

Automated Modeling of Nonlinear Dynamical Subsystems for Stable System Simulation

by

Yu-Chung Hsiao

Submitted to the Department of Electrical Engineering and Computer
Science

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2015

© Massachusetts Institute of Technology 2015. All rights reserved.

Author
Department of Electrical Engineering and Computer Science
May 20, 2015

Certified by
Luca Daniel
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by
Leslie A. Kolodziejksi
Chair, Department Committee on Graduate Theses

Automated Modeling of Nonlinear Dynamical Subsystems for Stable System Simulation

by

Yu-Chung Hsiao

Submitted to the Department of Electrical Engineering and Computer Science
on May 20, 2015, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Automated modeling techniques allow fast prototyping from measurement or simulation data and can facilitate many important application scenarios, for instance, shortening the time frame from subsystem design to system integration, calibrating models with higher-order effects, and providing protected models without revealing the intellectual properties of actual designs.

Many existing techniques can generate nonlinear dynamical models that are stable when simulated alone. However, such generated models oftentimes result in unstable simulation when interconnected within a physical network. This is because energy-related system properties are not properly enforced, and the generated models erroneously produce numerical energy, which in turn causes instability of the entire physical network. Therefore, when modeling a system that is unable to generate energy, it is essential to enforce passivity in order to ensure stable system simulation.

This thesis presents an algorithm that can automatically generate nonlinear passive dynamical models via convex optimization. Convex constraints are proposed to guarantee model passivity and incremental stability. The generated nonlinear models are suited to be interconnected within physical networks in order to enable the hierarchical modeling strategy.

Practical examples include circuit networks and arterial networks. It is demonstrated that our generated models, when interconnected within a system, can be simulated in a numerically stable way. The system dynamics of the interconnected models can be faithfully reproduced for a range of operations and show an excellent agreement with a number of system metrics. In addition, it is also shown via these two applications that the proposed modeling technique is applicable to multiple physical domains.

Thesis Supervisor: Luca Daniel
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

First and foremost, I would like to express my gratitude to my advisor, Prof. Luca Daniel, for his instruction and guidance for the past several years. His impressive ability to connect concepts has successfully fostered a multidisciplinary research environment, giving me enormous opportunities to learn and work with people from various backgrounds. This work could not have been possible without his support.

I would also like to thank my committee members, Prof. Alexandre Megretski and Prof. Pablo A. Parrilo for their valuable suggestions on this work. I would like to give special thanks to Alex for being an extremely patient and insightful guide on my journey through system theory and identification. His tremendous knowledge has greatly influenced my thinking and communication in mathematics.

I would like to acknowledge two research groups outside MIT that greatly contributed to my study in cardiovascular systems: Prof. Yuri Vassilevski's group at Institute of Applied Mathematics, Russian Academy of Sciences, and Prof. Hao-Ming Hsiao's Advanced Medical Device Laboratory at National Taiwan University. My appreciation also extends to their members who collaborated closely during my visit: Tatiana Dobroserdova, Timur Tscherbul, and Kuang-Huei Lee.

I would like to thank my colleagues in Computational Prototyping Group, whom I learned, shared, and spent a great time with. I would like to thank Bradley N. Bond for introducing me to the exciting field of system identification, Amit Hochman for being such an encouraging mentor, Tarek El-Moselhy for sharing many words of wisdom, Zohaib Mahmood for being a nice comrade marching together toward our degrees, Zheng Zhang for many constructive research suggestions, Tsui-Wei (Lily) Weng for always reminding me of the beauty of our hometown, Taiwan, Lei Zhang and Bo Kim for organizing great social events, Jose E. Cruz Serralles and Richard Zhang for many delightful conversations and suggestions on my presentations, and also Niloofar Farnoush and Omar Mysore for being excellent officemates.

I wish to express my gratitude to several individuals for their mentoring and encouragement during the whole process. I would like to thank Steven C. Wasserman,

whom I worked with in his lab as a teaching assistant during my first year study, for his generosity, patience, and encouragement. I want to thank Prof. Terry P. Orlando for his wisdom that guided me through different stages of my PhD study. I want to express my deep appreciation to Ching-Kuang Clive Tzuang, Professor Emeritus at National Taiwan University, for his long-term mentoring since college. I would like to thank Dr. Shang-Te Yang for inspirational conversations across various topics. I would like to especially thank Dr. Yuh-Chwen G. Lee for her long-term support and encouragement during the rough spots of this challenging journey.

Last but not the least, I would like to thank my family for their endless support, understanding, and love.

Contents

1	Introduction	13
1.1	Existing Techniques for Passive Model Generation	15
1.2	Proposed Method and Related Works	16
1.3	Scope of Applications	17
1.4	Outline and Contributions	17
2	Background	21
2.1	System theory	21
2.1.1	Dissipativity and Passivity	21
2.1.2	Incremental Stability	24
2.2	Sum-of-Squares Optimization	25
2.3	Additional Theorems	26
3	System Identification for Passive Nonlinear Dynamical Systems	29
3.1	Model Structure	29
3.2	Convex Constraints	30
3.2.1	Bijectiveness Guarantee	30
3.2.2	Passivity Guarantee	30
3.2.3	Incremental Stability Guarantee	31
3.3	Objective Functions and Optimization	32
4	Analysis of Convex Constraints	35
4.1	Passivity Guarantee	35

4.1.1	For LTI Systems	35
4.1.2	For First-Order Single-Port LTI Systems	38
4.1.3	For Nonlinear Systems with Polynomials e , f , and h	41
4.2	Incremental Stability Guarantee	43
4.2.1	For LTI systems	43
4.2.2	For Nonlinear Systems with Polynomials e , f , and h	44
4.3	Proofs	45
4.3.1	Proof of Theorem 4.1	45
4.3.2	Proof of Corollary 4.2	47
4.3.3	Proof of Theorem 4.3	48
5	Implementation	51
5.1	Basis Functions	51
5.2	Derivative Estimation	52
5.3	Data Normalization	52
5.4	Optimization Tools	53
5.5	Simulation Environment	53
6	Model Training Practice	55
6.1	Data Preparation	55
6.1.1	Intended Usage of the Model	55
6.1.2	Training Data	56
6.1.3	Validation Data	56
6.1.4	Test Data	56
6.2	Error Metrics	57
6.3	Model Complexity	57
6.3.1	Polynomial Degrees	57
6.3.2	Stiffness	58
6.4	Modeling Procedure	58

7 Case Studies: Circuit Networks	61
7.1 Diode RC Line	61
7.1.1 Training	62
7.1.2 Validation	62
7.1.3 Test within a Physical Network	63
7.2 Experiments on Effects of System Properties	66
7.2.1 Effects of Passivity Guarantee	67
7.2.2 Effects of Incremental Stability and Regularization	68
7.2.3 Effects of Weight κ and Regularization	68
7.3 Modeling from Noisy Data	76
7.4 Class E Power Amplifier	78
8 Case Studies: Arterial Networks	83
8.1 Scenario	83
8.2 Geometries and Parameters	85
8.3 Assumptions for Fluid Dynamics Simulation	86
8.4 Modeling of Aortic Bifurcation	87
8.5 Modeling for Iliac Bifurcation	93
8.6 Modeling via Interconnecting Local Models	95
9 Conclusion	103
9.1 Extensions of Techniques	103
9.1.1 Repeated Experiments and Noisy Data	103
9.1.2 Convex Constraint Related Topics	104
9.1.3 Model Quality Improvement	105
9.1.4 Concepts from Linear Systems	105
9.1.5 Dissipativity in Discrete Time	106
9.2 Further Development of Applications	107

List of Notations

\mathbb{N}_0	set of natural numbers including 0
\mathbb{R}_+	nonnegative real number
$[a; b]$	vertical concatenation of vector a and b
x'	transpose of x
\dot{x}	time derivative of x
$\mathbb{R}[x]$	set of polynomials in x with coefficients in \mathbb{R}
x_i	i -th entry of x
$[x]_d$	a vector of monomials up to degree d
$X \succ Y$	positive definiteness of $X - Y$
$X \succeq Y$	positive semidefiniteness of $X - Y$
$X \geq Y$	elementwise $x_{ij} \geq y_{ij}$
$\ x\ _Q$	weighted 2-norm of vector x as $\sqrt{x'Qx}, Q \succ 0$
$\ X\ _{F,Q}$	column weighted Frobenius (semi)-norm of matrix X as $\sqrt{\text{tr}(X'QX)}, Q \succ 0, (Q \succeq 0)$
$\text{tr}(X)$	trace of matrix X
X^\dagger	Moore-Penrose pseudoinverse of matrix X
$\text{diag}(d)$	a diagonal matrix with diagonals as entries in d
I_n	an n -by- n identity matrix

$0_{m \times n}$	an m -by- n zero matrix
0_n	an n -by- n zero matrix
1	a vector of all ones
$\ker(X)$	kernel space of X
$\deg(a)$	degree of polynomial a
$\deg_z(a)$	highest degree of variable z in polynomial a

Chapter 1

Introduction

Modeling dynamical systems is the foundation for solving complex engineering problems. Many mature and well-established physics-based models are “handcrafted” through years of trials and errors using domain knowledge. On the other hand, automated techniques have been proposed for fast prototyping and black-box modeling. These capabilities can facilitate many application scenarios, such as shortening the time frame from the design of individual components to their integration into a system, calibrating models with higher-order effects, and providing protected models without revealing the intellectual properties of actual designs.

The black-box approaches have achieved wide success in generating standalone models [5, 6, 20, 35, 38, 48, 58, 60, 70]. However, when models are interconnected in a physical network, such as electrical circuits, the law of energy conservation must be considered in order to avoid instability artifacts. Such a concept of energy conservation is centered on the notion of *passivity*, i.e., the inability of generating energy. Energy conservation ensures that a system that is formed by interconnecting passive *systems* is also passive. In the modeling context, the counterpart statement for interconnecting passive *models* has also been studied in a similar manner [75]. When passive systems fail to be represented as passive models, numerical energy may be generated. This excess artificial energy oftentimes causes numerical instability of the entire interconnected systems and may even halt the transient simulation due to convergence issues.

Passive systems cover a wide range of practical systems. Any systems that are lossless or energy-consuming (when *all* supplied power is considered, particularly including the power from sources) are *passive in the sense of energy consumption*. In electrical circuits, for instance, almost all circuit networks are passive when sources are excluded. This set includes linear RLC components, nonlinear transistors and diodes, and their combinations, such as power amplifiers or RF mixers. It is important to note that the traditional classification of active and passive circuits is different from the passivity discussed here. Many circuit networks are traditionally labeled as *active* because they can provide incremental or small-signal gain from designated inputs to outputs. However, when the supplied power from *all* ports is considered, the vast majority of such so-called *active* circuits are in fact *passive in the sense of energy consumption*. Therefore, such active circuit networks also belong to the set of passive systems.

This thesis concerns the problem of generating nonlinear passive dynamical models from measurement or simulation data. Several challenges contribute to the difficulty of such a task. First, many existing passive modeling techniques are based on matching the poles and zeros of transfer functions to frequency-domain data [15, 41, 62]. However, for *nonlinear* systems, the concept of frequency is not uniquely defined [11, 14]. Second, while model fitting for nonlinear systems is preferably performed in the time domain, the passivity condition is, in general, non-convex [28, 73] and leads to challenging optimization problems. Although convex relaxation techniques have been proposed to “convexify” the passivity condition for linear modeling [41, 62], the convex relaxation techniques for *nonlinear* modeling are relatively scarce, if there is any.

This thesis presents an algorithm that can automatically generate nonlinear passive dynamical models via convex optimization. Convex constraints are proposed to guarantee model passivity and incremental stability. The generated passive models are suited to be interconnected within physical networks for enabling the hierarchical modeling strategy. Two sets of practical examples are demonstrated: circuit networks and arterial networks. It can be seen via these two applications that the proposed

modeling technique is applicable to multiple physical domains.

1.1 Existing Techniques for Passive Model Generation

Various methods have been proposed for generating *linear* passive models based on the Positive Real Lemma (the Kalman-Yakubovich-Popov Lemma) [2], for example, [15, 22, 24, 42, 63]. However, the techniques for automatically generating *nonlinear* passive models are relative scarce [37, 79].

Automated passive modeling methods can be categorized into three major types based on the timing when passivity is enforced. Passivity can be enforced *before*, *during*, or *after* model parameters are identified.

Guaranteeing Passivity by Construction

Passivity is guaranteed by restricting model structures such that the model passivity is irrespective of the choice of model parameters. Examples include the synthesis of linear models using passive R, L, C, G components [17, 32, 77] and the specialized dynamic neural networks for nonlinear systems [37, 79].

Enforcing Passivity Simultaneously during Model Identification

This is typically done through formulating an optimization scheme such that the optimal model parameters are sought within a constrained set described by a passivity condition. Several existing methods belong to this category, for instance, [15, 41, 63] for linear modeling.

Enforcing Passivity by Post-Processing Generated Models

Models are first generated with limited or even without any passivity consideration. Then the model parameters are perturbed or post-processed in order to satisfy certain passivity criteria. Examples include [23, 25, 42, 72] for linear modeling.

To the best of our knowledge, the existing techniques for generating nonlinear passive models are currently restricted to the first type. Prior to this work, there seem no reported results for stable system simulations that incorporate interconnections of generated nonlinear models.

1.2 Proposed Method and Related Works

The proposed algorithm is formulated as a sum-of-squares optimization program and guarantees the model passivity using a convex passivity constraint. Such a passivity constraint is a convex relaxation of the dissipation inequality in [73]. The modeling strategy of the proposed method is to search for model parameters within a passivity constraint, and, therefore, belongs to the second type in the classification above. To enhance the model robustness, two additional mechanisms are proposed: incremental stability and ℓ_1 regularization. Incremental stability is a system property that requires that, when excited by the same input, any two state trajectories that start from different initial conditions converge to each other over time. In other words, an incrementally stable model is more robust against a perturbation on initial conditions. To guarantee incremental stability, an additional convex constraint is proposed as a convex relaxation of the incremental Lyapunov theory [3, 80] that implies incremental stability. The ℓ_1 regularization is used to empirically reduce the overfitting problem and perform basis function selections [65].

The approach of this work is closely related to the techniques that identify incrementally stable systems [8, 45, 66, 67]. All these four works proposed algorithms based on the sum-of-squares optimization method and involved various incremental stability guarantees and simulation error bounds. No passivity guarantee has been studied in these four works. In addition, most of such developments were focused on discrete-time systems, with the exception of [67], half of which is dedicated to continuous-time systems. This thesis, on the other hand, focuses on continuous-time systems and proposes a passivity guarantee that ensures stable simulation of interconnected models.

1.3 Scope of Applications

The scope of this work can be extended to a wide range of applications that are compatible with *bond* graphs [51], in which the instantaneous power delivered by a bi-directional link is defined as the product of the *across* and the *through* variables associated with the link. We list a few examples in Table 1.1. Detailed treatments and surveys can be found in [9, 16, 34, 47].

Table 1.1: Examples of physical networks

Domain	Across variable	Through variable
Electrical	Voltage	Current
Magnetic	Magnetomotive force	Magnetic flux
Mechanical, translation	Linear velocity	Force
Mechanical, rotation	Angular velocity	Torque
Hydraulic	Pressure	Flow
Thermal	Temperature	Heat flow

1.4 Outline and Contributions

This section outlines the organization of this thesis and summarizes the contributions.

- **Chapter 2** gives a brief summary of background that is relevant to the forthcoming discussions. Topics include the system theory and the sum-of-squares optimization. Among the system theory, passivity and incremental stability are presented. Some prior knowledge is assumed. For example, basic circuit analysis methods, state-space representations, and the basics of convex optimization are, therefore, not covered here.
- **Chapter 3** develops a convex optimization program that can identify a nonlinear passive dynamical model given a set of training data. A form of nonlinear state-space model is considered throughout this thesis. The convex optimization program minimizes the state-equation errors and the output-equation errors, as a heuristic to search for the best suited model parameters against the training

data. To ensure that the generated models are passive, the optimization program consists of a convex constraint that serves as the sufficient condition to passivity. A formal proof is provided to establish such sufficiency to the original passivity condition.

To regularize the behaviors of the generated models, the optimization program is further equipped with two mechanisms: incremental stability and the ℓ_1 regularization. Similar to the passivity condition, a convex sufficient condition is proposed to ensure that the generated models are incrementally stable. The regularization is used to reduce the overfitting problem. The ℓ_1 version of the regularization additionally reduces the number of basis functions in the generated models. The resulting convex optimization problem can be formulated as a sum-of-squares optimization program.

- **Chapter 4** presents a number of analyses of the convex passivity and incremental stability constraints developed in Chapter 3. The restrictiveness due to the sufficiency of the passivity condition is studied using the general LTI case, the first-order single-port LTI case, and the polynomial case. All these three cases show that the output equation in the model structure must algebraically depend on the input signals. Specifically, the analysis using the first-order single-port LTI system visualizes the feasible regions and shows that the convex passivity constraint excludes some low-loss and lossless systems. For the convex incremental stability constraint, it is shown that a certain restriction is imposed on the model structure when the models are represented by polynomial functions.
- **Chapter 5** provides the implementation details.
- **Chapter 6** outlines the model training practice. The modeling procedures include three phases: the training phase generates models, the validation phase determines the training and modeling parameters, and the test phase measures the performance of the generated model when interconnected within a physical network. A few practical considerations are also discussed in this chapter.

- **Chapter 7** presents the first set of examples: the application in circuit networks. Examples include a diode RC line and a Class E power amplifier. In both cases, stable simulation is performed with good accuracy when the generated models are interconnected within a physical network. Several numerical examples are provided to demonstrate the usefulness of the system properties, the effect of the ℓ_1 regularization, and the performance of model generation from noisy data.
- **Chapter 8** presents the second set of examples: the application in arterial networks. In particular, the segments from the abdominal aorta to the iliac arteries are considered. Such arteries are modeled in a hierarchical manner: generate the local bifurcation models first, and then form a global model by interconnecting the generated local models. Stable and accurate simulation is demonstrated for both the local and the global models.
- **Chapter 9** concludes this thesis and outlines several future research directions.

Chapter 2

Background

We briefly review some concepts in the system theory and the sum-of-squares (SOS) optimization.

2.1 System theory

2.1.1 Dissipativity and Passivity

The earliest development of passivity in systems and circuit networks can be traced back at least to 1950s. Most of the discussions during that period were focused on linear network theory [54, 68, 74, 78] in the form of

$$\int_{-\infty}^{\infty} v(t)' i(t) dt \geq 0, \quad (2.1)$$

where $v(t)$ and $i(t)$ are vectors of voltages and currents at the ports of a circuit network. The prime notation ($'$) denotes the transpose of the vector. Willems generalized this concept to *dissipativity* for nonlinear systems and explicitly involved the states of a state-space model in an inequality similar to (2.1).

Definition 2.1 (Dissipativity[73]). A dynamical system Σ

$$\dot{x}(t) = f(x(t), u(t)) \quad (2.2)$$

$$y(t) = h(x(t), u(t)) \quad (2.3)$$

for $t \geq 0$ with inputs $u \in \mathcal{U}$, states $x \in \mathcal{X}$, and outputs $y \in \mathcal{Y}$ is called *dissipative* with respect to a *supply rate* $\sigma(u, y)$ if there exists a nonnegative *storage function* $V_s : \mathcal{X} \mapsto \mathbb{R}_+$ such that the *dissipation inequality*

$$V_s(x(T)) - V_s(x(0)) \leq \int_0^T \sigma(u(t), y(t)) dt \quad (2.4)$$

is satisfied for all $T \geq 0$ and for all $u \in \mathcal{U}, x \in \mathcal{X}, y \in \mathcal{Y}$. In addition, if V_s is continuously differentiable, then (2.4) is equivalent to

$$\dot{V}_s(x(t)) \leq \sigma(u(t), y(t)) \quad (2.5)$$

for all $t \geq 0$ and for all $u \in \mathcal{U}, x \in \mathcal{X}, y \in \mathcal{Y}$. □

Dissipativity is a mathematical abstraction of energy conservation. Intuitively, the dissipativity inequality (2.4) states that the increase of internal energy is bounded by the external supplied energy, or from (2.5), the growth rate of internal energy is bounded by the external supplied power at any time instance. The dissipation inequality in (2.4) or (2.5) were approached in multiple ways in literature. Willems' definition [73] postulated that dissipativity holds whenever a storage function exists. Some others, for instance, Hill and Moylan [27, 28], began with the non-negative dissipation

$$\int_0^T \sigma(u(t), y(t)) dt \geq 0, \quad (2.6)$$

for all $T \geq 0$, $u \in \mathcal{U}$, $y \in \mathcal{Y}$, and then proved the existence of the available storage function V_a , which serves as a lower bound for V_s in (2.4). A range of definitions, treatments, and their relationships can be found in Chapter 4 of [10].

Definition 2.2 (Passivity[73]). A dynamical system Σ with inputs $u \in \mathcal{U}$, states $x \in$

\mathcal{X} , and outputs $y \in \mathcal{Y}$ in the form of (2.2) and (2.3) is called *passive* if Σ is dissipative with respect to $\sigma(u, y) = u'y$. More generally, let $\sigma(u, y) = u'y - \varepsilon u'u - \delta y'y$. Then the system Σ is called input-strict passive if $\varepsilon > 0$, output-strict passive if $\delta > 0$, and input-output strict passive if both $\varepsilon > 0$ and $\delta > 0$. \square

It should be noted that the passivity we discuss here is *in the sense of energy consumption*. This definition is different from the traditional classification of active and passive circuits, which concerns the ability of providing *incremental* gain *only* from designated inputs to outputs. On the contrary, the overall energy consumption, as we consider here, includes all energy exchange at all ports in the *non-incremental* sense.

The dissipation inequality in Definition 2.2 is symmetric about u and y , which reflects the fact that in physical networks, each terminal is bi-directional, i.e., simultaneously being an input and an output. For example, in electrical systems, inputs u and outputs y can be referred to as current and voltage signals, or vice versa, depending on the modeling convenience.

Passivity is important in physical networks because it is a closure property under interconnection: the interconnection of passive systems is also passive. A passive system that fails to be represented as a passive model may violate the closure property, potentially causing instability of the entire interconnected system as a numerical artifact. The following theorem states that the *admissible* interconnection of passive systems is passive. By admissible, we mean that the interconnection has negligible physical lengths, incurs zero energy loss, and forms a set of linear algebraic constraints such that, for circuits, both Kirchhoff's current and voltage laws (KCL and KVL) hold. For other physical networks, the same statement for the equivalents of the KCL and the KVL is applied.

Theorem 2.3 ([75]). *Let a dynamical system Σ with a state space \mathcal{X} be formed by admissibly interconnecting dynamical systems Σ_i with state spaces \mathcal{X}_i for $i = 1, \dots, k$ such that $\mathcal{X} = \mathcal{X}_1 \times \dots \times \mathcal{X}_k$. Then if $\Sigma_1, \dots, \Sigma_k$ are passive, Σ is passive.* \square

2.1.2 Incremental Stability

When the robustness of a dynamical system is considered, incremental stability is of particular interest. Incremental stability suggests that any two state trajectories tend to each other when the system is excited by the same input. Formally,

Definition 2.4 (Incremental stability [3, 80]). A dynamical system in the form of (2.2) with $u \in \mathcal{U}$ and $x \in \mathcal{X}$, is incrementally globally asymptotically stable if there exists a metric \mathbf{d} and a \mathcal{KL} function β such that

$$\mathbf{d}(x_1(t), x_2(t)) \leq \beta(\mathbf{d}(x_1(0), x_2(0)), t),$$

for all $t \geq 0$ and all trajectories x_1, x_2 of (2.2) subject to the same input $u \in \mathcal{U}$. \square

The function $\beta(r, s)$ of class \mathcal{KL} is a continuous function that belongs to class \mathcal{K} in r and strictly decreasing in s with $\beta(r, s) \rightarrow 0$ as $s \rightarrow \infty$. A class \mathcal{K} function $\alpha(r)$ is a continuous, strictly increasing function with $\alpha(0) = 0$. It is called a \mathcal{K}_∞ function when $\alpha(r) \rightarrow \infty$ as $r \rightarrow \infty$ [33]. It can be seen from the definition that over time, the “memory” of initial conditions fades and the “distance” between trajectories shrinks. Since there is no specialty in the time $t = 0$ for time-invariant systems, every time stamp $t = t_0$ can be treated as initial conditions for the trajectories $t \geq t_0$, suggesting that the system is in general more robust against noises and errors on both initial conditions and state trajectories.

A link between the incremental stability and the incremental Lyapunov function $V_s(x_1, x_2)$ has been established:

Theorem 2.5 (Lyapunov approach to incremental stability [3, 80, 81]). *A dynamical system (2.2) with $u \in \mathcal{U}$ and $x \in \mathcal{X}$, is incrementally globally asymptotically stable if there exists a metric \mathbf{d} , a continuously differentiable function $V_s(x_1, x_2)$, and \mathcal{K}_∞ functions $\underline{\alpha}, \bar{\alpha}, \mu$ such that*

$$\underline{\alpha}(\mathbf{d}(x_1, x_2)) \leq V_s(x_1, x_2) \leq \bar{\alpha}(\mathbf{d}(x_1, x_2)) \quad (2.7)$$

and

$$\frac{d}{dt}V_s(x_1, x_2) \leq -\mu(\mathbf{d}(x_1, x_2)), \quad (2.8)$$

for all t and trajectories x_1 and x_2 of (2.2) subject to the same input $u \in \mathcal{U}$. The converse holds when \mathcal{U} is compact and \mathbf{d} is continuous. \square

To aid our analysis, we state a version of the positive real lemma for a linear time-invariant (LTI) descriptor system $\Sigma(E, A, B, C, D)$

$$\begin{aligned} E\dot{x} &= Ax + Bu, \quad x(0) = 0 \\ y &= Cx + Du, \end{aligned} \quad (2.9)$$

where $u, y \in \mathbb{R}^m$, $x \in \mathbb{R}^n$, matrices E, A, B, C, D are of compatible sizes, and $E+E' \succeq 0$.

Theorem 2.6 (Positive Real Lemma for (2.9)). *The LTI descriptor system $\Sigma(E, A, B, C, D)$ in (2.9) is passive if there exists $P = P' \succ 0$ such that*

$$\begin{bmatrix} -E'PA - A'PE & C' - E'PB \\ C - B'PE & D + D' \end{bmatrix} \succeq 0. \quad (2.10)$$

\square

2.2 Sum-of-Squares Optimization

A sum-of-squares (SOS) optimization program minimizes a linear objective function within a feasible set defined by a set of SOS constraints:

$$\begin{aligned} \underset{\theta \in \mathbb{R}^m}{\text{minimize}} \quad & c'\theta \\ \text{subject to} \quad & p_i(x; \theta) : \text{SOS}(x), \quad i = 1, \dots, N, \end{aligned} \quad (2.11)$$

where $\text{SOS}(x)$ denotes the set of sum-of-squares of polynomials of variables $x \in \mathbb{R}^n$, and $p_i(x; \theta_i) \in \mathbb{R}[x]$ of degree $2d$ are affine in θ . A polynomial $p(x)$ is SOS if there

exist $q_j(x) \in \mathbb{R}[x]$ such that

$$p(x) = \sum_j q_j^2(x), \quad \forall x. \quad (2.12)$$

Such a sum-of-squares decomposition is a sufficient condition to guarantee $p(x) \geq 0$ for all x .¹ In general, determining the nonnegativeness of a polynomial of degree greater than or equal to four is an NP-hard problem [7], whereas searching for $q_j(x)$ in (2.12) is a polynomial-time semidefinite programming (SDP) problem. SOS can be related to SDP by

$$p(x) = [x]_d' Q [x]_d, \quad Q = Q' \succeq 0, \quad (2.13)$$

for all x , where $[x]_d$ is a vector of monomials up to degree d . Finding a SOS decomposition in (2.12) is equivalent to searching for a constant symmetric positive-semidefinite matrix Q in (2.13), which is a SDP problem. If a polynomial is sparse, i.e., containing only few monomials, *Newton polytope* can be used to reduce the number of monomials in $[x]_d$, which in turn simplifies the corresponding SDP problem. The Newton polytope of a polynomial $p(x) \in \mathbb{R}[x_1, \dots, x_n]$ is the convex hull of the exponents of the monomials in $p(x)$. It has been shown in [55] that if a polynomial $p(x)$ is SOS decomposed as in (2.12), then the Newton polytope of q_j and p is related by

$$\mathcal{N}(q_j) \subseteq \frac{1}{2} \mathcal{N}(p), \quad (2.14)$$

where $\mathcal{N}(\cdot)$ denotes the Newton polytope of a polynomial. More details can be found in [40, 50, 64].

