$$

In many cases that arise naturally from process flowsheet simulation, model inputs and parameters (e.g., process design specifications, controllable inputs) define unique state conditions by continuity equations (i.e., conservation laws) as equality constraints that can be solved explicitly. Thus, in this paper, we assume that there exists a unique explicit closed-form function $\mathbf{z} : Y \rightarrow Z$ such that $\mathbf{h}(\mathbf{z}(\mathbf{y}), \mathbf{y}) = \mathbf{0}$ for every $\mathbf{y} \in Y$. Under this assumption, the equality constraints can be eliminated and (5.3) can be simplified as:

$$\begin{aligned} \min_{\mathbf{y} \in Y} \phi(\mathbf{z}(\mathbf{y}), \mathbf{y}) & \quad (5.4) \\ \text{s.t. } \mathbf{g}(\mathbf{z}(\mathbf{y}), \mathbf{y}) &\leq \mathbf{0}. \end{aligned}$$

Since the equality constraints of (5.3) are entirely eliminated in this formulation, there is a (significant) reduction in problem dimensionality. Although we assumed the existence of explicit functions for the proposed approach, this does not restrict the method. The section **Extension to Implicit Forms** discusses how this assumption may be relaxed allowing for the proposed approach to be applied to more general hybrid models with implicit forms.

Remark 8. *Note that the uniqueness assumption is required for the elimination of the coupling equality constraints from the original problem formulation (5.2). This assumption and approach have been commonly made in practice for addressing design under uncertainty problems (e.g., [165, 192, 193, 194, 195, 196], among others), and is not presented as a new approach here. However, in case of nonunique parametric solutions (e.g., multiple steady states), without special consideration this approach would effectively restrict the feasible set. For example, consider the model $h(\hat{z}, y) = \hat{z}^2 - y = 0$. This problem has explicit closed-form solutions $z_1(y) = \sqrt{y}$ and $z_2(y) = -\sqrt{y}$ with $y = 0$ a bifurcation point. To ensure that a global solution of the original problem is obtained, both parametric solution branches z_1 and z_2 must be considered or else feasible solutions may be ignored. Within the context of hybrid models, nonuniqueness may be less of a concern since DDMs are trained as explicit input-output mappings representing a system or phenomena of interest and, when coupled to FPMs, are likely to force adherence to a single solution branch. For problems where this is not the case, each solution branch of the FPM would need to be considered separately, as in the simple example above. For parametric dynamical systems, relatively mild assumptions ensure the existence and uniqueness of parametric solution trajectories[55].*

Reduced-space approaches to deterministic global optimization originated from Epperly and Pistikopoulos [197], who detailed a convergent branch-and-bound (B&B) algorithm that branched only on a subset of the decision variables. This *reduced-space* formulation approach was subsequently generalized for many different problem and model types (e.g., [198, Sec. 4.1], wilhelm2019global[199, 200, 201]). This approach avoids the introduction and explicit handling of auxiliary variables and equality constraints through intermediate calculations by

treating the independent input variables \mathbf{y} as the only decision variables of the optimization problem. Since $n_y \ll n_z$ in most process systems engineering problems, (5.4) represents a significantly lower-dimensionality problem than (5.3). Due to the curse of dimensionality in deterministic global optimization, this reduction in dimensionality often translates to a significant reduction in the solution time.

In general, formulations (5.2)-(5.4) are nonconvex optimization problems that are solved to guaranteed global optimality via a variation of the spatial B&B algorithm [202, 203, 204]. This consists of a presolve step, followed by the successive solution of lower- and upper-bounding problems with intermediate domain reduction. An upper bound is typically determined by solving the original nonconvex problem to either feasibility or local optimality. This is distinct from the subproblems encountered in domain reduction and the lower bounding routines that construct and solve relaxations of the nonconvex problem through the computation of convex relaxations of the nonconvex objective and constraint functions [205].

Within this reduced-space context, researchers have addressed the construction of convex and concave relaxations of factorable functions [199, 206, 207, 208] (i.e., a function defined by a finite recursive composition of sums, products, and univariate transcendental functions) as well as specific classes of functions that break the factorability assumption. Methods for computing relaxations of parametric solutions of differential equations [42, 55, 209] as well as implicit functions evaluated by fixed-point methods, have both been detailed [210]. Provided that relaxations of intermediate terms may be computed, these relaxations may be readily composed in a generalized framework [208]. For instance, relaxations of the solutions of parametric differential equations may be computed provided that convex/concave relaxations of the right-hand side function are known, and then composed with an algebraic

objective or constraint term. As such, this modeling framework is generally applicable to the preponderance of hybrid model architectures.

There are several existing deterministic global optimization solvers capable of addressing general problems formulated as (5.3). These include commercially licensed offerings such as BARON [203] and ANTIGONE [94], as well as open-source offerings such as EAGO [52] and MAiNGO [211]. Due to limitations in how problems are represented and how relaxations of nonconvex functions are constructed, BARON [203] and ANTIGONE [94] cannot address formulation (5.4). Alternatively, EAGO [52] and MAiNGO [211] were developed with this class of problems in mind with more flexible modeling requirements and advanced methods for constructing relaxations of nonconvex functions. Due to the high dimensionality of formulation (5.3) and the curse of dimensionality in deterministic global optimization, excessively long run times are expected for solving (5.3). It has been demonstrated through several examples [183, 197, 201, 206, 207, 210, 212] that an equivalent *reduced-space* problem (5.4) can dramatically reduce the run time of a compatible algorithm by dramatically reducing the number of variables branched on. Moreover, the elimination of equality constraints from (5.3) plays a particularly important role in ensuring that SIPs of interest in the subsequent section, are formulated in a readily solvable manner.

5.4 Semi-Infinite Optimization with Hybrid Models

In this section, the foundations for incorporating hybrid models into SIP formulations are formalized. First, consider the input variables of a hybrid model partitioned as $\mathbf{y} = (\mathbf{x}, \mathbf{p})$ with the corresponding domain $Y = X \times P$ with $\mathbf{x} \in X \in \mathbb{IR}^{n_x}$ and $\mathbf{p} \in P \in \mathbb{IR}^{n_p}$. Then,

the general form of an SIP governed by a hybrid model can be expressed as:

$$\begin{aligned}
\phi^* = & \quad \min_{\mathbf{x} \in X} \phi(\mathbf{x}) & (5.5) \\
\text{s.t. } & g(\hat{\mathbf{z}}, \mathbf{x}, \mathbf{p}) \leq 0, \forall \mathbf{p} \in P \\
& \mathbf{h}(\hat{\mathbf{z}}, \mathbf{x}, \mathbf{p}) = \mathbf{0}, \forall \mathbf{p} \in P.
\end{aligned}$$

In this formulation, \mathbf{x} represents a vector of decision variables, $\hat{\mathbf{z}}$ represents a vector of internal state variables governed by the hybrid model equations (as introduced previously), and \mathbf{p} represents a vector of parameters. The objective function $\phi : X \rightarrow \mathbb{R}$ depends solely on the variables $\mathbf{x} \in X$ and the constraints $g : Z \times X \times P \rightarrow \mathbb{R}$ and $\mathbf{h} : Z \times X \times P \rightarrow \mathbb{R}^{n_z}$ are parameterized by $\mathbf{p} \in P$. Note that the hybrid model \mathbf{h} is defined as in the previous section. It is assumed that the objective function ϕ , semi-infinite inequality constraint function g , and equality constraint function \mathbf{h} are factorable and continuous on their domains. Note that we have made no assumptions about the smoothness of g . Therefore, multiple performance constraints g_1, \dots, g_n , may be handled trivially by reformulation into a single constraint $g = \max_i g_i$.

As in formulation (5.3), the state variables $\hat{\mathbf{z}}$ in (5.5) are governed by continuous hybrid model functions $\mathbf{h}(\hat{\mathbf{z}}, \mathbf{x}, \mathbf{p}) = \mathbf{0}$ for each $(\mathbf{x}, \mathbf{p}) \in X \times P$. Similar to formulation (5.4), we assert that the hybrid model functions $\mathbf{h}(\hat{\mathbf{z}}, \mathbf{x}, \mathbf{p}) = \mathbf{0}$ can be solved explicitly. Thus, the state variables can be expressed as an explicit input-output mapping $\mathbf{z} : X \times P \rightarrow \mathbb{R}^{n_z}$ such that $\mathbf{h}(\mathbf{z}(\mathbf{x}, \mathbf{p}), \mathbf{x}, \mathbf{p}) = \mathbf{0}$ for every $(\mathbf{x}, \mathbf{p}) \in X \times P$. Under these assumptions, (5.5) can be

reformulated as:

$$\begin{aligned} \phi^* = & \min_{\mathbf{x} \in X} \phi(\mathbf{x}) \\ \text{s.t. } & g(\mathbf{z}(\mathbf{x}, \mathbf{p}), \mathbf{x}, \mathbf{p}) \leq 0, \forall \mathbf{p} \in P. \end{aligned} \tag{5.6}$$

The SIP formulation covers classes of robust design and optimization under uncertainty problems of specific interest in this work.

Remark 9. *Note that the assumption of uniqueness of $\mathbf{z}(\cdot) \in Z$ for every $\mathbf{y} \in Y$ (i.e., $\forall (\mathbf{x}, \mathbf{p}) \in X \times P$) was discussed in the previous section as a requirement for the elimination of the coupling equality constraints. Within the SIP context for robustness verification, caution must be exercised to ensure that uniqueness can be verified. For systems with multiple solution branches present, an SIP must be solved with respect to each physically meaningful solution branch. However, as remarked in the previous section, nonuniqueness is expected to be rare for systems of interest with hybrid modeling approaches and not an issue for dynamical systems under relatively mild assumptions.*

Three problem types that fall under the general formulation (5.6) will be considered in this work for their relevance in the design of safety-critical systems. The first problem is the design under uncertainty feasibility problem. The goal with this problem is to confirm whether there exists a design that is robust to a worst-case realization of parametric uncertainty:

$$\begin{aligned} \eta^* = & \min_{\mathbf{d} \in D, \eta \in H} \eta \\ \text{s.t. } & \eta \geq g(\mathbf{z}(\mathbf{d}, \boldsymbol{\pi}), \mathbf{d}, \boldsymbol{\pi}), \forall \boldsymbol{\pi} \in \Pi. \end{aligned} \tag{5.7}$$

Here, $\mathbf{d} \in D \in \mathbb{I}\mathbb{R}^{n_d}$ represents a vector of design variables, $\boldsymbol{\pi} \in \Pi$ represents uncertain parameters in the hybrid model, and $\eta \in H \in \mathbb{I}\mathbb{R}$ represents a measure of robust feasibility. With respect to the SIP formulation (5.6), we have $\mathbf{x} = (\mathbf{d}, \eta)$, $X = D \times H$, $\mathbf{p} = \boldsymbol{\pi}$, and $P = \Pi$. If the optimal solution value of the feasibility problem satisfies $\eta^* \leq 0$, then a design exists that is robust to worst-case realizations of uncertainty.

The second problem of consideration is the robust optimal design problem. The objective function is directly defined as a cost function based on a technical or economic objective (e.g., total capital cost or process efficiency):

$$\begin{aligned} \phi^* = & \min_{\mathbf{d} \in D} \phi(\mathbf{d}) \\ \text{s.t. } & g(\mathbf{z}(\mathbf{d}, \boldsymbol{\pi}), \mathbf{d}, \boldsymbol{\pi}) \leq 0, \forall \boldsymbol{\pi} \in \Pi. \end{aligned} \quad (5.8)$$

Here, with respect to formulation (5.6), we have $\mathbf{x} = \mathbf{d}$, $\mathbf{p} = \boldsymbol{\pi}$, $X = D$, and $P = \Pi$. A global optimal solution of this problem will be an optimal system design that is robust to worst-case realizations of uncertainty (if such a design exists).

The last problem of consideration is the operation under uncertainty feasibility problem. This formulation is used to determine whether there exist control settings or recourse such that the system of interest will always satisfy the performance and/or safety specifications. The problem is formulated as:

$$\begin{aligned} \eta^* = & \max_{\boldsymbol{\pi} \in \Pi, \eta \in \mathbb{R}} \eta \\ \text{s.t. } & \eta \leq g(\mathbf{z}(\boldsymbol{\pi}, \mathbf{u}), \boldsymbol{\pi}, \mathbf{u}), \forall \mathbf{u} \in U. \end{aligned} \quad (5.9)$$

Here, we introduce control variables \mathbf{u} that can be manipulated in response to uncertainty realizations $\boldsymbol{\pi}$. With respect to the general SIP formulation (5.6), we have $\mathbf{x} = \boldsymbol{\pi}$, $\mathbf{p} = \mathbf{u}$, $X = \Pi$, and $P = U$. This formulation addresses the question of operational feasibility and verifies the (non)existence of feasible control actions to mitigate the effects of worst-case uncertainty. If $\eta^* \leq 0$, then a feasible recourse control action exists that mitigates the worst-case impacts of uncertainty on the process with respect to the performance/safety specifications.

A state-of-the-art method for solving SIPs to global optimality is discussed in the following section.

5.5 Global Solution of SIPs

In this section, the SIPres algorithm introduced by Mitsos [164] is presented with respect to the formulation (5.6) for hybrid model systems for completeness. The algorithm flowchart is illustrated in Figure 5.3 and relies on solving three nonconvex subproblems (formulated below) to global optimality at each iteration. The algorithm is guaranteed to converge to ϵ -optimality in finitely many iterations under the assumptions of continuity of ϕ and g and the existence of an SIP Slater point arbitrarily close to a minimizer.

Definition 5.5.1 (Lower-Bounding Problem [165]). *Given a finite number of constraints with respect to $\mathbf{p} \in P^{LBD}$ with $P^{LBD} \subset P$ a finite set, the lower-bounding problem is formu-*

lated as:

$$\begin{aligned} \phi^{LBD} = & \min_{\mathbf{x} \in X} \phi(\mathbf{x}) \\ \text{s.t. } & g(\mathbf{z}(\mathbf{x}, \mathbf{p}), \mathbf{x}, \mathbf{p}) \leq 0, \forall \mathbf{p} \in P^{LBD}. \end{aligned}$$

Definition 5.5.2 (Inner Program [165]). *Given a point $\bar{\mathbf{x}} \in X$, the inner program is formulated as:*

$$\bar{g}(\bar{\mathbf{x}}) = \max_{\mathbf{p} \in P} g(\mathbf{z}(\bar{\mathbf{x}}, \mathbf{p}), \bar{\mathbf{x}}, \mathbf{p}).$$

The inner program verifies feasibility of the point $\bar{\mathbf{x}}$ with respect to the original SIP. If $\bar{g}(\bar{\mathbf{x}}) \leq 0$, $\bar{\mathbf{x}}$ is feasible in (5.6).

Definition 5.5.3 (Upper-Bounding Problem [165]). *Given a finite number of constraints with respect to $\mathbf{p} \in P^{UBD}$ with $P^{UBD} \subset P$ a finite set, the upper-bounding problem is formulated as:*

$$\begin{aligned} \phi^{UBD} = & \min_{\mathbf{x} \in X} \phi(\mathbf{x}) \\ \text{s.t. } & g(\mathbf{z}(\mathbf{x}, \mathbf{p}), \mathbf{x}, \mathbf{p}) \leq -\epsilon^g, \forall \mathbf{p} \in P^{UBD}, \end{aligned}$$

where $\epsilon^g > 0$ is the restriction parameter [164], representing a parameter for perturbing the right-hand side of the semi-infinite constraint away from zero, thereby restricting the feasible set of the upper-bounding problem. Note that according to Mitsos [164], the upper-bounding problem should be solved to global optimality to obtain a global solution of the original SIP (5.6), but a valid upper bound $\phi^{UBD} \geq \phi^$ can be obtained by a local solution $\bar{\mathbf{x}}$ of the upper-*

bounding problem if its feasibility in the original SIP (5.6) is verified. That is, any SIP feasible point is a valid upper bound.

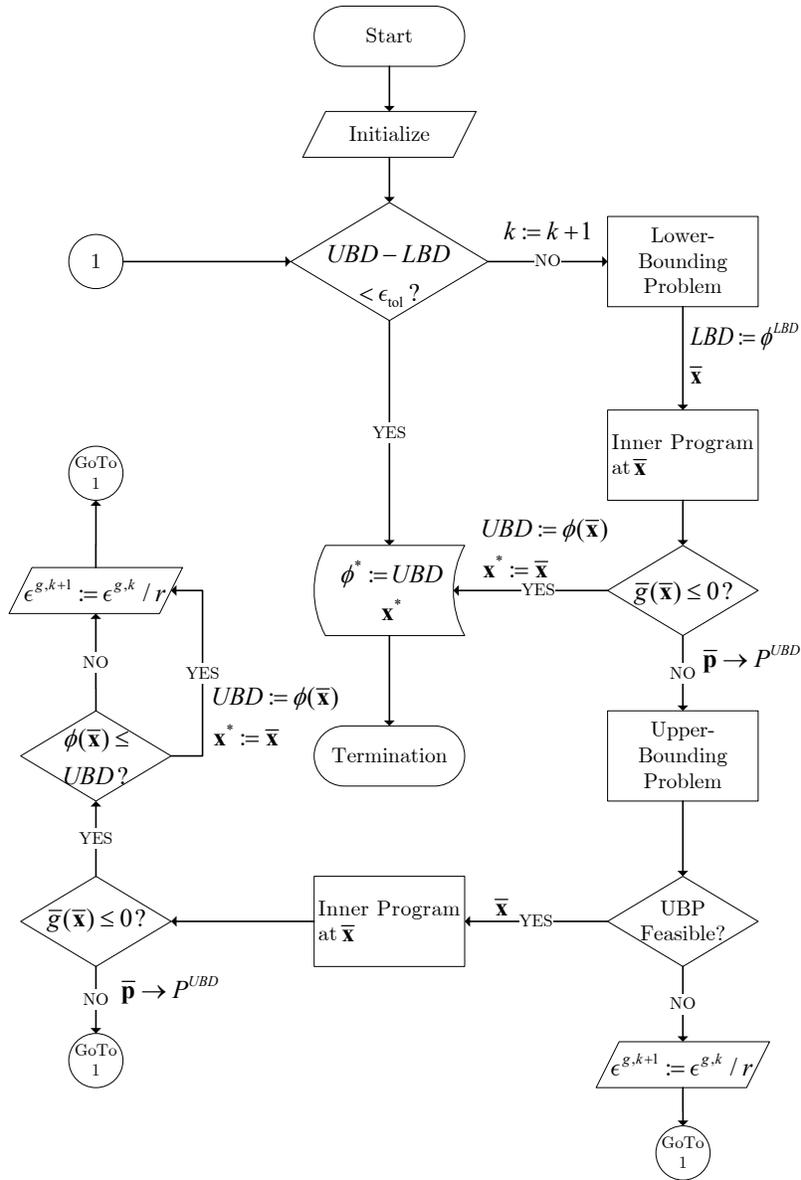


Figure 5.3: The SIPres algorithm is illustrated as a flowchart as adapted from that presented by Stuber and Barton [165].

5.6 Case Studies

All numerical experiments in this work were run on a single thread of an Intel Xeon E3-1270 v5 3.60/4.00GHz (base/turbo) processor with 16GB ECC RAM allocated to a virtual machine running the Ubuntu 18.04LTS operating system and Julia v1.6.1 [53]. Absolute and relative convergence tolerances for the B&B algorithm of 10^{-4} were specified for all example problems, unless otherwise noted and a maximum CPU time limit of the SIPres algorithm was set to 3600 seconds. The EAGO.jl package (v0.6.1) [52] was used to solve each optimization problem. Validated interval arithmetic was computed using the package IntervalArithmetic.jl [213]. The Intel MKL package (2019 Update 2) [214] was used to perform all LAPACK [215, 216] and BLAS [217] routines. The data used with and generated from the following numerical examples are openly available in the Git repository established at <https://github.com/PSORLab/RobustHybridModels> along with the corresponding problem formulations.

5.6.1 Case Study 1: Robust Feasibility of a Nitrification CSTR

In this case study, we consider the rigorous verification of robust feasibility of a continuous stirred-tank reactor (CSTR) undergoing nitrification reactions for wastewater treatment. The aim here is to verify the existence of a simple robust control policy that maintains the desired water quality specifications. The system involves a single continuously-flowing feed stream and a single continuously-flowing outlet stream, as shown in Figure 5.4. An air diffuser exists at the bottom of the tank to provide oxygen for oxidizing ammonium. The controller receives feedback signals from the conductivity sensor in the reactor and sends

a control signal to the valve on the air stream to increase or decrease the flow of air (i.e., aeration) into the CSTR that, in turn, controls the nitrification reactions. In practice, this aerobic nitrification step often precedes an anaerobic nitrification step.

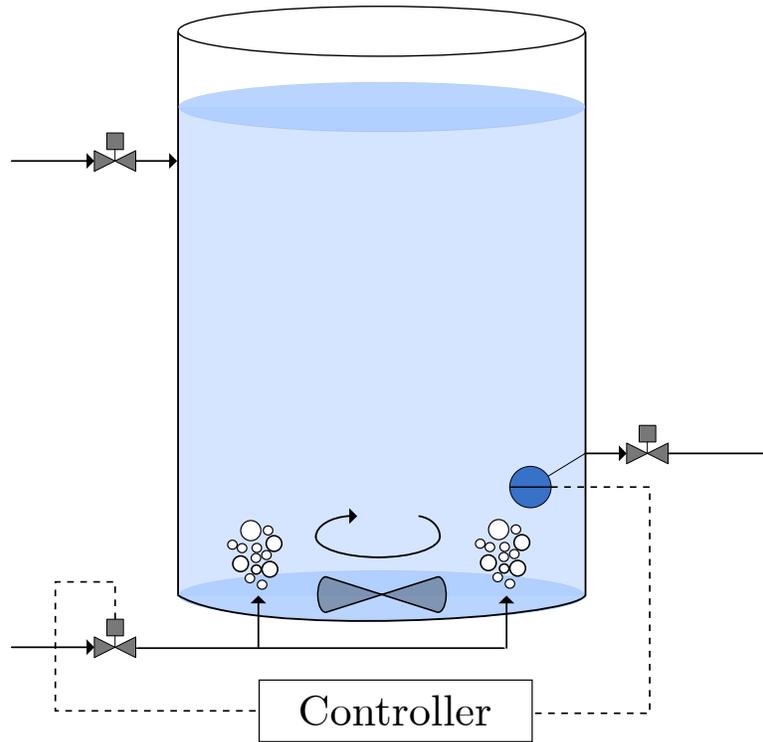
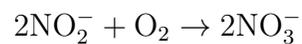


Figure 5.4: The dynamic nitrification CSTR system considered in the robust feasibility case study (Case Study 1), is shown. Under normal operation, oxygen is injected to control the nitrification reactions using a feedback controller utilizing measurements of the ammonium and dissolved oxygen concentrations in the outlet stream.

