

2016

Control and optimization methods in biomedical systems: from cells to humans

<https://hdl.handle.net/2144/17071>

"Downloaded from OpenBU. Boston University's institutional repository."

BOSTON UNIVERSITY
COLLEGE OF ENGINEERING

Dissertation

**CONTROL AND OPTIMIZATION METHODS IN
BIOMEDICAL SYSTEMS: FROM CELLS TO HUMANS**

by

QI ZHAO

B.S., University of Science and Technology Beijing, 2010
M.S., Boston University, 2012

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

2016

© 2016 by
QI ZHAO
All rights reserved

Approved by

First Reader

Ioannis Paschalidis, PhD
Professor of Electrical and Computer Engineering
Professor of Systems Engineering
Professor of Biomedical Engineering

Second Reader

Daniel Segrè, PhD
Professor of Biology
Professor of Biomedical Engineering
Professor of Physics

Third Reader

Pirooz Vakili, PhD
Associate Professor of Mechanical Engineering
Associate Professor of Systems Engineering

Third Reader

Hua Wang, PhD
Associate Professor of Mechanical Engineering
Associate Professor of Systems Engineering

*What the Great Learning teaches, is
to illustrate illustrious virtue;
and to rest in the highest excellence.*
One)

The Great Learning (Paragraph

Acknowledgments

Completing a Ph.D. degree is never an individual experience; rather it includes the efforts of some persons, whom I would like to thank sincerely.

First of all, I wish to express my gratitude to my advisor, Professor Paschalidis. I have been working with him for more than 5 years, and I learned a lot from him. Professor Paschalidis has excellent time management skills and he never tires of his passion for new challenges in research. I truly appreciate his consistent support, patience, and encouragement throughout my graduate study. His technical and editorial advice was essential to the completion of this dissertation.

I also wish to thank professor Daniel Segrè. I started working with him from 2012. His expertise in biology inspired me and motivated me to come up with a lot of amazing ideas in the research.

I would like to thank my committee members and chair, professor Hua Wang, professor Pirooz Vakili and professor Roberto Tron for reading the draft of my dissertation and providing many valuable comments that improved the quality of this dissertation.

I wish to acknowledge the support received from my lab mates and friends. We had many interesting and good-spirited discussions related to this research.

Last, but not the least, I would like to express special thanks to my wife, Yiwen Ding, and my parents for their understanding and love during my Ph.D. study. Their support and encouragement was in the end what made this dissertation possible.

This work is supported by the NSF under grants CNS-1239021 and IIS-1237022, by the NIH under grants R01-GM093147 and R01-GM089978, by the ARO under grants W911NF-11-1-0227 and W911NF-12-1-0390, by the ONR under grant N00014-10-1-0952, and by the DOE under grant DE-SC0004962.

Qi Zhao

Systems Engineering Division

CONTROL AND OPTIMIZATION METHODS IN BIOMEDICAL SYSTEMS: FROM CELLS TO HUMANS

QI ZHAO

Boston University, College of Engineering, 2016

Major Professor: Ioannis Paschalidis, PhD
Professor of Electrical and Computer Engineering
Professor of Systems Engineering
Professor of Biomedical Engineering

ABSTRACT

Optimization and control theory are well developed techniques to quantize, model, understand and optimize real world systems and they have been widely used in engineering, economics, and science. In this thesis, we focus on applications in biomedical systems ranging from cells to microbial communities, and to something as complex as the human body.

The first problem we consider is that of medication dosage control for drugs delivered intravenously to the patient. We focus specifically on a blood thinner (called bivalirudin) used in the post cardiac surgery Intensive Care Unit (ICU). We develop two approaches (a model-free and a model-based one) that predict the effect of bivalirudin. After obtaining the model and its best fit parameters by solving a non-linear optimization problem, we develop automatic dosage controllers that adaptively regulate its effect to desired levels. Our algorithms are validated using actual data from a large hospital in the Boston area.