2.3 Additional Theorems

We will utilize the following result in our later discussion.

Theorem 2.7 ([52]). *Let $T : \mathbb{R}^n \mapsto \mathbb{R}^n$ be a continuous function which is strongly*

¹The existence of a sum-of-squares decomposition is also a necessary condition for $p(x) \geq 0$ for all x when (i) $n = 1$, (ii) $2d = 2$, and (iii) $n = 2, 2d = 4$. See [7].

monotone for any x, y such that

$$(x - y)'(T(x) - T(y)) \geq \varepsilon \|x - y\|^2, \quad \varepsilon > 0. \quad (2.15)$$

Then $T(x) = 0$ has a unique solution. \square

If T is strongly monotone, it can be seen that $\tilde{T}(x) = T(x) - c$ for all $c \in \mathbb{R}^n$ is also strongly monotone and there exists a unique solution for every $T(x) = c$. One can, therefore, conclude that T is a bijection.

Chapter 3

System Identification for Passive Nonlinear Dynamical Systems

In this chapter, we propose a system identification algorithm that generates passive nonlinear dynamical systems via convex optimization. Given a set of training data, the optimization program minimizes the equation errors and searches for model parameters. We propose a sufficient passivity guarantee and a sufficient incremental stability guarantee that serve as convex constraints in the optimization program.

3.1 Model Structure

We focus on the following continuous-time state-space model

$$\begin{aligned} \frac{d}{dt}e(x(t)) &= f(x(t), u(t)), \quad x(0) = x_0, \\ y(t) &= h(x(t), u(t)), \end{aligned} \tag{3.1}$$

where $u \in \mathcal{U} \in \mathbb{R}^m$, $x \in \mathcal{X} \in \mathbb{R}^n$, and $y \in \mathcal{Y} \in \mathbb{R}^m$ are inputs, states, and outputs, respectively, and $e(\cdot)$ is bijective and continuously differentiable for all x with $\frac{d}{dx}e(x) = E(x)$. We further require e, f, h to be represented as linear combinations of basis functions ϕ_a

$$a = \sum_i \theta_{a,i} \phi_{a,i},$$

where $\theta_{a,i} \in \mathbb{R}$ and a is e , f , or h . The entire model (3.1), therefore, is linearly parameterized by $\theta = [\theta_e; \theta_f; \theta_h]$. We further denote $e = e_\theta, f = f_\theta, h = h_\theta$ to emphasize their parameter dependency if necessary.

3.2 Convex Constraints

In this section, we propose a set of convex constraints that are compatible with the SOS optimization.

3.2.1 Bijectiveness Guarantee

We require $e(\cdot)$ to be strongly monotone to ensure that $e(\cdot)$ is a bijection. By Theorem 2.7 and $\frac{d}{dx}e(x) = E(x)$, it can be seen that $e(\cdot)$ is a bijection if

$$v'E(x)v \geq \varepsilon \|v\|^2 \quad (3.2)$$

for all $x, v \in \mathbb{R}^n$.

3.2.2 Passivity Guarantee

Theorem 3.1. *Let Σ be a dynamical system in the form of (3.1), $\sigma(u, y)$ be a supply rate. If there exists a $Q = Q' \succ 0$ such that the inequality*

$$\begin{aligned} & 2\sigma(u, h(x, u)) + 2x'(e(x) - f(x, u)) - \|x\|_Q^2 \\ & - 2q'(e(x) + f(x, u)) + \|q\|_Q^2 \geq 0 \end{aligned} \quad (3.3)$$

is satisfied for all u, x, q , then Σ is dissipative with respect to $\sigma(u, y)$. \square

Proof. Use the inequality $2a'b \leq \|a\|_Q^2 + \|b\|_{Q^{-1}}^2$ and let $P = Q^{-1}$. Inequality (3.3) becomes

$$2\sigma(u, h(x, u)) + \|e - f\|_P^2 - 2q'(e + f) + \|q\|_Q^2 \geq 0 \quad (3.4)$$

Then minimize the left-hand side with respect to q , i.e., $q = P(e + f)$:

$$2\sigma(u, h(x, u)) + \|e - f\|_P^2 - \|e + f\|_P^2 \geq 0, \quad (3.5)$$

which is equivalent to (2.5) by setting $V(x) = \|e(x)\|_P^2$. ■

Corollary 3.2. *A dynamical system in the form of (3.1) is passive if there exists a $Q = Q' \succ 0$ such that the inequality (3.3) is satisfied for all u, x, q with $\sigma(u, y) = u'h(x, u)$, i.e.,*

$$\begin{aligned} & 2u'h(x, u) + 2x'(e(x) - f(x, u)) - \|x\|_Q^2 \\ & - 2q'(e(x) + f(x, u)) + \|q\|_Q^2 \geq 0 \end{aligned} \quad (3.6)$$

for all u, x, q . □

Proof. The proof follows immediately from Theorem 3.1 and Definition 2.2. ■

3.2.3 Incremental Stability Guarantee

To improve the robustness of the generated models, inspired by [66], we propose the following constraint as an incremental stability certificate.

Theorem 3.3. *Let Σ be a dynamical system in the form of (3.1) and $\varepsilon > 0$. If there exists a $Q = Q' \succ 0$ such that the inequality*

$$\begin{aligned} & 2\Delta'(\delta_e - \delta_f) - \|\Delta\|_Q^2 - 2v'(\delta_e + \delta_f) + \|v\|_Q^2 \\ & - 2\varepsilon w'\delta_e + \|w\|_{\varepsilon Q}^2 \geq 0 \end{aligned} \quad (3.7)$$

is satisfied for all u, x, Δ, v, w , where

$$\delta_e = e(x + \Delta) - e(x), \quad (3.8)$$

$$\delta_f = f(x + \Delta, u) - f(x, u), \quad (3.9)$$

then Σ is incrementally globally asymptotically stable with respect to $\mathbf{d}(x_1, x_2) = \|e(x_1) - e(x_2)\|_{Q^{-1}}$. □

Proof. Similar to the proof for Theorem 3.1, using the inequality $2a'b \leq \|a\|_Q^2 + \|b\|_{Q^{-1}}^2$ and minimizing the expression with respect to v and w , i.e., $v = P(\delta_e + \delta_f)$ and $w = P\delta_e$, gives

$$2\delta'_e P\delta_f \leq -\frac{1}{2}\varepsilon\|\delta_e\|_P^2, \quad (3.10)$$

where $P = Q^{-1} \succ 0$. Note that $\frac{d}{dt}\delta_e = \delta_f$ and let $\Delta = x_1 - x_2$ and $x = x_2$. Then,

$$\frac{d}{dt}\|e(x_1) - e(x_2)\|_P^2 \leq -\frac{1}{2}\varepsilon\|e(x_1) - e(x_2)\|_P^2, \quad (3.11)$$

for all x_1 and x_2 . Since $e(\cdot)$ is invertible, it can be checked that $\mathbf{d}(x_1, x_2) = \|e(x_1) - e(x_2)\|_P$ is a metric. Choosing $\mu(r) = \frac{1}{2}\varepsilon r^2$ and $V_s(x_1, x_2) = \|e(x_1) - e(x_2)\|_P^2$ shows that the system

$$\dot{x} = E(x)^{-1}f(x, u), \quad (3.12)$$

is incremental globally asymptotically stable, and so is (3.1). ■

3.3 Objective Functions and Optimization

Given a set of training data $\Xi = \{\tilde{u}(t_i), \tilde{x}(t_i), \tilde{y}(t_i), \tilde{\dot{x}}(t_i) \mid i = 1, \dots, N\}$ and a weight $\kappa > 0$, we consider the least squares of equation errors of (3.1)

$$J_\kappa(\Xi) = \sum_{i=1}^N \|E(\tilde{x}_i)\tilde{\dot{x}}_i - f(\tilde{x}_i, \tilde{u}_i)\|^2 + \kappa \sum_{i=1}^N \|\tilde{y}_i - h(\tilde{x}_i, \tilde{u}_i)\|^2, \quad (3.13)$$

where we use subscript i to denote the corresponding variable at t_i . It is worth noting that even if we are dealing with a continuous-time model, the training information is still sampled at discrete-time points. The generated models are expected to interpolate smoothly between data points. In addition, minimizing (3.13) only provides a minimizer for the local *equation errors*. The actual *simulation errors* $\int_0^{t_N} \|y(t) - \tilde{y}(t)\| dt$ may not be effectively minimized. Moreover, during the simulation of (3.1), it is possible that the states x deviate away from the training condition due to numerical errors. For all the above reasons, it is important to desensitize the

functions e , f , and h by regularizing the objective function with $\lambda \geq 0$

$$J_{\kappa,\lambda}(\Xi) = J_\kappa(\Xi) + \lambda \|\theta\|_p. \quad (3.14)$$

This ℓ_p regularization empirically alleviates the overfitting problem and tends to result in more uniform and comparable coefficients of basis functions. We are particularly interested in choosing $p = 1$ as it additionally introduces sparsity in θ for basis function selections [30, 65].

The objective function $J_{\kappa,\lambda}(\Xi)$ can then be cast into an SDP problem

$$\begin{aligned} & \underset{t,a,b,c \in \mathbb{R}, \theta}{\text{minimize}} \quad t + \lambda \mathbf{1}' c \\ & \text{subject to} \quad \begin{bmatrix} t & a & b \\ a & 1 & 0 \\ b & 0 & \kappa^{-1} \end{bmatrix} \succeq 0, \\ & \quad \begin{bmatrix} c - \theta \\ c + \theta \end{bmatrix} \geq 0, \\ & \quad a^2 \geq \sum_{i=1}^N \|E(\tilde{x}_i)\tilde{x}_i - f(\tilde{x}_i, \tilde{u}_i)\|^2, \\ & \quad b^2 \geq \sum_{i=1}^N \|\tilde{y}_i - h(\tilde{x}_i, \tilde{u}_i)\|^2. \end{aligned} \quad (3.15)$$

The last two inequalities are second-order cones, which are compatible with SDP through further transformations.

The complete optimization program consists of the objective function with the ℓ_1 regularization (3.14), the bijectiveness guarantee (3.2), the passivity guarantee (3.6), and the incremental stability guarantee (3.7). Formally, given a training data set Ξ ,

and parameters $\kappa > 0$ and $\lambda \geq 0$,

$$\begin{aligned}
& \underset{\theta, Q_1, Q_2}{\text{minimize}} \quad J_{\kappa, \lambda}(\Xi) \\
& \text{subject to} \quad (3.2) \text{ for bijectiveness of } e(\cdot), \\
& \quad (3.6) \text{ with } Q = Q_1 \text{ for passivity,} \\
& \quad (3.7) \text{ with } Q = Q_2 \text{ for incremental stability.}
\end{aligned} \tag{3.16}$$

It is worth noting that $c \geq 0$ is automatically fulfilled in (3.15) and the positive definiteness of Q_1 and Q_2 is implicitly implied by (3.6) when $u, x = 0$ and by (3.7) when $u, x, \Delta, w = 0$, respectively. The ℓ_1 regularization and the incremental stability constraint are not required for generating passive models. However, both mechanisms improve the robustness of the generated models, as will be shown by examples in Section 7.

Chapter 4

Analysis of Convex Constraints

In this chapter, we analyze some properties of the convex passivity guarantee (3.6) and the convex incremental stability guarantee (3.7).

4.1 Passivity Guarantee

4.1.1 For LTI Systems

We impose the passivity guarantee (3.6) on the LTI descriptor systems in (2.9) and compare the corresponding conditions with the positive real lemma in Theorem 2.6. A major conclusion is that $D + D'$ needs to be full-rank in the generated models. Hence, the lossless cases are excluded in the feasible set.

Theorem 4.1. *Let $\Sigma(E, A, B, C, D)$ be an LTI descriptor state-space system with $x(0) = 0$ as in (2.9). The passivity constraint in (3.6) is equivalent to*

$$\begin{bmatrix} E - A + E' - A' - Q & C' - B & -(E + A)' \\ C - B' & D + D' & -B' \\ -(E + A) & -B & Q \end{bmatrix} \succeq 0, \quad (4.1)$$

for some $Q = Q' \succ 0$. The positive semidefiniteness of (4.1) with $P = Q^{-1}$ further

implies

$$-E'PA - A'PE \succeq \frac{1}{2}(E - A - Q)'P(E - A - Q) \quad (4.2)$$

$$D + D' \succeq 0, \quad (4.3)$$

$$\|B\|_{F,P}^2 \leq \text{tr}(D + D'), \quad (4.4)$$

$$\|C - B'\|_{F,(D+D')^\dagger}^2 \leq \text{tr}(E - A + E' - A' - Q). \quad (4.5)$$

Furthermore, if $D + D'$ is not full-rank, i.e., $\text{rank}(D + D') = m_r < m$, then there exists an invertible $U \in \mathbb{R}^{m \times m}$ such that

$$BU = \begin{bmatrix} \bar{B} & 0_{n \times (m-m_r)} \end{bmatrix}, \quad C'U = \begin{bmatrix} \bar{C}' & 0_{n \times (m-m_r)} \end{bmatrix} \quad (4.6)$$

for some $\bar{B}, \bar{C}' \in \mathbb{R}^{n \times m_r}$. And if $u(t) \in \ker(D + D')$, then

$$u(t)'y(t) = 0. \quad (4.7)$$

□

Proof. See Section 4.3.1. ■

In the theorem, $\text{tr}(\cdot)$ and $(\cdot)^\dagger$ denote the trace and the Moore-Penrose pseudoinverse of a matrix, respectively. We define $\|\cdot\|_{F,P}$ as the column weighted Frobenius norm when $P \succ 0$ or semi-norm when $P \succeq 0$ such that

$$\|X\|_{F,P}^2 = \text{tr}(X'PX) = \sum_{i=1}^n \|x_i\|_P^2, \quad (4.8)$$

where $X \in \mathbb{R}^{m \times n}$, $P \in \mathbb{R}^{m \times m}$, and $X = [x_1, x_2, \dots, x_n]$.

We can draw the following observations for LTI descriptor systems by comparing Theorem 4.1 with the positive realness in Theorem 2.6:

1. The constraint (4.2) implies the positive semidefiniteness of the first principal block matrix in Theorem 2.6 by noting $(E - A - Q)'P(E - A - Q) \succeq 0$, and (4.3)

is identical to the positive semidefiniteness of the second principal block matrix.

2. According to the Positive Real Lemma stated in Theorem 2.6, if $D + D'$ is rank-deficient and $\text{rank}(D + D') = m_r$, it can be derived that there also exists an invertible $U \in \mathbb{R}^{m \times m}$ such that

$$(C' - E'PB)U = \begin{bmatrix} R & 0_{n \times (m-m_r)} \end{bmatrix} \quad (4.9)$$

for some $R \in \mathbb{R}^{n \times m_r}$. This implies that $C'U$ and $E'PBU$ share some common columns. Theorem 4.1 requires a more restrictive condition (4.6), in which the common columns are identical to zeros.

3. A non-trivial model that satisfies the passivity constraint (3.6) has to be input-strict passive. This can be concluded from (4.6) or directly from (3.6) by checking the highest degree of u . In practice, a full-rank $D + D'$ is preferable. As suggested by (4.7), if $D + D'$ is not full-rank, there exist certain nonzero input signals such that the power consumption of the model is *instantaneously* zero for all time. Such behaviors unlikely exist in a practical system and, therefore, are of less interest. The matrix D can be physically interpreted as a resistive network attached between terminals, resulting in resistive loss.
4. The conditions (4.4) and (4.5) are extra constraints compared to the positive real lemma (2.10), stating that the range of the matrix B and the difference between B' and C are bounded by some metric of other parameters.

The additional restrictions (4.4) and (4.5) come from the convex relaxation (3.6) of the original passivity condition (2.5). To enlarge the feasible set, scaling the state equation by a positive factor, i.e., $\Sigma(E, A, B, C, D) = \Sigma(kE, kA, kB, C, D)$ with $k > 0$, provides an extra degree of freedom.

Corollary 4.2. *Let $\Sigma(kE, kA, kB, C, D)$ be an LTI descriptor state-space system as*

in (2.9). The passivity constraint in (3.6) is equivalent to

$$\begin{bmatrix} E - A + E' - A' - Q & C'/k - B & -(E + A)' \\ C/k - B' & (D + D')/k & -B' \\ -(E + A) & -B & Q \end{bmatrix} \succeq 0, \quad (4.10)$$

for some $Q = Q' \succ 0$. Accordingly,

$$\|\sqrt{k}B\|_{F,P}^2 \leq \text{tr}(D + D'), \quad (4.11)$$

$$\left\| \frac{1}{\sqrt{k}}C - \sqrt{k}B' \right\|_{F,(D+D')^\dagger}^2 \leq \text{tr}(E - A + E' - A' - Q). \quad (4.12)$$

□

Proof. See Section 4.3.2. ■

Therefore, we intentionally leave the state equation unnormalized in the optimization program (3.16). The scaling factor k is implicitly determined by the weight for the output equation errors κ and the regularization parameter λ as formulated in (3.13).

In comparison, state equation scaling does not affect the feasible region of the Positive Real Lemma (2.10). This can be seen from scaling E , A , and B in (2.10).

$$\begin{bmatrix} -(kE)'P(kA) - (kA)'P(kE) & C' - (kE)'P(kB) \\ C - (kB)'P(kE) & D + D' \end{bmatrix} \succeq 0. \quad (4.13)$$

The inequality (4.13) returns to the original condition (2.10) when k^2 is absorbed into the free variable P .

4.1.2 For First-Order Single-Port LTI Systems

The feasible region of the passivity constraint (3.6) can be visualized on a two-dimensional plane if we consider a first-order single-port LTI system. Because positive scaling of the input and the output preserves passivity, it is sufficient to study the

following normalized system

$$\dot{x} = -ax + u \quad (4.14)$$

$$y = x + du, \quad (4.15)$$

which we denote as $\Sigma(1, -a, 1, 1, d)$. The corresponding transfer function is

$$T(s) = \frac{1}{s+a} + d. \quad (4.16)$$

It can be seen that when $a \geq 0$ and $d \geq 0$, the system $\Sigma(1, -a, 1, 1, d)$ is passive.

The following theorem resolves the feasible region of the passivity constraint (3.6) for the system $\Sigma(1, -a, 1, 1, d)$

Theorem 4.3. *Let Σ be the first-order single-port LTI descriptor system in (4.14) and (4.15) with $a > 0$. The passivity constraint (3.6) is equivalent to*

$$\begin{bmatrix} 2+2a-q & 0 & -(1-a) \\ 0 & 2d & -1 \\ -(1-a) & -1 & q \end{bmatrix} \succeq 0, \quad (4.17)$$

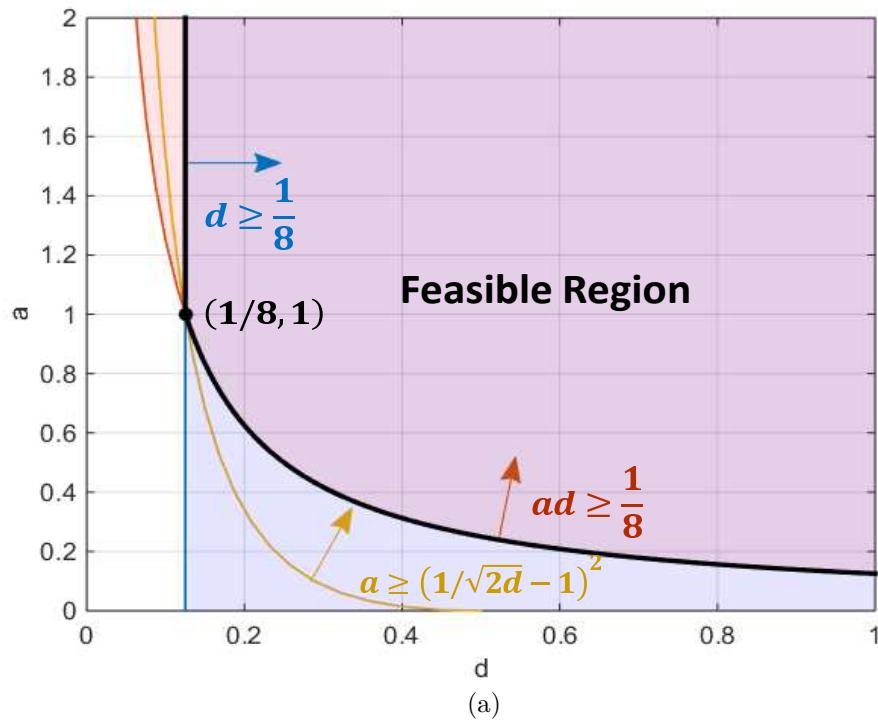
for some $q > 0$. The corresponding feasible region on the da -plane is bounded by

$$d \geq \frac{1}{8} \quad \text{and} \quad ad \geq \frac{1}{8}, \quad (4.18)$$

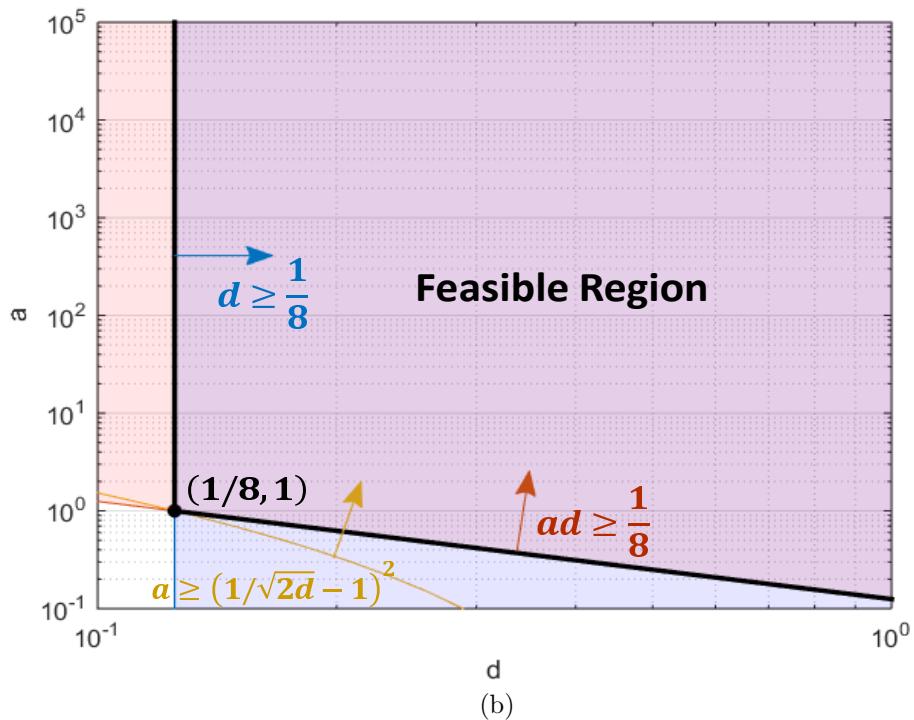
which is depicted in Figure 4-1. ■

Proof. See Section 4.3.3.

In comparison, the original feasible region due to the Positive Real Lemma is the first quadrant, i.e., $a, d \geq 0$. The passivity constraint (3.6) on the normalized first-order single-port LTI system (4.14) and (4.15) requires additional margins (4.18) from both axes. Therefore, some lossless or low-loss cases are excluded. A circuit analogy for such restrictions is made in Figure 4-2, in which both the series resistor R_s and



(a)



(b)

Figure 4-1: Feasible region for the convex passivity constraint (3.6) on the normalized first-order single-port LTI system (4.14) and (4.15) in (a) the linear scale and (b) the log-log scale.

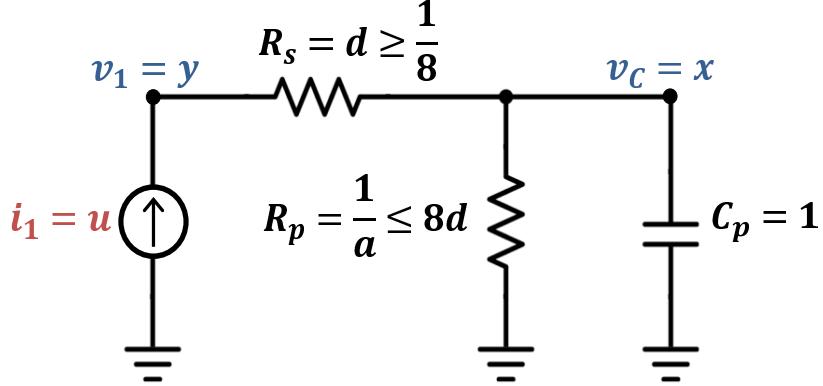


Figure 4-2: Circuit analogy of the normalized scalar LTI system in (4.14) and (4.15). The passivity constraint (3.6) requires the series resistor R_s to be bounded away from short-circuit and the parallel resistor R_p to be bounded away from open-circuit. The values of R_s and R_p determine the loss of the generated model.

the parallel resistor R_p are restricted to be finite according to (4.18).

4.1.3 For Nonlinear Systems with Polynomials e , f , and h

In this section, we assume e , f , and h in the model structure (3.1) are polynomials without the constant terms, i.e., $e(0) = 0$, $f(0,0) = 0$, and $h(0,0) = 0$. This representation makes the convex inequalities, such as (3.6) and (3.7), compatible with the sum-of-squares optimization, and, therefore, is used in our implementation. Analyzing the degrees of polynomials e , f , and h reveals useful insights of the generated models. We use $\deg(a)$ to denote the degree of polynomial a and use $\deg_z(a)$ to denote the highest degree of variable z in polynomial a .

Theorem 4.4. *Let Σ be a dynamical system in the form of (3.1) and assume there exists $Q = Q' \succ 0$ such that the passivity constraint (3.6) is satisfied for all u, x, q . Then $h(x, u) \neq h(x)$ and*

$$u'h(0, u) \geq 0, \quad (4.19)$$

for all u . Polynomial $f(x, 0)$ is either 0 or odd-degree, and

$$x'f(x, 0) \leq 0, \quad (4.20)$$

for all x . In addition, if $f(x, u) = \hat{f}(x) + g(u)$ and $\deg(h) = \deg(h(0, u))$, then

$$\deg(h) \geq \deg(g).$$

□

Proof. Let $q = -x$ and (3.6) becomes

$$u'h(x, u) + 2x'e(x) \geq 0, \quad (4.21)$$

for all u, x . If $h(x, u) = h(x)$, then (4.21) cannot be satisfied for all u . This shows that $h(x, u)$ must depend on u . The inequality (4.19) can be derived from (4.21) with $x = 0$.

For polynomial f , the convex passivity constraint (3.6) with $u = 0, q = x$ results in (4.20), which implies that $f(x, 0)$ can only be either odd-degree or 0.