Hybrid Model Formulation

The reaction mechanism for this nitrification process has two steps:



In the first step, the ammonium ions are oxidized to nitrite ions. In the second step, the nitrite ions are further oxidized to nitrate ions. Based on the molecular biological study, the ammonia oxidizing bacteria (AOB) predominates the first step and the nitrite oxidizing bacteria (NOB) carries out the second step [218, 219]. The dynamic species mass balances in the CSTR are given by the ODEs [220]:

$$\begin{aligned}
\frac{dC_{NH}}{dt} &= \frac{1}{V}(\dot{m}_{in}C_{in} - \dot{m}_{out}C_{NH}) - r_{AO} \cdot X_{AO} \\
\frac{dC_{NI}}{dt} &= r_{AO} \cdot X_{AO} - r_{NO} \cdot X_{NO} \\
\frac{dC_{NA}}{dt} &= r_{NO} \cdot X_{NO} \\
\frac{dC_O}{dt} &= -r_{AO} \cdot \Psi_{AO} \cdot X_{AO} - r_{NO} \cdot \Psi_{NO} \cdot X_{NO} + k_{la} \cdot (C_O^* - C_O),
\end{aligned} \tag{5.10}$$

where C_{NH} , C_{NI} , and C_{NA} are the concentrations (mg N/(L · s)) for NH_4^+ , NO_2^- , and NO_3^- , respectively, C_O is the oxygen concentration (mg O_2 /(L · s)), \dot{m}_{in} and \dot{m}_{out} are continuous inlet and outlet flow rates (L/s), C_{in} is the NH_4^+ concentration in the inlet stream, and V is the reactor volume (1000 L). The ammonium oxidation rate (mg N- NH_4^+ / (g VSS_{AO} · min)) is given by r_{AO} , X_{AO} is the concentration of AOB (mg VSS/L), r_{NO} is the nitrite oxidation rate (mg N- NO_2^- / (g VSS_{NO} · min)), X_{NO} is the concentration of NOB (mg VSS/L), Ψ_{AO} is the stoichiometric ratio between oxygen and ammonia (mg O_2 / mg N- NH_4^+), Ψ_{NO} is the stoichiometric ratio between oxygen and nitrite (mg O_2 / mg N- NO_2^-), k_{la} is the volumetric mass transfer coefficient (s^{-1}), and C_O^* is the dissolved oxygen saturation concentration (9.1 mg/L at 20 °C, [221]). The rate equation for nitrite oxidation r_{NO} can be expressed further

as:

$$r_{NO} = r_{NO,\max} \frac{C_{NI}}{K_{SNO} + C_{NI} + \frac{C_{NI}^2}{K_{INO}}} \cdot \frac{C_O}{K_{ONO} + C_O},$$

where $r_{NO,\max}$ is the maximum nitrite consumption rate (mg N-NO₂⁻ / (g VSS_{NO} · min)), K_{SNO} is the Monod constant of nitrite for NOB (mg N-NO₂⁻/L), K_{INO} is the inhibition constant of nitrite for NOB (mg N-NO₂⁻/L), and K_{ONO} is the Monod constant of oxygen for NOB (mg/L).

The aeration process is governed by the mass transfer of oxygen into the solution as the term $k_{la}(C_O^* - C_O)$ in (5.10), that is derived from the standard oxygen transfer rate (SOTR, mg/s) defined as: SOTR = $k_{la}C_O^*V$ [222]. Assuming that the air flow rate from the air diffuser is represented by Q (mg/s), the mass flow rate of oxygen W_O in the air stream can be computed from an empirical formula: $W_O = 0.2967Q$ [223]. Then, the standard oxygen transfer rate can be calculated as SOTR = SOTE · W_O , where SOTE is the standard oxygen transfer efficiency (%). Therefore, the aeration mass transfer coefficient can be rewritten as:

$$k_{la} = \frac{0.2967Q \cdot \text{SOTE}}{C_O^* \cdot V}.$$

The parameter values used in the model are listed in Table 5.1.

Since this is a complicated biological reaction system in a physicochemical environment, it is very hard to obtain accurate kinetic parameters under constantly varying conditions for FPMs. There are situations such that the biological parameters cannot be easily obtained and verified by experiments. Thus, we propose to use an ANN model to estimate the rate constant r_{AO} and account for the hybrid modeling approach in this study. The rate

Symbol	Definition	Value	Reference
V	Liquid volume (L)	1000	This study
\dot{m}_{in}	Inlet volumetric flow rate (L/s)	4.167	This study
\dot{m}_{out}	Outlet volumetric flow rate (L/s)	4.167	This study
C_O^*	Saturated oxygen concentration (mg O ₂ /L)	9.1	[221]
X_{AO}	Concentration of AOB (mg VSS/L)	505	[220]
X_{NO}	Concentration of NOB (mg VSS/L)	151	[220]
$r_{NO,max}$	Max. nitrite consumption rate (mg N-NO ₂ ⁻ / (g VSS _{NO} · min))	1.07	[220]
Ψ_{AO}	Stoich. ratio of oxygen to ammonia (mg O ₂ / mg N-NH ₄ ⁺)	2.5	[224]
Ψ_{NO}	Stoich. ratio of oxygen to nitrite (mg O ₂ / mg N-NO ₂ ⁻)	0.32	[224]
K_{SNO}	Monod constant of nitrite for NOB (mg N-NO ₂ ⁻ /L)	1.6	[224]
K_{INO}	Inhibition constant of nitrite for NOB (mg N-NO ₂ ⁻ /L)	13000	[224]
K_{ONO}	Monod constant of oxygen for NOB (mg/L)	1.5	[224]
SOTE	Standard oxygen transfer efficiency (%)	10	[222]

Table 5.1: The model parameters used for the nitrification CSTR case study are listed in this table.

constant r_{AO} is related to both ammonium concentration c_{NH} and oxygen concentration c_O .

Consequently, r_{AO} is calculated as an intermediate variable from the ANN and substituted into the dynamic hybrid model, as illustrated in Figure 5.5 to form a dynamic serial hybrid model.

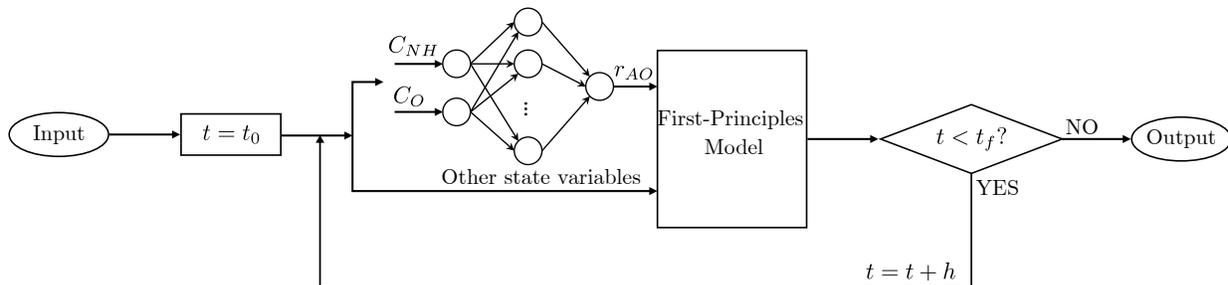


Figure 5.5: The hybrid model architecture used for the nitrification CSTR case study (Case Study 1), is illustrated. This model represents a DDM/FPM serial architecture with a reaction rate term modeled by an ANN.

Data-Driven Model Construction

A training data set was generated by evaluating a proposed empirical model for r_{AO} provided by Sánchez et al. [220], that relates r_{AO} to C_{NH} and C_O . The ANN model is developed to demonstrate a hybrid modeling approach in more complicated reacting systems. A preliminary investigation established physically plausible ranges for C_{NH} and C_O of $[0, 40]$ and $[0, 9.1]$, respectively, while values within $[0, 4]$ and $[0, 1]$, respectively, were typically observed from exploratory simulations of the system. For $(C_{NH}, C_O) \in [0, 4] \times [0, 1]$, r_{AO} varies significantly with respect to C_{NH} and C_O , while it is relatively flat outside this region. The empirical model was evaluated over two distinct Latin hypercube sampling (LHC) designs with 10^5 points each: one design on the domain $[0, 40] \times [0, 9.1]$, and the other on the domain $[0, 4] \times [0, 1]$, to ensure adequate sampling of the sensitive region. As with the previous examples, the data set was scaled using a min-max normalization and divided randomly into training (70%), validation (15%), and test (15%) sets. Training was performed using the Keras [225] module in the nightly version of Tensorflow [226] with the Adam optimizer. An early stopping protocol was performed using default parameters to prevent overfitting [227]. An ANN consisting of two hidden layers, each containing eight neurons, with the hyperbolic tangent activation function and a sigmoid output layer was used. This network was trained using a learning rate schedule that began with a value of 0.1 and was decreased by a factor of 0.5 every 100 epochs. This training protocol lead to loss values of 1.837×10^{-6} and 1.9708×10^{-6} for the training and validation sets, respectively.

SIP Formulation and Results

According to the standard of moderate municipal wastewater [100], the concentration of ammonium ions in the effluent is required to be below 30 mg N-NH₄⁺/L. Moreover, the dissolved oxygen concentration must be lower than 2 mg O₂/L to ensure that a viable operating window exists for a secondary anaerobic denitrification step [228]. Thus, the upper specification limits for ammonium ions (USL_{NH}) and dissolved oxygen (USL_O) are set to 30 mg N-NH₄⁺/L and 2 mg O₂/L, respectively. The CSTR is initially operating at steady state with a constant concentration (31 mg N-NH₄⁺/L) in the inlet stream. At some moment ($t = t_0$), a concentration shock is observed in the inlet stream within a short operating window (20 s) of the treatment process. It is our desire to operate the process in an open-loop manner, and so the objective here is to determine whether there exists a feasible design for the valve setting for air flow rate that is robust to worst-case realizations of uncertainty at the end of the simulation horizon ($t = t_f = 100$ s). The design variable is taken to be the air flow rate $d = Q \in D = [440, 2000]$, which can be interpreted as the valve setting. The uncertainty comes from the disturbance in the inlet stream $\pi = C_{in} \in \Pi = [31.0, 40.0]$. Thus, the design under uncertainty feasibility problem accounts for the following semi-infinite constraints:

$$C_{NH}(t_f, d, \pi) - \text{USL}_{\text{NH}} \leq 0, \quad \forall \pi \in \Pi, \quad (5.11)$$

$$C_O(t_f, d, \pi) - \text{USL}_O \leq 0, \quad \forall \pi \in \Pi.$$

The two semi-infinite constraints present in (5.11) are then reformulated as a single nonsmooth semi-infinite constraint, and an epigraph rearrangement of the problem is made to yield the following SIP:

$$\begin{aligned} \eta^* = & \min_{d \in D, \eta \in H} \eta & (5.12) \\ \text{s.t. } & \max\{C_{NH}(t_f, d, \pi) - \text{USL}_{NH}, C_O(t_f, d, \pi) - \text{USL}_O\} \leq \eta, \quad \forall \pi \in \Pi. \end{aligned}$$

This formulation corresponds with the design under uncertainty feasibility problem (5.7). Again, the term η represents a measure of robust feasibility. If the optimal solution value of the feasibility problem (5.12) satisfies $\eta^* \leq 0$, then a design exists that is robust to worst-case realization of uncertainty. For this problem, a relative convergence tolerance of 10^{-3} for the SIP-feasibility problem (5.12) was used. An explicit Euler method was used to integrate (5.10) with a stepsize of $h = 10$ s. To avoid domain violations and associated difficulties that arise from overestimation of C_{O_2} , a positive value of C_{O_2} was enforced by setting $C_{O_2} = \max(C_{O_2}, \epsilon)$ with $\epsilon = 10^{-10}$ at each time step. The SIPres algorithm [164] (see Figure 5.3) was used to solve the SIP given in (5.12). The SIPres algorithm solves (5.12) after a single iteration in 21.86 CPU seconds with an optimal solution $\eta^* = 0.288$, illustrating that a robust design does not exist for this system with respect to the given performance/safety specifications.

This motivates a search for alternative approaches to verify robustness. Namely, we seek to determine if a robust operation is feasible. We consider the same uncertainty source $\pi \in \Pi$ and the control variable is taken as $u = Q \in U = [440, 2000]$. We aim to establish a robust operation problem to verify if a control recourse exists that mitigates the impacts of

uncertainty. The semi-infinite constraints in this problem are:

$$C_{NH}(t_f, \pi, u) - \text{USL}_{NH} \leq 0, \quad \forall u \in U, \quad (5.13)$$

$$C_O(t_f, \pi, u) - \text{USL}_O \leq 0, \quad \forall u \in U.$$

Accordingly, the operation under uncertainty feasibility problem can be expressed as the following SIP:

$$\eta^* = \max_{\pi \in \Pi} \eta \quad (5.14)$$

$$\text{s.t. } \eta \leq \max\{C_{NH}(t_f, \pi, u) - \text{USL}_{NH}, C_O(t_f, \pi, u) - \text{USL}_O\}, \quad \forall u \in U.$$

Again, the SIPres algorithm [164] with convex/concave envelopes of activation functions[229] was used to solve the SIP (5.14). The SIPres algorithm terminates with $\eta^* = 0.288$ after a single iteration in 21.14 CPU seconds. As a consequence, we see that a control setting recourse is not feasible given the provided specifications. Moreover, the presented formulations with dynamic hybrid models demonstrate the applicability of robustness verification approaches to relatively complicated processes with dynamic governing equations.

5.6.2 Case Study 2: Worst-Case Design of Subsea Production Facilities - Mitigation of Domain Violations

In Stuber et al. [14], the worst-case design of a subsea oil production facility (illustrated in Figure 5.6) was formulated as an operation under uncertainty feasibility problem and solved using several novel methodologies. Namely, the problem was reformulated as an SIP with

implicit functions embedded. The subsea separator model uses transcendental functions with definitions on narrow domains that result in numerical difficulties when simulating and optimizing the system. For the purposes of this paper, the interest is not in the application itself, but in the model as representative of a broader class of industrially-relevant examples plagued by numerical simulation and convergence issues caused by domain violations. Within this context, it is of interest to explore how hybrid modeling approaches might be used to improve the robustness of an FPM and solvers (i.e., improve the reliable convergence to accurate solutions).

Domain violations are ubiquitous across process systems engineering applications and pose major challenges to researchers and practitioners of simulation and optimization [230, 231, 232]. Within the broader context of numerical simulation, domain violations are encountered when a solver attempts to evaluate an expression at a point outside of its defined domain (e.g., divide by zero or square-root a negative number). Hybrid models may pose additional challenges as they may also suffer from violations of their domains of validity. That is, a solver may attempt to evaluate a DDM at a point outside of the domain of inputs for which the DDM is considered to be “valid” (i.e., accurately represents reality). When considering the optimization of hybrid models, domain violations may be frequently encountered when such domains may not be explicitly known and accounted for with appropriate constraints, without prior analysis.

In Stuber et al. [14], a method of forward-backward interval constraint propagation on the DAG [198, 233], interval contractor methods [43], a novel convex/concave relaxation algorithm [210], and a novel algorithm for solving SIPs [165] were all necessary to solve this problem. While these methods adequately address the problem in question, the broad

and robust applicability of this approach to more general SIPs is wanting. We should note, however, that this approach reduces the problem in question from a GSIP to that of a standard SIP. This, combined with a desire to generalize the prior results to allow for the incorporation of more complex physical phenomena, further motivates our interest in this example.

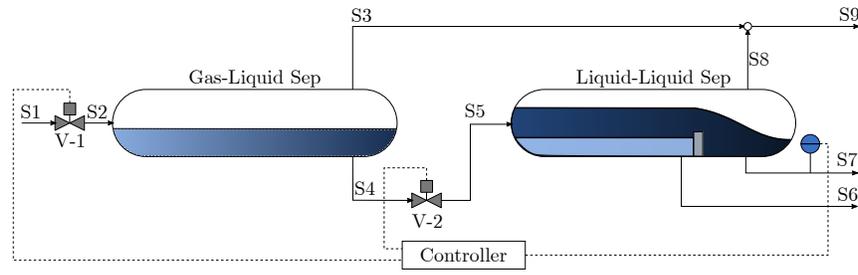


Figure 5.6: The process flow diagram for the subsea separator (adapted from Stuber et al. [14]), is presented in this figure. This system is considered in the subsea separator case study for the use of hybrid models to overcome numerical domain violation issues. A mixture of gas, oil, and water is fed to the system in S1. Gas is separated from the oil-water mixture in the gas-liquid separator and oil is separated from water in the liquid-liquid separator.

Hybrid Model Formulation

In this study, the focus is on a modification of the gas-liquid/liquid-liquid separation train problem presented in Case 3 of Stuber et al. [14]. To model the performance of gas separation in each separator, simple exponential decay models based on mean gas bubble sizes were assumed [14]. The relationship between inlet and outlet gas quantities may be expected to change in meaningful ways when a population-based model of bubble sizes is incorporated along with information about the equipment’s geometry. Moreover, for bubbly mixtures, overflow can occur in volumes less than those considered by solely taking into account liquid levels, provided a large gas concentration is present in the inlet. In practice, this type of problem is typically characterized using a mixture of computational fluid dynamics software

and empirical investigation.

We propose simplifying the published model by using an ANN surrogate model to relate the input variables to the gas-liquid separator and the control variable for the second valve (V-2) to the system outputs. This serves to eliminate the domain violation issue inherent in the model, as the activation functions considered lack domain restrictions, and allow the system-level model to be readily generalized to incorporate information from computational experiments generated by CFD models, or elsewhere. The inputs, outputs, and expected ranges of each variable in each ANN are summarized in Table 5.2. As the development of CFD models is often time consuming, equipment specific, and not the central focus of this work, we will forgo this and instead illustrate how this approach works at the system level. We use the prior mass balances and process specifications for the gas-liquid separator (GLS) and the liquid-liquid separator (LLS). The governing equations for the first valve (V-1), and the gas mixer will be left unaltered. The equations governing V-1 lead to the following simplified relationships:

$$\begin{aligned}\xi_{W,1} &= 1 - \xi_{G,1} - \xi_{O,1}, \\ SG_{mix}^{-1} &= \frac{\xi_{G,1}}{SG_G} + \frac{\xi_{W,1}}{SG_W} + \frac{\xi_{O,1}}{SG_O}, \\ \dot{m}_2 &= u_1 C_{v1} \sqrt{SG_{mix}^{-1} (P_{well} - P_{GLS}) + \epsilon_d} \\ \xi_2 &= \xi_1.\end{aligned}\tag{5.15}$$

These equations specify that the mass fractions in the input stream ($\xi_{W,1}$, $\xi_{G,1}$, $\xi_{O,1}$) sum to one, provide a formula relating specific gravity of the mixture SG_{mix} to the specific gravity

of individual components (SG_G , SG_W , SG_O), and relate the mass flow rate through the valve \dot{m}_2 to valve position (u_1), valve coefficient (C_{v1}) and a specified pressure difference ($P_{well} - P_{GLS}$) between the GLS and the wellhead. A small number $\epsilon_d = 10^{-6}$ is added to the argument of the $\sqrt{\cdot}$ function to avoid the introduction of numerically ill-posed gradients that present computational issues for local NLP subproblems encountered during global optimization.

Simple algebraic substitutions of the equations governing V-2 and the LLS behavior lead to the following algebraic expression:

$$\xi_{G,7} = \xi_{G,4} \exp \left(-\dot{m}_4 k_{LLS} \frac{V_{LLS}}{\rho_4 + \epsilon_d} \right). \quad (5.16)$$

While additional expressions are required to fully determine all stream characteristics in the flowsheet, the LLS performance specification (5.16) is sufficient to construct the SIP constraint. This specification relates the inlet gas mass fraction $\xi_{G,4}$, density ρ_4 , and mass flow rate \dot{m}_4 to the oil product stream gas mass fraction $\xi_{G,7}$ by means of a performance constant k_{LLS} . Due to downstream equipment specifications, the oil product stream gas mass fraction may not exceed the value $G_{max} = 0.05$. The full model can be found in Stuber et al. [14] with the analysis of the DAG in Stuber [198, Sec. 8.1].

Variable	Lower	Upper	Unit	Layer
\dot{m}_2	8.228	19.517	kg/s	Input
u_2	0.35	0.8	-	Input
$\xi_{G,2}$	0.35	0.5	-	Input
$\xi_{W,2}$	0.1	0.25	-	Input
\dot{m}_4	541.364	845.881	kg/s	Output
H_{GLS}	0.462165	0.7992	m	Output
$\xi_{G,4}$	9.463053×10^{-3}	0.36	-	Output
P_4	4.00264×10^6	4.01079×10^6	Pa	Output
ρ_4	584.6	1376.6	kg/m ³	Output

Table 5.2: The state variables for the subsea separator case study are listed in this table along with their corresponding bounds, units, and identification of whether they are classified as inputs or outputs for the hybrid model. Bounds directly specified by Stuber et al. [14] were used if available. Otherwise, natural interval extensions of known quantities were used to compute necessary values. The parameters C_{v1} , SG_G , SG_W , SG_O , g_a , P_{well} , P_{LLS} , P_{GLS} , k_{GLS} , L_{GLS} , and R_{GLS} take the values previously specified in Stuber et al. [14].

Data-Driven Model Construction

Training data was generated by repeatedly solving a feasibility problem equivalent to the nonlinear system:

$$\begin{aligned}
(\xi_{G,2} - 1)\dot{m}_2 - (\xi_{G,4} - 1)\dot{m}_4 &= 0 \\
u_2^2 C_{v2}^2 \rho_W^o (P_4 - P_{LLS}) - \rho_4 \dot{m}_4^2 &= 0 \\
(P_4 - P_{GLS}) - \rho_4 g_a H_{GLS} &= 0 \\
\xi_{G,2} \exp\left(-k_{GLS} \rho_4 \frac{(\xi_{G,4} - 1)V_{GLS}(H_{GLS})}{(\xi_{G,2} - 1)\dot{m}_2}\right) - \xi_{G,4} &= 0 \\
\rho_4 \frac{\xi_{G,4}}{SG_G} + \rho_4 \frac{\xi_{G,2}(\xi_{G,4} - 1)}{SG_W(\xi_{G,2} - 1)} + \rho_4 \frac{\xi_{G,2}(\xi_{G,4} - 1)(1 + \xi_{W,2} - \xi_{G,2})}{SG_O(\xi_{G,2} - 1)} - \rho_W^o &= 0 \\
V_{GLS} - L_{GLS} \left((H_{GLS} - R_{GLS}) \sqrt{(2R_{GLS}H_{GLS} - H_{GLS}^2)} + R_{GLS} \cos^{-1} \left[1 - \frac{H_{GLS}}{R_{GLS}} \right] \right) &= 0
\end{aligned} \tag{5.17}$$

that is parameterized by $\mathbf{w} = (\dot{m}_2, u_2, \xi_{G,2}, \xi_{W,2}) \in W$. Ipopt [234] was used to solve (5.17) with a multistart approach using 16 initial guesses chosen via an LHC sampling procedure for each set of parameters considered. An LHC sampling procedure was then performed over a range of valid values given in Table 5.2 to generate 10^5 data points used to train the DDM. As noted in Stuber et al. [14], the implicit function characterized by (5.17) may not exist for some realization of uncertainty and control variables. Values that yielded a locally-infeasible result were labelled accordingly, while the solutions of the feasible problems were saved. Of the 10^5 points generated, 6,742 infeasible points were evaluated.