In the second problem, we introduce a cellular objective function inference mechanism in metabolic networks. We develop an inverse optimization method, called

InvFBA (Inverse Flux Balance Analysis), to infer the objective functions of growing cells by using their reaction fluxes. InvFBA can be seen as an inverse version of FBA (Flux Balance Analysis) which predicts the distribution of the cell's reaction fluxes by using a hypothetical objective function. The objective functions can be linear, quadratic and non-parametric. The efficiency of the InvFBA approach matches the structure of the FBA and ensures scalability to large networks and optimality of the solution. After testing our algorithm on simulated *E. coli* data and time-dependent *S. oneidensis* fluxes inferred from gene expression data, we apply our inverse approach to flux measurements in long-term evolved *E. coli* strains, revealing objective functions that provide insight into metabolic adaptation trajectories.

In the final problem in this thesis, we formulate a novel resource allocation problem in microbial ecosystems. We consider a given number of microbial species living symbiotically in a community and a list of all metabolic reactions present in the community, expressed in terms of the metabolite proportions involved in each reaction. We are interested in allocating reactions to organisms so that each organism maintains a minimal level of growth and the community optimizes certain objectives, such as maximizing growth and/or the uptake of specific compounds from the common environment. We leverage tools from Flux Balance Analysis (FBA) and formulate the problem as a mixed integer linear programming problem. We test our method in a toy model involving two organisms that can only survive through cross-feeding, demonstrating that the method can recover this interaction. We also test the method in a community of two simplified bacteria described in terms of their core, simplified metabolic network. We demonstrate that the method can obtain syntrophic cross-feeding species that would be very difficult to design manually.

Contents

1	Introduction	1
1.1	Overview	1
1.2	Literature Review	2
1.3	Our Contributions	4
1.4	Thesis Organization	5
2	Predicting and Evaluating the Effect of Bivalirudin in Cardiac Surgical Patients	7
2.1	Background	7
2.2	Regularized Regression	10
2.2.1	Preliminaries	10
2.2.2	The Predictors	12
2.2.3	Results	15
2.3	An Explicit Dynamic System Model	18
2.3.1	The Model	18
2.3.2	Parameter Identification	21
2.4	An Adaptive Model: Extended Kalman Filter	24
2.5	Conclusions	28
3	Adaptive Control of Bivalirudin in the Cardiac Intensive Care	30
3.1	Introduction	30
3.2	Dynamic System Model Formulation	31
3.2.1	The Model	31

3.2.2	Parameter Identification	33
3.3	Bivalirudin Control Scheme	33
3.3.1	Model Reference Control (MRC)	35
3.3.2	Indirect Model Reference Adaptive Control (MRAC)	41
3.3.3	Direct Model Reference Adaptive Control MRAC	47
3.4	Conclusions	51
4	Learning Cellular Objectives from Fluxes by Inverse Optimization	52
4.1	Introduction	52
4.2	Inverse Process of FBA	55
4.2.1	Inverse FBA with Linear Objective Function	56
4.2.2	Linear Objective Variability Analysis	62
4.2.3	Inverse FBA with Quadratic Objective Function	63
4.2.4	Inverse FBA with Non-parametric Objective Function	65
4.3	Numerical Examples	68
4.3.1	Inference of Linear Objective Functions in Simulated <i>E. coli</i> Fluxes	68
4.3.2	Recovering Objectives and Fluxes From Noisy Data	71
4.3.3	Inference of a Linear Objective Function with Time-dependent Fluxes	74
4.3.4	The Inference of Linear Objective Functions in <i>E. coli</i> Strains that Underwent Long-term Evolutionary Experiments	77
4.3.5	The Inference of a Quadratic Objective Function with Time Series Fluxes	79
4.3.6	The Inference of a Kernelized Objective Function on <i>E. coli</i> Data	81
4.4	Conclusions	82

5	Optimal Allocation of Metabolic Functions among Organisms in a Microbial Ecosystem	84
5.1	Problem Formulation	86
5.1.1	Community-level Flux Balance Analysis	86
5.1.2	Community Stoichiometric Partitioning	90
5.2	An Artificial Toy Model	91
5.2.1	Model Description	91
5.2.2	FBA at the Community Level	92
5.2.3	Identification of the Putative Vector \mathbf{t}	94
5.3	A Simplified Bacterial Community	95
5.3.1	Model Description	95
5.3.2	Community Analysis	96
5.4	Conclusions	99
6	Conclusions	100
6.1	Summary	100
6.2	Future Works	102
A	LASSO Version of InvFBA	103
B	Noise Generation	104
C	Inference of Fluxes from Experimentally Measured Branching Ratios	105
	References	107
	Curriculum Vitae	113