Last, we prove $\deg(h) \geq \deg(g)$ under the assumption $f(x, u) = \hat{f}(x) + g(u)$. Without loss of generality, assume $h(x, u) = H_1(x, u)x + h_2(u)$ and $\deg(h) = \deg(h_2)$. Substituting $q = x$ into (3.6) gives

$$u'h_2(u) - 2x'g(u) + u'H_1(x, u)x - 2x'\hat{f}(x) \geq 0, \quad (4.22)$$

for all u, x . We would like to examine the coefficients for the highest degree of u and determine whether (4.22) can be satisfied. Since $\deg(h) = \deg(h_2)$, $\deg_u(u'H_1(x, u)) < \deg(u'h_2(u))$. Therefore, the third term is not decisive in determining the non-negativity for all u . Consider the first two terms. Assume $\deg(g) > \deg(h)$. Then $\deg(g(u)) \geq \deg(u'h_2(u))$. The terms with the highest n -th degree of u can be collected from $u'h_2(u) - 2x'g(u)$ and factored as $p(x)'u^n$, where the vector of polynomials $p(x)$ is linear in x . Since it is possible to choose an appropriate $x = x_0$ such that $p(x_0)'u^n$ tends to negative infinity as u approaches infinity, (4.22) cannot be satisfied for all u . This concludes that $\deg(h) \geq \deg(g)$. ■

Since the convex passivity constraint (3.6) is a sufficient condition for the original passivity inequality (2.5) with $V_s(x) = \frac{1}{2}\|e(x)\|_P^2$, it can be expected that some aforementioned conditions may also be applicable to (2.5). In fact, the conditions that (4.19) and $\deg(f_2) \leq \deg(h)$ hold when $f(x, u) = \hat{f}(x) + g(u)$ and $\deg(h) =$

$\deg(h(0, u))$ can be identically derived using similar arguments. The condition similar to (4.20) is

$$e'(x)Pf(x, 0) \leq 0, \quad (4.23)$$

for all x . In addition, the polynomial $h(0, u)$ must odd-degree, or $h(x, u) = h(x)$, the latter of which requires f to be affine in u as $f(x, u) = \hat{f}(x) + G(x)u$ such that

$$e'(x)PG(x) = h'(x). \quad (4.24)$$

The conditions in (4.23) and (4.24) together are called the Hill-Moylan condition [71] with respect to $V_s(x) = \frac{1}{2}\|e(x)\|_P^2$. Last, polynomial $f(x, 0)$ is similarly required to be either odd-degree or 0.

4.2 Incremental Stability Guarantee

4.2.1 For LTI systems

The structural similarity between the passivity guarantee (3.6) and the incremental stability guarantee (3.7) stimulates the interest in comparing their feasibility conditions. In the following, we compare the conditions of (3.6) and (3.7) on the LTI descriptor systems (2.9).

Theorem 4.5. *Let Σ be an LTI descriptor state-space system (E, A, B, C, D) as in (2.9) and $M = E - A + E' - A' - Q$. If Σ satisfies (3.7), then*

$$\begin{bmatrix} M & -(E + A)' & -\varepsilon E' \\ -(E + A) & Q & 0_n \\ -\varepsilon E & 0_n & \varepsilon Q \end{bmatrix} \succeq 0 \quad (4.25)$$

is satisfied. The positive semidefiniteness of (4.25) implies

$$-E'PA - A'PE \succeq \frac{1}{2}(E - A - Q)'P(E - A - Q), \quad (4.26)$$

$$E + E' - (A + A') \succeq Q + \varepsilon E'PE. \quad (4.27)$$

□

The proof is similar to Theorem 4.1 and, therefore, omitted. By comparing Theorem 4.1 and 4.5, it can be seen that (4.2) and (4.26) are identical, and (4.27) is more restrictive than the positive semidefiniteness of the first diagonal block matrix in (4.1) by an additional term $\varepsilon E'PE$. In other words, for LTI systems, the incremental stability constraint (3.7) imposes a similar yet more restrictive condition on E and A compared to the passivity constraint (3.6). Note that this similarity is limited to the LTI case and does not extend to general nonlinear systems.

4.2.2 For Nonlinear Systems with Polynomials e , f , and h

Similar to the convex passivity constraint, we assume e , f , and h in the model structure (3.1) are polynomials without the constant terms, i.e., $e(0) = 0$, $f(0, 0) = 0$, and $h(0, 0) = 0$. We would like to examine the degrees of polynomials e, f, h in the convex incremental stability constraint (2.7) and identify the corresponding restrictions on the model structure (3.1).

Theorem 4.6. *Let Σ be a dynamical system in the form of (3.1) and assume there exists $Q = Q' \succ 0$ such that the incremental stability constraint (3.7) is satisfied for all u, x, Δ, v, w . Then $f(x, u) = \hat{f}(x) + g(u)$ for some polynomials \hat{f} and g .*

□

Proof. Let $w = 0$ in (3.7). Rearranging the resulting inequality gives

$$2(\Delta - v)' \delta_e(x, \Delta) - 2(\Delta + v)' \delta_f(x, \Delta, u) - \Delta' Q \Delta + v' Q v \geq 0, \quad (4.28)$$

for all u, x, Δ, v . We would like to examine the highest degree of u in (4.28) in order identify the restrictions on f . Assume, to the contrary, that $f(x, u)$ is not of the form $\hat{f}(x) + g(u)$. Then there exist x_0, Δ_0 such that $f(x_0 + \Delta_0, u) - f(x_0, u)$ is a polynomial of u with a positive degree d , i.e., $\deg_u(\delta_f(x_0, \Delta_0, u)) = d > 0$. In this

case, the coefficient at u^d in

$$-2(\Delta_0 + v)' \delta_f(x_0, \Delta_0, u) \quad (4.29)$$

is a non-constant linear function of v and, hence, can be made negative by an appropriate selection of $v = v_0$ such that (4.29) tends to negative infinity as u approaches infinity, which contradicts (4.28). \blacksquare

4.3 Proofs

4.3.1 Proof of Theorem 4.1

1. Substituting $e(x) = Ex$, $f(x, u) = Ax + Bu$, and $h(x, u) = Cx + Du$ into (3.6) gives, for all x, u, q ,

$$\begin{bmatrix} x \\ u \\ q \end{bmatrix}' \begin{bmatrix} M & C' - B & -(E + A)' \\ C - B' & D + D' & -B' \\ -(E + A) & -B & Q \end{bmatrix} \begin{bmatrix} x \\ u \\ q \end{bmatrix} \geq 0, \quad (4.30)$$

which implies the positive semidefiniteness of (4.1).

2. The constraint (4.3) follows immediately from the positive semidefiniteness of the second diagonal block matrix.
3. Consider the principal submatrix

$$\begin{bmatrix} D + D' & -B' \\ -B & Q \end{bmatrix} \succeq 0 \quad (4.31)$$

Use the Schur complement and the fact $Q \succ 0$. Then

$$D + D' - B'PB \succeq 0. \quad (4.32)$$

Taking the trace results in (4.4).

4. For (4.2), consider the principal submatrix

$$\begin{bmatrix} M & -(E+A)' \\ -(E+A) & Q \end{bmatrix} \succeq 0. \quad (4.33)$$

The Schur complement gives

$$M - (E+A)'P(E+A) \succeq 0. \quad (4.34)$$

Rearranging terms, we have

$$-2E'PA - 2A'PE \succeq (E-A-Q)'P(E-A-Q), \quad (4.35)$$

which proves (4.2). Note that (4.2) also implies $M \succeq 0$.

5. For the principal submatrix

$$\begin{bmatrix} M & C' - B \\ C - B' & D + D' \end{bmatrix} \succeq 0, \quad (4.36)$$

because neither M nor $D + D'$ is strictly positive definite by the assumption, extra steps have to be taken to reach (4.5). Assume $\text{rank}(D + D') = m_r$ and let $\hat{m} = m - m_r$. Since $D + D' \succeq 0$ is symmetric, there exists an orthogonal matrix U such that

$$\tilde{D} = U'(D + D')U = \text{diag}(\bar{D}, 0_{\hat{m}}), \quad (4.37)$$

where $\bar{D} = \text{diag}(d_1, \dots, d_{m_r}) \succ 0$. Apply the congruent transformation $\tilde{U} = \text{diag}(I_n, U)$ to (4.36)

$$\tilde{U}' \begin{bmatrix} M & C' - B \\ C - B' & D + D' \end{bmatrix} \tilde{U} = \begin{bmatrix} M & \tilde{C}' - \tilde{B} \\ \tilde{C} - \tilde{B}' & \tilde{D} \end{bmatrix}, \quad (4.38)$$

where $\tilde{C} = U'C$ and $\tilde{B} = BU$. Because $\tilde{D}_{ii} = 0$ for $i > m_r$ and the positive semidefiniteness of (4.36) and (4.31), \tilde{B} and \tilde{C} can be further expressed as

$$\tilde{B} = \begin{bmatrix} \bar{B} & 0_{n \times \hat{m}} \end{bmatrix}, \quad \tilde{C}' = \begin{bmatrix} \bar{C}' & 0_{n \times \hat{m}} \end{bmatrix}, \quad (4.39)$$

which shows (4.6).

6. Taking the Schur complement of the m_r -by- m_r leading principal submatrix of (4.36) gives

$$M - (\bar{C}' - \bar{B})\bar{D}^{-1}(\bar{C} - \bar{B}') \succeq 0. \quad (4.40)$$

This is equivalent to

$$\begin{aligned} M &\succeq \begin{bmatrix} \bar{C}' - \bar{B} & 0_{n \times \hat{m}} \end{bmatrix} \begin{bmatrix} \bar{D}^{-1} \\ 0_{\hat{m}} \end{bmatrix} \begin{bmatrix} \bar{C} - \bar{B}' \\ 0_{\hat{m} \times n} \end{bmatrix} \\ &= (C' - B)(D + D')^\dagger(C - B'), \end{aligned} \quad (4.41)$$

which implies (4.5).

7. For the last statement (4.7), if $u = 0$ and $x(0) = 0$, then $u'y = 0$ for all $t \geq 0$ is trivial. If $u \neq 0$ and $(D + D')u = 0$, then $\text{rank}(D + D') = m_r < m$. Use the U in (4.37) and let $u = U\tilde{u}$, where the entries \tilde{u}_i are nonzero only when $i > m_r$.

Then

$$u'y = u'Cx + \frac{1}{2}u'(D + D')u = \tilde{u}'\tilde{C}x + \frac{1}{2}\tilde{u}'\tilde{D}\tilde{u} = 0. \quad (4.42)$$

The last equality holds because of (4.37) and (4.6). ■

4.3.2 Proof of Corollary 4.2

Proof. Substituting kE, kA, kB into (4.1) and absorbing the factor $1/k$ into the matrix Q gives (4.10). When $D + D'$ is full-rank, the inequalities (4.11) and (4.12)

can be obtained by taking the trace of the Schur complement on the submatrices

$$\begin{bmatrix} (D + D')/k & -B' \\ -B & Q \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} E - A + E' - A' - Q & C'/k - B \\ C/k - B' & (D + D')/k \end{bmatrix}, \quad (4.43)$$

respectively. When $D + D'$ is rank-deficient, the same argument can be made as in Step 5 and 6 in the proof of Theorem 4.1. \blacksquare

4.3.3 Proof of Theorem 4.3

Proof. Let M be the matrix in (4.17) and denote the principal minors of M with α rows and columns by M_α , where α is a multi-index. For example, $M_1 = 2 + 2a - q$, M_{12} is the determinant of the first two rows and columns of M , and $M_{123} = \det(M)$. The matrix M is positive semidefinite if and only if its all principal minors are non-negative, i.e., $M_1, M_2, M_3, M_{12}, M_{23}, M_{13}, M_{123} \geq 0$. For $M_{23} \geq 0$ to be satisfied, d must be strictly greater than 0. The constraints due to the non-negativity of each minor and the assumptions of the theorem can be summarized as $a, q, d > 0$ and

$$\max\left(\frac{1}{2d}, (1 - \sqrt{a})^2\right) \leq q \leq (1 + \sqrt{a})^2, \quad (4.44)$$

$$(1 - a)^2 \leq (2 + 2a - q)(q - \frac{1}{2d}), \quad (4.45)$$

where (4.44) is from $M_{13}, M_{23} \geq 0$ and (4.45) is from $M_{123} \geq 0$.

The next step is to eliminate the free variable q such that the feasible region is maximized. Consider maximizing the right-hand side of (4.45) with respect to q . Note that due to (4.44), both factors $(2 + 2a - q)$ and $(q - 1/(2d))$ are non-negative. It can be checked that the maximizer $q^* = 1 + a + 1/(4d)$ satisfies (4.44). Therefore, with q^* substituted, (4.45) becomes

$$(8d - 1)(8ad - 1) \geq 0. \quad (4.46)$$

This inequality determines two possible regions

$$d \geq \frac{1}{8} \quad \text{and} \quad ad \geq \frac{1}{8}, \quad (4.47)$$

$$d \leq \frac{1}{8} \quad \text{and} \quad ad \leq \frac{1}{8}. \quad (4.48)$$

The region bounded by (4.48) is ruled out because of (4.44), which implies

$$a \geq \left(\frac{1}{\sqrt{2d}} - 1 \right)^2. \quad (4.49)$$

The resulting feasible regions are, therefore, bounded by (4.49) and (4.47). It can be checked that the region bounded by (4.49) is included in the set of (4.47). Therefore, the final feasible region is defined by (4.47). Figure 4-1 depicts (4.49), (4.47), and the feasible region.

■

Chapter 5

Implementation

5.1 Basis Functions

We use monomials $x^\alpha \in \mathbb{R}$ as the basis functions for $e(x)$, where $x \in \mathbb{R}^n$ and α is an n -tuple multi-index with $\alpha_i \in \mathbb{N}_0, \alpha_i \geq 0$ such that

$$x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}. \quad (5.1)$$

The degree of x^α is $|\alpha| = \sum_i \alpha_i$. The i -th element in $e(x)$ of degree p is defined as a linear combination of all monomials x^α with degrees $0 < |\alpha| \leq p$, i.e.,

$$e_i(x) = \sum_{0 < |\alpha| \leq p} \theta_{e_i, \alpha} x^\alpha. \quad (5.2)$$

Note that here we explicitly exclude the constant term. The monomials for $f(x, u)$ and $h(x, u)$ are defined similarly with both variables u and x .

5.2 Derivative Estimation

The time derivatives of states $\tilde{x}(t_i)$ can be estimated from state data $\tilde{x}(t_i)$. The basic case is to use the central difference scheme

$$\tilde{x}(t_i) \approx \frac{\tilde{x}(t_{i+1}) - \tilde{x}(t_{i-1})}{t_{i+1} - t_{i-1}}. \quad (5.3)$$

When noise is present, robust numerical differentiation methods, such as the Savitzky-Golay filter [57] or the smooth noise-robust differentiator [29], can be applied.

5.3 Data Normalization

Properly normalizing data can improve the numerical property of the optimization problem (3.16). This is especially true for the identification process that involves the time derivatives at extremely high or low frequencies. In such a case, if without data normalization, certain descending directions during the optimization may not be updated efficiently, hence taking an exceedingly long time to converge, or even simply stall because of the arithmetic underflow of floating-point numbers.

Since the actual time is not present explicitly in the objective function in (3.13), the time derivatives can be individually normalized for *each* state such that \tilde{x} and $\tilde{\dot{x}}$ are at the same orders of magnitude

$$E_{\theta(\eta)}(\tilde{x})\tilde{\dot{x}}_\eta \triangleq (E_\theta(\tilde{x})H)(H^{-1}\tilde{\dot{x}}) = E_\theta(\tilde{x})\tilde{\dot{x}}, \quad (5.4)$$

where $H = \text{diag}(\eta)$, $\eta_i = \|\tilde{\dot{x}}_i\|/\|\tilde{x}_i\|$, in which the 2-norm is performed on discrete-time samples, and the notation $\theta(\eta)$ emphasizes that θ depends on η . The original unscaled θ is denoted by $\theta = \theta(\mathbf{1})$. Therefore, the estimate of unscaled $E_{\theta(\mathbf{1})}(x)$ can be obtained by

$$E_{\theta(\mathbf{1})}(x) = E_{\theta(\eta)}(x)H^{-1}.$$

For input and output normalization, it is desirable to choose suitable units such that the numerical values of inputs and outputs are commensurate in the same orders

of magnitude. In addition, simultaneously upscaling and downscaling an input-output pair preserves passivity

$$\hat{u}'\hat{y} = (K^{-1}u)'Ky = u'y, \quad (5.5)$$

where $K = \text{diag}(k_1, \dots, k_m)$, $k_i > 0$. Such k_i can be empirically determined by inspecting training data, for instance, $k_i = \sqrt{\|\tilde{u}_i\|/\|\tilde{y}_i\|}$, in which the 2-norm is performed on discrete-time samples. Besides, scaling *all* inputs or *all* outputs by a positive constant, i.e., $\hat{u} = \alpha u$ or $\hat{y} = \alpha y$ with $\alpha > 0$, also preserves passivity. Formally, if a system with inputs u and outputs y is passive with respect to a storage function $V_s(x) \geq 0$ for all x , i.e., $\int_0^T u'y dt \geq V_s(x(T)) - V_s(x(0))$ for all $T \geq 0$, then

$$\int_0^T \hat{u}'\hat{y} dt = \alpha \int_0^T u'y dt \geq \alpha V_s(x(T)) - \alpha V_s(x(0)), \quad (5.6)$$

for all $T \geq 0$. Hence, the system with inputs αu and outputs y or the system with inputs u and outputs αy is passive with respect to the storage function $\alpha V_s(x)$.

5.4 Optimization Tools

We use Yalmip [39] to prepare the SOS optimization problem (3.16) into an SDP program. The resulting SDP problem is then solved using Mosek [1]. Both of the tools are interfaced with MATLAB [43].

5.5 Simulation Environment

The polynomials e , f , and h generated by (3.16) are converted into Simscape Language [44] and compiled into Simscape models. The resulting compiled models are interconnected in Simscape and simulated using MATLAB ordinary differential equation solvers.

Chapter 6

Model Training Practice

This chapter outlines the general procedures to train nonlinear passive models using the proposed method (3.16).

6.1 Data Preparation

We use training data to train models, validation data to determine the proper parameters κ and λ in (3.16), and test data to test the quality of the resulting model.

6.1.1 Intended Usage of the Model

The first step to prepare data is to determine the intended signaling and the intended environment for the system to operate. Unlike modeling linear systems, for which transfer functions depend only on frequencies and delays, modeling nonlinear systems requires the additional information, such as amplitudes, waveform types, and interconnection conditions, which may also be nonlinear and dynamical. It is also unlike modeling standalone systems, for which users can directly specify input signals. When a model is interconnected within a physical network, the actual input signals are altogether determined by the overall system in connection. Because enumerating all combinations of signals and interconnections are generally prohibited, one must consider a confined range of operations such that the system behaviors within that

range can be thoroughly explored.

6.1.2 Training Data

Training data are collected from operating the system of interest with the intended signaling and within the intended environments. Since polynomials are typically not ideal for extrapolation, the training data should include the input signals with maximal amplitudes. The nonlinear behaviors should be specifically investigated, whereas the linear parts can typically be interpolated well. The resulting training data Ξ_{Trng} include K sets of inputs $\tilde{u}(t_i)$, states $\tilde{x}(t_i)$, outputs $\tilde{y}(t_i)$, and the time derivatives of states $\tilde{\dot{x}}(t_i)$, which can be measured directly or estimated from $\tilde{x}(t_i)$. Each set is designed to explore some aspect of the system behaviors,

$$\Xi_{\text{Trng}} = \{(\tilde{u}_i, \tilde{x}_i, \tilde{y}_i, \tilde{\dot{x}}_i)_k \mid i = 1, \dots, N_k, k = 1, \dots, K\}. \quad (6.1)$$

6.1.3 Validation Data

Validation data are used to evaluate the performance of the models generated using the training data set Ξ_{Trng} . The quality of models can be measured, for instance, using the \mathcal{L}_2 or \mathcal{L}_∞ errors defined later in (6.2) against the validation data Ξ_{Val} . Therefore, the validation data Ξ_{Val} should be chosen based on the same assumption of the intended signaling and the intended environments but are different from what have been used in the training data set Ξ_{Trng} .

6.1.4 Test Data

The resulting generated model from the validation phase is further tested when interconnected within a physical network. The overall interconnected system is fed with practical input signals and the corresponding responses are collected at both the outputs of the generated model and the outputs of the overall system. The former gives an idea about the local modeling errors, and the latter indicates the influence of such modeling errors on the entire system behaviors. Accordingly, the test data Ξ_{Test}

consist of the test physical network, the practical input signals, and the corresponding output signals from the local system being modeled and the overall system.

6.2 Error Metrics

We use the \mathcal{L}_2 error and the \mathcal{L}_∞ error to quantify the relative error between the model outputs $\hat{y}(t)$ and the reference $\tilde{y}(t)$ for $t \in [0, T]$

$$\text{Err}_{\mathcal{L}_p}(\hat{y}, \tilde{y}) = \frac{\|\hat{y} - \tilde{y}\|_{\mathcal{L}_p}}{\|\tilde{y}\|_{\mathcal{L}_p}}. \quad (6.2)$$

When $p = 2$,

$$\|y\|_{\mathcal{L}_2} = \sqrt{\int_0^T \|y\|_2^2 dt}. \quad (6.3)$$

When $p = \infty$,

$$\|y\|_{\mathcal{L}_\infty} = \sup_{t \in [0, T]} \|y\|_\infty. \quad (6.4)$$

Intuitively, the \mathcal{L}_2 error measures the *average* error across the period T , whereas the \mathcal{L}_∞ error measures the *maximum* error among the period T . In practice, since $\hat{y}(t)$ and $\tilde{y}(t)$ are normally not sampled at the same discrete-time points t_i , we use the cubic spline to interpolate $\hat{y}(t)$ at the sample time of $\tilde{y}(t)$.

6.3 Model Complexity

6.3.1 Polynomial Degrees

A model of higher degrees of polynomials e, f, h is generally more expressive to fit the training data but at the same time tend to overfit the training data. Besides, the number of coefficients θ of a polynomial grows rapidly with the degree d and the number of variables n as

$$\text{Dim}(\theta) = \binom{n+d}{d}. \quad (6.5)$$

Therefore, it is always desirable to train a model with lower degrees of polynomials as long as the model meets the error criteria.

6.3.2 Stiffness

Stiffness of a dynamical model simulated using a numerical method influences the size of time steps taken to integrate over time. Although there is no universal definition for the stiffness of a general ordinary differential equation, we can still empirically measure the stiffness of a model along a trajectory using the average of step sizes $\bar{\tau}$ and their normalized standard deviation $s_{\tau,n} = s_\tau / \bar{\tau}$. Unreasonably small averages or comparatively large variations signify possible stiff models. Given the same inputs and similar qualities of outputs, a model with larger average step sizes $\bar{\tau}$ and smaller deviations $s_{\tau,n}$ is generally preferred.

6.4 Modeling Procedure

The training procedure is flowcharted in Figure 6-1. After the data sets are prepared and a range of κ and λ are specified, the process begins with the linear case, i.e. $\deg(e, f, h) = (1, 1, 1)$.

In the training phase, a set of models $\Sigma_{\kappa_i, \lambda_j}(\Xi_{\text{Trng}})$ are generated for each pair of κ_i and λ_j using the proposed program (3.16). The generated models are simulated using the same training input and then forwarded to the next phase for validation if the simulation can be successfully performed. The models that are too stiff to complete the simulation are discarded.

In the validation phase, the generated models from the training phase are simulated and compared against the validation data. The model with the minimal errors, for instance, the minimal \mathcal{L}_2 errors as defined in (6.2), is adopted and the corresponding κ_i and λ_j are denoted as κ^* and λ^* , respectively. If such a model $\Sigma_{\kappa^*, \lambda^*}(\Xi_{\text{Trng}}, \Xi_{\text{Val}})$ is accurate enough, then the model is forwarded to the test phase. If not, the degrees of polynomials e, f, h is incremented and the training phase is restarted. In practice, it is typically preferable to increment the degrees of e and f first and retain a low-

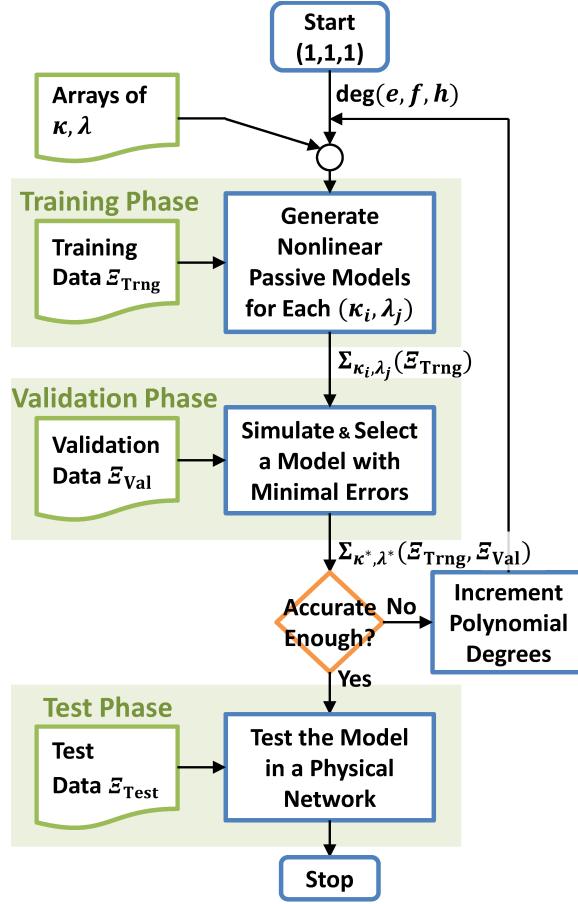


Figure 6-1: Flowchart for the model training procedure.

degree h . This is because a high-degree output equation h tends to cause an ill-posed interconnection that is algebraically difficult or impossible to solve.

In the test phase, the model $\Sigma_{\kappa^*,\lambda^*}(\Xi_{\text{Trng}}, \Xi_{\text{Val}})$ from the validation phase is converted into a Simscape model and tested within an interconnected network against the test data to evaluate the model performance.

Chapter 7

Case Studies: Circuit Networks

In this chapter, we use our proposed algorithm to generate nonlinear passive models for two circuit examples: a diode RC line and a Class E power amplifier. Several system properties of the generated models are studied through numerical examples using the diode RC line as the test platform. It is worth emphasizing that power amplifiers are *passive in the sense of energy consumption*, as we explained in Chapter 1, and, hence, can be handled by the proposed algorithm.

7.1 Diode RC Line

Figure 7-1 shows a diode RC line consisting of multiple identical stages modified from [12]. Our goal is to identify a model for a *single-stage* subcircuit and expect that the single-stage model can be cascaded into an accurate diode RC line model. It should be noted from Figure 7-1 that a single stage of the diode RC line is a three-terminal subsystem. Each terminal has its own voltage and current. The degrees of freedom are reduced by two due to the KCL and the KVL algebraic constraints. Therefore, by treating Terminal 3 in the figure as the common node, we consider only two currents, i_1 , i_2 , and two voltages, $v_{13} = v_1 - v_3 = v_1$, $v_{23} = v_2 - v_3 = v_2$, as the inputs and outputs of a single-stage subsystem.