The approach to training the ANNs for this problem, parallel the previous examples. The data set was scaled using a min-max normalization and divided randomly into training (70%), validation (15%), and test (15%) sets. Training was performed using the Keras [225] module in the nightly version of Tensorflow [226] with the Adam optimizer. The surrogate ANN consisted of four inputs, two dense layers, twelve neurons per layer, and utilized the SiLU activation function. A sigmoid output layer was used to ensure that the output results remained within the range of the training data. The surrogate model had min-max-scaled mean-squared-error (MSE) values of 7.74×10^{-5} and 2.2506×10^{-4} on the training and test sets, respectively. The validity constraint consists of an ANN with four inputs, two hidden layers, two neurons per layer, and utilizes the SiLU activation functions with a single hyperbolic tangent output layer that is trained using a binary cross-entropy loss function. This achieved a binary accuracy greater than 99.0% on both the test and training sets. Weights and offsets for both the surrogate model and the validity constraint can be found in the Git repository. Both the surrogate and classifier ANNs used a learning rate schedule that began with a value of 0.1 and was decreased by a factor of 0.5 every 100 epochs. We note

here that, due to the nature of the application, no classifier can be expected to be exactly accurate as the valid and invalid regions adjoin one another.

SIP Formulation and Results

Any ANN can only be expected to provide valid results when interpolating and special consideration must be given to exclude invalid operating parameters. In general, two distinct outcomes must be considered: either a domain violation arises from a purely numerical consideration (e.g., instability) or one that corresponds to a nonphysical operating condition (e.g., negative density). In the former case, the accuracy of the hybrid model should be verified to guarantee the results for the corresponding robust operation problem. In the latter case, restricting the model to a domain of validity is sufficient to ensure a guarantee of robustness.

Ensuring validity regions for surrogate models remains an active area of research within the optimization community. Some approaches include restricting the function evaluations to be within the convex hull of a finite number of sampled points [173, 235] or categorizing the data using a support vector machine [180, 236]. In either case, this restriction can be framed as a potentially nonconvex constraint $g_c : Z \times \Pi \times U \rightarrow \{-1, 1\}$ where $g_c(\hat{\mathbf{z}}, \boldsymbol{\pi}, \mathbf{u}) = -1$ indicates a valid model for $(\hat{\mathbf{z}}, \boldsymbol{\pi}, \mathbf{u}) \in Z \times \Pi \times U$. We note that the forms addressed pertain to standard optimization formulations and the extension of these approaches to multilevel programs has yet to be developed. In keeping with surrogate modeling frameworks adopted in this paper, we choose to make use of a second ANN, $f_c^{ANN} : Z \times \Pi \times U \rightarrow \mathbb{R}$, in addition to the surrogate model, to perform a binary classification task via logistic regression.

The binary classification task is performed as follows. Provided that $f_c^{ANN}(\hat{\mathbf{z}}, \boldsymbol{\pi}, \mathbf{u}) \leq 0$,

the input features is classified as $g_c(\hat{\mathbf{z}}, \boldsymbol{\pi}, \mathbf{u}) = -1$ (valid classification). In a corresponding manner, the classification ANN predicts that the the input features will be classified as $g_c(\hat{\mathbf{z}}, \boldsymbol{\pi}, \mathbf{u}) = +1$ (invalid classification) due to a domain violation $f_c^{ANN}(\hat{\mathbf{z}}, \boldsymbol{\pi}, \mathbf{u}) > 0$. With this validity constraint, the robust feasibility constraint takes the logical form:

$$\forall \boldsymbol{\pi} \in \Pi, \exists \mathbf{u} \in U : g(\hat{\mathbf{z}}, \boldsymbol{\pi}, \mathbf{u}) \leq 0 \wedge g_c(\hat{\mathbf{z}}, \boldsymbol{\pi}, \mathbf{u}) \leq 0 \wedge \mathbf{h}(\hat{\mathbf{z}}, \boldsymbol{\pi}, \mathbf{u}) = \mathbf{0}. \quad (5.18)$$

For this problem, the state variables $\hat{\mathbf{z}}$ can be calculated as an explicit function $\mathbf{z} : \Pi \times U \rightarrow Z$ such that $\mathbf{h}(\mathbf{z}(\boldsymbol{\pi}, \mathbf{u}), \boldsymbol{\pi}, \mathbf{u}) = \mathbf{0}$ for every $(\boldsymbol{\pi}, \mathbf{u}) \in \Pi \times U$. The robust operation problem can then be formulated as an SIP with a nonsmooth semi-infinite constraint:

$$\begin{aligned} \eta^* &= \max_{\boldsymbol{\pi} \in \Pi, \eta \in H} \eta & (5.19) \\ \text{s.t. } \quad & \eta \leq \max \{g(\mathbf{z}(\boldsymbol{\pi}, \mathbf{u}), \boldsymbol{\pi}, \mathbf{u}), g_c(\mathbf{z}(\boldsymbol{\pi}, \mathbf{u}), \boldsymbol{\pi}, \mathbf{u})\}, \forall \mathbf{u} \in U. \end{aligned}$$

Alternatively, (5.19) may be reformulated as an SIP with a disjunctive constraint or as a mixed-integer SIP. Note that this form is identical to the structure encountered when relaxing a GSIP and the reader is directed to Mitsos and Tsoukalas [168] for a discussion of the numerical eccentricities associated with solving that problem class. The robust design

problem for the subsea separator may then be formally stated as:

$$\begin{aligned}
\eta^* &= \max_{\boldsymbol{\pi} \in \Pi, \eta \in H} \eta & (5.20) \\
\text{s.t. } \eta &\leq \max \{ \xi_{G,\tau}(\boldsymbol{\pi}, \mathbf{u}) - G^{max}, g_c(\mathbf{z}(\boldsymbol{\pi}, \mathbf{u}), \boldsymbol{\pi}, \mathbf{u}) \}, \quad \forall \mathbf{u} \in U \\
U &= [0.35, 0.8]^2 \\
\Pi &= [0.35, 0.5].
\end{aligned}$$

We note that the valid region of the developed binary classifier is bounded by a 0-sublevel set, which is potentially a disconnected and nonconvex set, and therefore the following equivalence can be established:

$$\{(\boldsymbol{\pi}, \mathbf{u}) \in \Pi \times U : g_c(\mathbf{z}(\boldsymbol{\pi}, \mathbf{u}), \boldsymbol{\pi}, \mathbf{u}) = -1\} \Leftrightarrow \{(\boldsymbol{\pi}, \mathbf{u}) \in \Pi \times U : g_t(\mathbf{z}(\boldsymbol{\pi}, \mathbf{u}), \boldsymbol{\pi}, \mathbf{u}) \leq 0\},$$

with $g_t(\cdot, \cdot, \cdot) \equiv f_c^{ANN}(\cdot, \cdot, \cdot)$. By construction, g_t is continuous on its domain, and so this reformulation ensures that the semi-infinite constraint is continuous, and in turn, ensures that the convex/concave relaxations used in the subproblem of the SIPres algorithm [164] exhibit desirable convergence properties [237]. Under this equivalence, the SIP (5.20) is

reformulated as:

$$\begin{aligned}
 \eta^* &= \max_{\boldsymbol{\pi} \in \Pi, \eta \in H} \eta & (5.21) \\
 \text{s.t. } & \eta - \max \{ \xi_{G,7}(\boldsymbol{\pi}, \mathbf{u}) - G^{max}, g_t(\mathbf{z}(\boldsymbol{\pi}, \mathbf{u}), \boldsymbol{\pi}, \mathbf{u}) \} \leq 0, \quad \forall \mathbf{u} \in U \\
 & U = [0.35, 0.8]^2 \\
 & \Pi = [0.35, 0.5].
 \end{aligned}$$

We first solved this hybrid model using the SIPres [164] routine provided in EAGO v0.6.1 [52, 141] and using the convex/concave envelope of SiLU described in a forthcoming work [229]. The SIP was solved to an absolute tolerance of 10^{-3} . The algorithm terminated in 3 iterations, taking 2.9 CPU seconds when using the envelope of SiLU when computing relaxations. The SIPres algorithm terminated after an optimal value was found in the lower-bounding problem and the maximal value of the corresponding lower-level problem was found to be nonpositive with a value of $\eta^* = -6.6 \times 10^{-4}$. In contrast, the original method in Stuber et al. [14] provided a solution value of -5.77×10^{-3} for this case study. However, it is worth noting that the method proposed by Stuber et al. [14] has an early-termination criterion whereby the algorithm terminates with a feasible suboptimal solution as soon as robustness is verified. Thus, the solution value obtained by Stuber et al. [14] is an upper bound on the global solution. Despite this, we notice that $\eta^* > -5.77 \times 10^{-3}$, seemingly in violation of the upper bound for the full mechanistic model [14]. Since the hybrid model utilizes an ANN to approximate the original equations exhibiting numerical issues (i.e., domain violations), such discrepancies are anticipated. The level of confidence in the solution lies in the accuracy of

the trained model versus the constraint satisfaction and algorithm convergence tolerances. In practice, it may be possible to verify SIP feasibility of an optimal solution with respect to the full mechanistic model. However, this depends entirely on the existence and complexity of such a model. For this case, the results verify that both models ensure the robust feasibility of this operation. A performance normalization was used based on CPU single-core IPC using the Cinebench R15 (Maxon, Newbury Park, CA) single-core benchmark to enable a fair comparison of the performance of the approach in this work versus Stuber et al. [14]. The normalized results indicate a 70-fold performance improvement over the original solution time of 549.3 CPU seconds reported by Stuber et al. [14]. In this particular case, we expect this improvement to be genuine as prior comparisons of Julia/EAGO to C++/MC++ implementations differed only by at most a factor of three [238]. However, the degree of computational performance improvement for the surrogate modeling approach relative to the original work of Stuber et al. [14] will undoubtedly be model-specific. As such, we make no broad claim of superior performance for this method. However, this example does illustrate that the use of surrogate modeling represents a viable approach to eliminate the need to apply specialized parametric interval analysis [43, 198], constraint propagation techniques [14], and implicit relaxation [210] methods when addressing bilevel optimization problems with coupling equality constraints, by replacing these models with a formulation that can be readily addressed with standard global optimization solvers.

5.7 Extension to Implicit Forms

In the **Optimization of Hybrid Models** section, the assumption was made that a unique explicit closed-form function $\mathbf{z} : Y \rightarrow Z$ exists such that $\mathbf{h}(\mathbf{z}(\mathbf{y}), \mathbf{y}) = \mathbf{0}$ for every $\mathbf{y} \in Y$. This assumption was also made within the context of SIPs in the section **Semi-Infinite Optimization with Hybrid Models**. As mentioned in those sections, the explicit closed-form solution assumptions do not necessarily restrict the applicability of the approach. Stuber et al. [210] originally developed a theory for considering implicit functions within deterministic global optimization formulations. This was explored further within the context of SIPs by Stuber and Barton [165]. Summarily, Stuber and Barton [165] extended the SIP approach for solving (5.5) to the more general case that the equality constraints do not admit an explicit closed-form solution. In this section, we discuss the conditions under which these assumptions may be relaxed and extend the applicability to a broader class of hybrid models that may involve implicit functions, including implicit ANNs and general nonlinear mechanistic models.

The conditions for considering implicit functions are established as follows. In the previous sections, the only requirements of the equality constraints $\mathbf{h}(\hat{\mathbf{z}}, \mathbf{x}, \mathbf{p}) = \mathbf{0}$ of (5.5) representing a hybrid model, were that they are factorable and continuous. Here, we have the additional requirement that $\mathbf{h} : Z \times X \times P \rightarrow \mathbb{R}^{n_z}$ is continuously differentiable on its domain. Then, it must be assumed that there exists an implicit function $\mathbf{z} : X \times P \rightarrow Z$ such that $\mathbf{h}(\mathbf{z}(\mathbf{x}, \mathbf{p}), \mathbf{x}, \mathbf{p}) = \mathbf{0}$ for every $(\mathbf{x}, \mathbf{p}) \in X \times P$. For the appropriate theories and methods [165, 210] to hold, and therefore to be applicable to hybrid models, it must again be assumed that such a function \mathbf{z} is unique in the set Z . In other words, such a Z must exist

within which \mathbf{z} is unique on $X \times P$.

Conditions for guaranteeing uniqueness of \mathbf{z} in Z on $X \times P$ may be inferred from the structure of the feed-forward ANN (as an explicit input-output mapping) and under the conditions stipulated by the *semilocal implicit function theorem* (Neumaier [43, Thm. 5.1.3]). Furthermore, existence and uniqueness tests associated with parametric interval methods (e.g., interval Newton [43], Krawczyk [239], Hansen-Sengupta [43]) may be used to verify this condition. Note that this does not require that \mathbf{h} has a unique solution, and in the event that multiple solution branches of \mathbf{h} exist in $Z \times X \times P$, bisection-based methods may be sufficient to identify a partition such that the existence and uniqueness of an implicit function can be guaranteed for each element of the partition (see Stuber et al. [210, App. 1] and Stuber [198, Sec. 3.5] for discussions on this). Formal treatment of cases in which $Z \times X \times P$ encloses bifurcation points and/or multiple solution branches of \mathbf{h} remains an active area of research.

5.8 Conclusion

In this work, we formalized the foundations for SIPs with hybrid first-principles and data-driven models. Particular attention was paid to surrogate modeling via ANNs as the data-driven sub-models. A reduced-space SIP formulations with implicit functions embedded was proposed. The formulations of SIPs with hybrid models were demonstrated through three common types of robust design and optimization under uncertainty problems. The SIPres algorithm for SIPs with implicit functions [165] was used for solving two case studies to demonstrate practicability and superiority of our approaches.

In our first case study, we illustrated how an SIP containing a hybrid model may be used to solve robust feasibility problems pertinent to a continuous nitrification CSTR for wastewater treatment. The use of hybrid models in this application allowed a data-driven approach to describe kinetic rate parameters in biological systems. A reformulation to combine two semi-infinite constraints on ammonium and dissolved oxygen concentrations in the effluent was implemented. The SIP framework presented herein was shown to be sufficiently general such that it may readily address dynamic robust feasibility problems within the context of hybrid models.

The robust simulation of a horizontal gas-liquid and liquid-liquid separator train was revisited in the second case study. This problem is especially challenging as the modeling equations are plagued by numerical issues caused by domain violations. The domain violation problem was addressed with a novel approach that incorporates validity constraints and replaces the problematic models encountering domain violations with an ANN. This problem demonstrates how the application of hybrid models may overcome numerical difficulties often encountered when simulating complicated process systems models. Moreover, the incorporation of validity constraints naturally leads to a nonsmooth SIP formulation that may readily be reformulated as a mixed-integer problem, a disjunctive formulation, or a GSIP.

One interesting application of this work is the solution of problems that require representation by multiple distinct models. These may arise when modeling dynamical systems stemming from transport phenomena whose underlying physics change markedly for different realizations of decision and uncertainty values. In this case, we can generalize the approach detailed here to associate each model with a region of validity and a nonsmooth SIP formula-

tion of the optimization problem. The use of specialized forms of validity constraints should be considered, such as mixed-integer linear formulations, as an alternative to the general nonlinear formulation used herein. This may allow for the use of specialized algorithms that address GSIP formulations for larger and more complex applications.

Chapter 6

Conclusions

Modeling spatiotemporal systems is paramount to engineered systems due to its capability to provide accurate simulation, precise process control, and efficient system designs. Optimization provides immense opportunities for model validation, optimal control, and optimal design with respect to spatiotemporal systems. The work in this thesis investigated several studies using optimization to solve challenging engineering problems involving spatiotemporal systems in different engineering fields, to address their practicability in real-world applications. Chapter 1 summarized this high-level vision.

Chapter 2 investigated a challenging topic in global optimization theory with respect to spatiotemporal systems: solving PDE-constrained global optimization problems. An algorithm for bounding numerical solutions of transient parametric PDEs was presented. This algorithm combines a mixed IA/ AA approach and the differential inequalities method to overcome the dependency problem and reduce conservatism of the calculated state bounds. This method still requires a formal proof to certify its rigorousness. However, several examples were implemented to demonstrate the performance and empirical validity of this

method and to demonstrate its application in deterministic global optimization with PDE constraints. However, even with the most efficient bounding PDE method developed, the observed computational cost is still very high and proved to be relatively ineffective for solving some global optimization problems embedded with stiff PDE systems. Future studies of this work should continue to focus on effective algorithms to reduce conservatism of the global bounds, while reducing computational complexity as much as possible. Extending the relaxation method using DI [24] to PDE systems can be a promising direction since this method can provide significantly tighter relaxations and has been proven to be rigorous for ODE systems.

In Chapter 3, an application of spatiotemporal systems was addressed for a traditional chemical engineering application: water/wastewater treatment systems. Non-ideal mixing models were developed to accurately simulate solution properties and mixture heterogeneity. These models have advantages because their formulations are simple with only a few parameters. Thus, they are very fast to simulate compared to traditional CFD models. Moreover, deterministic global optimization ensured that the best-fit parameter values were obtained to validate the model. In addition, these models were established based on physical principles which provide a better understanding on mechanisms and can serve as a platform for simulating solution properties for other chemical species with similar physical principles. Furthermore, it is demonstrated that these simple models can be used to enable advanced closed-loop control strategies for better operation of wastewater treatment plants, which is another major contribution of this study. In particular, advanced MPC and EMPC strategies were studied for a nitrification step of a biological nutrient removal operation to demonstrate their performance and capability for disturbance rejection and optimal operation utilizing

high-resolution sensor data. The established model-based technology platform, consisting of different control strategies and architectures, addressed the practicability for precise control of water and wastewater treatment processes.

Chapter 4 explored the application of quantitative and formal methods to a spatiotemporal system in cancer research. This work established a systematic theoretical framework for modeling transport phenomena in a tumor that is validated using global optimization. In particular, in addition to the rigorous bounding approaches developed in Chapter 2, machine learning methods were used to establish surrogate models for solving challenging global optimization problems. Utilizing the validated model, the relative contributions of convective and diffusive transvascular transport were quantified. The novel findings of this analysis elucidated the influence of TME-normalizing therapy on the governing drug and nutrient transport mechanisms. Furthermore, systematic *in silico* model-based approaches were proposed for determining optimal dose selection for TME-normalizing therapy design and anticancer macromolecule size design. The information obtained through these approaches will aid in the development of more comprehensive models (e.g., incorporating the influence of vascular density) and provided deeper insight into the mechanistic transport phenomena in tumors to guide drug development and delivery. Since quantifying the impacts of uncertainty is critical for decision-making in cancer therapeutics, investigating robust design approaches in this field would be interesting future work.

Chapter 5 addressed worst-case design problems using hybrid modeling methods for safety-critical system under uncertainty. SIP formulations were proposed in this work for solving these problems. A major contribution of this study is that we formalized SIP formulations with hybrid first-principles and data-driven models of various architectures. Hybrid

models may be advantageous because they not only have the ability to capture unknown phenomena using data-driven approaches to provide practical solutions for modeling partially unknown systems, but also to reduce computational complexity for more efficient simulations. The formulations and solution strategies for different types of worst-case design problems with embedded hybrid models were demonstrated. Particular attention was paid to the hybrid models that use ANNs with novel activation functions that are of interest for deep learning. In the first case study of this work, we addressed how an SIP containing a hybrid model can be used to solve robust feasibility problems pertinent to a nitrification CSTR wastewater treatment process (which is related to the nitrification system in Chapter 3). The SIP framework presented in this case study addressed the methods for solving robust feasibility problems within the context of spatiotemporal systems governed by dynamic hybrid models. Another case study addressed the worst-case design of subsea production facilities. The most challenging part of this problem is that the system governing equations are plagued by domain violation issues. An approach incorporating validity constraints with an ANN model has the potential to overcome this numerical difficulty, and this approach demonstrated how hybrid modeling methods can help solve domain violation problems in complicated process system applications.

Appendix A

Supplementary Information

A.1 Supplementary Information for Chapter 3

A.1.1 Development of pH Mixing Models

The development of pH mixing models for three independent studies with shock injection in the high, middle, and low zone are introduced in this section. The high zone injection model is introduced in Section 3.2.2 in the Chapter 3, established as:

$$\begin{aligned}\frac{dH_1}{dt} &= \frac{1}{v_1 V} (H_0 + \kappa_1 H_4 - \kappa_1 H_1), \\ \frac{dH_2}{dt} &= \frac{1}{v_2 V} (\kappa_1 H_1 + \kappa_2 H_4 - (\kappa_1 + \kappa_2) H_2), \\ \frac{dH_3}{dt} &= \frac{1}{v_3 V} ((\kappa_1 + \kappa_2) H_2 + \kappa_3 H_4 - (\kappa_1 + \kappa_2 + \kappa_3) H_3), \\ \frac{dH_4}{dt} &= \frac{1}{(1 - v_1 - v_2 - v_3) V} ((\kappa_1 + \kappa_2 + \kappa_3) H_3 - (\kappa_1 + \kappa_2 + \kappa_3) H_4).\end{aligned}\tag{A.1}$$

In this model, H_i represents the pH corresponding to each zone ($i = 1, 2, 3, 4$) in the reactor, v is the volume fraction of different zones, and κ is the superficial transport coefficient indicating the “flow rate” of between adjacent zones that accounts for both reaction and convection transport (L/s). As illustrated in Figure 3.3(d) of the main manuscript, the flow rates were set as: $f_{4,1} = \kappa_1, f_{4,2} = \kappa_2, f_{4,3} = \kappa_3, f_{1,2} = \kappa_1, f_{2,3} = \kappa_1 + \kappa_2, f_{3,4} = \kappa_1 + \kappa_2 + \kappa_3$ based on the principle of mass conservation (assuming constant density). H_0 represents the shock input of KOH, which is equal to H_v during the injection period (0-8s) defined as:

$$H_0 = \begin{cases} H_v, & 0 < t < 8 \\ 0, & 8 < t < 200 \end{cases} \quad (\text{A.2})$$

In this model, the parameters $\kappa_1, \kappa_2, \kappa_3$ and H_v in addition to the volume fractions v_1, v_2 and v_3 are considered to be uncertain parameters requiring estimation by deterministic global optimization.

As for the case of KOH shock injection in the middle zone (Figure 3.3(e) of the main manuscript), the model was established as:

$$\begin{aligned} \frac{dH_1}{dt} &= \frac{1}{v_1 V} (\kappa_1 H_2 - \kappa_1 H_1), \\ \frac{dH_2}{dt} &= \frac{1}{v_2 V} (H_0 + (\kappa_1 + \kappa_2) H_4 - (\kappa_1 + \kappa_2) H_2), \\ \frac{dH_3}{dt} &= \frac{1}{v_3 V} (\kappa_2 H_2 + \kappa_3 H_4 - (\kappa_2 + \kappa_3) H_3), \\ \frac{dH_4}{dt} &= \frac{1}{(1 - v_1 - v_2 - v_3) V} (\kappa_1 H_1 + (\kappa_2 + \kappa_3) H_3 - (\kappa_1 + \kappa_2 + \kappa_3) H_4). \end{aligned} \quad (\text{A.3})$$

In this model, the corresponding flow rates were defined as: $f_{2,1} = \kappa_1, f_{2,3} = \kappa_2, f_{4,3} = \kappa_3,$

$f_{1,4} = \kappa_1$, $f_{4,2} = \kappa_1 + \kappa_2$, and $f_{3,4} = \kappa_1 + \kappa_2 + \kappa_3$. The initial injection shock of KOH also followed the previous principle (A.2). The volumes of the pH sensor zones were assumed to remain the same fraction values based on the injection position, which can be confirmed from the global optimization results of the high-zone injection case. Thus, the volume fractions were not estimated in this model. The superficial transport coefficients κ_1 , κ_2 , κ_3 and the input H_v were considered as uncertain parameters in the system required to be determined by global optimization.

The model corresponding to the KOH shock injection in the low zone (Figure 3.3(f) of the main manuscript) was established as:

$$\begin{aligned}
\frac{dH_1}{dt} &= \frac{1}{v_1 V} ((\kappa_1 + \kappa_2)H_2 - (\kappa_1 + \kappa_2)H_1), \\
\frac{dH_2}{dt} &= \frac{1}{v_2 V} (\kappa_1 H_3 + \kappa_2 H_4 - (\kappa_1 + \kappa_2)H_2), \\
\frac{dH_3}{dt} &= \frac{1}{v_3 V} (H_0 + (\kappa_1 + \kappa_3)H_4 - (\kappa_1 + \kappa_3)H_3), \\
\frac{dH_4}{dt} &= \frac{1}{(1 - v_1 - v_2 - v_3)V} ((\kappa_1 + \kappa_2)H_1 + \kappa_3 H_3 - (\kappa_1 + \kappa_2 + \kappa_3)H_4).
\end{aligned} \tag{A.4}$$

Here, based on the conservation law, the flow rates are set as: $f_{3,2} = \kappa_1$, $f_{4,2} = \kappa_2$, $f_{3,4} = \kappa_3$, $f_{2,1} = \kappa_1 + \kappa_2$, $f_{1,4} = \kappa_1 + \kappa_2$, and $f_{4,3} = \kappa_1 + \kappa_3$. Similarly, we can obtain the volume fractions of the sensor zones by high zone injection optimization results. Furthermore, we solve the established global optimization problem to verify the superficial transport coefficient κ_1 , κ_2 , κ_3 and the shock H_v .

A.1.2 Global Optimization Formulation and Validation for Temperature Mixing Model

Based on the energy balance for the batch system, a single equation is used to model the bulk fluid temperature:

$$\frac{dT}{dt} = -\frac{UA_c}{V\rho C_p}(T - T_c). \quad (\text{A.5})$$

Here, T is the temperature of the water solution ($^{\circ}\text{C}$) and A_c is the cross-sectional area of the cylindrical container (m^2) across which heat transfer occurs. The height of the container can be derived as: $L = V/A_c = 0.16\text{m}$. ρ and C_p are respectively the density (kg/m^3) and heat capacity ($\text{kJ}/(\text{kg}\cdot^{\circ}\text{C})$) of the water ($\rho = 998.19, C_p = 4.18$). T_c is the temperature of the inner face of the bottom of the reactor and U is the overall heat transfer coefficient of the system ($\text{kW}/(\text{m}^2\cdot^{\circ}\text{C})$). In this model, the inner wall on the bottom of the reactor is considered as the heated surface for convective heat transfer. Thus, U is simplified to the convective heat transfer coefficient for the liquid in the tank h_l . The initial condition of this system was set as $T(t = 0) = T_0 = 18.5^{\circ}\text{C}$.

The temperature mixing model (A.5) can be solved analytically to obtain the explicit closed-form solution as:

$$T = T_c + (T_0 - T_c) \exp\left[-\frac{Ut}{L\rho C_p}\right]. \quad (\text{A.6})$$

Therefore, the global optimization problem for the heat transfer model can be easily solved

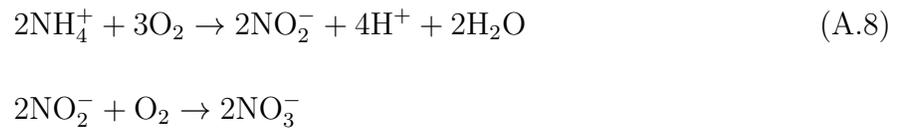
by directly using the analytical solution (A.6):

$$\begin{aligned} & \min_{T_i \in \mathcal{T} \subset \mathbb{R}, \mathbf{p} \in \Pi \subset \mathbb{R}^{n_p}} \sum_{i=1}^N \sum_{j=1}^3 (T_i - T_{i,j}^{data}) \\ \text{s.t. } & T_i = T_c + (T_0 - T_c) \exp\left(-\frac{Ut_i}{h\rho C_p}\right), \quad i = 1, \dots, N \end{aligned} \quad (\text{A.7})$$

The objective function was defined as the SSE between the temperature calculated from the model and the data in different sensor zones. The unknown parameters are the heat transfer coefficient $U \in [0.5, 5.0]$ and the temperature of the heated surface $T_c \in [20, 200]$.

A.1.3 Development of A Continuous Flow Conductivity Transport Model for Nitrification CSTR

In this section, details on the case study of wastewater treatment simulations in a continuous flow nitrification reactor (Figure 3.2(c)) are introduced. The CSTR volume is set as 1000 L. Based on the standard of moderate municipal wastewater [100], the operating setpoint is set at 280 $\mu\text{S}/\text{cm}$ corresponding to the standard concentration. To meet this specification, biological nitrification processes are applied to oxidize ammonia in wastewater [219]. In this tank reactor, a continuous atmospheric air stream is supplied at the bottom for removing excess ammonia through the following two-step nitrification process:



First, ammonium ions are oxidized to nitrite ions in the nitrification step, and then the nitrite ions are further oxidized to nitrate ions in the nitrification step. In practice, the first step is carried out by ammonia oxidizing bacteria (AOB) and the second step is carried out by nitrite oxidizing bacteria (NOB) [219]. In general, 4.5 mg of oxygen is required to fully nitrify 1 mg of N-NH_4^+ . Molecular biology studies indicate that AOB predominates in the nitrification process [220]. Thus, a simplified kinetic model for oxygen consumption is considered in this chapter. It is assumed that there are volatile suspended solids (VSS) in the system that are necessary for nitrification and the initial biomass concentration was in the range of 1.4 to 1.6 g-VSS/L. The ammonium consumption rate $r_{\text{NH}_4^+}$ counted as nitrogen (mg N- NH_4^+ /(L·min)) in the reaction is given by:

$$r_{\text{NH}_4^+} = -r_{\text{AOB}}X_{\text{AOB}} \quad (\text{A.9})$$

where X_{AOB} is the concentration of AOB (mg VSS/L), and r_{AOB} is the ammonium oxidation rate (mg N- NH_4^+ /(g VSS·min)) that is given by:

$$r_{\text{AOB}} = r_{\text{AOB},\text{max}} \frac{c_{\text{NH}_4^+}}{K_{\text{SAOB}} + c_{\text{NH}_4^+} + \frac{c_{\text{NH}_4^+}^2}{K_{\text{IAOB}}}} \frac{c_{\text{O}}}{K_{\text{OAOB}} + c_{\text{O}}} \quad (\text{A.10})$$

In this formula, $r_{\text{AOB},\text{max}}$ is the maximum nitrification rate (mg N- NH_4^+ /(g VSS·min)), K_{SAOB} is the Monod constant of ammonia for AOB (mg N- NH_4^+ /L), K_{IAOB} is the inhibition constant of ammonia for AOB ((mg N- NH_4^+ /L)²), c_{O} is the dissolved oxygen concentration (mg/L), and K_{OAOB} is the Monod constant of oxygen for AOB (mg/L). In this project, it is assumed that oxygen is growth limiting since ammonium is always in excess (around 30 mg N- NH_4^+ /L) and the corresponding attributing coefficient is around 1. Thus, the ammonium oxidation

rate can be simplified to $r_{\text{AOB}} = r_{\text{AOB,max}}c_{\text{O}}/(K_{\text{OAOB}} + c_{\text{O}})$, which is only related to oxygen concentration.

The oxygen balance for the nitrification reaction is given by:

$$\frac{dc_{\text{O}}}{dt} = -r_{\text{AOB}}\psi_{\text{AOB}}X_{\text{AOB}} + k_{\text{la}} \cdot (c_{\text{O}}^* - c_{\text{O}}) \quad (\text{A.11})$$

where ψ_{AOB} is the stoichiometric ratio between oxygen and ammonia (mg O₂/mg N-NH₄⁺), the term $k_{\text{la}}(c_{\text{O}}^* - c_{\text{O}})$ represents the rate of mass transfer of oxygen into the liquid from air bubbles, k_{la} is the volumetric mass transfer coefficient (s⁻¹) [98], and c_{O}^* is the saturated dissolved oxygen concentration (9.1 mg/L at 20 °C) [99]. The standard oxygen transfer rate (SOTR, mg/s) is defined as $\text{SOTR} = k_{\text{la}}c_{\text{O}}^*V$ that represents the amount of oxygen transferred per second at 20 °C. The standard oxygen transfer efficiency (SOTE, %) refers to the ratio of oxygen in inlet air stream dissolved in liquid at 20 °C that is given by $\text{SOTE} = \text{SOTR}/W_{\text{O}}$, with W_{O} the mass flow of oxygen in the air stream (mg/s). W_{O} can be calculated by an empirical formula: $W_{\text{O}} = 0.2967Q$, where Q is the airflow rate at 20 °C. Therefore, the oxygen transfer rate can be rewritten as:

$$\frac{dc_{\text{O}}}{dt} = -r_{\text{AOB}}\psi_{\text{AOB}}X_{\text{AOB}} + \frac{0.2967Q \cdot \text{SOTE}}{V} \left(1 - \frac{c_{\text{O}}}{c_{\text{O}}^*}\right) \quad (\text{A.12})$$

All the relevant parameter values are summarized in Table A.5.

The modified conductivity mixing model is introduced in Section 3.2.4:

$$\begin{aligned}\frac{dC_i}{dt} &= \frac{1}{4V}(k_i(C_4 - C_1) + \dot{m}_{in,i}C_{in,i} - \dot{m}_{out,i}C_i) + R_{NH_4^+}, \quad , i = 1, 2, 3, \\ \frac{dC_4}{dt} &= \frac{1}{4V}k_4(C_1 + C_2 + C_3 - 3C_4) + R_{NH_4^+},\end{aligned}\tag{A.13}$$

where $\dot{m}_{in,i}$ and $\dot{m}_{out,i}$ are continuous inlet and outlet flow rate at zone i (L/s), respectively, $C_{in,i}$ represents the conductivity of the inlet stream at zone i ($\mu\text{S}/\text{cm}$), and $R_{NH_4^+}$ is the reaction rate law for NH_4^+ consumption counted as conductivity ($\mu\text{S}/\text{cm}/\text{s}$). When substituting the reaction rate $r_{NH_4^+}$ (A.9) to the modified model (A.13), the ammonium concentration should be converted to conductivity for consistency. The converting relationship is linear and can be directly derived as $C_{\text{NH}_4\text{Cl}} = 10.78 + 1.323E5c_{\text{NH}_4\text{Cl}}$ by linear regression of the data from CRC handbook listed in Table A.3 [220, 240]. Ideally, the mass transfer coefficients k_1 , k_2 , k_3 and k_4 should be able to account for all situations with inlet conductivity shocks at high, middle, or low zones since in a real system we would not have *a priori* knowledge of the heterogeneity before measurement. However, as we can see from the previous optimization results, the shock experiments resulted in slightly differing values for these coefficients as indicated in Table A.1(a). Thus, the mass transfer coefficients were taken as the average of the optimal mass transfer coefficients from the original conductivity transport models for high injection and middle injection cases. The low-zone injection case was excluded here as its corresponding optimal value for k_2 differed dramatically from the optimal values for the other cases; most likely due to the very close proximity to the adjacent mixing zone. Then the mass transfer coefficients were scaled based on the nitrification CSTR volume. We conduct simulations by substituting the average mass transfer coefficients and verify that

as a whole, the simulated profiles conformed to the original experimental data (even for the low zone injection case, the simulated profile can be adjusted to coincide with the experimental profile by using the average C_v from high zone and middle zone injection cases to replace C_v from low zone injection case). The mass transfer coefficients of this continuous flow conductivity model are listed in Table A.5. The modified continuous flow conductivity model (A.13) combined with the oxygen transfer model (A.12) can be deemed as a new system of ODEs for simulating a segment of a continuous biological nutrient removal step with heterogeneous mixing. The explicit Euler method is employed for integrating the combined system of ODEs to simulate the conductivity profiles in each zone.

A proportional-integral (PI) controller was designed to control the aeration for high-zone influent conductivity shock rejection. The overall control law is defined as:

$$u(t) = K_p e(t) - K_i \int_0^t e(t') dt' \quad (\text{A.14})$$

where $u(t)$ is the control variable that is equal to the aeration airflow rate Q (mg/s) in this study. $e(t) = \text{SP} - C_1(t)$ is the error value as the difference between the desired setpoint (SP) and a measured process variable, which is the high-zone conductivity in this study $C_1(t)$. K_p and K_i are the coefficients for the proportional and integral terms, respectively. These two coefficients are obtained by fitting dynamic input and output data to a first-order plus dead-time model and then tuned. First, the Internal Model Control (IMC) rules are used for tuning. Then the parameters are further adjusted manually until the closed-loop systems perform as desired. The final values of the PI parameters used in this study are $K_p = -74.941$ and $K_i = -0.07419$.

A.1.4 Global Optimization Formulation for Conductivity and pH Mixing Model

For the conductivity and pH models, analytical solutions of the corresponding systems of ordinary differential equations (ODEs) do not exist. Thus, numerical methods must be applied to compute the numerical solutions of the systems of ODEs. The time domain I was discretized into $N = (t_f - t_0)/\Delta t$ time steps and then the explicit Euler algorithm was used for discretizing the ODEs to obtain a discrete-time system. Consequently, the discrete-time dynamic optimization problem was obtained by:

$$\min_{\mathbf{x}_i \in X \subset \mathbb{R}^{n_x}, \mathbf{p} \in \Pi \subset \mathbb{R}^{n_p}} \sum_{\tau=1}^{N_\tau} \sum_{j=1}^3 (x_{\tau,j} - x_{\tau,j}^{data})^2 \quad (\text{A.15})$$

$$\text{s.t. } \mathbf{x}_0 = \mathbf{z}_0$$

$$\mathbf{x}_{i+1} = \mathbf{x}_i + h\mathbf{f}(\mathbf{x}_i, \mathbf{p}), \quad i = 0, 1, \dots, N - 1$$

where the sum-of-squared errors (SSE) as the objective function was sought to minimize between the state variable $x_{\tau,j}$ calculated from the model and the experimental data $x_{\tau,j}^{data}$ at τ time node for all of three sensor zones. Here, the initial condition was taken as a constant value \mathbf{z}_0 , and a representative subset of the raw data was taken to construct the objective function for simplification of the mathematical structure and acceleration of global optimization calculation. Specifically, the index of the data point with the highest measured value was chosen in the dataset at the injection sensor zone (i_{max}). The time period before i_{max} was considered as the “injection period” and the time period after i_{max} is considered as the “dilution period”. Five equidistant points (i_{max} not included) in the injection period,

15 equidistant points (i_{max} not included) in the dilution period and i_{max} were selected to compose the selected subset which had $N_\tau = 21$ nodes. The distance between each node was rounded to an integer for ease of obtaining the index set. The equality constraints of these discrete-time dynamic optimization problems are the numerical expressions of the discretized ODE system using explicit Euler over the entire time horizon, which had $n_p N$ equations in total. As for the conductivity models, the uncertain parameter vector was defined as $\mathbf{p} = (k_1, k_2, k_3, k_4, C_v)$ for high, middle, and low injection zones. The lower bounds and upper bounds of the parameters are listed in Table A.6 (a).

To validate the pH model with the experimental data, the global optimization problem was first solved for the high zone pH shock injection. The unknown parameters for this problem were defined as $\mathbf{p} = (v_1, v_2, v_3, \kappa_1, \kappa_2, \kappa_3, H_v)$. The uncertainty intervals for these parameters are listed in Table A.6 (b). After solving the problem for high zone injection case to global optimality, the optimal volume fractions were taken and fixed as the volume fractions for the middle and low zone injection as listed in Table A.1. Therefore, for the middle zone and low zone parameter estimation problems, we only estimated four parameters $\mathbf{p} = (\kappa_1, \kappa_2, \kappa_3, H_v)$ with uncertainty listed in Table A.6 (b).

A.1.5 Calibration Process for Revised Conductivity and pH Mixing Models for Additional Experiments

The calibration process for revised conductivity mixing models for KCl and MgSO₄ experiments is summarized in this section. The critical parameter C_v in the model was adjusted to mitigate mismatch with the experimental data. Since the shock conductivity dilutes im-

mediately after injection, C_v represents the conductivity inside the reactor tank instantly after injection in inverse proportion to injection time. Therefore, C_v was adjusted for the KCl and MgSO₄ experiments to be 4.5 times (36s/8s) and 18 times (36s/2s) that of the original NaCl experiments on the basis of the time to peak for conductivity. Furthermore, an additional modification was required to reconcile the difference between ionic species. Based on the dependence of electrical conductivity on concentration as mass percent (%), as listed in Table A.3, the units of shock concentrations for NaCl, KCl, and MgSO₄ (100g/L) were first converted to mass percentage and the corresponding conductivities were calculated by interpolation. Finally, C_v for the KCl and MgSO₄ experiments was further modified by the ratio of conductivity for KCl and MgSO₄ solution to NaCl solution (136.29/120.29 and 40.90/120.29), respectively.

As for calibration for pH mixing model for NaOH experiment, the H_v parameter is inversely proportional to the apparent injection time (8s for original KOH experiment; 1.9s for NaOH experiment). We adjusted the H_v based on the ratio of the apparent injection times for the KOH experiment to the NaOH experiments (8s/1.9s) and re-simulated using the modified H_v .

A.1.6 Development of Computational Fluid Dynamics (CFD) Models

Numerical finite element models were developed in COMSOL Multiphysics (COMSOL, Inc, Burlington, MA, USA). Flow in the reactor was modeled by the Navier-Stokes equations

using the Arbitrary Lagrangian-Eulerian (ALE) techniques [240]:

$$\frac{\partial \rho}{\partial t} - \frac{\partial x}{\partial t} \cdot \nabla \rho + \nabla \cdot (\rho u) = 0$$

$$\rho \left(\frac{\partial u}{\partial t} - \frac{\partial x}{\partial t} \cdot \nabla u \right) + \rho(u \cdot \nabla)u = \nabla \cdot [-pl + \tau] + F$$

Here, ρ is density, $u = v + \partial r/\partial t$, where v is the velocity vector in the rotating coordinate system, and r is the position vector, t is the mesh reference time, x is a function of angular velocity and time, p is pressure, τ is shear stress, and F is the external force vector. To simulate the flow in the reactor, a fixed domain and a rotating domain were defined and coupled using a continuity boundary condition on the common interior walls [112].

Table A.1: The global optimal parameter values obtained from (a) the conductivity parameter estimation problems, and (b) the optimal parameter values obtained from the pH parameter estimation problems.

Conductivity (a)	k_1^*	k_2^*	k_3^*	k_4^*	C_v^*	
High zone injection	2.376E-03	1.411E-03	1.504E-03	0.9477	428.46	
Middle zone injection	1.385E-03	2.914E-03	2.577E-03	1.8990	412.37	
Low zone injection	2.283E-03	2.9234	7.761E-03	0.5995	182.53	
pH (b)	v^*		κ_1^*	κ_2^*	κ_3^*	H_v^*
High zone injection	v_1^*	3.937E-01				
	v_2^*	1.990E-03	1.155E-02	3.464E-03	4.214E-03	0.1413
	v_3^*	5.412E-03				
Middle zone injection	v_1	1.990E-03				
	v_2	3.937E-01	3.012E-02	1.107E-02	2.174E-03	0.1458
	v_3	5.412E-03				
Low zone injection	v_1	1.990E-03				
	v_2	5.412E-03	2.142E-02	8.912E-03	6.346E-03	0.1469
	v_3	3.937E-01				

Table A.2: Time cost for solving conductivity and pH global optimization problems for independent high-zone, middle-zone and low-zone injection experiments, are reported in this table.

Time(s)	Conductivity model	pH model
High-zone injection	214.55	483.96
Middle-zone injection	4038.74	9.559
Low-zone injection	372.34	8.922

Table A.3: Electrical conductivity (mS/cm) for NaCl, KCl, and MgSO₄ at different mass percentages [241].

Mass percent	0.5%	1%	2%	5%	10%	15%	20%	25%
NaCl	8.2	16.0	30.2	70.1	126	171	204	222
KCl	8.2	15.7	29.5	71.9	143	208		
MgSO ₄	4.1	7.6	13.3	27.4	42.7	54.2	51.1	44.1

Table A.4: This table contains the data (in percentages) for environmental discharge and energy usage from the no control, open-loop control, PI control, MPC, EMPC1, and EMPC2 simulations.

Percentage		No Control	Open Loop	PI	MPC	EMPC1	EMPC2
High Zone Shock	Discharge	100	22.342	44.055	30.676	26.094	28.097
	Energy	45.818	100	61.137	57.849	63.438	60.317
Middle Zone Shock	Discharge	100	20.482	45.292	29.791	24.486	26.721
	Energy	45.818	100	52.552	56.624	62.476	59.086
Low Zone Shock	Discharge	100	21.254	45.498	30.148	25.165	27.294
	Energy	45.818	100	52.951	57.129	62.469	59.674
Multiple Shocks	Discharge	100	49.673	66.71	52.869	50.525	51.514
	Energy	45.818	100	80.271	70.185	75.507	72.578

Table A.5: The parameters for continuous flow conductivity mixing model are summarized in this table.