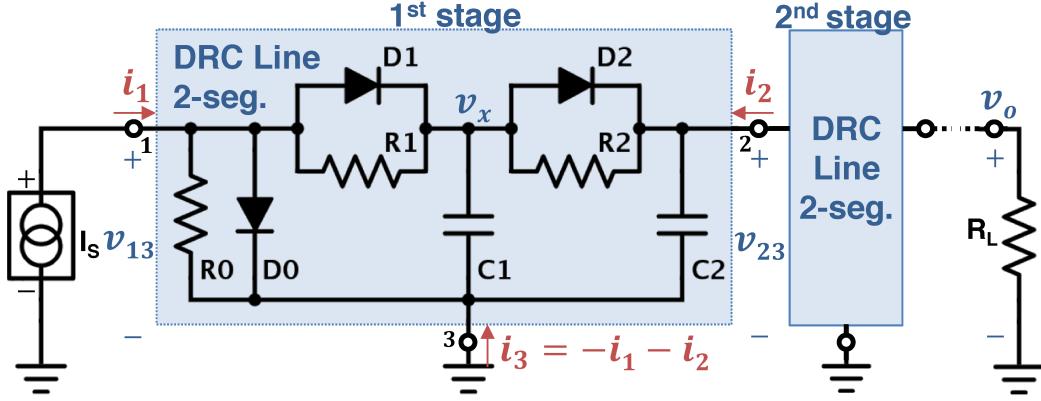


Figure 7-1: A diode RC line by cascading stages

7.1.1 Training

We assume that the circuit is intended to operate under sinusoid excitation between 1 GHz and 4 GHz and interconnected with identical stages. The last stage is terminated with a resistor around 3Ω . Accordingly, we devise a set of training input currents: (1) a large single-tone 1 A at 1 GHz, (2) a large single-tone 1 A at 4 GHz, and (3) a two-tone 0.6 A and 0.4 A current at non-harmonic 2 GHz and 2.5 GHz, respectively. Each of the excitation is fed to a two-stage setup as in Figure 7-1, terminated with either $R_L = 2\Omega$ or $R_L = 5\Omega$. The voltages v_1 , v_2 , and v_x in the figure are combined as the states x , from which the time derivatives \dot{x} are estimated using the central difference scheme. We further consider the inputs and outputs for both the 1st and 2nd stages in Figure 7-1 in order to incorporate the behaviors when cascading stages. Combining all the above factors produces $K = 12$ sets of training data (6.1). The total number of data points is $\sum_k N_k = 2408$. We train the models for κ and λ in a logarithmic scale across the range 10^{-3} to 10^2 and 10^{-8} to 10^{-1} , respectively.

7.1.2 Validation

For the validation set, we prepare a non-harmonic two-tone signal at 2 GHz and 3 GHz with both amplitudes 0.5 A, and an envelope-modulated 1 A sinusoid at 2 GHz, both of which are terminated with $R_L = 3\Omega$. Again, we consider both stages in Figure 7-1. The resulting validation data set, Ξ_{Val} , consists of four sets of input and output

Table 7.1: Performance of standalone single-stage diode RC models

Degree			$\log(\cdot)$		# of Coeff.	Optim. Time (s)	Simulation		Error (%) [*]	
e	f	h	κ	λ			$\mathcal{L}_{2,\text{mean}}$	$\mathcal{L}_{2,\text{(std)}}$	$\mathcal{L}_{\infty,\text{mean}}$	$\mathcal{L}_{\infty,\text{(std)}}$
1	1	1	0	-3	34	0.82	5.14	(1.16)	6.05	(2.19)
3	3	1	-1	-3	188	1.66	2.97	(0.82)	4.08	(1.58)
3	3	3	0	-3	320	1.73	1.65	(0.75)	2.30	(1.04)
5	5	3	-1	-8	845	33.1	0.80	(0.40)	1.31	(0.57)
5	5	5	-1	-7	1342	59.9	0.75	(0.39)	1.22	(0.51)
1	1	1	0	-3	34	0.82	5.14	(1.16)	6.04	(2.20)
3	3	1	0	-3	151	2.59	3.07	(1.04)	4.89	(1.91)
3	3	3	0	-6	263	2.79	1.48	(0.42)	2.16	(0.66)
5	5	3	0	-7	604	131	1.05	(0.37)	1.51	(0.45)
5	5	5	0	-7	978	155	1.03	(0.32)	1.45	(0.38)

[†] Upper: Passive models with ℓ_1 regularization but without incremental stability.

[‡] Lower: Passive models with ℓ_1 regularization and incremental stability.

* See (6.2) for the definition.

signals.

The performance of the validated best models for certain $\deg(e, f, h)$ are summarized in Table 7.1. The number of nonzero coefficients of a model is the dimension of θ , i.e., the total number of coefficients in e , f , and h . The optimization time is the time for solving the program in (3.16) for a single (κ, λ) pair on a desktop computer with an Intel i7-2600K CPU and 16 GB RAM. For simulation errors, we calculate the means and the standard deviations of both the \mathcal{L}_2 and the \mathcal{L}_∞ errors across the validation set. Figure 7-2 shows sample training signals that exhibit the nonlinear saturation due to large amplitudes. A two-tone signal and an envelope-modulated signal from the validation set are shown in Figure 7-3. It can be seen from the figure that when simulated alone, our generated model of $\deg(e, f, h) = (3, 3, 3)$ can faithfully reproduce both the training and the validation data.

7.1.3 Test within a Physical Network

We test our models by cascading them repetitively as in Figure 7-1. Figure 7-4 shows the test responses of cascading two, three, and four identical models of degrees $(3, 3, 3)$. The input current source is set to be a triangular wave up to the 5th harmonic

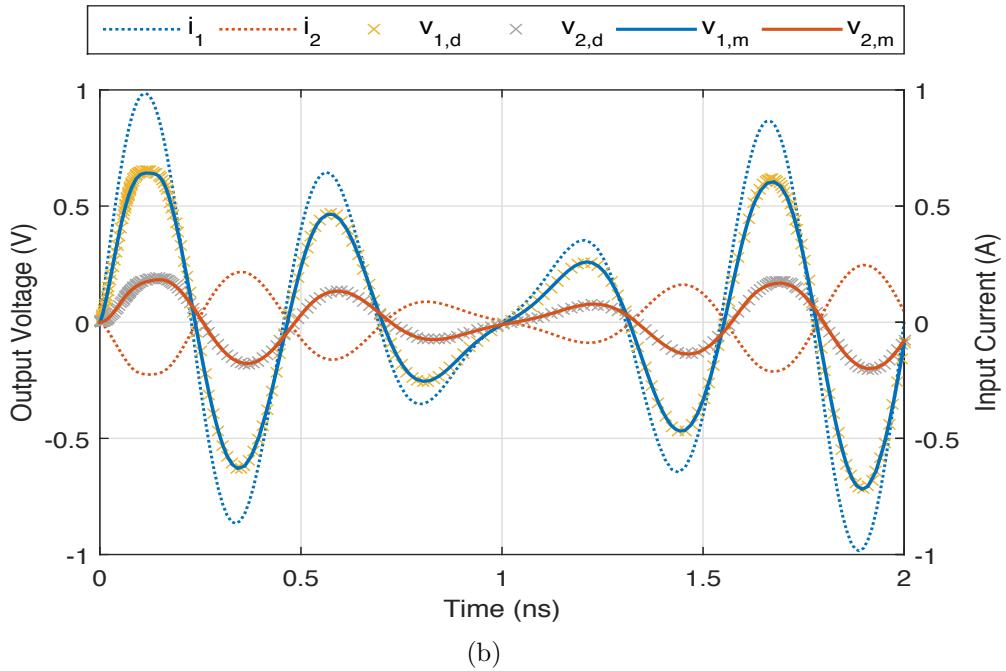
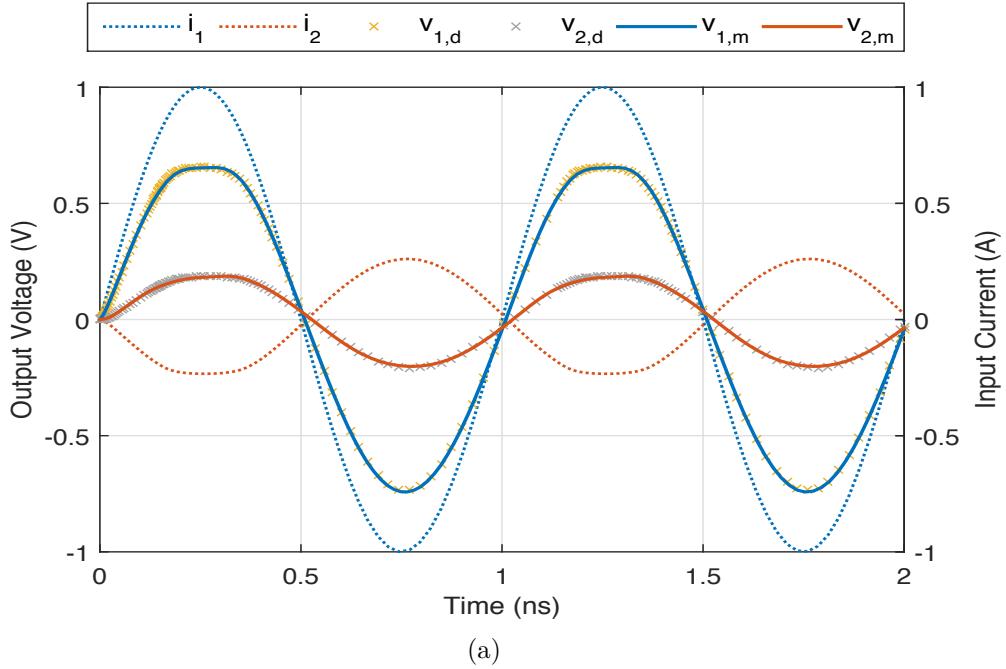


Figure 7-2: Sample training signals for a single-stage diode RC line. Crossed markers denote reference data, subscripted by d , and solid lines are generated from simulating our identified model of degrees $(3, 3, 3)$, subscripted by m . (a) A large single-tone sinusoid at 1 GHz with $R_L = 2\Omega$. (b) A two-tone signal with amplitudes 0.6 and 0.4 A at 2 and 2.5 GHz terminated with $R_L = 5\Omega$.

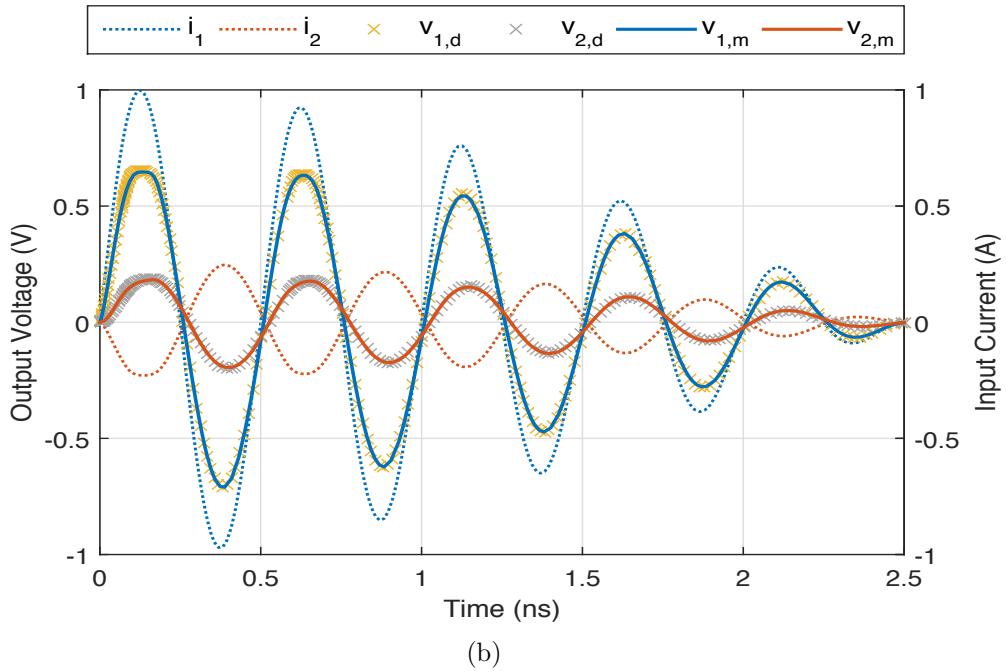
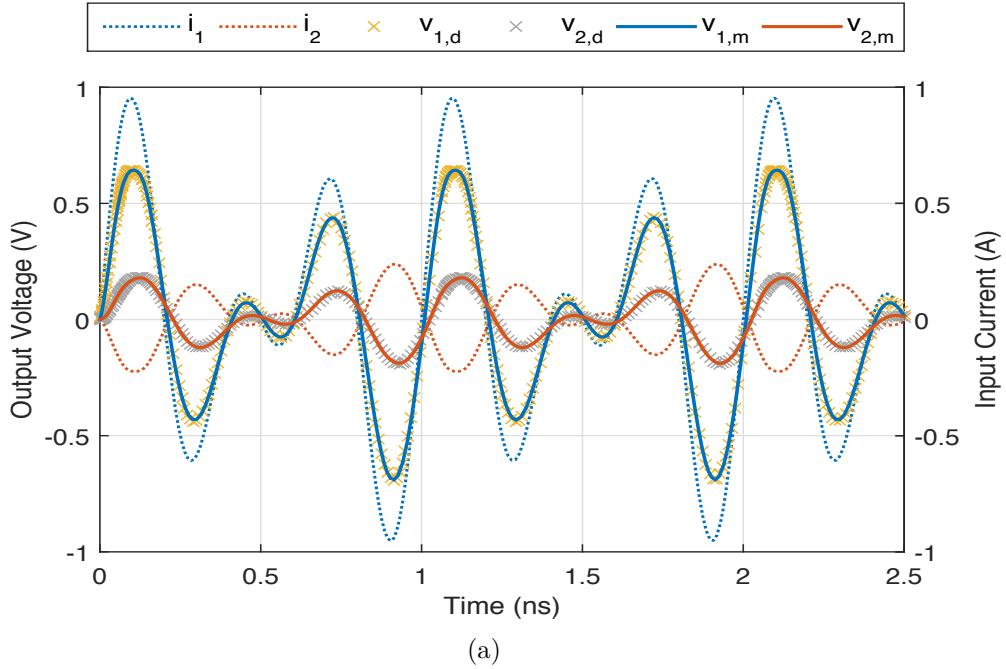


Figure 7-3: Sample validation signals for a single-stage diode RC line. Crossed markers denote reference data, subscripted by d , and solid lines are generated from simulating our identified model of degrees $(3, 3, 3)$, subscripted by m . (a) A two-tone signal with both amplitudes 0.5 A at 2 and 3 GHz terminated with $R_L = 3 \Omega$. (b) An envelope-modulated sinusoid at 2 GHz with $R_L = 3 \Omega$.

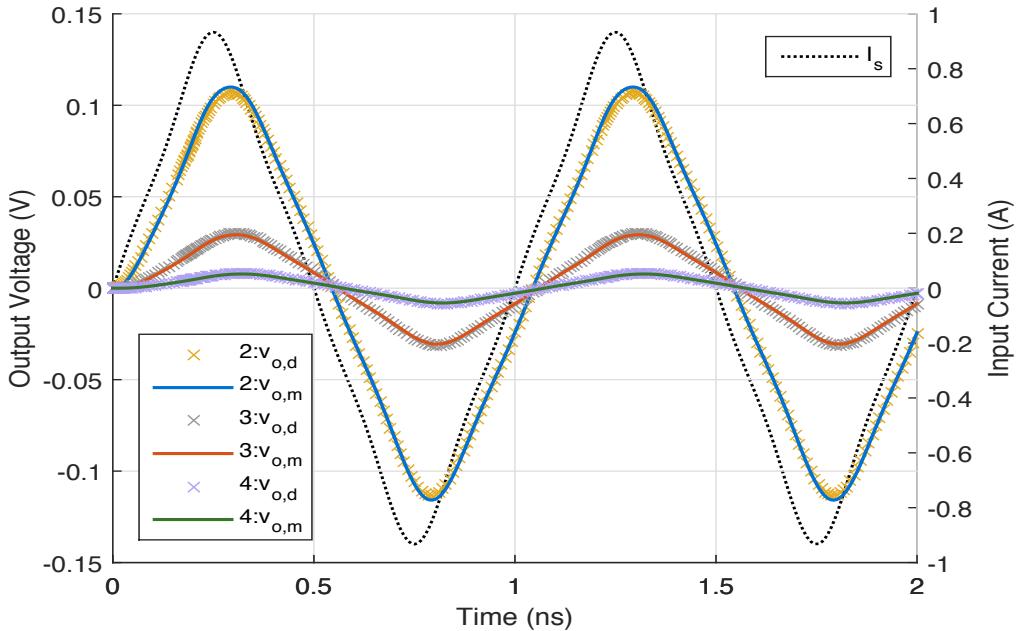


Figure 7-4: Test responses of cascading two, three, and four identical post-validation diode RC models of degrees $(3, 3, 3)$ with a load $R_L = 3 \Omega$. Subscripts d and m denote the references and the simulations of the generated models, respectively.

Table 7.2: Performance of interconnected multi-stage diode RC models

Degree			2-Stage Err. (%) [*]		3-Stage Err. (%) [*]		4-Stage Err. (%) [*]	
e	f	h	\mathcal{L}_2	\mathcal{L}_∞	\mathcal{L}_2	\mathcal{L}_∞	\mathcal{L}_2	\mathcal{L}_∞
1	1	1	4.77	5.90	6.58	7.69	8.35	9.46
3	3	3	2.15	3.30	2.86	2.56	5.72	5.14
5	5	5	1.85	3.61	1.57	1.92	2.70	3.13

* See (6.2) for the definition.

with the fundamental frequency at 1 GHz. It can be seen that although the model is originally trained for the two-stage case, the cascaded models with an extended number of stages still perform accurately. We summarize the test errors in Table 7.2.

7.2 Experiments on Effects of System Properties

In this section, we use the diode RC line as the test platform to study the system properties of the generated models.

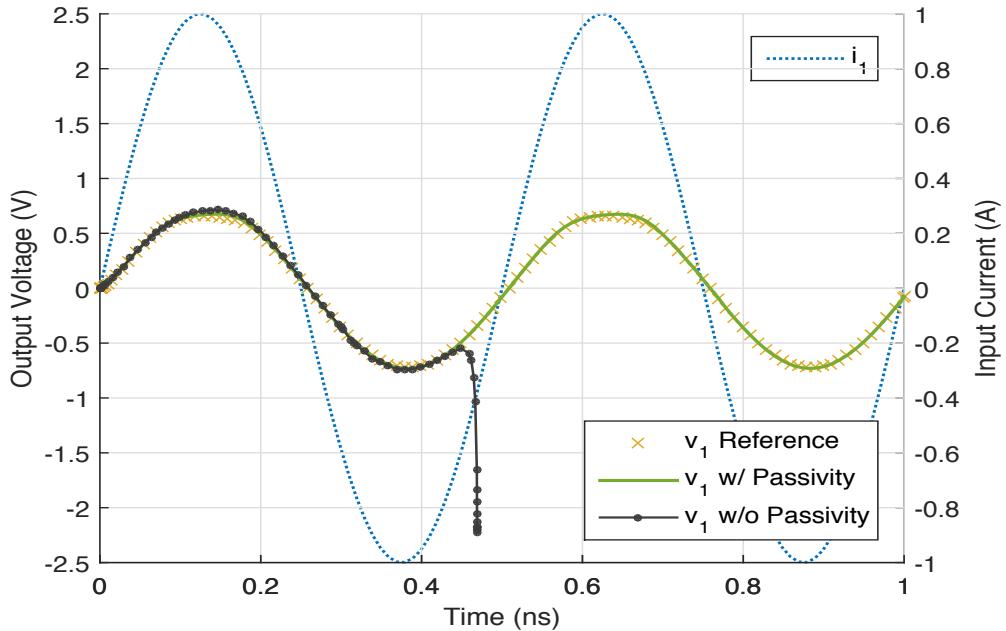


Figure 7-5: Effects of passivity guarantee. The models of degrees (5,5,5) with and without passivity are cascaded into a four-stage chain with $R_L = 3\Omega$. Simulations are performed using the trapezoidal rule with an identical accuracy setting of adaptive step sizes.

7.2.1 Effects of Passivity Guarantee

Figure 7-5 compares the simulation results of the models trained *with* and *without* the passivity guarantee (3.6). The models are of degrees (5,5,5) and cascaded into a four-stage chain terminated with $R_L = 3\Omega$ as the test system. Only the input current i_1 and the output voltage v_1 of the first stages are depicted. All training and validation procedures are identical except for whether the passivity constraint (3.6) is included in the optimization program (3.16). The resulting four-stage chain is simulated using the MATLAB `ode23t` trapezoidal rule with the same accuracy setting of adaptive step sizes.

When simulated alone, these two models, trained with and without the passivity guarantee, are both accurate for the input signal in Figure 7-2a with \mathcal{L}_2 errors 0.84% and 0.58%, respectively. However, when simulated as an interconnected four-stage chain, the model trained *without* the passivity guarantee deviates sharply from the references and fails to complete the simulation, whereas the model trained *with* the

passivity guarantee can faithfully reproduce the system behaviors with an \mathcal{L}_2 error 2.02%.

7.2.2 Effects of Incremental Stability and Regularization

Figure 7-6 compares the performance of the passive models of degrees (5, 5, 5) trained (a) without the ℓ_1 regularization and (b) with the ℓ_1 regularization. In each plot, the comparison is made for the models trained *with* and *without* the incremental stability. In Figure 7-6a, we use the same input current as in Figure 7-3b. It can be seen from the figure that without the regularization, the generated models are sensitive to the errors due to numerical integrations and result in drastic deviations from the reference. The deviation is even more severe for the model trained without incremental stability.

In Figure 7-6b, we use a single-tone sinusoid at 2 GHz from the training set except with a 25% larger amplitude. Both of the models are trained with the regularization and agree with the references, as has been presented in Figure 7-2a and Table 7.1. However, when the models are operated *beyond* the intended range (amplitude in this case) and begin to extrapolate, the incrementally stable model appears to be more stable and robust against such a perturbation. Being able to extrapolate smoothly is important because when models are interconnected within a physical network, users do not have the full control of input signals. Besides, our experiments in Table 7.1 show that the model qualities are less sensitive to the choice of κ and λ when the incremental stability is enforced (the lower half of the table). Therefore, in practice, the vectors κ and λ can be sampled at a coarser grid and, therefore, fewer models need to be generated.

7.2.3 Effects of Weight κ and Regularization

In this section, we conduct the experiments to assess the effects of the weight κ and the regularization parameter λ . We are interested in understanding their influences on model performance for different sizes of simulation time steps. Unlike the

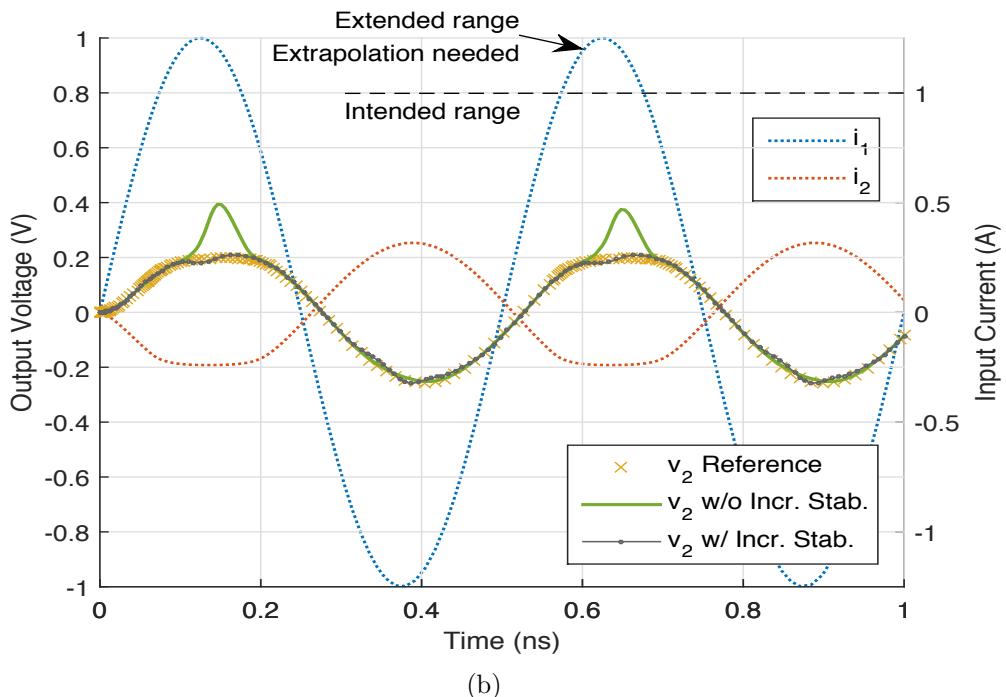
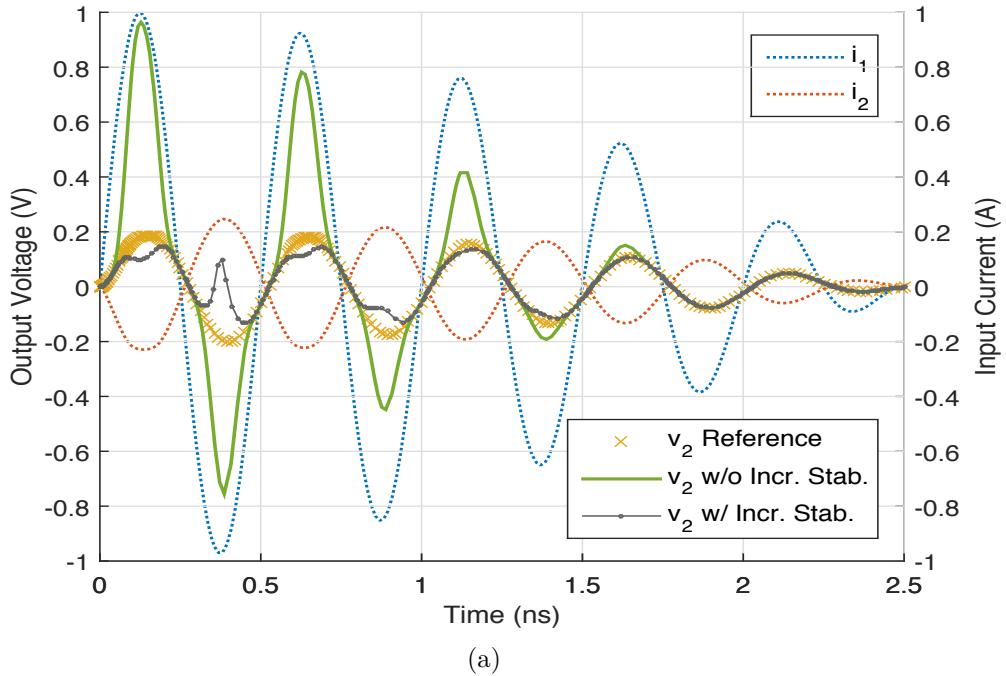


Figure 7-6: Effects of regularization and incremental stability. The single-stage models of degrees (5, 5, 5) are (a) trained without the regularization, and (b) trained with the regularization but operated beyond the intended range of amplitude. All models are simulated alone.

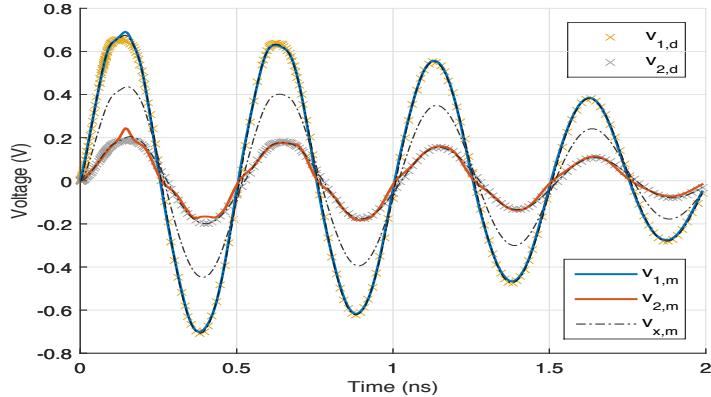
discrete-time systems, in which models are evaluated at fixed time points, the time points in the continuous-time systems are determined by various factors, such as simulation methods, adaptive algorithms for step sizing, and also other models that are interconnected within the system. A reasonable continuous-time dynamical model should behave accurately and consistently across a range of step sizes. Therefore, this is an extra complexity in modeling continuous-time systems compared with the discrete-time modeling.

The models generated by the program (3.16) can be classified based on the value of κ : small, medium, and large κ . When κ is small, the objective function (3.13) tends to overlook the output equation errors and focus on minimizing the state equation errors. When κ is large, the output equation tends to be overfitted to data whereas the state equation tends to be poorly fitted. The preferred case is the medium κ , for which both the state and the output equation errors are properly minimized.

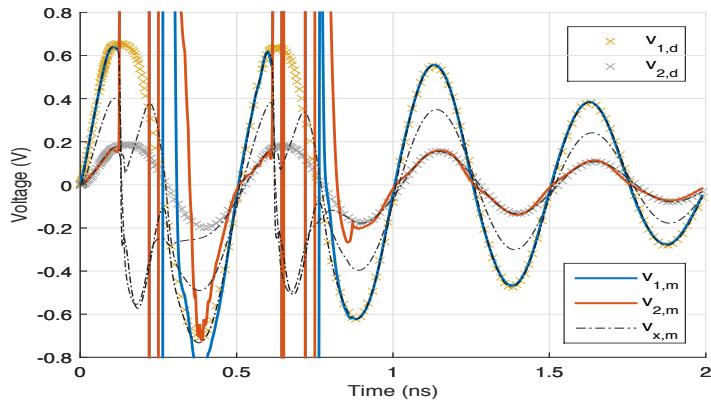
A generated model may suffer from the overfitting problems in either or both the state equation and the output equation. Such overfitting problems may be observed by simply changing simulation time steps. This is because, even when the input signals are identical to the training data, the model still needs to interpolate if the time steps are different during simulation. If the states of the model deviate significantly for *certain* choices of step sizes, the state equation may be overfitted. On the other hand, if the states behave uniformly for various step sizes but the output signals vary drastically, it is likely that the output equation is overfitted.

To demonstrate the effects of κ and how regularization helps alleviate the overfitting problems, we use two sets of experiments in Figure 7-7, 7-8, and Figure 7-9, 7-10. The models of degrees (5, 5, 5) in these figures are trained either with small or with medium κ , and either with or without regularization. The incremental stability constraint (3.7) is not included in these cases so that the effect of regularization can be fully revealed. The adaptive step sizes are determined by MATLAB `ode23t`, but the maximum step size in each experiment is controlled by our specified values.

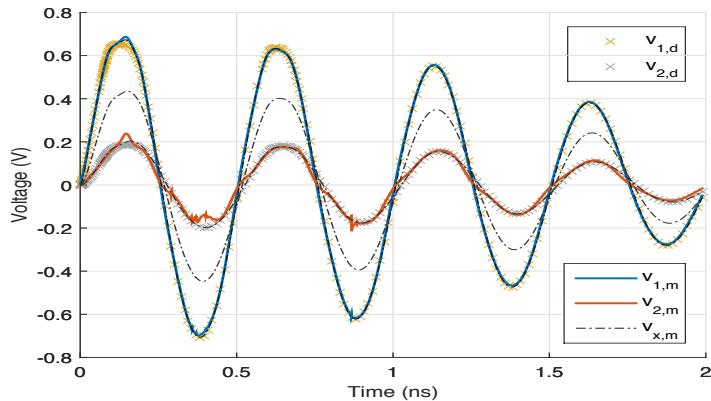
Figure 7-7 shows the simulation of a model trained with a small $\kappa = 10^{-5}$ and without regularization for various maximum step sizes. The input signals are the



(a) Max step size = 10 ps

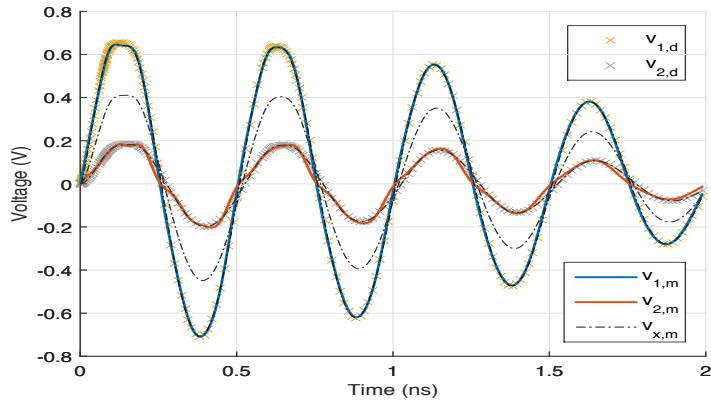


(b) Max step size = 1 ps

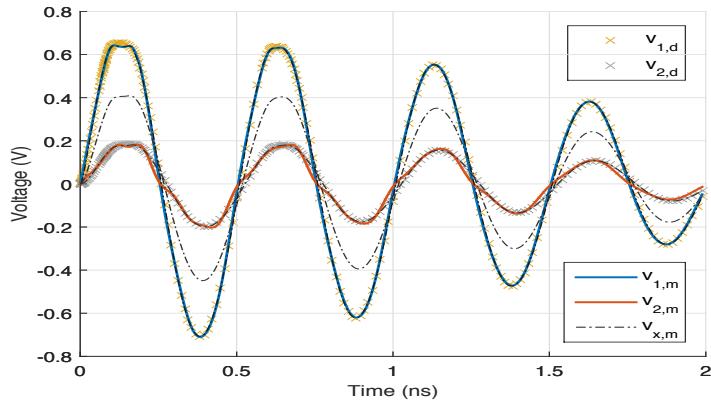


(c) Max step size = 0.1 ps

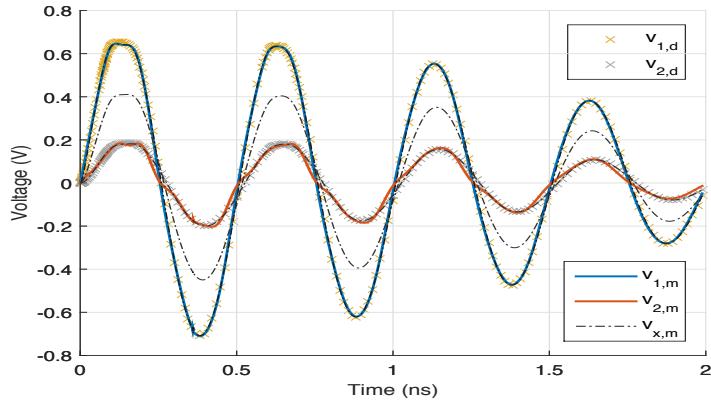
Figure 7-7: Overfitting in the state equation. A single-stage diode RC model is trained with a small $\kappa = 10^{-5}$ and without regularization. From (a) to (c), the model is simulated using `ode23t` with the inputs in Figure 7-3b at different maximum time steps. Subscripts d and m denote reference data and model simulation, respectively, and x , 1, and 2 denote states, port 1, and port 2. Since κ is small, the output equation is a poor fit. Without regularization, the oscillatory behavior only appears in specific step sizes as in (b), indicating the overfitting problem in the state equation.



(a) Max step size = 10 ps

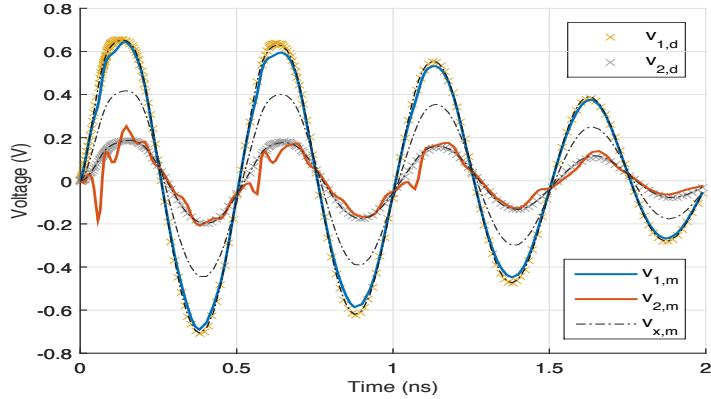


(b) Max step size = 1 ps

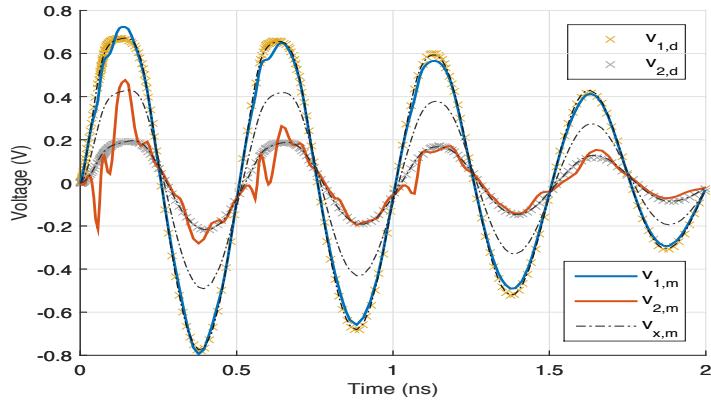


(c) Max step size = 0.1 ps

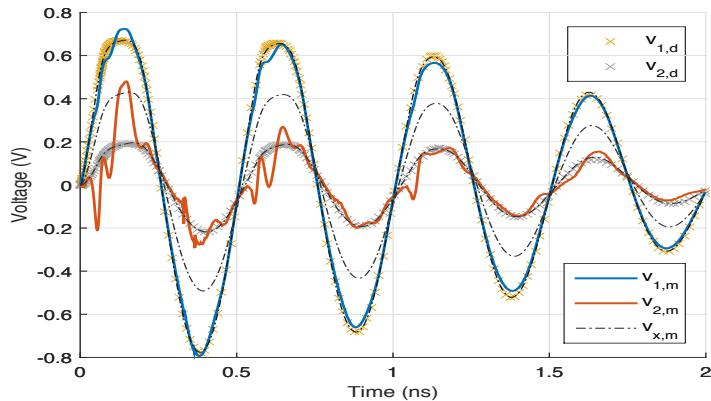
Figure 7-8: (*Continued*) Overfitting alleviated by regularization. A single-stage diode RC model is trained with a small $\kappa = 10^{-5}$ and with regularization $\lambda = 10^{-7}$. With the regularization, the state trajectories behave consistently across various step sizes, while the output equation is still a poor fit due to the small κ .



(a) Max step size = 100 ps, input as in Figure 7-3b

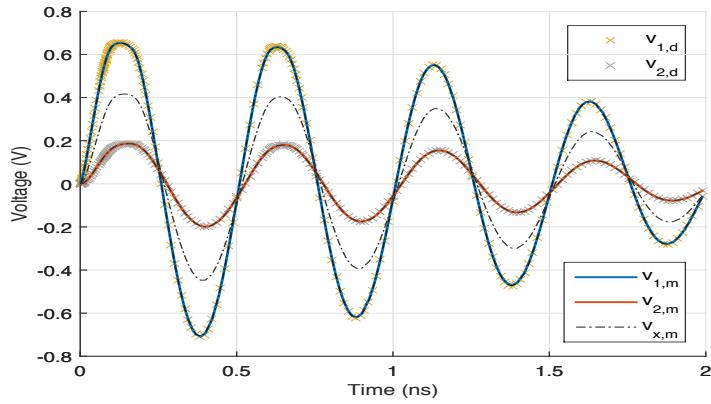


(b) Max step size = 100 ps, input with 10% larger amplitude

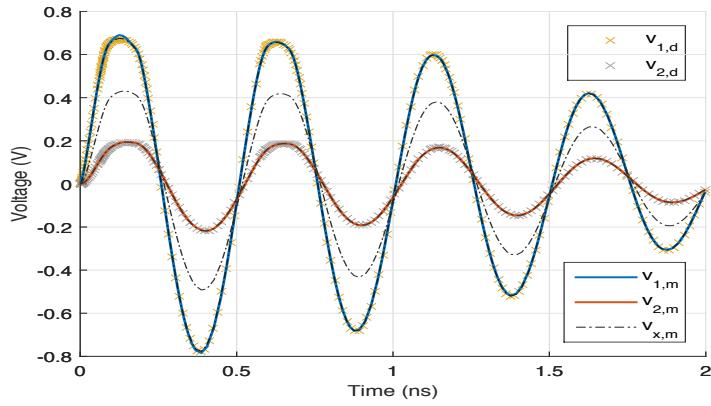


(c) Max step size = 0.1 ps, input with 10% larger amplitude

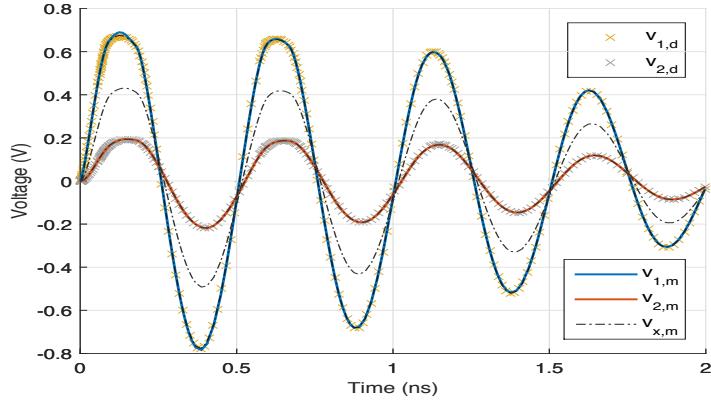
Figure 7-9: Overfitting in the output equation. A single-stage diode RC model is trained with a medium $\kappa = 0.1$ and without regularization. The same notations are used as in Figure 7-7. The input signals (not shown) in (b) and (c) are made 10% larger in amplitude but still within the intended operation range. The state trajectories are consistently accurate for various maximum step sizes and input amplitudes, but, in (b) and (c), it can be seen that at 0.4 ns, the output equation is sensitive to slight changes in states, indicating the overfitting problem in the output equations.



(a) Max step size = 100 ps, input as Figure 7-3b



(b) Max step size = 100 ps, input with 10% larger amplitude



(c) Max step size = 0.1 ps, input with 10% larger amplitude

Figure 7-10: (*Continued*) Overfitting alleviated by regularization. A single-stage diode RC model is trained with a medium $\kappa = 0.1$ and with regularization $\lambda = 10^{-7}$. After regularization, the output equation becomes smooth and accurate for various maximum step sizes and input amplitudes.

envelope-modulated sinusoids used in Figure 7-3b. It can be seen from Figure 7-7a and 7-7c, the states are relatively well-behaved for the maximum step sizes of 10 ps and the 0.1 ps, respectively. However, when the maximum step size is set to be 1 ps in Figure 7-7b, both the states and the outputs drastically deviate from the reference. Since the model is selectively sensitive to certain choices of maximum step sizes, one can conclude that the model suffers from the overfitting problem in its state equation.

Figure 7-8 shows the simulation of a model trained under the same conditions as in Figure 7-7 but with regularization $\lambda = 10^{-7}$. The experiment in Figure 7-8 shows that the state equation is not sensitive to step sizes after regularization, though the output equation is still a poor fit due to the small value of κ .

Figure 7-9 shows the simulation of a model trained with a medium $\kappa = 0.1$ without regularization. The input in Figure 7-9a is the same as in Figure 7-7, whereas the input in Figure 7-9b is enlarged by 10% in amplitude but still within the intended range. One can observe that in these two cases, the state trajectories are accurate and smooth, but the output signals appear oscillatory. In contrast to the cases in Figure 7-7, such oscillatory behaviors occur irrespective of certain step sizes. An evidence is shown in Figure 7-9c, where a small step size, 0.1 ps, is used. Although the states are almost identical in Figure 7-9b and 7-9c, a rapid oscillation occurs at 0.4 ns in port 2 in Figure 7-9c, indicating the overfitting problem in the output equation.

Figure 7-10 shows the exact counterpart of Figure 7-9 but with regularization $\lambda = 10^{-7}$. The oscillatory behaviors disappear and the output signals are accurate for various maximum step sizes and input amplitudes.

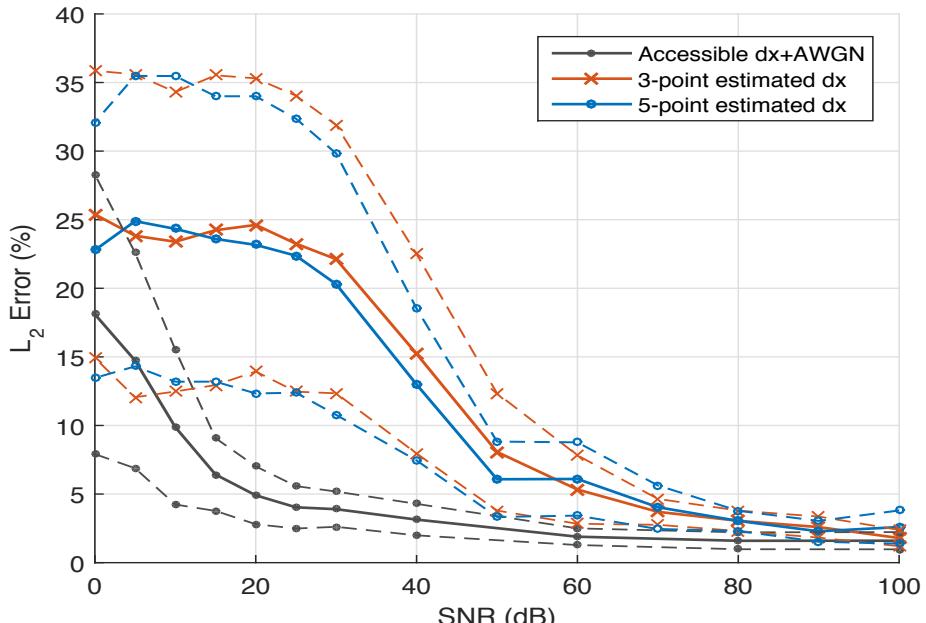
The case with a large κ generates models with a poorly fitted state equation and an overfitted output equation. Since a poorly fitted state equation only produces inaccurate states with errors accumulated over time, the resulting output is generally inaccurate. Accordingly, the experiment on the case with a large κ shows limited insights and, therefore, is omitted.

7.3 Modeling from Noisy Data

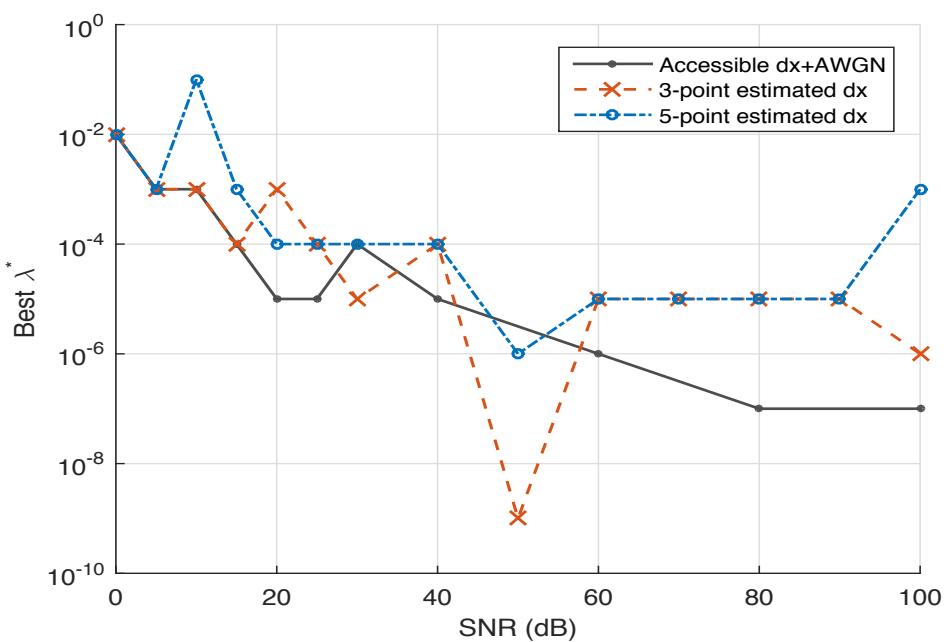
In this section, we test the ability of our proposed algorithm in handling noisy data. Noisy data are commonly encountered during measurement. In the following, we consider the case when the noise only occurs during measurement, and the noise is assumed to be Additive White Gaussian Noise (AWGN) on the state \tilde{x} and the output \tilde{y} . The input data \tilde{u} are assumed under the user's full control and, therefore, intact. Two scenarios are considered in this experiment: (1) state derivatives $\tilde{\dot{x}}$ are directly measurable and also corrupted by AWGN, and (2) state derivatives $\tilde{\dot{x}}$ are estimated from corrupted state data \tilde{x} .

1. Accessible state derivatives corrupted by AWGN: the k -th set of the training data (6.1) consists of data points $(\tilde{u}_i, \tilde{x}_i + n_x, \tilde{y}_i + n_y, \tilde{\dot{x}}_i + n_{\dot{x}})_k$, where $n_x, n_y, n_{\dot{x}}$ are vectors of independent normal random variables with zero means. Their variances depend on the signal levels at each port or each state such that the targeted Signal-to-Noise Ratio (SNR) is met individually.
2. Estimated time derivatives of states from noisy state data: the k -th set of the training data (6.1) consists of data points $(\tilde{u}_i, \tilde{x}_i + n_x, \tilde{y}_i + n_y, \hat{\dot{x}}_i)_k$, where n_x, n_y are vectors of independent normal random variables defined as in the previous case. The time derivatives of states $\hat{\dot{x}}_i$ are estimated from multiple neighboring states $\tilde{x}_j + n_x$, $|j - i| \leq \frac{N-1}{2}$, where $N \geq 3$ is an odd integer that represents the total number of data points used for estimation. When $N = 3$, the central difference scheme is adopted. When $N > 3$, we use the smooth noise-robust differentiators described in [29].

The training procedure of generating models for a single-stage diode RC line is the same as previous sections. The only difference here is that we replace the 12 sets of training data with the noise-corrupted version. For validation, we combine the 12 sets of the original noise-free training data and the four sets of the original noise-free validation data into a new validation data set consisting of 16 sets of signals. We set



(a) Model quality vs. SNR



(b) Optimal λ^* vs. SNR

Figure 7-11: Modeling from noise-corrupted data. The single-stage models of degrees (3, 3, 3) are generated from clean input signals, noisy states, noisy output signals, and (1) accessible time derivatives of states corrupted by noise, (2) estimated time derivatives of states from noisy state data with $N = 3$, and (3) estimated time derivatives of states from noisy state data using noise-robust differentiators with $N = 5$. All models are simulated alone.

a fixed $\kappa = 0.1$, sweep a range of λ , and determine the best λ^* for the model with the minimal mean of the \mathcal{L}_2 errors against the 16 validation sets. Figure 7-11a shows three sets of curves: the \mathcal{L}_2 errors of models generated from clean input signals, noisy states, noisy output signals, and (1) accessible time derivatives of states corrupted by noise, (2) estimated time derivatives of states using noisy state data with $N = 3$, and (3) estimated time derivatives of states using noisy state data using noise-robust differentiators [29] with $N = 5$. All models are of degree $(3, 3, 3)$ and simulated alone. Solid lines from the figure represent the mean of the \mathcal{L}_2 errors across the 16 validation sets, and the dashed curves represent the values less than one standard deviation away from the corresponding means. It can be seen from the figure that noise has a drastic impact on the case with estimated time derivatives of states. Estimating derivatives from more data points using the noise-robust differentiators improves the quality of models when the noise level is medium ($\text{SNR}=20\text{--}50\text{ dB}$), but introduces some errors when the noise level is very low ($\text{SNR}=100\text{ dB}$). Figure 7-11b shows the relation of the optimal λ^* to SNR. A rough tendency can be observed that the more noisy the data, the larger regularization is typically preferred.

7.4 Class E Power Amplifier

Class E power amplifier, shown in Figure 7-12, is a switch-type amplifier that attempts to separate the conduction angles of the drain current and voltage in order to achieve high efficiency [61]. Like Class C, D, and F, Class E power amplifier belongs to the category of “constant-envelope” amplifiers that only produce fixed-envelope carriers. We are interested in modeling the Class E power amplifier in Figure 7-12 using our proposed algorithm across a range of carrier frequencies. This amplifier is designed using the 45nm Silicon-On-Insulator (SOI) process technology. The generated model is intended to be operated with V_{IN} modulated as Quadrature Phase Shift Keying (QPSK) signals with amplitude 1 V, $V_{\text{DD}} = 1\text{ V}$, and $R_L = 50\Omega$.

In the training phase, we use QPSK signals with amplitude 1 V at frequencies 0.8, 1, and 1.2 GHz, total 2729 data points, and take $u = [v_{14}, v_{24}, v_{34}]' = [v_1, v_2, v_3]'$

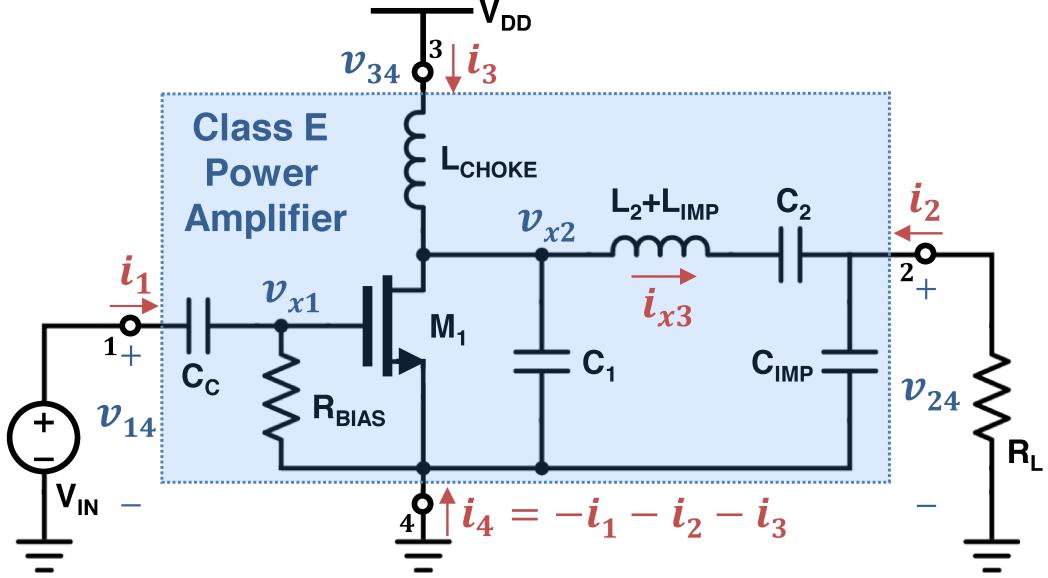


Figure 7-12: A Class E power amplifier. L_2, C_1, C_2 are designed for drain voltage and current shaping to achieve high efficiency. L_{IMP} and C_{IMP} are chosen for impedance matching.

as system inputs, $y = [i_1, i_2, i_3]'$ as outputs, and $x = [v_{x1}, v_{x2}, i_{x3}, v_{24}, i_3]'$ as states. The validation data are generated using the same setting except for the frequency at 0.9 and 1.1 GHz. In the test phase, the generated model is interconnected with $R_L = 50 \Omega$, $V_{\text{DD}} = 1 \text{ V}$, and V_{IN} driven by the same QPSK signals.

The optimization for each pair of (κ, λ) takes 163 s on average. The resulting model of $\deg(e, f, h) = (3, 3, 3)$, trained with $\kappa^* = 10^{-1}$ and $\lambda^* = 10^{-6}$, contains 1166 nonzero coefficients. This is not a large number even compared with a linearized model. As a comparison reference, if the transistor in Figure 7-12 is represented by the SPICE level 3 MOSFET model, the linearized small-signal representation of the original circuit consists of 20 states and total 529 coefficients. Although this is an underestimate of the complexity, it provides us a rough order of magnitude of how complex the original nonlinear circuit is.