Name	Symbol	Value	Reference
High zone mass transfer coefficient [L/s]	k_1	4.9478	This study
Middle zone mass transfer coefficient [L/s]	k_2	5.6907	This study
Low zone mass transfer coefficient [L/s]	k_3	5.3694	This study
Mixing zone mass transfer coefficient [L/s]	k_4	3745.6	This study
CSTR volume [L]	V	1000	This study
Continuous inlet flow rate [L/s]	\dot{m}_{in}	1.042	This study
Continuous outlet flow rate [L/s]	\dot{m}_{out}	1.042	[220]
Concentration of AOB [mg VSS/L]	X_{AOB}	505	[224]
Monod constant of ammonia for AOB [mg N-NH ₄ ⁺ /L]	K_{SAOB}	0.24	[224]
Inhibition constant of ammonia for AOB [mg N-NH ₄ ⁺ /L]	K_{IAOB}	6200	[224]
Monod constant of oxygen for AOB [mg/L]	K_{OAOB}	0.3	[224]
Stoichiometric ratio between oxygen and ammonia [mg O ₂ /mg N-NH ₄ ⁺]	ψ_{OAOB}	2.5	[224]
Saturated dissolved oxygen concentration [mg/L]	c_O^*	9.1	[99]
Standard oxygen transfer efficiency [%]	SOTE	10	[222]

Table A.6: The lower and upper bounds for uncertain parameters in nonideal heterogeneous mixing models (a: conductivity, b: pH)

Parameters (a)	High injection		Middle injection		Low injection	
	Lower bounds	Upper bounds	Lower bounds	Upper bounds	Lower bounds	Upper bounds
k_1	2.0E-3	2.5E-3	1E-3	5E-3	2.0E-3	2.5E-3
k_2	1.2E-3	1.7E-3	1E-3	5E-3	1.5	3.0
k_3	1.2E-3	1.7E-3	1E-3	5E-3	7.5E-3	8.0E-3
k_4	0.85	0.95	1.0	2.0	0.5	1.0
C_v	420	430	400	450	150	200
Parameters (b)	High injection		Middle injection		Low injection	
	Lower bounds	Upper bounds	Lower bounds	Upper bounds	Lower bounds	Upper bounds
	bounds	bounds	bounds	bounds	bounds	bounds
κ_1	5E-3	1.5E-2	1E-3	1E-1	5E-4	5E-2
κ_2	1E-3	1E-2	1E-3	1E-1	1E-4	1E-2
κ_3	1E-3	1E-2	1E-3	1E-1	1E-4	1E-2
H_v	1E-1	2E-1	1E-1	2E-1	1E-1	2E-1
v_1	0.25	0.55				
v_1	1E-3	1E-2				
v_2	1E-3	1E-2				

Table A.7: The physical properties of water at 20°C are presented below.

Physical property	Symbol	Value
Density [kg/m ³]	ρ	998.19
Viscosity [Pa·s]	μ	1.002E-3
Heat capacity [kJ/(kg·°C)]	C_p	4.18
Thermal conductivity [W/(m·°C)]	λ	0.5973
Prandtl number	Pr	7.2059

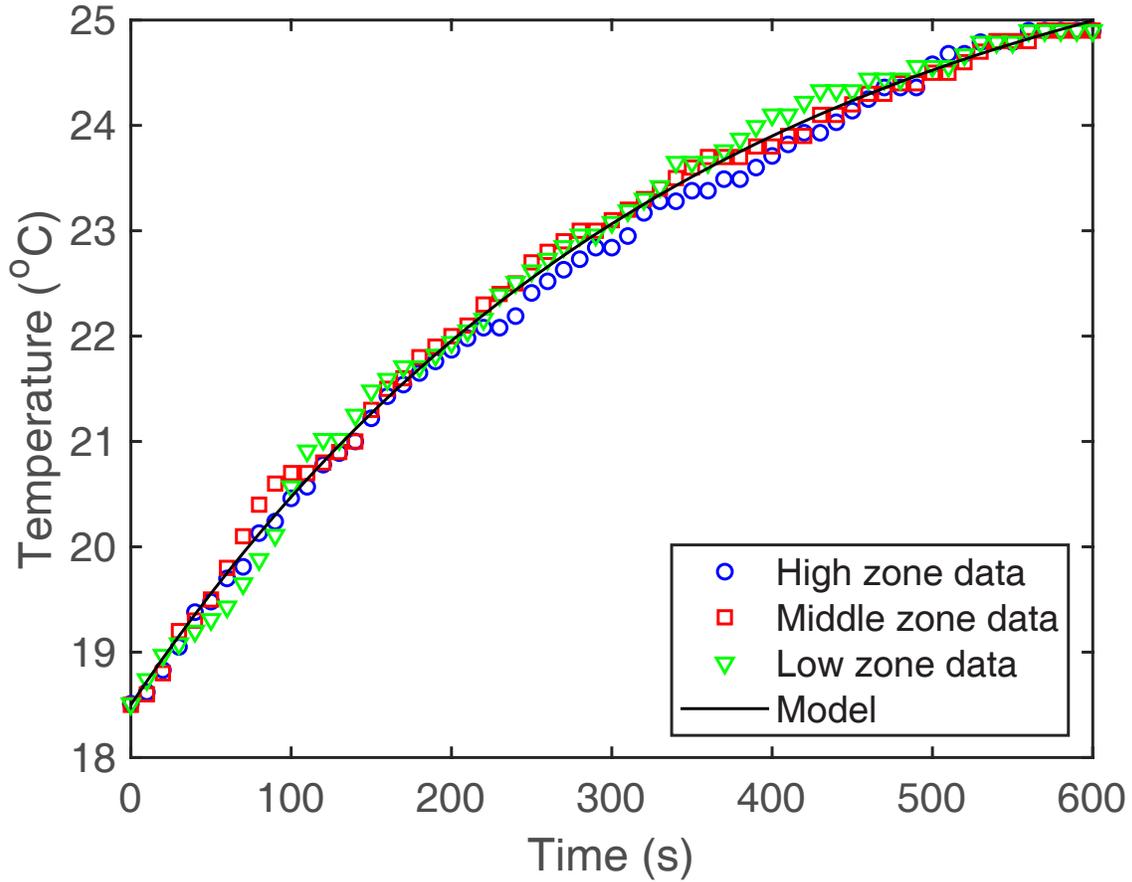


Figure A.1: The optimal temperature profile (black solid line) from the global optimization results is plotted versus the measured temperature sensor data in high zone (blue circle), middle zone (red square), and low zone (green triangle).

A.2 Supplementary Information for Chapter 4

A.2.1 Tumor Transport Model

The 1-dimensional (1D) tumor transport model proposed by Baxter and Jain[129, 130, 131, 132] is used in this study as a mechanistic foundation for studying transvascular exchange and extravascular transport in tumors. The real vasculature of the tumor is intricate and the cells between regions have large differences [242]. There is a necrotic region at the center of the tumor (i.e., most/all cells are dead). In contrast, the outer region of the tumor contains rapidly dividing cells requiring a large blood supply by abundant active blood vessels. Thus, actual solid tumors are spatially heterogeneous and it may be that some physiological parameters in this model are spatially dependent. In our work, we simplify the tumor microenvironment (TME) to be spatially homogeneous without lymphatics or extravascular bindings, which is helpful for certifying and evaluating the overall role of the interstitial fluid pressure (IFP) on fluid transport and penetration of nanocarriers in a tumor. The blood vessels, cells, extracellular matrix (ECM), and other microscopic structures, as illustrated in Figure A.2, are also not considered explicitly in the model because this level of granularity is not important at the length scales we are concerned with in this study. In addition, a main focus of our research is on studying the overall macromolecular solute concentrations in a tumor over a prescribed time horizon. Therefore, we utilize spatial averaging in the data and simulation results, which essentially homogenizes the macroscopic structures. In addition, it is also assumed that the vasculature is distributed continuously over the spatial domain rather than at discrete or localized positions.

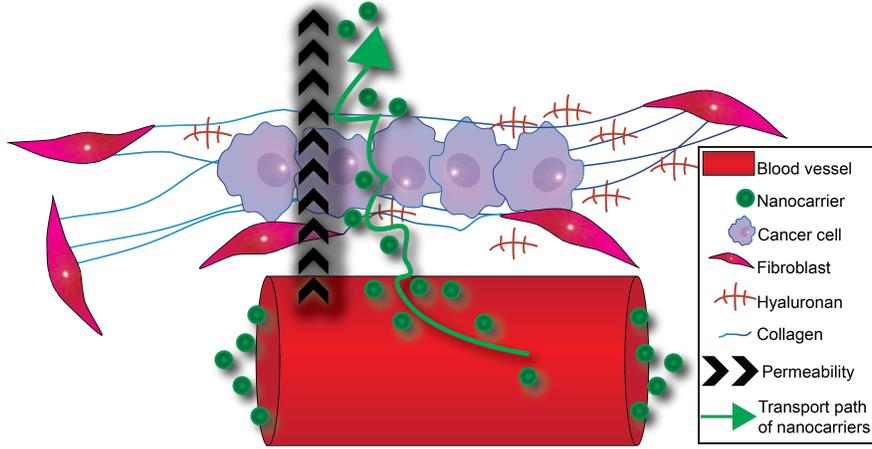


Figure A.2: A diagram of the tumor microenvironment illustrating fluid and solute transport from the blood vessels to the interstitium with high transvascular permeability [6].

Fluid Transport

The fluid transport in the interstitium of a tumor follows Darcy's law:

$$\mathbf{u} = -K\nabla\mathbf{p}. \quad (\text{A.16})$$

Here, \mathbf{u} is the interstitial fluid velocity (IFV) (cm/s), K is the hydraulic conductivity of tumor interstitium ($\text{cm}^2/\text{mm Hg}\cdot\text{sec}$), and \mathbf{p} is the IFP (mm Hg). We assume axisymmetric flow in the spherical coordinate, and (A.16) can be simplified to

$$u = -K\frac{dp}{dr},$$

where r is the radial position (cm).

The continuity equation for steady-state incompressible fluid flow in spherical coordinates

is given by:

$$\frac{1}{r^2} \frac{d(r^2 u)}{dr} = L_p \frac{S}{V} (p_v - p). \quad (\text{A.17})$$

Here, L_p is the hydraulic conductivity of the microvascular wall (cm/mm Hg-sec), S/V is the vascular surface area per unit volume (cm⁻¹), and p_v is the vascular pressure (mm Hg). Substituting (A.16) into the continuity equation (A.17), the steady-state fluid transport model is given by

$$\frac{1}{r^2} \frac{d}{dr} \left(r^2 \frac{dp}{dr} \right) = \frac{\alpha^2}{R^2} (p - p_{ss}), \quad (\text{A.18})$$

where

$$\alpha = R \sqrt{\frac{S L_p}{V K}}$$

is a dimensionless parameter representing the ratio of resistances of the fluid flow in the interstitium to across the vasculature, R is the radius of the spherical tumor (cm), and p_{ss} is the steady-state interstitial pressure where the efflux from the vessels equals the influx (mm Hg), and is equal to p_v in this study.

The boundary conditions consist of a no-flux symmetry condition at the center of the spherical tumor and a Dirichlet condition at the periphery, respectively, as:

$$\left. \frac{dp}{dr} \right|_{r=0} = 0,$$

$$p|_{r=R} = p_\infty,$$

where p_∞ denotes the surrounding tissue pressure (mm Hg).

Solute Transport

To describe and characterize the transport mechanism of nanocarriers in tumors, the macro-molecular solute transport model is governed by the convection-diffusion equation:

$$\frac{\partial c}{\partial t} + \frac{1}{r^2} \frac{\partial(r^2 uc)}{\partial r} = D \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial c}{\partial r} \right) + \phi_s, \quad (\text{A.19})$$

where c is the concentration of the solute in the interstitium of the tumor (g/mL), D is the diffusion coefficient (cm²/sec), and ϕ_s is the distributed source term based on a vessel pore model for transcapillary exchange [243], given by

$$\phi_s = L_p \frac{S}{V} (p_v - p)(1 - \sigma)c_v + P \frac{S}{V} (c_v - c) \frac{Pe}{e^{Pe} - 1} \quad (\text{A.20})$$

Here, $Pe = L_p(p_v - p)(1 - \sigma)/P$ is the Péclet number representing the ratio of the rates of convection to diffusion across the vascular wall, σ is the solute reflection coefficient, P is the vascular permeability of the solute through the vascular wall (cm/sec), and c_v is the solute concentration in tissue vessels (g/mL). Since the bolus injection model is applied, the vascular solute concentration decays exponentially with time as $c_v = c_o e^{-t/k_d}$, where c_o is the initial macromolecular solute concentration in the blood (g/mL), and k_d is the half-life circulation time of the nanocarriers (sec).

It is assumed that no macromolecular solutes exist in the tumor before injection, and

therefore the initial condition is $c(0, r) = 0$. The boundary conditions are defined as:

$$-D \left. \frac{\partial c}{\partial r} \right|_{r=0} + uc|_{r=0} = 0$$

$$c|_{r=R} = c_\infty,$$

where the interstitial concentration satisfies the no-flux condition at the center, is continuous across the tumor periphery, and equals c_∞ , representing the concentration (g/mL) in the normal tissue surrounding the tumor.

Pore Theory

We follow the pore theory developed in Bungay and Brenner [244]. The pores of the vessels are assumed to be cylindrical, in this case, we can evaluate the hydraulic conductivity of the tumor vessels L_p , the vascular permeability P , and the reflection coefficient σ by the pore theory

$$L_p = \frac{\gamma r_o^2}{8\mu L} \tag{A.21}$$

$$P = \frac{\gamma H D_o}{L}$$

$$\sigma = 1 - W$$

where γ is the fraction of the surface area occupied by pores, r_o is the pore radius (nm), μ is the blood viscosity (mm Hg-sec), L is the thickness of the vessel wall (μm), D_o is the diffusion coefficient of the nanocarrier in free solution at 37°C given by the Stokes-Einstein relationship $D_o = k_B T / (6\pi\mu r_p)$, H and W are respectively diffusive and convective hindrance

factors based on the ratio of the particle size over the pore size which are given in Bungay and Brenner [244]:

$$H = \frac{6\pi\Phi}{K_t},$$

$$W = \frac{\Phi(2 - \Phi)K_s}{2K_t},$$

where Φ is the partition coefficient defined as the ratio of the average intrapore concentration to that in the bulk solution at equilibrium. When the interactions between the solutes and pore wall are purely steric, the partition coefficient is taken as $\Phi = (1 - \lambda)^2$, where λ is the ratio of particle size (d_m , nm) to the pore size (d_o , nm). The K_t and K_s factors for the convective hindrance term W are defined as

$$K_t = \frac{9}{4}\pi^2\sqrt{2}(1 - \lambda)^{-5/2}\left[1 + \sum_{k=1}^2 \alpha_k(1 - \lambda)^k\right] + \sum_{k=0}^4 \alpha_{k+3}\lambda^k$$

$$K_s = \frac{9}{4}\pi^2\sqrt{2}(1 - \lambda)^{-5/2}\left[1 + \sum_{k=1}^2 \beta_k(1 - \lambda)^k\right] + \sum_{k=0}^4 \beta_{k+3}\lambda^k$$

The corresponding coefficients a_k and b_k are listed in Table A.8. As indicated by (A.21), the vascular permeability P depends on the particle size and vessel wall properties, such as pore size, thickness, charge, and arrangement. Larger particles will result in lower P and when the particle size is larger than the pore cut-off size, P becomes zero. The vascular hydraulic conductivity L_p relies on the morphology of the wall and the fraction of the wall surface occupied by active pores [115].

Solution Strategy

The fluid and solute transport models were solved numerically. First, the dimensionless form of the tumor radius, IFP, and solute concentration, were defined as:

$$\begin{aligned}\hat{r} &= \frac{r}{R}, \\ \hat{p} &= \frac{p - p_\infty}{p_{ss} - p_\infty}, \\ \hat{c} &= \frac{c - c_\infty}{c_o - c_\infty}.\end{aligned}$$

After reformulating the tumor transport model into dimensionless form, the centered finite difference method was used to discretize the spatial domain. The IFP profile is obtained by solving the fluid transport model (A.18). As for the solute transport model, the backward difference scheme was employed for discretization of the first partial derivative $\partial c/\partial r$. Then, the explicit Euler method was used to integrate the transient convection-diffusion equation with stepsize set as $h = 15$ s to obtain the medicine concentration profile over the tumor radius.

A.2.2 Simplification of Inequality Constraints

In this section, we demonstrate that the inequality constraints on the superficial IFP in (4) of the main manuscript can be expressed as linear constraints on the optimization variables, L_p and K , such that $K = \zeta L_p$, with $\zeta \in \mathbb{R}$.

First, the dimensionless analytical solution of the fluid transport model[129] (A.18) can be

derived as:

$$\hat{p} = \left(1 - \frac{\sinh(\hat{r}\alpha)}{\hat{r} \sinh(\alpha)} \right),$$

where α is given in (A.18).

Then, the IFP in the superficial region can be represented as:

$$\hat{p}_{peri} = \left(1 - \frac{\sinh(\hat{r}_{peri}\alpha)}{\hat{r}_{peri} \sinh(\alpha)} \right), \quad (\text{A.22})$$

where \hat{r}_{peri} is the dimensionless radius from the center towards the superficial region of a tumor.

Substituting (A.22) into the inequality constraints of (4) of the main manuscript results in:

$$\left(1 - \frac{\sinh(\hat{r}_{peri}\alpha)}{\hat{r}_{peri} \sinh(\alpha)} \right) \geq \hat{p}_{peri,\min}, \quad (\text{A.23})$$

$$\left(1 - \frac{\sinh(\hat{r}_{peri}\alpha)}{\hat{r}_{peri} \sinh(\alpha)} \right) \leq \hat{p}_{peri,\max}. \quad (\text{A.24})$$

If (A.23) is active, then the following equality holds:

$$1 - \frac{\sinh(\hat{r}_{peri}\alpha)}{\hat{r}_{peri} \sinh(\alpha)} = \hat{p}_{peri,\min}. \quad (\text{A.25})$$

Differentiating (A.25) with respect to L_p , yields the following expression:

$$\left(-\frac{\hat{r}_{peri} \sinh(\alpha) \cosh(\hat{r}_{peri}\alpha) - \sinh(\hat{r}_{peri}\alpha) \cosh(\alpha)}{\hat{r}_{peri} \sinh^2(\alpha)} \right) \left(\frac{d\alpha}{dL_p} \right) = 0. \quad (\text{A.26})$$

Since $\alpha > 0$ always holds ($L_p > 0$), it can be verified that $\left(-\frac{\hat{r}_{peri} \sinh(\alpha) \cosh(\hat{r}_{peri}\alpha) - \sinh(\hat{r}_{peri}\alpha) \cosh(\alpha)}{\hat{r}_{peri} \sinh^2(\alpha)}\right) >$

0. Therefore, if the constraint is active, then we must have:

$$\begin{aligned} \frac{d\alpha}{dL_p} &= 0 \\ \Rightarrow \frac{d\left(R\sqrt{\frac{SL_p}{VK}}\right)}{dL_p} &= 0 \end{aligned}$$

For this expression to hold, this means that α must be constant with respect to L_p . Since all parameters in α other than L_p and K are constants, K must necessarily be a scalar multiple of L_p .

This gives the following result:

$$K = \zeta L_p, \text{ for some } \zeta \in \mathbb{R} \text{ such that } \hat{p} = \hat{p}_{peri,\min}. \quad (\text{A.27})$$

By the same procedure, K must be a scalar multiple of L_p if (A.24) is active. Therefore, (A.23) and (A.24) can be simplified as, respectively:

$$K \leq \zeta_{\max} L_p, \quad (\text{A.28})$$

$$K \geq \zeta_{\min} L_p. \quad (\text{A.29})$$

The values of ζ_{\min} and ζ_{\max} are listed in Table 5 of the main manuscript. The values for ζ_{\min} are calculated according to the following procedure:

1. Choose two different values of L_p within the interval bounds.
2. Solve the nonlinear equation (A.25) with each value of L_p for the corresponding K values.
3. Compute ζ_{\min} as the slope of the secant line joining the two points on an L_p versus K plot.

The calculation of ζ_{\max} values follow analogously.

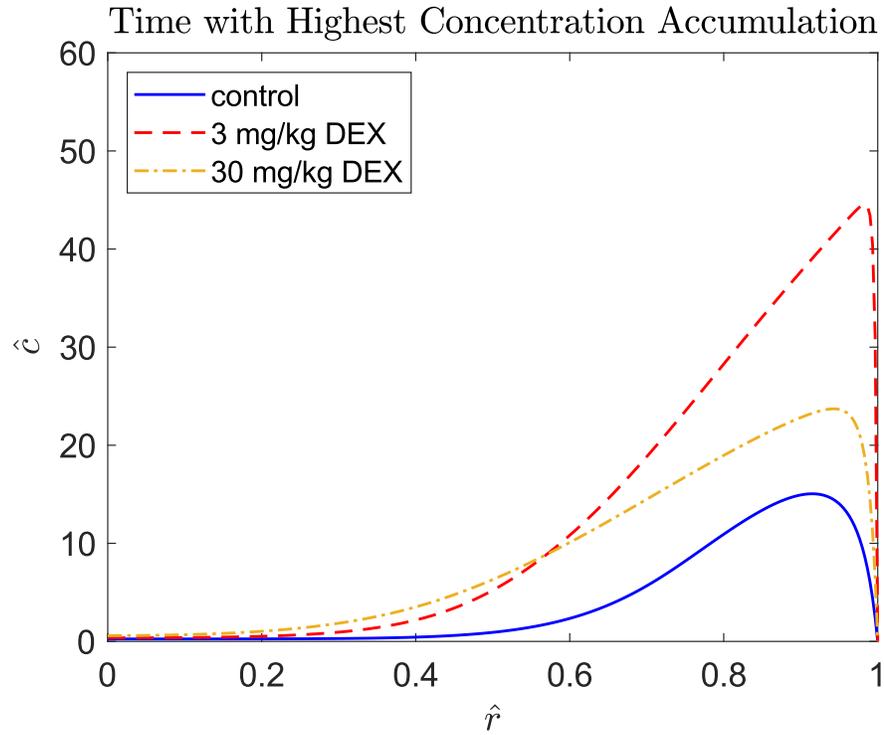


Figure A.3: The radial interstitial concentration profiles at the time corresponding to the highest spatially-averaged concentrations with respect to control (38.8 h), 3 mg/kg dexamethasone treatment (34.2 h), and 30 mg/kg DEX treatment (53.9 h) cases are presented.

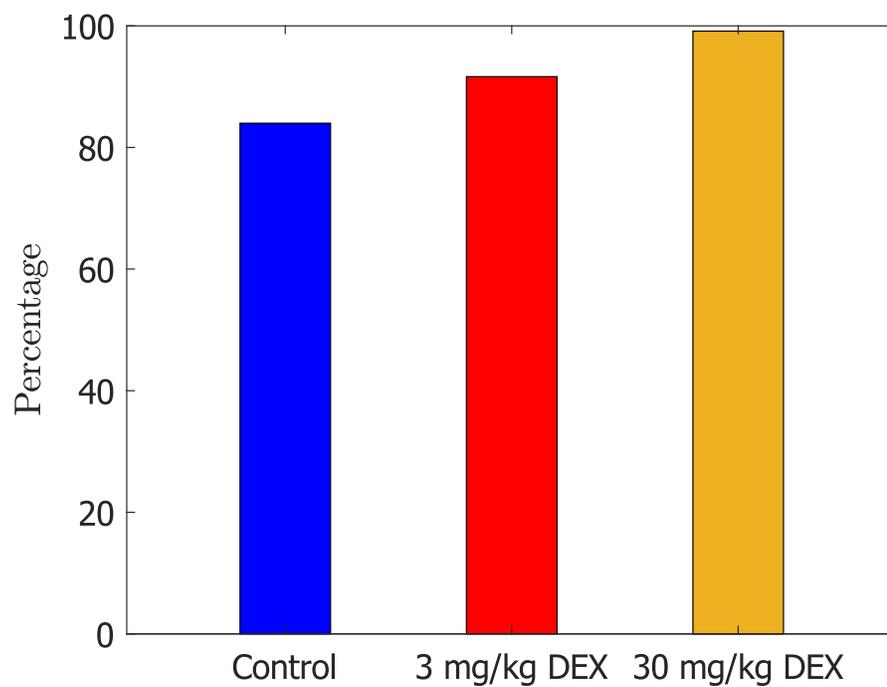


Figure A.4: The percentages of the spatially-averaged concentrations at 72 h over the highest spatially-averaged concentrations for the control, 3 mg/kg and 30 mg/kg dexamethasone (DEX) treatment cases are presented. The DEX treatment enhances the retention effect.