Figure 7-13 shows a sample transient simulation of the voltage across the load R_L using our generated model in Simscape (shown as Model in the figure) compared with the simulation of the original schematic in Cadence Spectre (as Reference). When operated with short symbol durations (20 carrier periods) as in Figure 7-13, the model performs with an \mathcal{L}_2 error 8.51% and an \mathcal{L}_{∞} error 27.0%. When the

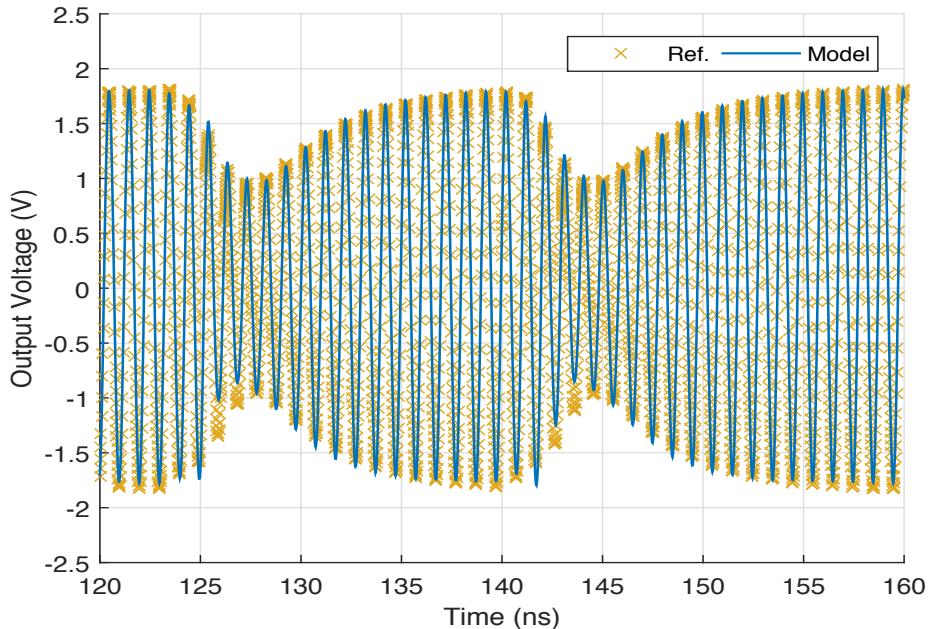


Figure 7-13: Transient output voltage signals at the load R_L with QPSK signaling.

Table 7.3: Performance metrics of the Class E power amplifier

Metric	Unit	Our Model	Reference	Diff.
Power Output	dBm	15.6	15.3	0.30
Power Gain	dB	12.9	13.5	-0.60
Power-Added Eff.	%	83.6	83.2	0.40
Overall Efficiency	%	84.3	83.9	0.47

model is fed with long symbol durations (over 100 carrier periods), the \mathcal{L}_2 error is reduced to 3.47 %. It can be concluded that most of the errors occur during symbol transitions. The constellation of the QPSK symbols is shown in Figure 7-14. Other performance metrics are summarized in Table 7.3. In this table, the power-added efficiency is conventionally defined as $(P_{\text{out}} - P_{\text{in}})/P_{\text{dc}}$, and the overall efficiency as $P_{\text{out}}/(P_{\text{dc}} + P_{\text{in}})$, where P_{out} is the power consumed by R_L , P_{in} is the power injected from V_{IN} , and P_{dc} is the power supplied from the voltage source V_{DD} .

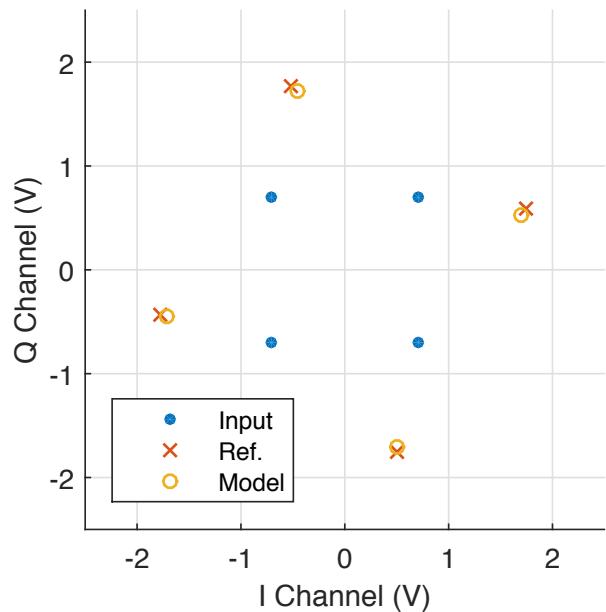


Figure 7-14: Constellation of the QPSK symbols. References are generated by simulating the actual circuit schematic with the input symbols depicted in blue dots. The maximum error in magnitude is 3.5% and the maximum error in angle is 1.7° .

Chapter 8

Case Studies: Arterial Networks

In this chapter, we attempt to use the proposed algorithm (3.16) to generate models for a set of human arteries. We demonstrate that the generated artery models can be further interconnected into a large-scale arterial network, and the simulation of the entire model is also stable and accurate. This capability enables the hierarchical modeling strategy. The training, validation, and test data in this chapter are generated using ANSYS Fluent [4], a Computational Fluid Dynamics (CFD) software package.

8.1 Scenario

Diagnosing medical conditions based on non-invasive (or minimally invasive) measurements requires simultaneous modeling for both (1) local pathological arteries and (2) global arterial networks in order to correlate the available measurements with the actual pathologies. For instance, diagnosing atherosclerosis or an aneurysm requires the detailed understanding of the pressure and flow inside the bifurcation segments. Such information is typically not measurable at pathological sites, but may still be attainable if it can be inferred from other measurements. Therefore, it is crucial to develop accurate yet efficient global arterial models such that the correlations between the pathologies and the available measurements can be established. The final diagnosis can be obtained by solving an inverse problem for the pathological parameters,

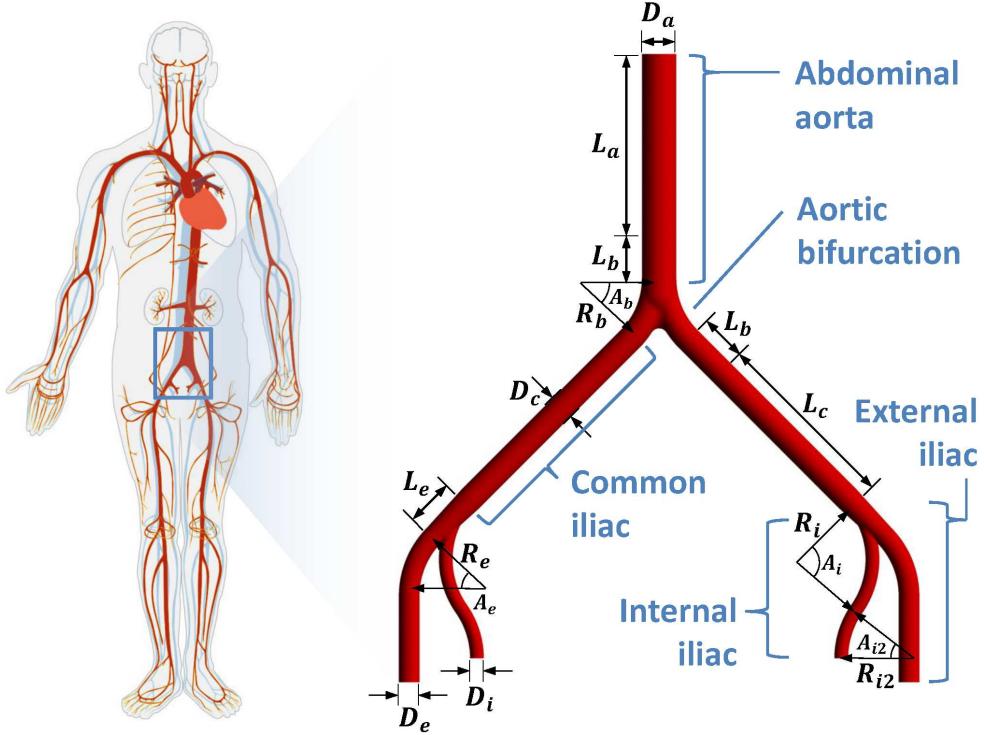


Figure 8-1: Geometry of the human abdominal aorta and iliac arteries.

for instance, aneurysm internal diameter, arterial wall thickness, plaque stiffness, etc.

For this strategy to be effective, the model for such a large-scale arterial network must be compact, computationally tractable, and field-solver-accurate. Our technique for modeling passive systems enables the hierarchical modeling capability: generating models for local sub-networks, such as branches and bifurcations, and interconnecting them to form a global network. This approach allows full exploitation of artery geometries without compromise due to the shape complexity. In addition, because the entire modeling efforts are subdivided into local model generations, the corresponding finite-element problems for generating training data are at a tractable size. Therefore, the full set of partial differential equations for fluid dynamics, such as viscosity and turbulence, can be fully employed to capture all types of nonlinearities without simplification.

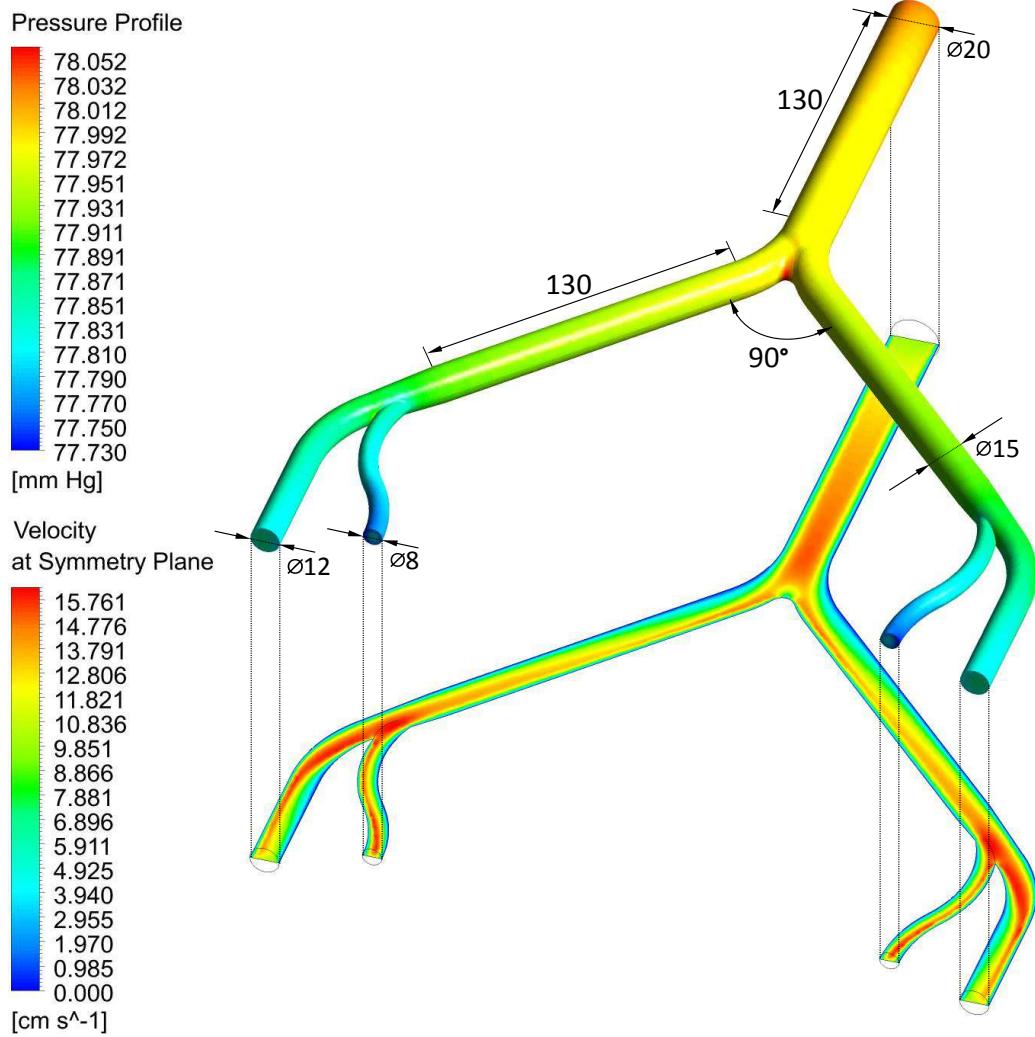


Figure 8-2: Pressure and velocity profiles solved by Fluent. The structure consists of 189k elements. The computation time for simulating the period of a single heart beat consisting of 200 data points is about 5 hours on a 4 GHz single-threaded machine.

8.2 Geometries and Parameters

We are interested in generating models for the subset of the arterial network shown in Figure 8-1. This set of arteries includes the abdominal aorta, common iliac, external and internal iliac arteries. The corresponding pressure and velocity profiles solved by Fluent are shown in Figure 8-2. We plan to generate artery models component by component using the decomposition suggested in Figure 8-3. Since it is known that the straight artery can be properly modeled as a linear system [53, 59], in the

Table 8.1: Parameters of arterial networks in Figure 8-1

Part	Parameter	Symbol	Value	Unit
Abdominal aorta	Length	L_a	100	mm
	Diameter	D_a	20	mm
Aortic bifurcation	Branch length	L_b	30	mm
	Bending radius	R_b	50	mm
	Bending angle	A_b	45	deg.
	Length	L_c	100	mm
Common iliac	Diameter	D_c	15	mm
	Diameter transition	L_e	30	mm
External iliac	Diameter	D_e	12	mm
	Bending radius	R_e	50	mm
	Bending angle	A_e	45	deg.
	Diameter	D_i	8	mm
Internal iliac	Bending radius	R_i	50	mm
	Bending angle	A_i	80	deg.
	Bending radius 2	R_{i2}	50	mm
	Bending angle 2	A_{i2}	35	deg.
Blood	Density		1060	kg/m ³
	Viscosity		0.0027	kg/m-s

following sections, we will focus on generating models for (1) the aortic bifurcation, and (2) the bent bifurcation of the common iliac artery. The geometry and fluid parameters are listed in Table 8.1. We artificially choose the artery dimensions based on the data in [31].

8.3 Assumptions for Fluid Dynamics Simulation

In the computational fluid dynamics (CFD) simulation, we assume that blood is a incompressible Newtonian fluid and the vessel walls are rigid and non-elastic. The incompressible fluid assumption holds when the vessel diameter is more than two orders of magnitude greater than the size of a red cell, which is about 6–8 μm in diameter [18]. The Newtonian assumption is generally accepted when the diameter of an artery is greater than or equal to 1 mm [13, 76]. The rigid wall assumption, on the other hand, is a simplification and typically used only in preliminary studies.

We further assume that the flows and pressures at the junction between the 3D geometries and lumped elements are constant valued across the junction as cross-sectional averages. This constant value assumption is a rough approximation for velocity profiles, as it can be seen from Figure 8-2. In fact, arteries other than the aorta are typically not long enough to develop a specific wavefront before encountering other artery branches [36]. More discussions on junction coupling can be found in [53]. In addition, the transient simulation of pressures and flows in Fluent is performed at uniform time grids.

It is worthy noting that the assumptions for CFD simulation listed above are only for the validity of the *generated data*. The capability of our proposed algorithm, the modeling procedures, and the hierarchical modeling strategy described in the following sections are irrespective of the specific choice of CFD solvers and any assumptions above.

8.4 Modeling of Aortic Bifurcation

In this section, we describe the procedure to train a model for the aortic bifurcation with $L_b = 30$ mm in each branch, as depicted in Figure 8-4. The outlets at both common iliac arteries are connected with a resistive load R_L in series with a constant pressure $P_0 = 75$ mmHg. The load R_L relates the flow q with the pressure p across the terminals by

$$p = \frac{k\rho}{2\pi^2 r^4} |q| q, \quad (8.1)$$

where r is the radius of the vessel, ρ is the density of the fluid, and k is a dimensionless coefficient. The quadratic dependency in the flow quantity q models the inertial loss of capillaries or porous media [21, 46]. The inlet from the abdominal aorta is fed with a transient flow $q(t)$. This configuration is set up in Fluent in order to generate the training, validation, and test data. In the test phase, we use the identical configuration in MATLAB to compare the performance of the generated model against the test data.

For the inlet flow $q(t)$, we use the real patient data from MICCAI 2013 CFD Challenge [19]. The subject was a 71 kg, 177 cm tall, 17-year old male with a mild

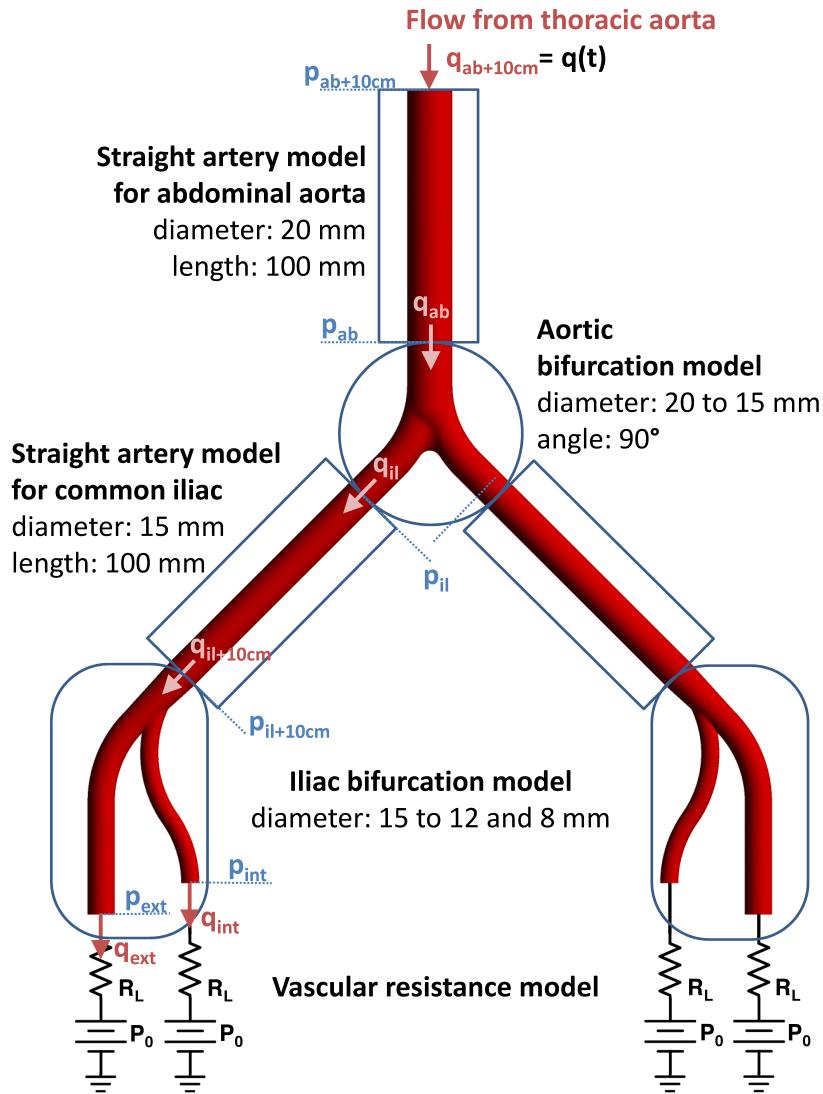


Figure 8-3: Decomposition into local models and the boundary setup for the simulation in Figure 8-2.

thoracic aortic coarctation. Two sets of flows are recorded: the patient at rest, and the patient under stress. The stress condition is induced by isoprenaline. When the patient was at rest, the heart rate was 47 beats per minute (cardiac cycle $T = 1.277$ sec) and the cardiac output was 3.71 L/min. When the patient was under stress, the heart rate was increased to 141 beats per minute (cardiac cycle $T = 0.425$ sec) and the corresponding cardiac output was 13.53 L/min. The flow profiles for these two conditions at the descending aorta are depicted in Figure 8-5.

To increase the amount of training data, we artificially scale down the amplitudes

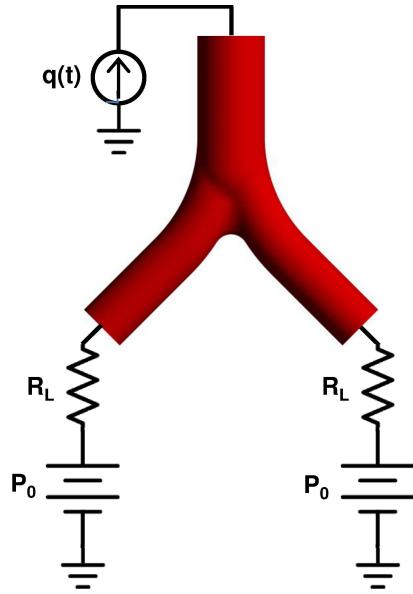


Figure 8-4: Configuration for the aortic bifurcation modeling.

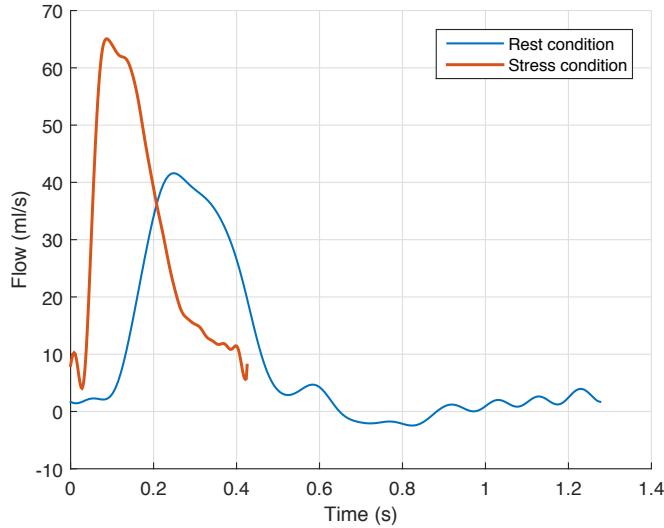


Figure 8-5: Inlet flows for a single heartbeat when the patient is at rest and under stress [19].

of both flow amplitudes by 20% as two additional input signals. The resulting training inlet flows include (1) the rest condition, (2) stress condition, (3) 80% amplitude of the rest condition, and (4) 80% amplitude of the stress condition. The corresponding pressures and flows at inlets and two common iliac outlets are computed using Fluent. In this case, we take pressures as model input u , flows as both model state x and output y . Each set of training data consists of a single heartbeat simulated as 200

Table 8.2: Performance of aortic bifurcation model

Error (%) [*]		
Rest condition	\mathcal{L}_2	\mathcal{L}_∞
Abdominal pressure	1.87	1.88
Common iliac pressure	0.70	0.62
Common iliac flow	1.90	1.02
Stress condition	\mathcal{L}_2	\mathcal{L}_∞
Abdominal pressure	3.26	4.72
Common iliac pressure	0.35	0.31
Common iliac flow	0.69	0.71

* See (6.2) for the definition.

time points, total 800 points for four set of signals.

For the validation set, we artificially generate two flow signals: (1) the rest condition with a heart rate shifted from 47 to 60 beats per minute, and (2) the stress condition with a heart rate shifted from 141 to 120 beats per minute. The corresponding pressures and flows are simulated in the same way as in generating the training data. The resulting validation set consists of two sets of data, each of which consists 200 time points, 400 points in total.

We treat the aortic bifurcation in Figure 8-4 as a three-port system. This is because, unlike three-terminal circuit elements, arteries with elastic walls may deform along with the beat cycle and temporarily accommodate more, or less, fluid in the structure. Therefore, the amount of blood at the inlets and outlets are not conserved instantaneously. Although the elasticity of artery walls is not considered in this test example, we still treat the bifurcation as a three-port system for the compatibility of the future development.

We train a model of degree $(3, 3, 1)$ using the aforementioned training data. The validation set determines the best κ and λ as $\kappa^* = 10^{-2}$ and $\lambda^* = 10^{-6}$. The resulting generated model consists of 259 nonzero coefficients. When simulated alone, the generated model exhibits an \mathcal{L}_2 error 6.29% and an \mathcal{L}_∞ error 4.85% against the validation set. In the test phase, we interconnect the generated model in the configuration identical to Figure 8-4 and feed the model with $q(t)$ equal to the flow

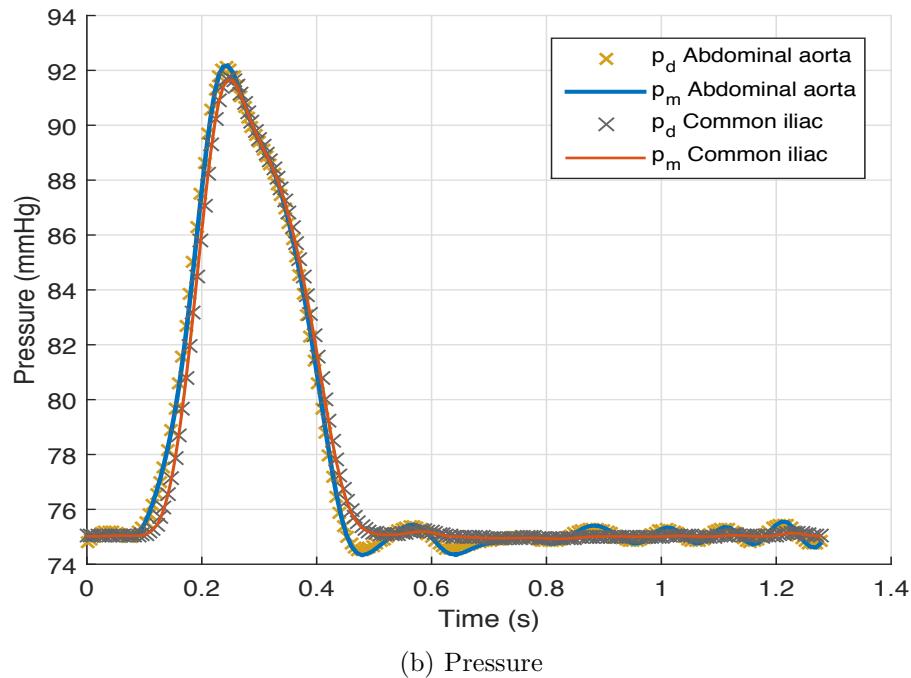
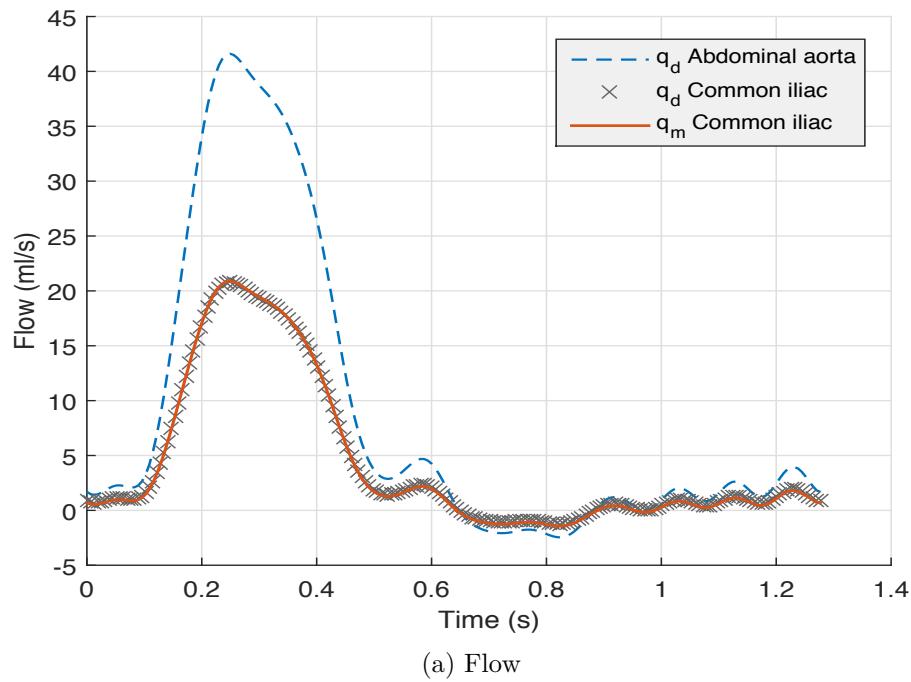


Figure 8-6: Test results of the generated aortic bifurcation model configured as in Figure 8-4 with $q(t)$ from the *rest* condition. The subscript d denotes the reference data and m denotes the simulation results from our generated model.