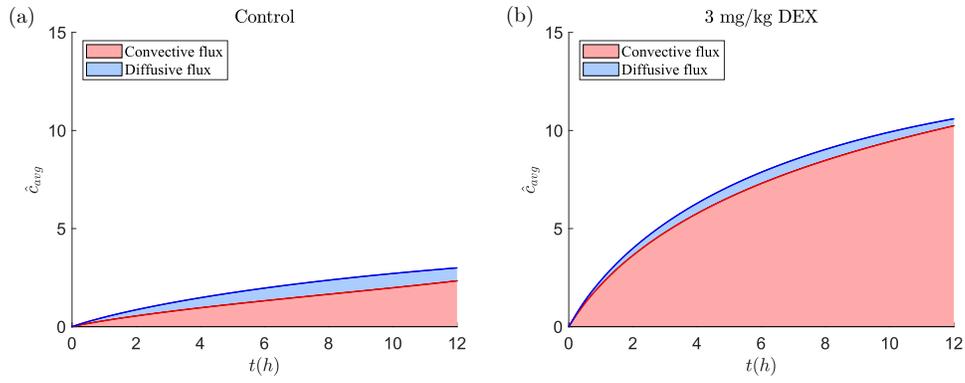


Figure A.5: The contributions from convective and diffusive flux to spatially-averaged concentrations versus time for (a) control; (b) 3 mg/kg dexamethasone (DEX) treatment cases are presented. The profiles are plotted with a 12-hour horizon because the diffusive flux becomes extremely small after that. The contribution from convective flux becomes more dominant after DEX treatment.

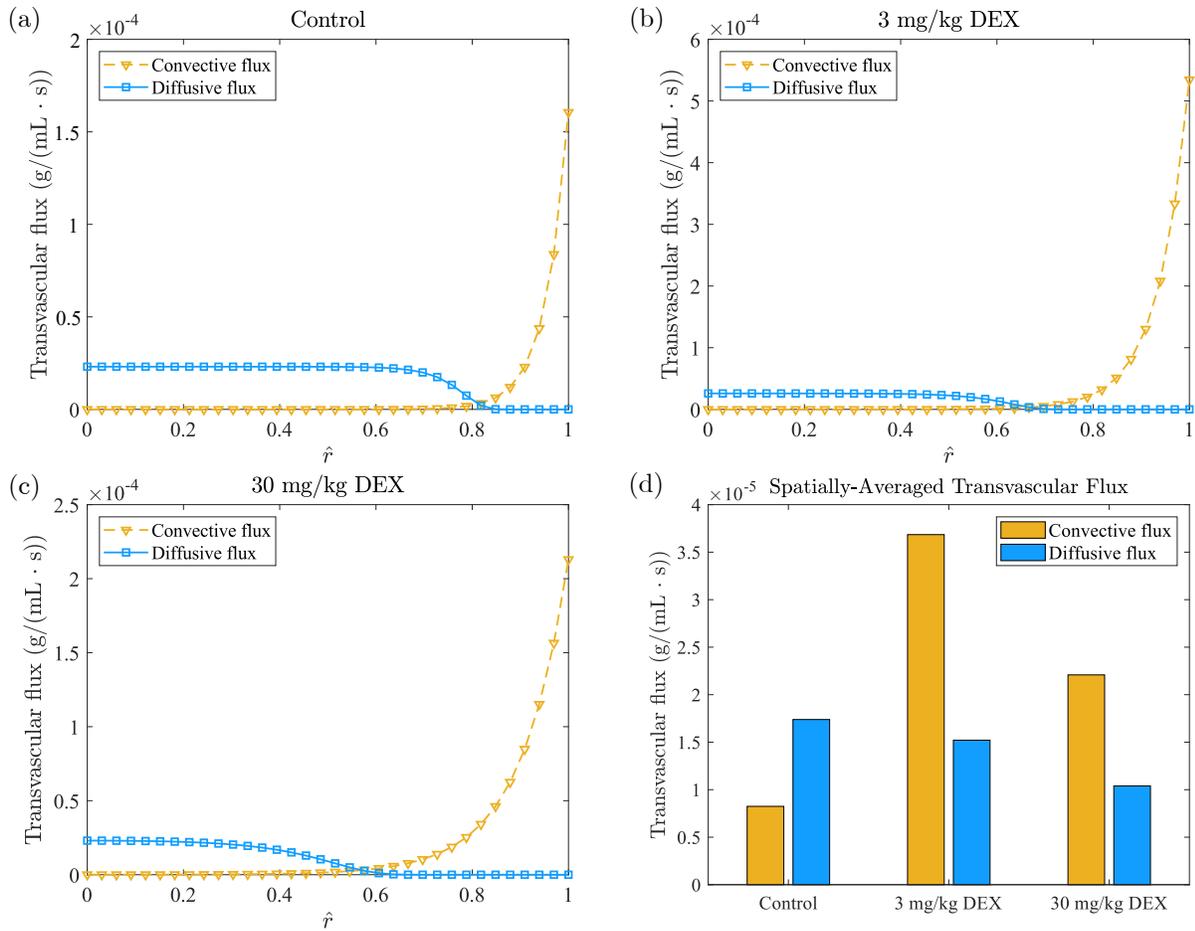


Figure A.6: The transvascular flux profiles over the dimensionless radius \hat{r} for the (a) control; (b) 3 mg/kg dexamethasone (DEX) treatment; and (c) 30 mg/kg DEX treatment cases with 70 kDa dextran one-hour post-administration are presented. (d) The spatially-averaged convective and diffusive fluxes at one-hour post-administration for different doses of DEX are presented in this bar plot.

Table A.8: Hydrodynamic Coefficients for Cylindrical Pore Model [243]

k	1	2	3	4	5	6	7
α_k	-73/60	77293/50400	-22.5083	-5.6117	-0.3363	-1.216	1.647
β_k	7/60	-2227/50400	4.0180	-3.9788	-1.9215	4.392	5.006

Table A.9: The bounds on input variables, L_p and K , used for the surrogate model construction are listed in this table.

Bounds Variable	Control		Treatment	
	Lower bound	Upper bound	Lower bound	Upper bound
L_p (cm/mm Hg-sec)	1.00×10^{-7}	1.75×10^{-6}	5.00×10^{-7}	3.50×10^{-6}
K (cm ² /mm Hg-sec)	1.00×10^{-7}	1.00×10^{-6}	7.00×10^{-7}	4.00×10^{-6}

Table A.10: The benchmark metrics for development time and performance of artificial neural network surrogate models of difference cases (70 kDa - control; 70 kDa - treatment; 500 kDa - control; 500 kDa - treatment) are tabulated. “70 kDa” and ”500 kDa” denote molecular weights of nanocarriers. “Treatment” denotes both 3 mg/kg and 30 mg/kg dexamethasone (DEX) treatment.

Time Metrics		
Case	Data Generation (s)	Training (s)
70 kDa - Control	3272	473
70 kDa - Treatment	3072	538
500 kDa - Control	3210	474
500 kDa - Treatment	3761	539
Performance Metrics		
Case	Mean-Squared Error	Mean-Percent Error (%)
70 kDa - Control	5.49×10^{-7}	0.339
70 kDa - Treatment	2.32×10^{-7}	0.102
500 kDa - Control	3.23×10^{-7}	0.467
500 kDa - Treatment	1.55×10^{-7}	0.096

Table A.11: Data for diffusion coefficients [245] and blood half-life circulation time [246] with respect to nanocarrier sizes are presented in this table.

Particle size	12nm	60nm	125nm	250nm
Diffusion coefficient (cm ² /s)	2×10^{-7}	5×10^{-8}	6×10^{-9}	1×10^{-9}
Half-life circulation time (min)	1480	995	582	500*

*represents extrapolation from data

Table A.12: This table provides benchmark metrics of time and performance (data generation time, training time, mean-squared error and mean-percent error) for ANN surrogate model development in (12) of the main manuscript.

	Time Metrics		Performance Metrics	
	Data Generation (s)	Training (s)	Mean-Squared Error	Mean-Percent Error (%)
$\hat{C}_{\text{avg}}^{\text{ANN}}$	2847	180	7.49×10^{-7}	0.172
$\hat{C}_{\text{peri}}^{\text{ANN}}$	2849	66	5.22×10^{-7}	0.276

Table A.13: Physiological parameter values for the tumor transport model are listed in this table. “70 kDa” and ”500 kDa” denote molecular weights of nanocarriers.

Parameter	Definition	Value	Reference
S/V (cm^{-1})	Vascular density	200	[247]
D (cm^2/sec)	Diffusion coefficient	2×10^{-7} (70 kDa); 1.375×10^{-7} (500 kDa)	[245]
p_v (mm Hg)	Vascular pressure	25	[149]
k_d (min)	Blood circulation time	1480 (70 kDa); 1278 (500 kDa)	[246]
μ (mm Hg-sec)	Blood viscosity	3×10^{-5}	[133]
L (μm)	Vessel wall thickness	5	[248]
γ	Fraction of pore area	1×10^{-3}	[121]

Appendix B

Related Publications and Presentations

B.1 Journal Articles

Wang, C., Wilhelm, M. E., and Stuber, M. D. Semi-infinite Optimization with Hybrid Models, *Industrial & Engineering Chemistry Research*, 61(15), 5239-5254 (2022).

Wang, C., Morgenstern, S., Martin, J. D., and Stuber, M. D. Optimal Therapy Design with Tumor Microenvironment Normalization, *AIChE Journal*, in press

Wilhelm, M. E., Wang, C., and Stuber, M. D. Convex and Concave Envelopes of Artificial Neural Network Activation Functions for Deterministic Global Optimization, under review.

Wang, T¹., Wang, C¹., Xu, Z., Cui, C., Wang, X., Demitrack, Z., Dai, Z., Bagtzoglou, A., Stuber, M. D., and Li, B. Precise control of water and wastewater treatment

systems with nonideal heterogeneous mixing models and high-fidelity sensing, *Chemical Engineering Journal*, 430(3), 132819 (2022). (1 co-first authorship)

Martin, J. D., Panagi, M., Wang, C., Khan, T. T., Martin, M. R., Voutouri, C., Toh, K., Papageorgis, P., Mpekris, F., Polydorou, C., Ishii, G., Takahashi, S., Gotohda, N., Suzuki, T., Wilhelm, M. E., Melo, V. A., Quader, S., Norimatsu, J., Lanning, R. M., Kojima, M., Stuber, M. D., Stylianopoulos, T., Kataoka, K., and Cabral, H. Dexamethasone Increases Cisplatin-Loaded Nanocarrier Delivery and Efficacy in Metastatic Breast Cancer by Normalizing the Tumor Microenvironment. *ACS Nano*. 13(6), 6396-6408 (2019)

B.2 Conference Proceedings

Wang C., Wihelm, M. E., and Stuber, M. D. Robust Optimization with Hybrid First-Principles Data-Driven Models. In *Proceedings, AIChE 2021, Boston, MA, Nov 10, 2021*.

Wang, C., Wang, T., Li, B., and Stuber, M. D. System Visualization Using Real-Time Data-Driven Models Derived From High-Resolution Sensor Profiling. In *Proceedings, AIChE 2020, Virtual, Nov 17, 2020*.

Wang, C. and Stuber, M. D. Recent Advances in Bounding Transient PDE Models With Parametric Uncertainty. In *Proceedings, AIChE 2020, Virtual, Nov 17, 2020*.

Wang, C. and Stuber, M. D. Robust Simulation of Transient PDE Models under Uncertainty. In *Proceedings, AIChE 2019, Orlando, FL, Nov 15, 2019*.

Wang, C., Martin, J. D., Cabral, H., and Stuber, M. D. Rigorous Parameter Estimation for Model Validation in Oncological Systems. In Proceedings, AIChE 2018, Pittsburgh, PA, Oct 29, 2018.

Bibliography

- [1] D. E. Rosner, *Transport processes in chemically reacting flow systems*. Stoneham, MA: Butterworth-Heinemann, 1986.
- [2] R. J. Kee, M. E. Coltrin, and P. Glarborg, *Chemically reacting flow: theory and practice*. Hoboken, NJ: John Wiley & Sons, 2003.
- [3] M. T. Reagan, H. N. Najm, R. G. Ghanem, and O. M. Knio, “Uncertainty quantification in reacting-flow simulations through non-intrusive spectral projection,” *Combustion and Flame*, vol. 132, no. 3, pp. 545–555, feb 2003.
- [4] A. S. M. Jonayat and B. G. Thomas, “Transient thermo-fluid model of meniscus behavior and slag consumption in steel continuous casting,” *Metallurgical and Materials Transactions B*, vol. 45, no. 5, pp. 1842–1864, jul 2014.
- [5] K. Recknagle, R. Williford, L. Chick, D. Rector, and M. Khaleel, “Three-dimensional thermo-fluid electrochemical modeling of planar SOFC stacks,” *Journal of Power Sources*, vol. 113, no. 1, pp. 109–114, jan 2003.
- [6] J. D. Martin, M. Panagi, C. Wang, T. T. Khan, M. R. Martin, C. Voutouri, K. Toh, P. Papageorgis, F. Mpekris, C. Polydorou, G. Ishii, S. Takahashi, N. Gotohda,

- T. Suzuki, M. E. Wilhelm, V. A. Melo, S. Quader, J. Norimatsu, R. M. Lanning, M. Kojima, M. D. Stuber, T. Stylianopoulos, K. Kataoka, and H. Cabral, “Dexamethasone increases cisplatin-loaded nanocarrier delivery and efficacy in metastatic breast cancer by normalizing the tumor microenvironment,” *ACS Nano*, vol. 13, no. 6, pp. 6396–6408, 2019.
- [7] L. T. Biegler, O. Ghattas, M. Heinkenschloss, and B. van Bloemen Waanders, *Large-scale PDE-constrained optimization*. Springer, 2003.
- [8] C. Clason and B. Kaltenbacher, “On the use of state constraints in optimal control of singular PDEs,” *Systems & Control Letters*, vol. 62, no. 1, pp. 48–54, 2013.
- [9] P. Kolvenbach, O. Lass, and S. Ulbrich, “An approach for robust PDE-constrained optimization with application to shape optimization of electrical engines and of dynamic elastic structures under uncertainty,” *Optimization and Engineering*, vol. 19, no. 3, pp. 697–731, 2018.
- [10] M. Sharma, M. Hahn, S. Leyffer, L. Ruthotto, and B. van Bloemen Waanders, “Inversion of convection–diffusion equation with discrete sources,” *Optimization and Engineering*, 2020.
- [11] S.-L. Wu and T. Zhou, “Diagonalization-based parallel-in-time algorithms for parabolic PDE-constrained optimization problems,” *ESAIM: Control, Optimisation and Calculus of Variations*, vol. 26, p. 88, 2020.
- [12] J. W. Pearson, M. Porcelli, and M. Stoll, “Interior-point methods and precondition-

- ing for PDE-constrained optimization problems involving sparsity terms,” *Numerical Linear Algebra with Applications*, vol. 27, no. 2, 2019.
- [13] M. D. Stuber and P. I. Barton, “Robust simulation and design using semi-infinite programs with implicit functions,” *International Journal of Reliability and Safety*, vol. 5, no. 3/4, p. 378, 2011.
- [14] M. D. Stuber, A. Wechsung, A. Sundaramoorthy, and P. I. Barton, “Worst-case design of subsea production facilities using semi-infinite programming,” *AIChE Journal*, vol. 60, no. 7, pp. 2513–2524, 2014.
- [15] B. V. B. Waanders, R. Bartlett, K. Long, P. Boggs, and A. Salinger, “Large scale non-linear programming for PDE constrained optimization,” Tech. Rep., oct 2002.
- [16] Y. S. Choi, “Simultaneous analysis and design in pde-constrained optimization,” Ph.D. dissertation, Stanford University, 2012. [Online]. Available: http://web.stanford.edu/group/SOL/dissertations/Youngsoo_thesis.pdf
- [17] I. Papamichail and C. S. Adjiman, “A rigorous global optimization algorithm for problems with ordinary differential equations,” *Journal of Global Optimization*, vol. 24, no. 1, pp. 1–33, 2002.
- [18] C. S. Adjiman, I. P. Androulakis, C. D. Maranas, and C. A. Floudas, “A global optimization method, α bb, for process design,” *Computers & Chemical Engineering*, vol. 20, pp. S419–S424, 1996.
- [19] A. B. Singer and P. I. Barton, “Global solution of optimization problems with

- parameter-embedded linear dynamic systems,” *Journal of Optimization Theory and Applications*, vol. 121, no. 3, pp. 613–646, 2004.
- [20] G. P. McCormick, “Computability of global solutions to factorable nonconvex programs: Part I — convex underestimating problems,” *Mathematical Programming*, vol. 10, no. 1, pp. 147–175, 1976.
- [21] A. B. Singer and P. I. Barton, “Global optimization with nonlinear ordinary differential equations,” *Journal of Global Optimization*, vol. 34, no. 2, pp. 159–190, 2006.
- [22] A. B. Singer, J. W. Taylor, P. I. Barton, and W. H. Green, “Global dynamic optimization for parameter estimation in chemical kinetics,” *The Journal of Physical Chemistry A*, vol. 110, no. 3, pp. 971–976, jan 2006.
- [23] J. K. Scott and P. I. Barton, “Convex relaxations for nonconvex optimal control problems,” in *2011 50th IEEE Conference on Decision and Control and European Control Conference*. IEEE, 2011, pp. 1042–1047.
- [24] —, “Improved relaxations for the parametric solutions of odes using differential inequalities,” *Journal of Global Optimization*, vol. 57, no. 1, pp. 143–176, Sep 2013.
- [25] J. K. Scott, B. Chachuat, and P. I. Barton, “Nonlinear convex and concave relaxations for the solutions of parametric ODEs,” *Optimal Control Applications and Methods*, vol. 34, no. 2, pp. 145–163, 2012.
- [26] K. Shen and J. K. Scott, “Rapid and accurate reachability analysis for nonlinear dynamic systems by exploiting model redundancy,” *Computers & Chemical Engineering*, vol. 106, pp. 596–608, nov 2017.

- [27] A. M. Sahlodin and B. Chachuat, “Discretize-then-relax approach for state relaxations in global dynamic optimization,” in *Computer Aided Chemical Engineering*. Elsevier, 2010, pp. 427–432.
- [28] —, “Discretize-then-relax approach for convex/concave relaxations of the solutions of parametric odes,” *Applied Numerical Mathematics*, vol. 61, no. 7, pp. 803–820, 2011.
- [29] J. K. Scott, M. D. Stuber, and P. I. Barton, “Generalized McCormick relaxations,” *Journal of Global Optimization*, vol. 51, no. 4, pp. 569–606, 2011.
- [30] A. Sahlodin and B. Chachuat, “Convex/concave relaxations of parametric ODEs using taylor models,” *Computers & Chemical Engineering*, vol. 35, no. 5, pp. 844–857, may 2011.
- [31] K. Makino and M. Berz, “Remainder differential algebras and their applications,” in *Computational Differentiation: Techniques, Applications, and Tools*, M. Berz, C. Bischof, G. Corliss, and A. Griewank, Eds. Philadelphia, PA: SIAM, 1996, pp. 63–74.
- [32] M. Berz, “From taylor series to taylor models,” in *AIP Conference Proceedings CONF-961208*, vol. 405, AIP. American Institute of Physics, 1997, pp. 1–23.
- [33] M. Berz and K. Makino, “Performance of taylor model methods for validated integration of ODEs,” in *Applied Parallel Computing. State of the Art in Scientific Computing*, Springer. Springer Berlin Heidelberg, 2006, pp. 65–73.
- [34] N. S. Nedialkov, K. R. Jackson, and G. F. Corliss, “Validated solutions of initial value

- problems for ordinary differential equations,” *Applied Mathematics and Computation*, vol. 105, no. 1, pp. 21–68, oct 1999.
- [35] M. Neher, “From interval analysis to Taylor models—An overview,” *Proc. IMACS 2005*, 2005.
- [36] B. Houska, M. E. Villanueva, and B. Chachuat, “A validated integration algorithm for nonlinear ODEs using Taylor models and ellipsoidal calculus,” in *52nd IEEE Conference on Decision and Control*. IEEE.
- [37] —, “Stable set-valued integration of nonlinear dynamic systems using affine set-parameterizations,” vol. 53, no. 5, pp. 2307–2328.
- [38] Y. Lin and M. A. Stadtherr, “Validated solutions of initial value problems for parametric ODEs,” *Applied Numerical Mathematics*, vol. 57, no. 10, pp. 1145–1162, oct 2007.
- [39] G. W. Harrison, “Dynamic models with uncertain parameters,” in *Proceedings of the first international conference on mathematical modeling*, vol. 1. University of Missouri Rolla, 1977, pp. 295–304.
- [40] W. Walter, *Differential and integral inequalities*. New York, NY: Springer-Verlag, 1970, vol. 55.
- [41] H. Smith, “Monotone dynamical systems: An introduction to the theory of competitive and cooperative systems,” *Bulletin (New Series) of the American Mathematical Society*, vol. 33, pp. 203–209, mar 2008.

- [42] J. K. Scott and P. I. Barton, “Bounds on the reachable sets of nonlinear control systems,” *Automatica*, vol. 49, no. 1, pp. 93–100, 2013.
- [43] A. Neumaier, *Interval Methods for Systems of Equations*. Cambridge, UK: Cambridge University Press, 1991, vol. 37.
- [44] R. E. Moore, *Methods and Applications of Interval Analysis*. Society for Industrial and Applied Mathematics, jan 1979, vol. 2.
- [45] J. L. D. Comba and J. Stolfi, “Affine arithmetic and its applications to computer graphics,” in *Proceedings of VI SIBGRAPI (Brazilian Symposium on Computer Graphics and Image Processing)*, 1993, pp. 9–18.
- [46] J. Stolfi and L. H. de Figueiredo, “An introduction to affine arithmetic,” *TEMA - Tendências em Matemática Aplicada e Computacional*, vol. 4, no. 3, pp. 297–312, dec 2003.
- [47] L. H. de Figueiredo and J. Stolfi, “Affine arithmetic: Concepts and applications,” *Numerical Algorithms*, vol. 37, no. 1-4, pp. 147–158, dec 2004.
- [48] R. J. LeVeque, *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. SIAM, 2007.
- [49] W. Gautschi, *Numerical analysis*. Springer Science & Business Media, 2012.
- [50] G. D. Smith, *Numerical solution of partial differential equations: finite difference methods*. Oxford, UK: Oxford University Press, 1985.

- [51] X. Yang and J. K. Scott, “Efficient reachability bounds for discrete-time nonlinear systems by extending the continuous-time theory of differential inequalities,” in *2018 Annual American Control Conference (ACC)*. IEEE, jun 2018, pp. 6242–6247.
- [52] M. E. Wilhelm and M. D. Stuber, “EAGO.jl: easy advanced global optimization in Julia,” *Optimization Methods and Software*, pp. 1–26, 2020.
- [53] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, “Julia: A fresh approach to numerical computing,” *SIAM Review*, vol. 59, no. 1, pp. 65–98, 2017.
- [54] J. U. Smith, N. J. Bradbury, and T. M. Addiscott, “SUNDIAL: A PC-based system for simulating nitrogen dynamics in arable land,” *Agronomy Journal*, vol. 88, no. 1, pp. 38–43, jan 1996.
- [55] M. E. Wilhelm, A. V. Le, and M. D. Stuber, “Global optimization of stiff dynamical systems,” *AIChE Journal*, vol. 65, no. 12, nov 2019.
- [56] J. Makungu, H. Haario, and W. C. Mahera, “A generalized 1-dimensional particle transport method for convection diffusion reaction model,” *Afrika Matematika*, vol. 23, no. 1, pp. 21–39, 2012.
- [57] K. V. Gernaey, M. C. van Loosdrecht, M. Henze, M. Lind, and S. B. Jørgensen, “Activated sludge wastewater treatment plant modelling and simulation: state of the art,” *Environmental Modelling & Software*, vol. 19, no. 9, pp. 763–783, 2004.
- [58] S. R. Qasim, “Wastewater treatment plants. planning, design and operation seconded,” 1999.

- [59] M. Benosman, “Model-based vs data-driven adaptive control: An overview,” *International Journal of Adaptive Control and Signal Processing*, vol. 32, no. 5, pp. 753–776, 2018.
- [60] E. Aydın, Y. Arkun, G. Is, M. Mutlu, and M. Dikbas, “Plant-wide optimization and control of an industrial diesel hydro-processing plant,” *Computers & Chemical Engineering*, vol. 87, pp. 234–245, 2016.
- [61] I. TAKACS, “A dynamic model of the clarification-thickening process,” *Water Research*, vol. 25, no. 10, pp. 1263–1271, 1991.
- [62] P. Roeleveld and M. van Loosdrecht, “Experience with guidelines for wastewater characterisation in the netherlands,” *Water Science and Technology*, vol. 45, no. 6, pp. 77–87, 2002.
- [63] V. Kumar, M. Kumar, and R. Prasad, Eds., *Phytobiont and Ecosystem Restitution*. Springer Singapore, 2018.
- [64] N. Banadda, I. Nhapi, and R. Kimwaga, “A review of modeling approaches in activated sludge systems,” *African Journal of Environmental Science and Technology*, vol. 5, no. 6, pp. 397–408, 2011.
- [65] S. G. Zimmermann, M. Wittenwiler, J. Hollender, M. Krauss, C. Ort, H. Siegrist, and U. von Gunten, “Kinetic assessment and modeling of an ozonation step for full-scale municipal wastewater treatment: Micropollutant oxidation, by-product formation and disinfection,” *Water Research*, vol. 45, no. 2, pp. 605–617, 2011.

- [66] L. Corominas, X. Flores-Alsina, L. Snip, and P. A. Vanrolleghem, “Comparison of different modeling approaches to better evaluate greenhouse gas emissions from whole wastewater treatment plants,” *Biotechnology and Bioengineering*, vol. 109, no. 11, pp. 2854–2863, 2012.
- [67] O. Bernard, Z. Hadj-Sadok, D. Dochain, A. Genovesi, and J.-P. Steyer, “Dynamical model development and parameter identification for an anaerobic wastewater treatment process,” *Biotechnology and Bioengineering*, vol. 75, no. 4, pp. 424–438, 2001.
- [68] N. Bertola, L. Palladino, A. Bevilacqua, and N. Zaritzky, “Optimisation of the design parameters in an activated sludge system for the wastewater treatment of a potato processing plant,” *Journal of Food Engineering*, vol. 40, no. 1-2, pp. 27–33, 1999.
- [69] D. Wang, J. Liu, and R. Srinivasan, “Data-driven soft sensor approach for quality prediction in a refining process,” *IEEE Transactions on Industrial Informatics*, vol. 6, no. 1, pp. 11–17, 2010.
- [70] N. Wahab, J. Balderud, M. Rahmat, and R. Katebi, “Data-driven adaptive model-based predictive control with application in wastewater systems,” *IET Control Theory & Applications*, vol. 5, no. 6, pp. 803–812, 2011.
- [71] A. Fenu, G. Guglielmi, J. Jimenez, M. Spèrandio, D. Saroj, B. Lesjean, C. Brepols, C. Thoeye, and I. Nopens, “Activated sludge model (ASM) based modelling of membrane bioreactor (MBR) processes: A critical review with special regard to MBR specificities,” *Water Research*, vol. 44, no. 15, pp. 4272–4294, 2010.

- [72] C. Rosen and J. Lennox, “Multivariate and multiscale monitoring of wastewater treatment operation,” *Water Research*, vol. 35, no. 14, pp. 3402–3410, 2001.
- [73] L. Belanche, J. Valdés, J. Comas, I. Roda, and M. Poch, “Prediction of the bulking phenomenon in wastewater treatment plants,” *Artificial Intelligence in Engineering*, vol. 14, no. 4, pp. 307–317, 2000.
- [74] X. Wei and A. Kusiak, “Short-term prediction of influent flow in wastewater treatment plant,” *Stochastic Environmental Research and Risk Assessment*, vol. 29, no. 1, pp. 241–249, 2014.
- [75] B. Bidar, J. Sadeghi, F. Shahraki, and M. M. Khalilipour, “Data-driven soft sensor approach for online quality prediction using state dependent parameter models,” *Chemometrics and Intelligent Laboratory Systems*, vol. 162, pp. 130–141, 2017.
- [76] E. Levlin, “Conductivity measurements for controlling municipal waste-water treatment,” in *Proceedings of a polish-Swedish-Ukrainian seminar*, 2010, pp. 51–62.
- [77] A. W. P. C. SYSTEMS, “Why pH is important in wastewater treatment,” <https://www.alarcorp.com/blog/2016/08/02/ph-important-wastewater-treatment>, 2010, accessed: 2018-10-02.
- [78] S. Ahsan, M. A. Rahman, S. Kaneco, H. Katsumata, T. Suzuki, and K. Ohta, “Effect of temperature on wastewater treatment with natural and waste materials,” *Clean Technologies and Environmental Policy*, vol. 7, no. 3, pp. 198–202, 2005.
- [79] J. E. Falk and R. M. Soland, “An algorithm for separable nonconvex programming problems,” *Management Science*, vol. 15, no. 9, pp. 550–569, 1969.

- [80] R. Horst and H. Tuy, *Global optimization: Deterministic approaches third ed.* Springer Science & Business Media, 2013.
- [81] C. S. Adjiman and C. A. Floudas, “Rigorous convex underestimators for general twice-differentiable problems,” *Journal of Global Optimization*, vol. 9, no. 1, pp. 23–40, 1996.
- [82] A. B. Singer, “Global dynamic optimization,” Ph.D. dissertation, Massachusetts Institute of Technology, 2004.
- [83] N. Andrei, “Interior point filter line search: IPOPT,” in *Continuous Nonlinear Optimization for Engineering Applications in GAMS Technology*. Basel, Switzerland: Springer International Publishing, 2017, pp. 415–435.
- [84] A. Waechter, C. Laird, F. Margot, and Y. Kawajir, “Introduction to IPOPT: A tutorial for downloading, installing, and using IPOPT,” 2009. [Online]. Available: <https://projects.coin-or.org/Ipopt/browser/stable/3.10/Ipopt/doc/documentation.pdf?format=raw>
- [85] M. Schmidt and H. Lipson, “Distilling free-form natural laws from experimental data,” *Science*, vol. 324, no. 5923, pp. 81–85, 2009.
- [86] Z. Xu, W. Zhou, Q. Dong, Y. Li, D. Cai, Y. Lei, A. Bagtzoglou, and B. Li, “Flat flexible thin milli-electrode array for real-time in situ water quality monitoring in distribution systems,” *Environmental Science: Water Research & Technology*, vol. 3, no. 5, pp. 865–874, 2017.
- [87] Z. Xu, Q. Dong, B. Otieno, Y. Liu, I. Williams, D. Cai, Y. Li, Y. Lei, and B. Li, “Real-time in situ sensing of multiple water quality related parameters using micro-electrode

- array (MEA) fabricated by inkjet-printing technology (IPT),” *Sensors and Actuators B: Chemical*, vol. 237, pp. 1108–1119, 2016.
- [88] H. S. Fogler, *Essentials of Chemical Reaction Engineering: fourth ed.* Prentice Hall Profesional, 2006.
- [89] R. B. Bird, “Transport phenomena,” *Applied Mechanics Reviews*, vol. 55, no. 1, pp. R1–R4, 2002.
- [90] M. E. Tuckerman, D. Marx, and M. Parrinello, “The nature and transport mechanism of hydrated hydroxide ions in aqueous solution,” *Nature*, vol. 417, no. 6892, pp. 925–929, 2002.
- [91] P. W. Atkins and J. De Paula, *Physical chemistry.* Oxford university press, Oxford UK, 1998.
- [92] M. Tuckerman, K. Laasonen, M. Sprik, and M. Parrinello, “Ab initio molecular dynamics simulation of the solvation and transport of hydronium and hydroxyl ions in water,” *The Journal of Chemical Physics*, vol. 103, no. 1, pp. 150–161, 1995.
- [93] R. B. Bird, W. E. Stewart, and E. N. Lightfoot, *Transport phenomena.* John Wiley & Sons, 2006, vol. 1.
- [94] 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. [Online]. Available: <https://doi.org/10.1007/s10898-014-0166-2>

- [95] M. R. Bussieck and A. Meeraus, “General algebraic modeling system (GAMS),” in *Applied Optimization*. Springer US, 2004, pp. 137–157.
- [96] I. Dunning, J. Huchette, and M. Lubin, “JuMP: A modeling language for mathematical optimization,” *SIAM Review*, vol. 59, no. 2, pp. 295–320, 2017.
- [97] M. Gavrilescu and M. Macoveanu, “Process engineering in biological aerobic wastewater treatment,” *Acta Biotechnologica*, vol. 19, no. 2, pp. 111–145, 1999.
- [98] S. Judd, *The MBR book: principles and applications of membrane bioreactors for water and wastewater treatment*. Elsevier, 2010.
- [99] L. Pawlowski, “Standard methods for the examination of water and wastewater, 18th edition,” *Science of The Total Environment*, vol. 142, no. 3, pp. 227–228, 1994.
- [100] M. Henze, M. C. van Loosdrecht, G. A. Ekama, and D. Brdjanovic, *Biological Wastewater Treatment*. IWA publishing, 2008.
- [101] O. Bello, Y. Hamam, and K. Djouani, “Coagulation process control in water treatment plants using multiple model predictive control,” *Alexandria Engineering Journal*, vol. 53, no. 4, pp. 939–948, 2014.
- [102] W. Shen, X. Chen, M. Pons, and J. Corriou, “Model predictive control for wastewater treatment process with feedforward compensation,” *Chemical Engineering Journal*, vol. 155, no. 1-2, pp. 161–174, 2009.
- [103] B. Holenda, E. Domokos, Á. Rédey, and J. Fazakas, “Dissolved oxygen control of

- the activated sludge wastewater treatment process using model predictive control,” *Computers & Chemical Engineering*, vol. 32, no. 6, pp. 1270–1278, 2008.
- [104] B. W. Bequette, *Process control: modeling, design, and simulation*. Prentice Hall Professional, 2003.
- [105] M. Ellis, H. Durand, and P. D. Christofides, “A tutorial review of economic model predictive control methods,” *Journal of Process Control*, vol. 24, no. 8, pp. 1156–1178, 2014.
- [106] J. Zeng and J. Liu, “Economic model predictive control of wastewater treatment processes,” *Industrial & Engineering Chemistry Research*, vol. 54, no. 21, pp. 5710–5721, 2015.
- [107] S. Revollar, P. Vega, M. Francisco, R. Vilanova, and I. Santin, “Optimization of economic and environmental objectives in a non linear model predictive control applied to a wastewater treatment plant,” in *2016 20th International Conference on System Theory, Control and Computing (ICSTCC)*. IEEE, oct 2016.
- [108] R. Horst and P. M. Pardalos, *Handbook of global optimization*. Springer Science & Business Media, 2013, vol. 2.
- [109] J. B. Bateman, C. L. Stevens, W. B. Mercer, and E. L. Carstensen, “Relative humidity and the killing of bacteria: the variation of cellular water content with external relative humidity or osmolality,” *Journal of General Microbiology*, vol. 29, no. 2, pp. 207–219, 1962.

- [110] C. J. Geankoplis, A. A. Hersel, and D. H. Lepek, *Transport processes and separation process principles*. Prentice hall Boston, MA, 2018, vol. 4.
- [111] D. A. Tillman, D. N. Duong, and N. S. Harding, “Modeling and fuel blending,” in *Solid Fuel Blending*. Elsevier, 2012, pp. 271–293.
- [112] Z. Xu, F. M. Dehkordy, Y. Li, Y. Fan, T. Wang, Y. Huang, W. Zhou, Q. Dong, Y. Lei, M. D. Stuber, A. Bagtzoglou, and B. Li, “High-fidelity profiling and modeling of heterogeneity in wastewater systems using milli-electrode array (MEA): Toward high-efficiency and energy-saving operation,” *Water Research*, vol. 165, p. 114971, 2019.
- [113] P. D. Christofides, J. F. Davis, N. H. El-Farra, D. Clark, K. R. D. Harris, and J. N. Gipson, “Smart plant operations: Vision, progress and challenges,” *AIChE Journal*, vol. 53, no. 11, pp. 2734–2741, 2007.
- [114] J.-P. Ylén *et al.*, *Measuring, modelling and controlling the pH value and the dynamic chemical state*. Helsinki University of Technology, 2001.
- [115] R. K. Jain and T. Stylianopoulos, “Delivering nanomedicine to solid tumors,” *Nature Reviews Clinical Oncology*, vol. 7, no. 11, pp. 653–664, 2010.
- [116] R. K. Jain, “Antiangiogenesis strategies revisited: From starving tumors to alleviating hypoxia,” *Cancer Cell*, vol. 26, no. 5, pp. 605–622, 2014.
- [117] J. D. Martin, H. Cabral, T. Stylianopoulos, and R. K. Jain, “Improving cancer immunotherapy using nanomedicines: progress, opportunities and challenges,” *Nature Reviews Clinical Oncology*, vol. 17, no. 4, pp. 251–266, 2020.

- [118] J. D. Martin, G. Seano, and R. K. Jain, “Normalizing function of tumor vessels: Progress, opportunities, and challenges,” *Annual Review of Physiology*, vol. 81, no. 1, pp. 505–534, 2019.
- [119] J. D. Martin, T. Miyazaki, and H. Cabral, “Remodeling tumor microenvironment with nanomedicines,” *WIREs Nanomedicine and Nanobiotechnology*, 2021.
- [120] H. Cabral, Y. Matsumoto, K. Mizuno, Q. Chen, M. Murakami, M. Kimura, Y. Terada, M. R. Kano, K. Miyazono, M. Uesaka, N. Nishiyama, and K. Kataoka, “Accumulation of sub-100 nm polymeric micelles in poorly permeable tumours depends on size,” *Nature Nanotechnology*, vol. 6, no. 12, pp. 815–823, 2011.
- [121] V. P. Chauhan, T. Stylianopoulos, J. D. Martin, Z. Popović, O. Chen, W. S. Kamoun, M. G. Bawendi, D. Fukumura, and R. K. Jain, “Normalization of tumour blood vessels improves the delivery of nanomedicines in a size-dependent manner,” *Nature Nanotechnology*, vol. 7, no. 6, pp. 383–388, apr 2012. [Online]. Available: <http://dx.doi.org/10.1038/nnano.2012.45>
- [122] T. Stylianopoulos, L. L. Munn, and R. K. Jain, “Reengineering the physical microenvironment of tumors to improve drug delivery and efficacy: From mathematical modeling to bench to bedside,” *Trends in Cancer*, vol. 4, no. 4, pp. 292–319, apr 2018.
- [123] R. K. Jain, R. T. Tong, and L. L. Munn, “Effect of vascular normalization by antiangiogenic therapy on interstitial hypertension, peritumor edema, and lymphatic metastasis: insights from a mathematical model,” *Cancer research*, vol. 67, no. 6, pp. 2729–2735, 2007.

- [124] V. P. Chauhan, J. D. Martin, H. Liu, D. A. Lacorre, S. R. Jain, S. V. Kozin, T. Stylianopoulos, A. S. Mousa, X. Han, P. Adstamongkonkul, Z. Popović, P. Huang, M. G. Bawendi, Y. Boucher, and R. K. Jain, “Angiotensin inhibition enhances drug delivery and potentiates chemotherapy by decompressing tumour blood vessels,” *Nature Communications*, vol. 4, no. 1, p. 2516, oct 2013.
- [125] T. Stylianopoulos, J. D. Martin, V. P. Chauhan, S. R. Jain, B. Diop-Frimpong, N. Bardeesy, B. L. Smith, C. R. Ferrone, F. J. Hornicek, Y. Boucher *et al.*, “Causes, consequences, and remedies for growth-induced solid stress in murine and human tumors,” *Proceedings of the National Academy of Sciences*, vol. 109, no. 38, pp. 15 101–15 108, 2012.
- [126] M. A. Socinski, R. M. Jotte, F. Cappuzzo, F. Orlandi, D. Stroyakovskiy, N. Nogami, D. Rodríguez-Abreu, D. Moro-Sibilot, C. A. Thomas, F. Barlesi, G. Finley, C. Kelsch, A. Lee, S. Coleman, Y. Deng, Y. Shen, M. Kowanetz, A. Lopez-Chavez, A. Sandler, and M. Reck, “Atezolizumab for first-line treatment of metastatic nonsquamous NSCLC,” *New England Journal of Medicine*, vol. 378, no. 24, pp. 2288–2301, 2018.
- [127] J. E. Murphy, J. Y. Wo, D. P. Ryan, J. W. Clark, W. Jiang, B. Y. Yeap, L. C. Drapek, L. Ly, C. V. Baglini, L. S. Blazzkowsky, C. R. Ferrone, A. R. Parikh, C. D. Weekes, R. D. Nipp, E. L. Kwak, J. N. Allen, R. B. Corcoran, D. T. Ting, J. E. Faris, A. X. Zhu, L. Goyal, D. L. Berger, M. Qadan, K. D. Lillemoe, N. Talele, R. K. Jain, T. F. DeLaney, D. G. Duda, Y. Boucher, C. F.-D. Castillo, and T. S. Hong, “Total neoadjuvant therapy with FOLFIRINOX in combination with losartan followed by

- chemoradiotherapy for locally advanced pancreatic cancer,” *JAMA Oncology*, vol. 5, no. 7, p. 1020, 2019.
- [128] V. P. Chauhan, T. Stylianopoulos, Y. Boucher, and R. K. Jain, “Delivery of molecular and nanoscale medicine to tumors: Transport barriers and strategies,” *Annual Review of Chemical and Biomolecular Engineering*, vol. 2, no. 1, pp. 281–298, 2011.
- [129] L. T. Baxter and R. K. Jain, “Transport of fluid and macromolecules in tumors. i. role of interstitial pressure and convection,” *Microvascular Research*, vol. 37, no. 1, pp. 77–104, jan 1989.
- [130] —, “Transport of fluid and macromolecules in tumors. ii. role of heterogeneous perfusion and lymphatics,” *Microvascular research*, vol. 40, no. 2, pp. 246–263, 1990.
- [131] —, “Transport of fluid and macromolecules in tumors: Iii. role of binding and metabolism,” *Microvascular research*, vol. 41, no. 1, pp. 5–23, 1991.
- [132] —, “Transport of fluid and macromolecules in tumors. iv. a microscopic model of the perivascular distribution,” *Microvascular research*, vol. 41, no. 2, pp. 252–272, 1991.
- [133] J. W. Baish, P. A. Netti, and R. K. Jain, “Transmural coupling of fluid flow in microcirculatory network and interstitium in tumors,” *Microvascular research*, vol. 53, no. 2, pp. 128–141, 1997.
- [134] P. W. Sweeney, A. d’Esposito, S. Walker-Samuel, and R. J. Shipley, “Modelling the transport of fluid through heterogeneous, whole tumours in silico,” *PLOS Computational Biology*, vol. 15, no. 6, p. e1006751, 2019.

- [135] E. Begoli, T. Bhattacharya, and D. Kusnezov, “The need for uncertainty quantification in machine-assisted medical decision making,” *Nature Machine Intelligence*, vol. 1, no. 1, pp. 20–23, 2019.
- [136] D. Deshpande, J. G. Pasipanodya, S. G. Mpagama, P. Bendet, S. Srivastava, T. Koeuth, P. S. Lee, S. M. Bhavnani, P. G. Ambrose, G. Thwaites, S. K. Heysell, and T. Gumbo, “Levofloxacin pharmacokinetics/pharmacodynamics, dosing, susceptibility breakpoints, and artificial intelligence in the treatment of multidrug-resistant tuberculosis,” *Clinical Infectious Diseases*, vol. 67, no. suppl_3, pp. S293–S302, 2018.
- [137] S. D. Nelson, C. G. Walsh, C. A. Olsen, A. J. McLaughlin, J. R. LeGrand, N. Schutz, and T. A. Lasko, “Demystifying artificial intelligence in pharmacy,” *American Journal of Health-System Pharmacy*, vol. 77, no. 19, pp. 1556–1570, 2020.
- [138] K. McBride and K. Sundmacher, “Overview of surrogate modeling in chemical process engineering,” *Chemie Ingenieur Technik*, vol. 91, no. 3, pp. 228–239, 2019.
- [139] M. Pirdashti, S. Curteanu, M. H. Kamangar, M. H. Hassim, and M. A. Khatami, “Artificial neural networks: applications in chemical engineering,” *Reviews in Chemical Engineering*, vol. 29, no. 4, pp. 205–239, 2013.
- [140] R. Horst and H. Tuy, *Global Optimization*. Springer Berlin Heidelberg, 1996.
- [141] M. Wilhelm and M. D. Stuber, “Easy advanced global optimization (eago): An open-source platform for robust and global optimization in julia.” AICHE, 2017, aICHE Annual Meeting.