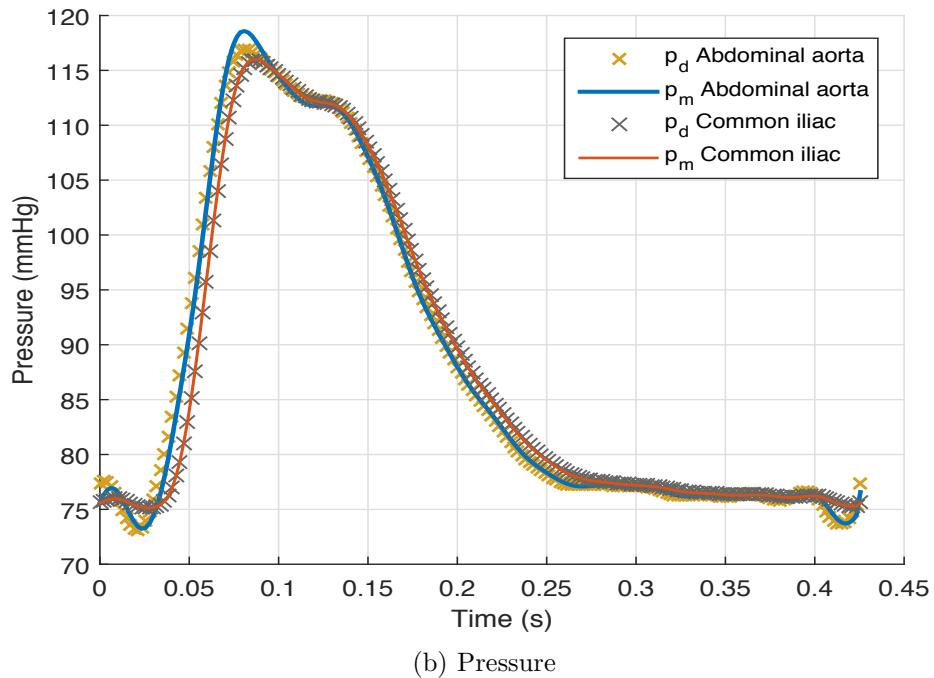
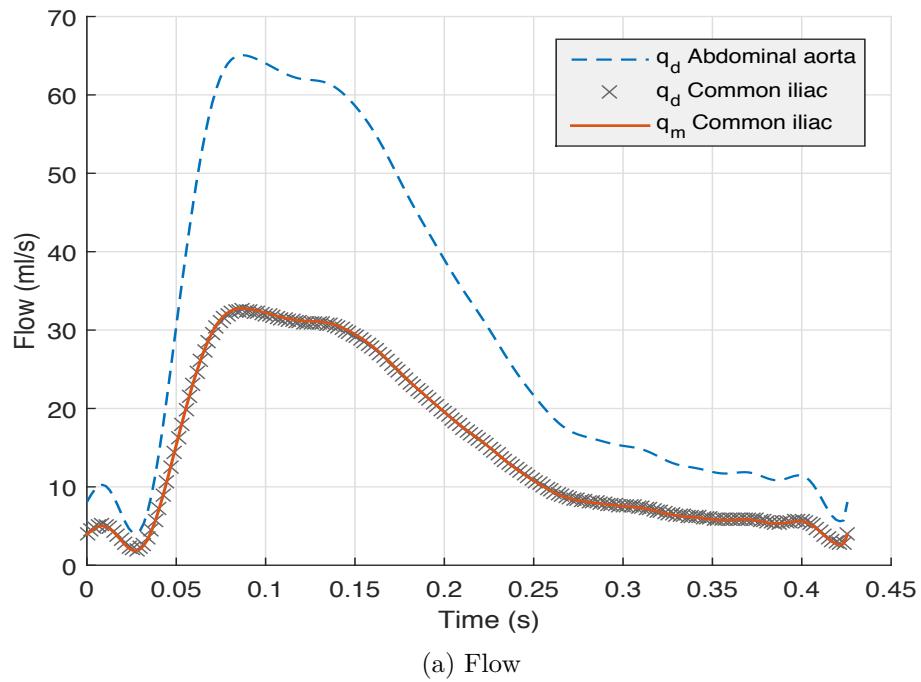


Figure 8-7: Test results of the generated aortic bifurcation model configured as in Figure 8-4 with $q(t)$ from the *stress* condition. The subscript d denotes the reference data and m denotes the simulation results from our generated model.

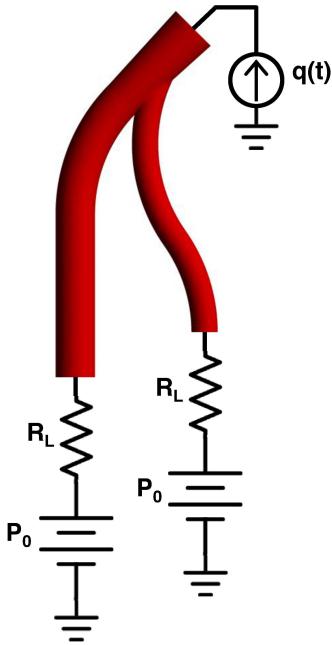


Figure 8-8: Configuration for the iliac bifurcation modeling.

when the patient is (1) at rest and (2) under stress. The flows and pressures compared with the reference test data are plotted in Figure 8-6 and 8-7. Only one side of the two common iliac arteries is shown because the geometry is symmetrical, and so are the pressure and the flow. The corresponding errors are summarized in Table 8.2. In comparison, if the model is assumed linear with degree (1, 1, 1), the \mathcal{L}_2 and \mathcal{L}_∞ errors are 23.2% and 33.0%, respectively.

8.5 Modeling for Iliac Bifurcation

The procedure for modeling the iliac bifurcation is similar to the previous section. Figure 8-8 shows the boundary condition setup for generating the training, validation, and test data using Fluent. The parameters for P_0 , R_L are identical to the aortic bifurcation case. The inlet flow $q(t)$ is adjusted by a factor 83% to accommodate the fact that the iliac arteries are thinner compared with the aorta and also a higher velocity is expected. These two effects can be observed from the complete CFD simulation in Figure 8-2. The factor 83% is a rough estimate, using 1.5 faster velocity times the ratio of the cross-sectional areas. The intended range of training signals

Table 8.3: Performance of iliac bifurcation model

	Error (%) [*]	
Rest condition	\mathcal{L}_2	\mathcal{L}_∞
Inlet pressure	1.68	2.13
External iliac pressure	1.91	1.83
External iliac flow	0.88	0.77
Internal iliac pressure	1.86	1.79
Internal iliac flow	1.62	1.55
Stress condition	\mathcal{L}_2	\mathcal{L}_∞
Inlet pressure	3.15	4.70
External iliac pressure	1.90	1.90
External iliac flow	0.26	0.52
Internal iliac pressure	1.85	1.89
Internal iliac flow	0.82	1.72

* See (6.2) for the definition.

has to be adjusted because the end goal is to interconnect this model to the rest of arteries in Figure 8-3.

The resulting training and validation data are generated in the same way as for the aortic bifurcation except that the inlet flow $q(t)$ is scaled. The training data consist of four sets of data, each of which contains 200 points, total 800 data points. The validation data consist of two sets of data, 200 points for each, and total 400 data points.

We train a model of degree $(3, 3, 1)$ and use the validation set to determine κ and λ . The best values are $\kappa^* = 10^{-2}$ and $\lambda^* = 10^{-7}$. The resulting generated model consists of 308 nonzero coefficients. When simulated alone, the generated model exhibits an \mathcal{L}_2 error 6.29% and an \mathcal{L}_∞ error 4.85% against the validation set. In the test phase, we interconnect the generated model with R_L and P_0 as in Figure 8-8 and use the inlet flow $q(t)$ from the rest and the stress condition with amplitudes scaled by 83%. The flows and pressures compared with the reference test data for the rest condition are plotted in Figure 8-9 and 8-10. Due to the small size of the structure, the pressure responses for each port are largely overlapping, and, therefore, plotted individually into three subfigures. The corresponding errors at each port are

summarized in Table 8.3. In comparison, if the model is assumed linear with degree $(1, 1, 1)$, the \mathcal{L}_2 and \mathcal{L}_∞ errors are 43.8% and 40.5%, respectively.

8.6 Modeling via Interconnecting Local Models

In this section, we interconnect the aortic bifurcation model, the iliac bifurcation model, and the straight artery models to form the entire model for the arterial network in Figure 8-1. Both of the bifurcation models are generated in previous sections. The interconnection of the models follow the decomposition suggested in Figure 8-3. The terminating resistance R_L and the constant pressure P_0 are set up in the same way as the aortic bifurcation modeling in Section 8.4.

The straight vessels are treated as 1-port LTI state-space systems. Since straight vessels are nearly lossless with almost zero feedthrough term, i.e., $d \approx 0$ in the (a, b, c, d) first-order LTI state-space model, the analysis in Theorem 4.3 has shown that the passivity guarantee (3.6) does not include such a case in the feasible region, as depicted in Figure 4-1. Alternatively, for these simple scalar cases, we use a least-square objective function (3.13) with the constraints $a \leq 0, d \geq 0$ to identify the passive models. The resulting models for both 10-cm vessels in 20-mm and 15-mm diameters exhibit less than 1% errors in both the \mathcal{L}_2 and the \mathcal{L}_∞ senses.

The flows from thoracic aorta are similarly prepared in two cases: the patient (1) at rest and (2) under stress, as depicted in Figure 8-5. We use Fluent to simulate the flows and pressures of the geometry in Figure 8-3 for one heartbeat with the boundary conditions set up accordingly. The simulated terminal pressures and flows are stored as the reference data. Separately, we feed the same flow as used for Fluent to our entire model of the arterial network and collect the pressures and flows at the terminals. Their comparisons are shown in Figure 8-11–8-14 for both the rest and the stress cases. We summarize the performance in Table 8.4.

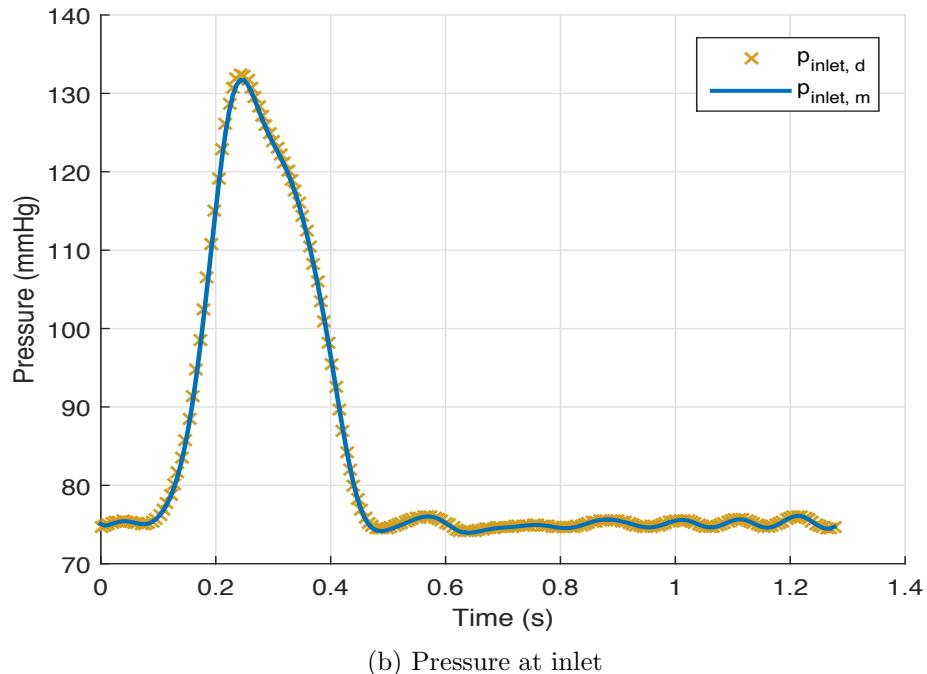
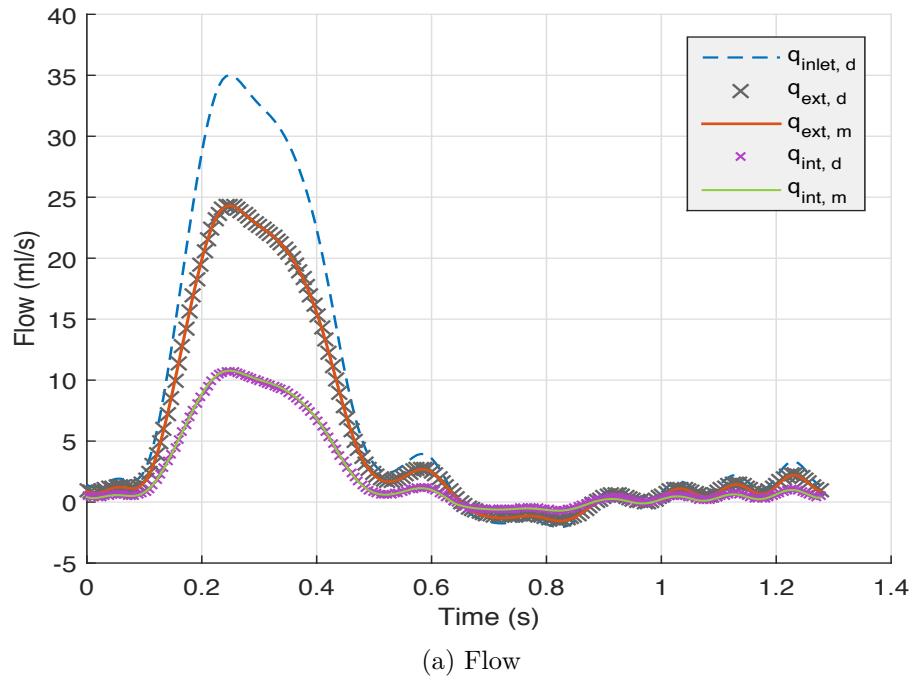
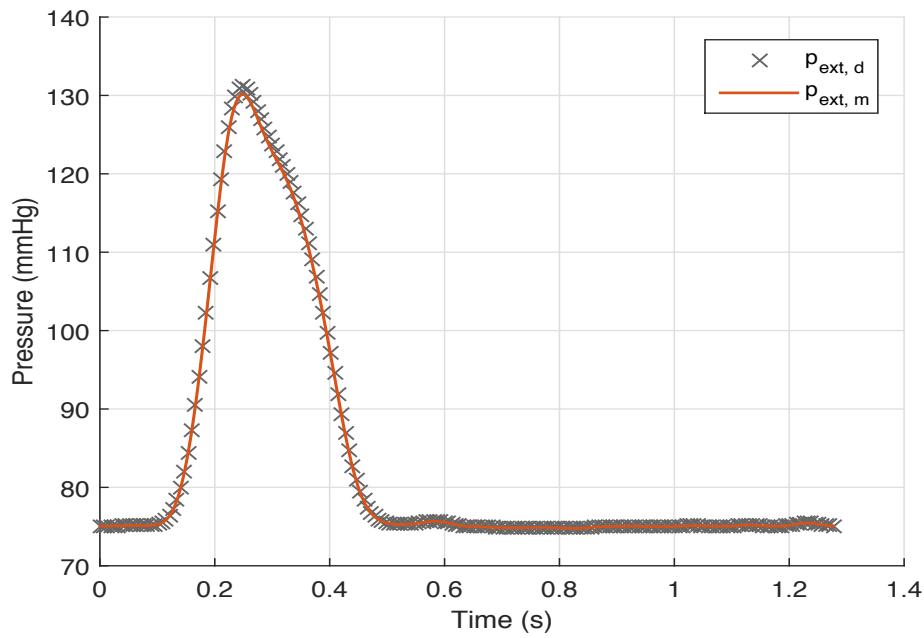
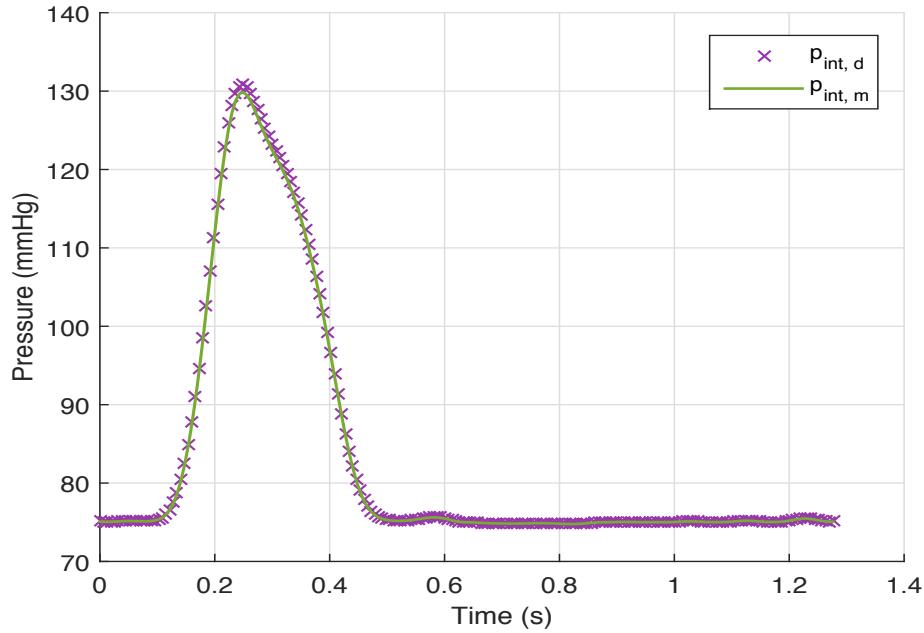


Figure 8-9: Test results of the generated iliac bifurcation model configured as in Figure 8-8 with $q(t)$ from the rest condition. The subscript d denotes the reference data, and m denotes the simulation results from our generated model. The subscript ext is for the external iliac artery, and int is for the internal iliac artery.



(a) Pressure at external iliac artery



(b) Pressure at internal iliac artery

Figure 8-10: (*Continued*) Test results of the generated iliac bifurcation model configured as in Figure 8-8 with $q(t)$ from the rest condition.

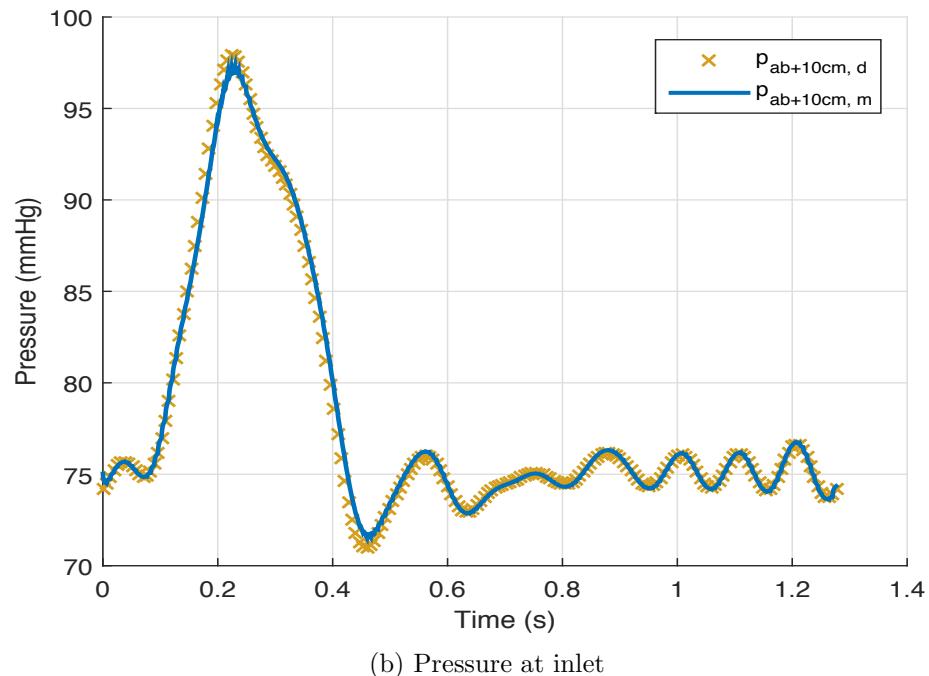
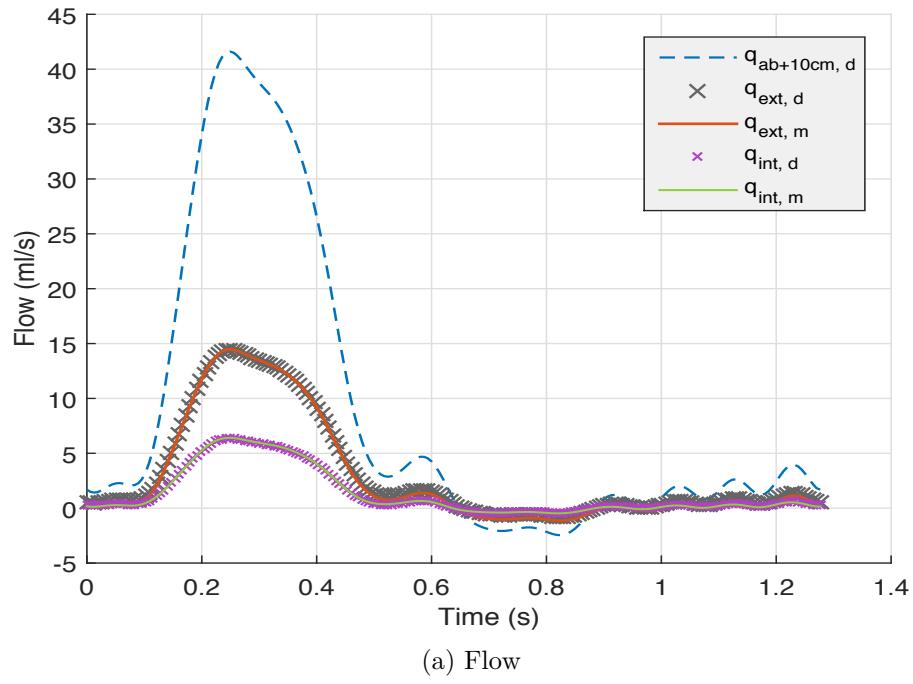
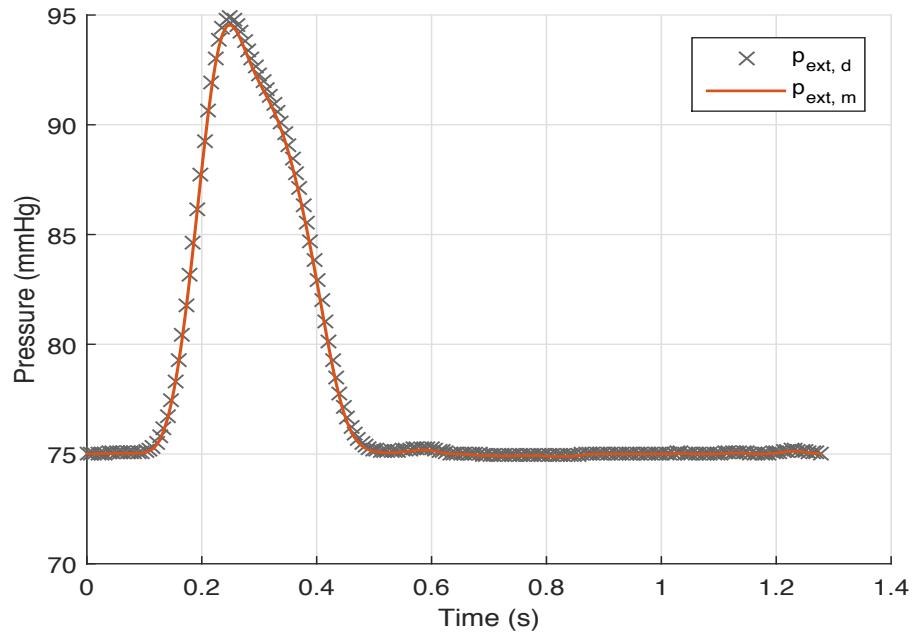
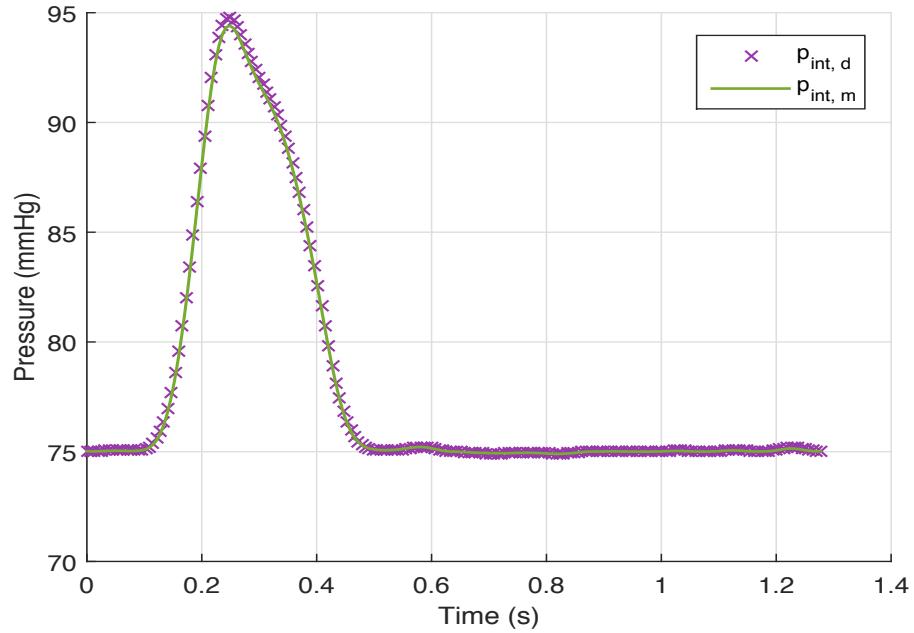


Figure 8-11: Test results of the entire model for arterial networks configured as in Figure 8-3 with $q(t)$ from the *rest* condition. The subscript d stands for the reference data, and m for the simulation results from our generated model. Other subscripts indicate the port locations depicted in Figure 8-3.



(a) Pressure at external iliac artery



(b) Pressure at internal iliac artery

Figure 8-12: (*Continued*) Test results of the entire model for arterial networks configured as in Figure 8-3 with $q(t)$ from the *rest* condition.

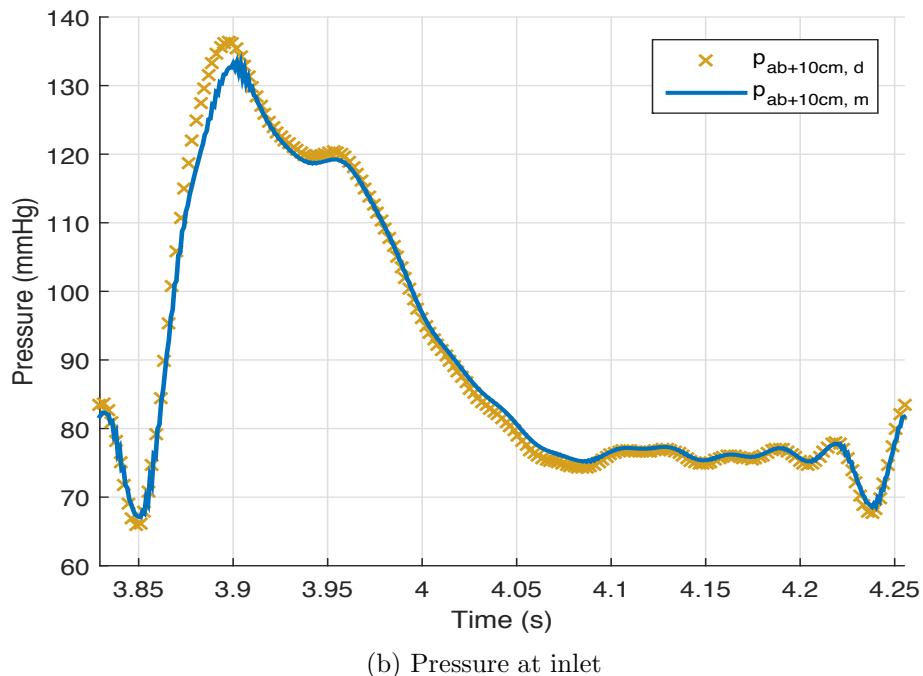
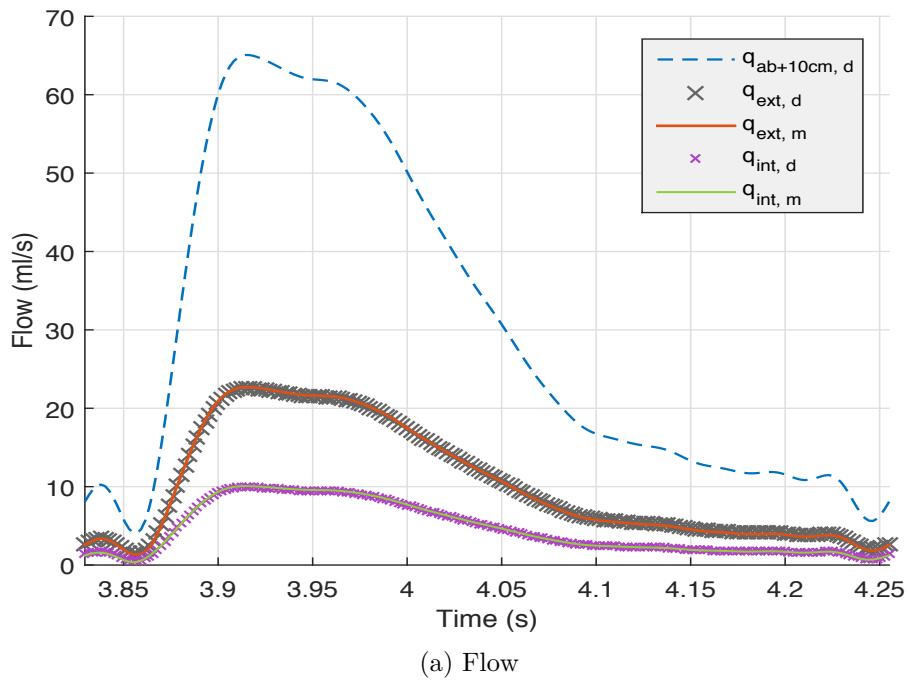
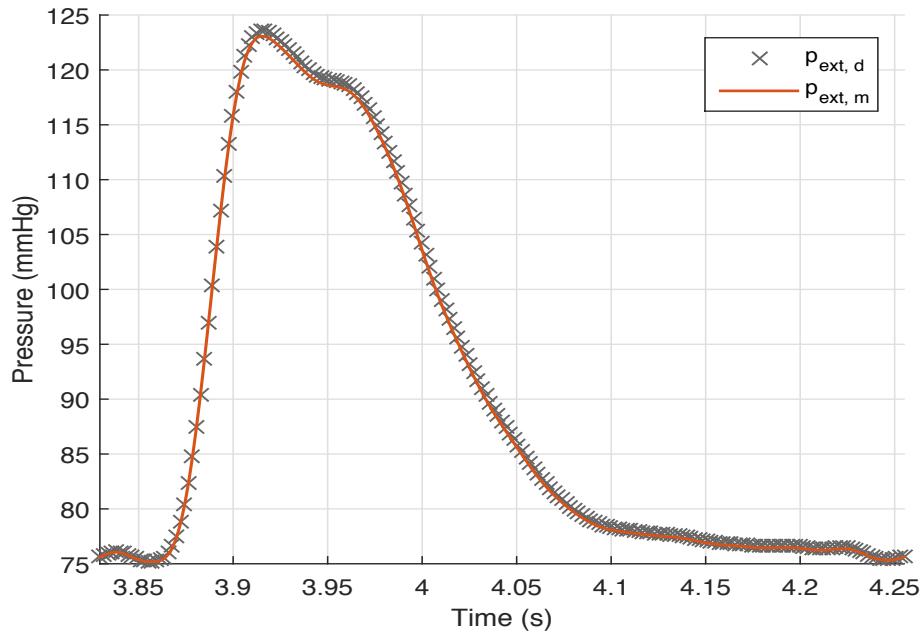
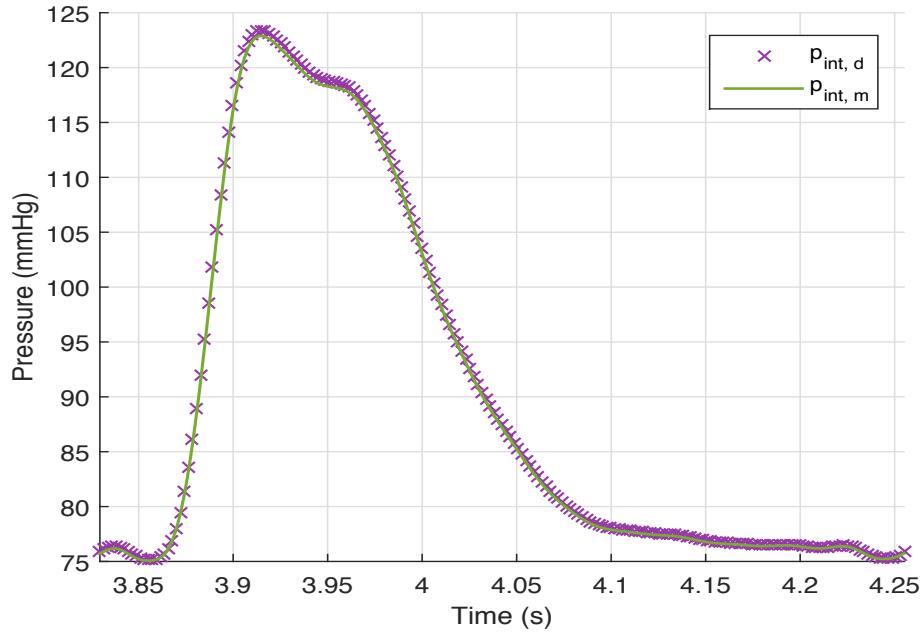


Figure 8-13: Test results of the entire model for arterial networks configured as in Figure 8-3 with $q(t)$ from the *stress* condition. The subscript d stands for the reference data, and m for the simulation results from our generated model. Other subscripts indicate the port locations depicted in Figure 8-3.



(a) Pressure at external iliac artery



(b) Pressure at internal iliac artery

Figure 8-14: (*Continued*) Test results of the entire model for arterial networks configured as in Figure 8-3 with $q(t)$ from the *stress* condition.

Table 8.4: Performance of entire arterial network model

Rest condition	Error (%) [*]	
	\mathcal{L}_2	\mathcal{L}_{∞}
Abdominal aorta pressure $p_{ab+10cm}$	5.75	5.79
External iliac pressure p_{ext}	1.36	1.41
External iliac flow q_{ext}	4.44	2.84
Internal iliac pressure p_{int}	1.27	1.29
Internal iliac flow q_{int}	2.50	2.34
Stress condition	\mathcal{L}_2	\mathcal{L}_{∞}
Abdominal aorta pressure $p_{ab+10cm}$	6.89	12.68
External iliac pressure p_{ext}	1.47	1.18
External iliac flow q_{ext}	1.05	1.26
Internal iliac pressure p_{int}	1.44	1.68
Internal iliac flow q_{int}	1.34	2.38

* See (6.2) for the definition.

Chapter 9

Conclusion

This thesis provided an automated modeling technique for generating nonlinear and passive dynamical models from input-state-output data. The proposed technique was based on the strategy of searching for model parameters within a convex set described by our passivity guarantee. The robustness of the generated models was further enhanced with additional incremental stability guarantee and the regularization in the objective function. Two types of applications were provided in this thesis: circuit networks and arterial networks, showing that the proposed technique is applicable to multiple physical domains. In the following, I would like to discuss the extensions of the proposed techniques and their further development of applications as the future research topics.

9.1 Extensions of Techniques

9.1.1 Repeated Experiments and Noisy Data

It is inevitable to encounter noise-corrupted data when it comes to measurements. Some preliminary experiments on generating models from noisy data can be found in Section 7.3. Yet, it is quite common that multiple experiments are performed with an identical experimental setting in order to record data repetitively. The average value across experiments can be used, but the variance of measurements may also

contain important information. Such data correlation should be treated specially to further improve the quality of the models. This notion has been studied in [66] for the discrete-time systems with a provable improvement using their proposed strategy. For continuous-time systems, a similar method can possibly be developed. An extra challenge is immediately expected: the sampling times across experiments in the continuous-time system may not be perfectly aligned as in the discrete-time case. This direction of research directly enhances the usability of the proposed technique.

9.1.2 Convex Constraint Related Topics

Lossless or Nearly Lossless Systems

As indicated by the analysis in Section 4.1, the current passivity guarantee requires the output equation $h(x, u)$ to explicitly depend on input u . The analysis on LTI systems has specifically shown that the feedthrough term of the state-space model must be nonzero. In other words, the current technique is limited to generating nonlinear systems with a sufficient loss. The difficulty arises because knowing whether a system is lossy enough *a priori* is not a straightforward task. Therefore, it is always beneficial to enlarge the feasible set of the passivity guarantee (3.6) and make it as close as to the set of the original passivity inequality (2.5).

Interpolated Models

The proposed technique in this thesis focuses on identifying *convex* models via convex optimization. The convexity property ensures that the convex combination of convex models that satisfy certain convex constraints is also a convex model that satisfies the same set of constraints. To be more specific, if two models, described by the parameters θ_0 and θ_1 , respectively, are passive due to the convex constraint of the passivity guarantee (3.6), then the model described by the parameters $\theta = \alpha\theta_1 + (1 - \alpha)\theta_0$, for $\alpha \in [0, 1]$, is also passive due to (3.6). This observation enables the possibility of generating new passive models without going through the training procedure. An immediate usage of this strategy is to provide geometrically parameterized models. For

instance, an aortic bifurcation model with a 40-degree bifurcation can be generated algebraically through the convex combination of a model with a 35-degree bifurcation and another model with a 45-degree bifurcation, without extra simulations, measurements, or optimizations. Some additional mechanisms may need to be introduced to ensure smooth transitions in model parameters.

9.1.3 Model Quality Improvement

Data Reweighting

Data points are not equally important. This can be seen from Figure 8-13b, where the model, generated from the data at uniform time grids, does not respond fast enough to capture the steep slope. A possible fix is to reweigh the data points in the objective function (3.13) such that the important features, like slopes and extreme values, can be faithfully maintained in the generated models.

Model Stiffness

Higher-degree polynomials tend to be sensitive to large input arguments, and sometimes the generated models are too sensitive to be practically usable. In addition, since $E(x)$ and $f(x, u)$ are determined independently, it is possible that there exist factors that can be nearly cancelled out, resulting in an oscillatory neighborhood. This spurious pole-zero pair problem has been studied in the Padé approximation [69]. Although the regularization in (3.14) empirically helps alleviate the stiffness problem, it may not have an effect on the spurious pair problem. It is desirable to establish certain mechanisms to regularize the behaviors of higher-degree polynomials.

9.1.4 Concepts from Linear Systems

Broader Concepts of Passivity

This thesis utilizes the passivity condition based on the choice of supply rate $\sigma(u, y) = u'y$. Other choices are possible, for instance, $\sigma(u, y) = \|u\|^2 - \|y\|^2$, which is referred

to as the bounded \mathcal{L}_2 gain, or the passivity in the scattering sense. This concept can be seen as the extension of the scattering parameters used in the analysis of microwave or optical systems. The immediate benefit of using the bounded \mathcal{L}_2 gain is that the $h(x, u)$ does not have to depend on u in order to have a feasible dissipativity constraint (3.3). Therefore, the dissipativity constraint (3.3) with the bounded \mathcal{L}_2 gain may be less restrictive than the passivity constraint (3.6). However, it is not clear whether the scattering wave transformation, which transforms voltages and currents into incident and reflective waves in the frequency domain, is physically meaningful in the time domain. It is also not clear whether a state-space model with scattering inputs and outputs can practically represent the same set of systems as the passivity with $\sigma(u, y) = u'y$. Further studies along this direction is of both practical and theoretical interests.

Classes of Nonlinear Loads

In linear systems, the concept of impedance is well defined and, at a fixed single frequency, a single complex number is sufficient to describe the loading condition. For nonlinear systems, the definition of impedance is not clear unless the system is sufficiently narrow-banded. This poses a problem for nonlinear modeling because without proper descriptions of loading conditions, it is difficult to thoroughly explore the nonlinearity of a system under intended interconnections. One possible solution is to narrow down the generality and define an application-specific impedance, such as X-parameters for power amplifiers [56]. This topic deserves further studies.

9.1.5 Dissipativity in Discrete Time

Many system identification techniques have been developed in the discrete-time domain. Their rapid advances even overshadow the development of the continuous-time methods. In fact, the continuous-time techniques occurred earlier in the last century [20]. To gain access to the established discrete-time techniques, one option is to transfer those techniques to the continuous-time domain. The alternative is to

move the passive model generation developed in this work and also the simulation platform to the discrete-time domain. The discrete-time version of the dissipative system theory has been analogously proposed [26]. Its relationship to the continuous-time passive systems was studied under the zero-order hold conversion [49]. However, whether this result can be extended to a more general setting and how this relationship can be applied to the system identification scenario are still unanswered questions.

9.2 Further Development of Applications

One of the major incentives of this research is the application in cardiovascular systems. Chapter 8 demonstrated that our hierarchical modeling strategy has the potential to model large-scale, even full-body, arterial networks with good accuracy. This capability is potentially useful for the computational fluid dynamics (CFD) community and also for cardiologists. For the CFD community, one of the major challenges in cardiovascular simulation is to set up the proper boundary conditions that reflect the actual pressures and flows in circulation. A viable solution is to couple the CFD problem of local arteries with our compact models for the global arterial network. On the other hand, cardiologists concern the problem of correlating the non-invasive measurable information to the pathological conditions. Our technique is potentially useful in constructing a patient-specific circulation model as a simulation platform to study such correlations. An inverse problem can then be formulated to infer the pathological parameters, as briefly described in Section 8.1.

Bibliography

- [1] E. D. Andersen and K. D. Andersen. The mosek interior point optimizer for linear programming: An implementation of the homogeneous algorithm. In H. Frenk, K. Roos, T. Terlaky, and S. Zhang, editors, *High Performance Optimization*, volume 33 of *Applied Optimization*, pages 197–232. Springer US, 2000.
- [2] B. D. O. Anderson and S. Vongpanitlerd. *Network analysis and synthesis: a modern systems theory approach*. Prentice-Hall, Englewood Cliffs, NJ, 1973.
- [3] D. Angeli. A Lyapunov approach to incremental stability properties. *Automatic Control, IEEE Transactions on*, 47(3):410–421, Mar 2002.
- [4] ANSYS. *ANSYS Academic Research, Release 16.0*. Canonsburg, PA.
- [5] K. J. Astrom and P. Eykhoff. System identification—a survey. *Automatica*, 7(2):123–162, 1971.
- [6] S. A. Billings. Identification of nonlinear systems—a survey. *Control Theory and Applications, IEE Proceedings D*, 127(6):272–285, Nov 1980.
- [7] G. Blekherman, P. A. Parrilo, and R. R. Thomas. *Semidefinite Optimization and Convex Algebraic Geometry*. MOS-SIAM series on optimization. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2013.
- [8] B. N. Bond, Z. Mahmood, Y. Li, R. Sredojevic, A. Megretski, V. Stojanovi, Y. Avniel, and L. Daniel. Compact modeling of nonlinear analog circuits using system identification via semidefinite programming and incremental stability certification. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 29(8):1149–1162, Aug 2010.
- [9] F. H. Branin and K. Huseyin. *Problem analysis in science and engineering*. New York : Academic Press, 1977.
- [10] B. Brogliato. *Dissipative Systems Analysis and Control: Theory and Applications*. Communications and Control Engineering. Springer, 2007.
- [11] C. I. Byrnes and A. Isidori. A frequency domain philosophy for nonlinear systems, with applications to stabilization and to adaptive control. In *Decision and Control, 1984. The 23rd IEEE Conference on*, pages 1569–1573, Dec 1984.

- [12] Y. Chen and J. White. A quadratic method for nonlinear model order reduction. In *Modeling and Simulation of Microsystems, 2000 International Conference on*, pages 477–480, 2000.
- [13] Y. I. Cho and K. R. Kensey. Effects of the non-newtonian viscosity of blood on flows in a diseased arterial vessel. part 1: Steady flows. *Biorheology*, (28):241–62, 1991.
- [14] L. O. Chua and C.-Y. Ng. Frequency domain analysis of nonlinear systems: general theory. *Electronic Circuits and Systems, IEE Journal on*, 3(4):165–185, July 1979.
- [15] C. P. Coelho, J. R. Phillips, and L. M. Silveira. A convex programming approach to positive real rational approximation. In *Computer Aided Design, 2001. IC-CAD 2001. IEEE/ACM International Conference on*, pages 245–251, 2001.
- [16] J. J. van Dixhoorn and F. J. Evans. *Physical structure in systems theory: network approaches to engineering and economics*. Academic Press, 1974.
- [17] A. Dounavis, R. Achar, and M. S. Nakhla. Efficient passive circuit models for distributed networks with frequency-dependent parameters. *Advanced Packaging, IEEE Transactions on*, 23(3):382–392, Aug 2000.
- [18] W. Dzwinel, K. Boryczko, and D. A. Yuen. A discrete-particle model of blood dynamics in capillary vessels. *Journal of Colloid and Interface Science*, 258(1):163–173, 2003.
- [19] C. A. Figueiredo, T. Mansi, P. Sharma, and N. Wilson. 2nd CFD challenge: Predicting patient-specific hemodynamics at rest and stress through an aortic coarctation, 2013.
- [20] H. Garnier and L. Wang. *Identification of Continuous-time Models from Sampled Data*. Advances in Industrial Control. Springer, 2008.
- [21] J. Geertsma. Estimating the coefficient of inertial resistance in fluid flow through porous media. *Society of Petroleum Engineers Journal*, 14(5), Oct 1974.
- [22] I. Goethals, T. Van Gestel, J. Suykens, P. Van Dooren, and B. De Moor. Identification of positive real models in subspace identification by using regularization. *Automatic Control, IEEE Transactions on*, 48(10):1843–1847, Oct 2003.
- [23] S. Grivet-Talocia. Passivity enforcement via perturbation of hamiltonian matrices. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 51(9):1755–1769, 2004.
- [24] S. Grivet-Talocia and A. Ubolli. A comparative study of passivity enforcement schemes for linear lumped macromodels. *Advanced Packaging, IEEE Transactions on*, 31(4):673–683, 2008.

- [25] B. Gustavsen. Fast passivity enforcement for pole-residue models by perturbation of residue matrix eigenvalues. *Power Delivery, IEEE Transactions on*, 23(4):2278–2285, 2008.
- [26] W. M. Haddad and V. Chellaboina. *Nonlinear Dynamical Systems and Control*. Princeton, 2008.
- [27] D. Hill and P. Moylan. Connections between finite-gain and asymptotic stability. *Automatic Control, IEEE Transactions on*, 25(5):931–936, Oct 1980.
- [28] D. J. Hill and P. J. Moylan. Dissipative dynamical systems: Basic input-output and state properties. *Journal of the Franklin Institute*, 309(5):327–357, 1980.
- [29] P. Holoborodko. Smooth noise robust differentiators. <http://www.holoborodko.com/pavel/numerical-methods/numerical-derivative/smooth-low-noise-differentiators/>, 2008.
- [30] Y.-C. Hsiao and L. Daniel. Sparse basis pursuit on automatic nonlinear circuit modeling. In *ASIC (ASICON), 2013 IEEE 10th International Conference on*, pages 1–4, Oct 2013.
- [31] H. Kahraman, M. Ozaydin, E. Varol, S. M. Aslan, A. Dogan, A. Altinbas, M. Demir, O. Gedikli, G. Acar, and O. Ergene. The diameters of the aorta and its major branches in patients with isolated coronary artery ectasia. *Texas Heart Institute Journal*, 33(4):463–468, 2006.
- [32] T. Kato, K. Inoue, and D. Kagawa. Lumped equivalent model synthesis for a passive element with frequency-dependent and/or temperature-dependent characteristics for emc simulation. In *Power Electronics and Motion Control Conference, 2009. IPMEC '09. IEEE 6th International*, pages 963–969, May 2009.
- [33] H. K. Khalil. *Nonlinear Systems*. Prentice Hall, 3rd edition, 2002.
- [34] B. E. Koenig and R. L. Tummala. Enterprise models for the design and management of manufacturing systems. In G. Fandel, T. Gulledge, and A. Jones, editors, *New Directions for Operations Research in Manufacturing*, pages 476–499. Springer Berlin Heidelberg, 1992.
- [35] E. B. Kosmatopoulos, M. M. Polycarpou, M. A. Christodoulou, and P. A. Ioannou. High-order neural network structures for identification of dynamical systems. *Neural Networks, IEEE Transactions on*, 6(2):422–431, Mar 1995.
- [36] D. N. Ku. Blood flow in arteries. *Annual Review of Fluid Mechanics*, 29(1):399–434, 1997.
- [37] C. Li and X. Liao. Passivity analysis of neural networks with time delay. *Circuits and Systems II: Express Briefs, IEEE Transactions on*, 52(8):471–475, 2005.
- [38] L. Ljung. *System identification*. Springer, 1998.

- [39] J. Lofberg. Yalmip : a toolbox for modeling and optimization in matlab. In *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, pages 284–289, 2004.
- [40] J. Lofberg. Pre- and post-processing sum-of-squares programs in practice. *IEEE Transactions on Automatic Control*, 54(5):1007–1011, 2009.
- [41] Z. Mahmood, B. Bond, T. Moselhy, A. Megretski, and L. Daniel. Passive reduced order modeling of multiport interconnects via semidefinite programming. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, pages 622–625, 2010.
- [42] Z. Mahmood, S. Grivet-Talocia, A. Chinea, G. C. Calafio, and L. Daniel. Efficient localization methods for passivity enforcement of linear dynamical models. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 33(9):1328–1341, Sept 2014.
- [43] MathWorks. *MATLAB R2014b*. Natick, Massachusetts.
- [44] MathWorks. *Simscape 3.12*. Natick, Massachusetts.
- [45] A. Megretski. Convex optimization in robust identification of nonlinear feedback. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 1370–1374, Dec 2008.
- [46] B. K. Mishra, P. Pradhan, and T. C. Panda. Flow of blood in a porous medium and its effect on heat transfer rate. *Int J Mol Med Adv Sci*, 6(1):8–13, 2010.
- [47] J. Montbrun-Di Filippo, M. Delgado, C. Brie, and H. M. Paynter. A survey of bond graphs: theory, applications and programs. *Journal of the Franklin Institute*, 328(5):565–606, 1991.
- [48] O. Nelles. *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*. Springer Science & Business Media, 2001.
- [49] Y. Oishi. Passivity degradation under the discretization with the zero-order hold and the ideal sampler. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 7613–7617, Dec 2010.
- [50] P. A. Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. PhD thesis, California Institute of Technology, 2000.
- [51] H. M. Paynter. *Analysis and Design of Engineering Systems: Class Notes for M.I.T. Course 2.751*. M.I.T. Press, 1961.
- [52] A. Poznyak. *Advanced Mathematical Tools for Control Engineers: Volume 1: Deterministic Systems*. Advanced Mathematical Tools for Automatic Control Engineers. Elsevier Science, 2010.

- [53] A. Quarteroni. Cardiovascular mathematics. In *Proceedings of the International Conference of Mathematicians*, volume I, pages 479–512. EMS, 2006.
- [54] G. Raisbeck. A definition of passive linear networks in terms of time and energy. *Journal of Applied Physics*, 25(12):1510–1514, 1954.
- [55] B. Reznick. Extremal PSD forms with few terms. *Duke mathematical journal*, 45(2):363–374, 1978.
- [56] D. E. Root, J. Verspecht, D. Sharrit, J. Wood, and A. Cognata. Broad-band poly-harmonic distortion (PHD) behavioral models from fast automated simulations and large-signal vectorial network measurements. *Microwave Theory and Techniques, IEEE Transactions on*, 53(11):3656–3664, Nov 2005.
- [57] A. Savitzky and M. J. E. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36(8):1627–1639, 1964.
- [58] J. Schoukens and R. Pintelon. *Identification of Linear Systems: A Practical Guideline to Accurate Modeling*. Elsevier Science, 2014.
- [59] Y. Shi, P. Lawford, and R. Hose. Review of zero-D and 1-D models of blood flow in the cardiovascular system. *Biomed Eng Online*, 10(33), 2011.
- [60] T. Soderstrom and P. Stoica. *System identification*, volume 2. Prentice Hall London, 1989.
- [61] N. O. Sokal and A. D. Sokal. Class E—a new class of high-efficiency tuned single-ended switching power amplifiers. *Solid-State Circuits, IEEE Journal of*, 10(3):168–176, June 1975.
- [62] K. C. Sou, A. Megretski, and L. Daniel. Convex relaxation approach to the identification of the wiener-hammerstein model. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 1375–1382, 2008.
- [63] K. C. Sou, A. Megretski, and L. Daniel. A quasi-convex optimization approach to parameterized model order reduction. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 27(3):456–469, 2008.
- [64] B. Sturmfels. Polynomial equations and convex polytopes. *The American mathematical monthly*, 105(10):907–922, 1998.
- [65] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society (Series B)*, 58(1):267–288, 1996.
- [66] M. M. Tobenkin, I. R. Manchester, and A. Megretski. Stable nonlinear identification from noisy repeated experiments via convex optimization. In *American Control Conference (ACC), 2013*, pages 3936–3941, June 2013.

- [67] M. M. Tobenkin, I. R. Manchester, J. Wang, A. Megretski, and R. Tedrake. Convex optimization in identification of stable non-linear state space models. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 7232–7237, Dec 2010.
- [68] J. S. Toll. Causality and the dispersion relation: Logical foundations. *Phys. Rev.*, 104:1760–1770, Dec 1956.
- [69] L. N. Trefethen. *Approximation theory and approximation practice*. SIAM, 2013.
- [70] H. Unbehauen and G. P. Rao. A review of identification in continuous-time systems. *Annual Reviews in Control*, 22(0):145–171, 1998.
- [71] A. van der Schaft. *L₂-Gain and Passivity Techniques in Nonlinear Control*. Springer, 2nd edition, 2000.
- [72] T. Wang, Y. Li, and Z. Ye. Robust passive macro-model generation with local compensation. *Microwave Theory and Techniques, IEEE Transactions on*, 60(8):2313–2328, Aug 2012.
- [73] J. C. Willems. Dissipative dynamical systems part I: General theory. *Archive for rational mechanics and analysis*, 45(5):321–351, 1972.
- [74] T. T. Wu. Causality and the radiation condition. *Craft Lab., Harvard Univ., Cambridge, Mass., Tech. Rept*, (211), Nov 1954.
- [75] J. L. Wyatt, Jr., L. O. Chua, J. Gannett, I. Goknar, and D. Green. Energy concepts in the state-space theory of nonlinear n-ports: Part I—passivity. *Circuits and Systems, IEEE Transactions on*, 28(1):48–61, Jan 1981.
- [76] F. Yilmaz and M. Y. Gundogdu. A critical review on blood flow in large arteries; relevance to blood rheology, viscosity models, and physiologic conditions. *Korea-Australia Rheology Journal*, 20(4):197–211, 2008.
- [77] T. Yioultsis, A. Woo, and A. C. Cangellaris. Passive synthesis of compact frequency-dependent interconnect models via quadrature spectral rules. In *Computer Aided Design, 2003. ICCAD-2003. International Conference on*, pages 827–834, Nov 2003.
- [78] D. C. Youla, L. Castriota, and H. J. Carlin. Bounded real scattering matrices and the foundations of linear passive network theory. *Circuit Theory, IRE Transactions on*, 6(1):102–124, Mar 1959.
- [79] W. Yu and X. Li. Some stability properties of dynamic neural networks. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 48(2):256–259, 2001.
- [80] M. Zamani and R. Majumdar. A Lyapunov approach in incremental stability. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 302–307, Dec 2011.

- [81] M. Zamani, N. van de Wouw, and R. Majumdar. Backstepping controller synthesis and characterizations of incremental stability. *Systems Control Letters*, 62(10):949–962, 2013.