- [142] C. Wang and M. D. Stuber, “Recent advances in bounding transient pde models with parametric uncertainty,” in *AICHE Annual Meeting 2020*, 2020.
- [143] C. Rackauckas and Q. Nie, “Differentials.jl—a performant and feature-rich ecosystem for solving differential equations in julia,” *Journal of Open Research Software*, vol. 5, no. 1, 2017.
- [144] I. Sobol', “On the distribution of points in a cube and the approximate evaluation of integrals,” *USSR Computational Mathematics and Mathematical Physics*, vol. 7, no. 4, pp. 86–112, 1967.
- [145] L. Bessi and C. Rackauckas, “Surrogates.jl,” 2019. [Online]. Available: <https://github.com/SciML/Surrogates.jl>
- [146] M. Innes, E. Saba, K. Fischer, D. Gandhi, M. C. Rudilosso, N. M. Joy, T. Karmali, A. Pal, and V. Shah, “Fashionable modelling with flux,” *CoRR*, vol. abs/1811.01457, 2018. [Online]. Available: <https://arxiv.org/abs/1811.01457>
- [147] M. Innes, “Flux: Elegant machine learning with Julia,” *Journal of Open Source Software*, vol. 3, no. 25, p. 602, 2018.
- [148] M. Han, Z. Shi, and W. Wang, “Modeling dynamic system by recurrent neural network with state variables,” in *Advances in Neural Networks - ISNN 2004*. Springer Berlin Heidelberg, 2004, pp. 200–205.
- [149] Y. Boucher and R. K. Jain, “Microvascular pressure is the principal driving force for interstitial hypertension in solid tumors: implications for vascular collapse,” *Cancer research*, vol. 52, no. 18, pp. 5110–5114, 1992.

- [150] V. Subbiah, J. E. Grilley-Olson, A. J. Combest, N. Sharma, R. H. Tran, I. Bobe, A. Osada, K. Takahashi, J. Balkissoon, A. Camp, A. Masada, D. J. Reitsma, and L. A. Bazhenova, “Phase Ib/II trial of NC-6004 (nanoparticle cisplatin) plus gemcitabine in patients with advanced solid tumors,” *Clinical Cancer Research*, vol. 24, no. 1, pp. 43–51, 2017.
- [151] F. Yuan, M. Dellian, D. Fukumura, M. Leunig, D. A. Berk, V. P. Torchilin, and R. K. Jain, “Vascular permeability in a human tumor xenograft: molecular size dependence and cutoff size,” *Cancer Research*, vol. 55, no. 17, pp. 3752–3756, 1995.
- [152] P. Papageorgis, C. Polydorou, F. Mpekris, C. Voutouri, E. Agathokleous, C. P. Kapnissi-Christodoulou, and T. Stylianopoulos, “Tranilast-induced stress alleviation in solid tumors improves the efficacy of chemo- and nanotherapeutics in a size-independent manner,” *Scientific Reports*, vol. 7, no. 1, 2017.
- [153] U.-P. Wen and S.-T. Hsu, “Linear bi-level programming problems—a review,” *Journal of the Operational Research Society*, vol. 42, no. 2, pp. 125–133, 1991.
- [154] O. Ben-Ayed, “Bilevel linear programming,” *Computers & Operations Research*, vol. 20, no. 5, pp. 485–501, 1993.
- [155] G. Anandalingam and T. L. Friesz, “Hierarchical optimization: An introduction,” *Annals of Operations Research*, vol. 34, no. 1, pp. 1–11, 1992.
- [156] L. N. Vicente and P. H. Calamai, “Bilevel and multilevel programming: A bibliography review,” *Journal of Global optimization*, vol. 5, no. 3, pp. 291–306, oct 1994.

- [157] S. Dempe, “Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints,” *Optimization*, vol. 52, no. 3, pp. 333–359, 2003.
- [158] 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.
- [159] C. A. Floudas and Z. H. Gümüş, “Global optimization in design under uncertainty: feasibility test and flexibility index problems,” *Industrial & Engineering Chemistry Research*, vol. 40, no. 20, pp. 4267–4282, 2001.
- [160] A. Mitsos, P. Lemonidis, and P. I. Barton, “Global solution of bilevel programs with a nonconvex inner program,” *Journal of Global Optimization*, vol. 42, no. 4, pp. 475–513, 2008.
- [161] E. N. Pistikopoulos, “Perspectives in multiparametric programming and explicit model predictive control,” *AIChE journal*, vol. 55, no. 8, pp. 1918–1925, 2009.
- [162] R. Oberdieck, N. A. Diangelakis, I. Nascu, M. M. Papathanasiou, M. Sun, S. Avraamidou, and E. N. Pistikopoulos, “On multi-parametric programming and its applications in process systems engineering,” *Chemical Engineering Research and Design*, vol. 116, pp. 61–82, 2016.
- [163] J. W. Blankenship and J. E. Falk, “Infinitely constrained optimization problems,” *Journal of Optimization Theory and Applications*, vol. 19, no. 2, pp. 261–281, 1976.
- [164] A. Mitsos, “Global optimization of semi-infinite programs via restriction of the right-hand side,” *Optimization*, vol. 60, no. 10-11, pp. 1291–1308, 2011.

- [165] M. D. Stuber and P. I. Barton, “Semi-infinite optimization with implicit functions,” *Industrial & Engineering Chemistry Research*, vol. 54, no. 1, pp. 307–317, 2014.
- [166] H. Djelassi and A. Mitsos, “A hybrid discretization algorithm with guaranteed feasibility for the global solution of semi-infinite programs,” *Journal of Global Optimization*, vol. 68, no. 2, pp. 227–253, 2017.
- [167] A. Tsoukalas and B. Rustem, “A feasible point adaptation of the blankenship and falk algorithm for semi-infinite programming,” *Optimization Letters*, vol. 5, no. 4, pp. 705–716, 2010.
- [168] A. Mitsos and A. Tsoukalas, “Global optimization of generalized semi-infinite programs via restriction of the right hand side,” *Journal of Global Optimization*, vol. 61, no. 1, pp. 1–17, 2014.
- [169] H. Djelassi, M. Glass, and A. Mitsos, “Discretization-based algorithms for generalized semi-infinite and bilevel programs with coupling equality constraints,” *Journal of Global Optimization*, vol. 75, no. 2, pp. 341–392, 2019.
- [170] H. Djelassi and A. Mitsos, “Global solution of semi-infinite programs with existence constraints,” *Journal of Optimization Theory and Applications*, vol. 188, no. 3, pp. 863–881, 2021.
- [171] H. Djelassi, A. Mitsos, and O. Stein, “Recent advances in nonconvex semi-infinite programming: Applications and algorithms,” *EURO Journal on Computational Optimization*, vol. 9, p. 100006, 2021.

- [172] S. Zendehboudi, N. Rezaei, and A. Lohi, “Applications of hybrid models in chemical, petroleum, and energy systems: A systematic review,” *Applied Energy*, vol. 228, pp. 2539–2566, 2018.
- [173] O. Kahrs and W. Marquardt, “The validity domain of hybrid models and its application in process optimization,” *Chemical Engineering and Processing: Process Intensification*, vol. 46, no. 11, pp. 1054–1066, 2007.
- [174] B. Lennartson, M. Tittus, B. Egardt, and S. Pettersson, “Hybrid systems in process control,” *IEEE Control Systems*, vol. 16, no. 5, pp. 45–56, 1996.
- [175] F. J. Doyle, C. A. Harrison, and T. J. Crowley, “Hybrid model-based approach to batch-to-batch control of particle size distribution in emulsion polymerization,” *Computers & Chemical Engineering*, vol. 27, no. 8-9, pp. 1153–1163, 2003.
- [176] D. C. Psychogios and L. H. Ungar, “A hybrid neural network-first principles approach to process modeling,” *AIChE Journal*, vol. 38, no. 10, pp. 1499–1511, 1992.
- [177] D. S. Lee, C. O. Jeon, J. M. Park, and K. S. Chang, “Hybrid neural network modeling of a full-scale industrial wastewater treatment process,” *Biotechnology and Bioengineering*, vol. 78, no. 6, pp. 670–682, 2002.
- [178] M. L. Thompson and M. A. Kramer, “Modeling chemical processes using prior knowledge and neural networks,” *AIChE Journal*, vol. 40, no. 8, pp. 1328–1340, 1994.
- [179] G. Bollas, S. Papadokonstadakis, J. Michalopoulos, G. Arampatzis, A. Lappas, I. Vasalos, and A. Lygeros, “Using hybrid neural networks in scaling up an FCC model from

- a pilot plant to an industrial unit,” *Chemical Engineering and Processing: Process Intensification*, vol. 42, no. 8-9, pp. 697–713, 2003.
- [180] A. M. Schweidtmann, J. M. Weber, C. Wende, L. Netze, and A. Mitsos, “Obey validity limits of data-driven models through topological data analysis and one-class classification,” *Optimization and Engineering*, pp. 1–22, 2021.
- [181] M. Belgiu and L. Drăguț, “Random forest in remote sensing: A review of applications and future directions,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 114, pp. 24–31, apr 2016.
- [182] C. A. Henao and C. T. Maravelias, “Surrogate-based superstructure optimization framework,” *AIChE Journal*, vol. 57, no. 5, pp. 1216–1232, 2010.
- [183] A. M. Schweidtmann and A. Mitsos, “Deterministic global optimization with artificial neural networks embedded,” *Journal of Optimization Theory and Applications*, vol. 180, no. 3, pp. 925–948, 2018.
- [184] M. von Stosch, R. Oliveira, J. Peres, and S. F. de Azevedo, “Hybrid semi-parametric modeling in process systems engineering: Past, present and future,” *Computers & Chemical Engineering*, vol. 60, pp. 86–101, 2014.
- [185] H. C. Aguiar and R. M. Filho, “Neural network and hybrid model: a discussion about different modeling techniques to predict pulping degree with industrial data,” *Chemical Engineering Science*, vol. 56, no. 2, pp. 565–570, 2001.
- [186] L. El Ghaoui, F. Gu, B. Travacca, A. Askari, and A. Y. Tsai, “Implicit deep learning,” *arXiv preprint arXiv:1908.06315*, vol. 2, 2019.

- [187] S. Bai, J. Z. Kolter, and V. Koltun, “Deep equilibrium models,” *arXiv preprint arXiv:1909.01377*, 2019. [Online]. Available: <https://arxiv.org/pdf/1909.01377.pdf>
- [188] E. Winston and J. Z. Kolter, “Monotone operator equilibrium networks,” *arXiv preprint arXiv:2006.08591*, 2020.
- [189] F. Gu, H. Chang, W. Zhu, S. Sojoudi, and L. E. Ghaoui, “Implicit graph neural networks,” *arXiv preprint arXiv:2009.06211*, 2020.
- [190] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, “Neural ordinary differential equations,” 2019.
- [191] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
- [192] B. M. Kwak and E. J. Haug, “Optimum design in the presence of parametric uncertainty,” *Journal of Optimization Theory and Applications*, vol. 19, no. 4, pp. 527–546, 1976.
- [193] K. P. Halemane and I. E. Grossmann, “Optimal process design under uncertainty,” *AIChE Journal*, vol. 29, no. 3, pp. 425–433, 1983.
- [194] G. M. Ostrovsky, L. E. K. Achenie, Y. Wang, and Y. M. Volin, “A new algorithm for computing process flexibility,” *Industrial & Engineering Chemistry Research*, vol. 39, no. 7, pp. 2368–2377, jul 2000. [Online]. Available: <http://dx.doi.org/10.1021/ie9905207>

- [195] V. D. Dimitriadis and E. N. Pistikopoulos, “Flexibility analysis of dynamic systems,” *Industrial & Engineering Chemistry Research*, vol. 34, no. 12, pp. 4451–4462, dec 1995.
- [196] W. T. Hale, M. E. Wilhelm, K. A. Palmer, M. D. Stuber, and G. M. Bollas, “Semi-infinite programming for global guarantees of robust fault detection and isolation in safety-critical systems,” *Computers & Chemical Engineering*, vol. 126, pp. 218–230, jul 2019.
- [197] T. G. W. Epperly and E. N. Pistikopoulos, “A reduced space branch and bound algorithm for global optimization,” *Journal of Global Optimization*, vol. 11, no. 3, pp. 287–311, 1997.
- [198] M. D. Stuber, “Evaluation of process systems operating envelopes,” Ph.D. dissertation, Massachusetts Institute of Technology, 2012.
- [199] A. Mitsos, B. Chachuat, and P. I. Barton, “McCormick-based relaxations of algorithms,” *SIAM Journal on Optimization*, vol. 20, no. 2, pp. 573–601, 2009.
- [200] A. Wechsung, “Global optimization in reduced space,” Ph.D. dissertation, Massachusetts Institute of Technology, 2014.
- [201] D. Bongartz and A. Mitsos, “Deterministic global optimization of process flowsheets in a reduced space using McCormick relaxations,” *Journal of Global Optimization*, vol. 69, no. 4, pp. 761–796, Dec 2017.
- [202] H. S. Ryoo and N. V. Sahinidis, “A branch-and-reduce approach to global optimization,” *Journal of Global Optimization*, vol. 8, no. 2, pp. 107–138, 1996.

- [203] 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.
- [204] R. Horst and H. Tuy, *Global Optimization: Deterministic Approaches*. Berlin, Germany: Springer Science & Business Media, 2013.
- [205] Y. Puranik and N. V. Sahinidis, “Domain reduction techniques for global NLP and MINLP optimization,” *Constraints*, vol. 22, no. 3, pp. 338–376, 2017.
- [206] K. A. Khan, H. A. J. Watson, and P. I. Barton, “Differentiable McCormick relaxations,” *Journal of Global Optimization*, vol. 67, no. 4, pp. 687–729, 2016. [Online]. Available: <http://dx.doi.org/10.1007/s10898-016-0440-6>
- [207] K. A. Khan, M. Wilhelm, M. D. Stuber, H. Cao, H. A. J. Watson, and P. I. Barton, “Corrections to: Differentiable McCormick relaxations,” *Journal of Global Optimization*, vol. 70, no. 3, pp. 705–706, 2018. [Online]. Available: <https://doi.org/10.1007/s10898-017-0601-2>
- [208] J. K. Scott, M. D. Stuber, and P. I. Barton, “Generalized McCormick relaxations,” *Journal of Global Optimization*, vol. 51, no. 4, pp. 569–606, 2011.
- [209] Y. Song and K. A. Khan, “Optimization-based convex relaxations for nonconvex parametric systems of ordinary differential equations,” *Mathematical Programming*, pp. 1–45, 2021.
- [210] M. D. Stuber, J. K. Scott, and P. I. Barton, “Convex and concave relaxations of implicit functions,” *Optimization Methods and Software*, vol. 30, no. 3, pp. 424–460, 2015.

- [211] D. Bongartz, J. Najman, S. Sass, and A. Mitsos, “MAiNGO: McCormick based algorithm for mixed integer nonlinear global optimization,” in *Technical Report*. Process Systems Engineering (AVT. SVT), RWTH Aachen University, 2018. [Online]. Available: <https://git.rwth-aachen.de/avt-svt/public/maingo>
- [212] R. B. Kearfott, J. Castille, and G. Tyagi, “A general framework for convexity analysis in deterministic global optimization,” *Journal of Global Optimization*, vol. 56, no. 3, pp. 765–785, 2013.
- [213] D. P. Sanders, L. Benet, lucaferranti, K. Agarwal, B. Richard, J. Grawitter, E. Gupta, M. F. Herbst, M. Forets, yashrajgupta, E. Hanson, B. van Dyk, C. Rackauckas, R. Vasani, S. Micluța-Câmpeanu, S. Olver, T. Koolen, C. Wormell, F. A. Vázquez, J. TagBot, K. O’Bryant, K. Carlsson, M. Piibeleht, Reno, R. Deits, T. Holy, M. Kaluba, and matsueushi, “Juliaintervals/intervalarithmetic.jl: v0.18.2,” May 2021. [Online]. Available: <https://doi.org/10.5281/zenodo.4739394>
- [214] G. Fedorov, K. T. Nguyen, P. Harrison, and A. Singh, *Intel Math Kernel Library 2019 Update 2 Release Notes*, 2019. [Online]. Available: <https://software.intel.com/en-us/mkl>
- [215] E. Anderson, Z. Bai, C. Bischof, L. S. Blackford, J. Demmel, J. Dongarra, J. D. Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users’ Guide*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1999.
- [216] E. Wang, Q. Zhang, B. Shen, G. Zhang, X. Lu, Q. Wu, and Y. Wang, “Intel math

- kernel library,” in *High-Performance Computing on the Intel® Xeon Phi™*. New York, NY: Springer International Publishing, 2014, pp. 167–188.
- [217] L. S. Blackford, A. Petitet, R. Pozo, K. Remington, R. C. Whaley, J. Demmel, J. Dongarra, I. Duff, S. Hammarling, G. Henry, and M. Heroux, “An updated set of basic linear algebra subprograms (BLAS),” *ACM Transactions on Mathematical Software*, vol. 28, no. 2, pp. 135–151, 2002.
- [218] S. Juretschko, G. Timmermann, M. Schmid, K.-H. Schleifer, A. Pommerening-Röser, H.-P. Koops, and M. Wagner, “Combined molecular and conventional analyses of nitrifying bacterium diversity in activated sludge: Nitrosococcus mobilis and nitrospira-like bacteria as dominant populations,” *Applied and Environmental Microbiology*, vol. 64, no. 8, pp. 3042–3051, 1998.
- [219] S. Siripong and B. E. Rittmann, “Diversity study of nitrifying bacteria in full-scale municipal wastewater treatment plants,” *Water Research*, vol. 41, no. 5, pp. 1110–1120, 2007.
- [220] O. Sánchez, E. Aspé, M. C. Martí, and M. Roeckel, “The effect of sodium chloride on the two-step kinetics of the nitrifying process,” *Water Environment Research*, vol. 76, no. 1, pp. 73–80, 2004.
- [221] R. B. Baird, A. D. Eaton, and E. W. Rice, Eds., *Standard Methods for the Examination of Water and Wastewater*. Washington, D.C.: American Public Health Association, American Water Works Association, and Water Environment Foundation, 2017.

- [222] L. Uby, “Next steps in clean water oxygen transfer testing – a critical review of current standards,” *Water Research*, vol. 157, pp. 415–434, 2019.
- [223] Z. He, A. Petiraksakul, and W. Meesapya, “Oxygen-transfer measurement in clean water,” *The Journal of KMITNB*, vol. 13, no. 1, pp. 14–19, 2003.
- [224] U. Wiesmann, “Biological nitrogen removal from wastewater,” in *Biotechnics/Wastewater*. Heidelberg, Germany: Springer Berlin Heidelberg, 1994, pp. 113–154.
- [225] F. Chollet *et al.*, “Keras,” 2015, software available from keras.io. [Online]. Available: <https://keras.io>
- [226] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015, software available from tensorflow.org. [Online]. Available: <https://www.tensorflow.org/>
- [227] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY: Springer-Verlag, 2006.
- [228] W. Higgins and P. Jim Kern, “Too much air? understanding the critical role

- of aeration systems,” 2016. [Online]. Available: https://www.tpomag.com/editorial/2016/03/too_much_air_understanding_the_critical_role_of_aeration_systems
- [229] M. E. Wilhelm, C. Wang, and M. D. Stuber, “Convex and concave envelopes of artificial neural network activation functions for deterministic global optimization,” under review.
- [230] M. Avriel and B. Golany, *Mathematical programming for industrial engineers*. CRC Press, 1996, vol. 20.
- [231] X. Wang, G. Du, M. Wu, Q. Song, and Y. Chen, “Two-step robust design of LLC converter with corner judgement and greedy algorithm,” *IEEE Transactions on Power Electronics*, pp. 1–1, 2021.
- [232] L. J. Fernández-Alcázar, R. Kononchuk, and T. Kottos, “Enhanced energy harvesting near exceptional points in systems with (pseudo-)PT-symmetry,” *Communications Physics*, vol. 4, no. 1, 2021.
- [233] H. Schichl and A. Neumaier, “Interval analysis on directed acyclic graphs for global optimization,” *Journal of Global Optimization*, vol. 33, no. 4, pp. 541–562, 2005.
- [234] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2005.
- [235] J. Bae, H. ji Lee, D. H. Jeong, and J. M. Lee, “Construction of a valid domain for a hybrid model and its application to dynamic optimization with controlled exploration,” *Industrial & Engineering Chemistry Research*, vol. 59, no. 37, pp. 16 380–16 395, 2020.

- [236] M. Quaglio, E. S. Fraga, E. Cao, A. Gavriilidis, and F. Galvanin, “A model-based data mining approach for determining the domain of validity of approximated models,” *Chemometrics and Intelligent Laboratory Systems*, vol. 172, pp. 58–67, 2018.
- [237] J. Najman and A. Mitsos, “Convergence analysis of multivariate McCormick relaxations,” *Journal of Global Optimization*, vol. 66, no. 4, pp. 597–628, 2016.
- [238] M. Wilhelm and M. D. Stuber, “Recent Advances in the EaGO Platform: Global and Robust Optimization in Julia,” 2018, aIChE Annual Meeting. [Online]. Available: <https://www.aiche.org/conferences/aiche-annual-meeting/2018/proceeding/paper/51g-recent-advances-eago-platform-global-and-robust-optimization-julia>
- [239] R. Krawczyk, “Interval iterations for including a set of solutions,” *Computing*, vol. 32, no. 1, pp. 13–31, Mar 1984.
- [240] F. Duarte, R. Gormaz, and S. Natesan, “Arbitrary lagrangian–eulerian method for navier–stokes equations with moving boundaries,” *Computer Methods in Applied Mechanics and Engineering*, vol. 193, no. 45-47, pp. 4819–4836, 2004.
- [241] D. R. Lide, *CRC handbook of chemistry and physics*. CRC press, 2004, vol. 85.
- [242] J. Nagy, S.-H. Chang, S.-C. Shih, A. Dvorak, and H. Dvorak, “Heterogeneity of the tumor vasculature,” *Seminars in Thrombosis and Hemostasis*, vol. 36, no. 03, pp. 321–331, 2010.
- [243] W. M. Deen, “Hindered transport of large molecules in liquid-filled pores,” *AICHE Journal*, vol. 33, no. 9, pp. 1409–1425, sep 1987.

- [244] P. M. Bungay and H. Brenner, “The motion of a closely-fitting sphere in a fluid-filled tube,” *International Journal of Multiphase Flow*, vol. 1, no. 1, pp. 25–56, oct 1973.
- [245] A. Pluen, Y. Boucher, S. Ramanujan, T. D. McKee, T. Gohongi, E. di Tomaso, E. B. Brown, Y. Izumi, R. B. Campbell, D. A. Berk, and R. K. Jain, “Role of tumor-host interactions in interstitial diffusion of macromolecules: Cranial vs. subcutaneous tumors,” *Proceedings of the National Academy of Sciences*, vol. 98, no. 8, pp. 4628–4633, mar 2001.
- [246] Z. Popović, W. Liu, V. P. Chauhan, J. Lee, C. Wong, A. B. Greytak, N. Insin, D. G. Nocera, D. Fukumura, R. K. Jain, and M. G. Bawendi, “A nanoparticle size series for in vivo fluorescence imaging,” *Angewandte Chemie*, vol. 122, no. 46, pp. 8831–8834, sep 2010.
- [247] F. Yuan, M. Leunig, S. K. Huang, D. A. Berk, D. Papahadjopoulos, and R. K. Jain, “Microvascular permeability and interstitial penetration of sterically stabilized (stealth) liposomes in a human tumor xenograft,” *Cancer research*, vol. 54, no. 13, pp. 3352–3356, 1994.
- [248] R. K. Jain, “Transport of molecules across tumor vasculature,” *CANCER AND METASTASIS REVIEW*, vol. 6, no. 4, pp. 559–593, dec 1987.