List of Tables

2.1	Performance of the regularized polynomial regression (test set). . . .	18
2.2	Optimal parameters values.	23
2.3	Performance of the dynamic system model.	24
2.4	EKF vs. dynamic model.	28
3.1	Performance of the Model Reference Control (test set).	41
3.2	Performance of the Inaccurate Model Reference Control (test set). . .	42
3.3	Performance of the Indirect Model Reference Adaptive Control (test set).	47
3.4	Performance of the Direct Model Reference Adaptive Control (test set).	50
3.5	Performances of the three control laws (test set).	51
4.1	InvFBA results from ℓ_1 and ℓ_0 norm regularization of simulated <i>E. coli</i> data under different growth conditions.	70
4.2	InvFBA results from ℓ_1 and ℓ_0 regularization of simulated <i>E. coli</i> data under different growth conditions (Maximization of ATP synthase). .	72
4.3	InvFBA results from ℓ_1 and ℓ_0 regularization of simulated <i>E. coli</i> data under glucose minimal medium (Minimization of glucose uptake). . .	72
4.4	Inferred parameter values.	80
5.1	Biomass growth rate vs. the number of active reactions in the community.	97

List of Figures

2·1	The inputs (predictors) and the output (response) of the regression engine.	13
2·2	Performance of the regularized polynomial regression for $d = 1$ and $d = 2$ for different numbers M of time series inputs in the test set. . .	17
2·3	The single-state linear model includes a gain, $1/\beta_3$, representing the elimination time constant of bivalirudin from the body. The constant β_2 provides for the translation from serum concentration to the site-effect (PTT).	19
2·4	Illustrating the performance of the dynamic system model of Section 2.3 for a particular patient. The blue “o” represent predicted PTT values from our model and the red “*” represent actual measured values.	21
2·5	Illustrating the performance of the EKF algorithm for a particular patient. The blue “o” represent predicted PTT values from our model and the red “*” represent actual measured values. The top figure plots estimated and measured PTT and the bottom figure plots the running RMSE at each step. Additionally, blue dot line represents the average RME of EKF adaptive method while green dot line represents the fixed parameters solve by previous section. It is obvious that EKF adaptive methods fits the individual model better.	25
2·6	The EKF algorithm for recursively estimating model parameters for an individual patient.	26

2·7	Illustrating the evolution of model parameter values $\mathbf{z} = (\beta_1, \dots, \beta_{13}, x(0))$ during the course of EKF algorithm.	27
3·1	In this dynamic model, the bivalirudin infusion rate $u(t)$ is the only controllable input. $d(t)$ is the linear combination of the rest of the inputs.	33
3·2	Model Reference Adaptive Control (MRAC) structure.	34
3·3	The effect of the MRC law derived for and applied to one randomly selected patient.	36
3·4	The MRC law derived for one patient but applied to another patient.	41
3·5	The performance of the indirect MRAC law.	44
3·6	The performance of direct MRAC law.	48
4·1	The feasible set of the FBA problem is a polyhedron – the intersection of points satisfying the linear constraints of (4.1). \mathbf{x}^j , $j = 1, \dots, 4$, denote extreme points of the feasible set. Depending on the objective function coefficient \mathbf{c} , one of these extreme points is optimal. In some cases, a particular \mathbf{c} can lead to multiple extreme points being optimal (purple case), in which case the whole edge/facet containing them is optimal. Notice, that for each optimal extreme point there are many \mathbf{c} 's that could have led to its optimality. These “valid” \mathbf{c} 's belong to a cone (cf. the four colored cones drawn).	54

4.2 **(Left):** The union of these cones is all of \mathbb{R}^n ($n = 2$, in the figure). If we use an ℓ_2 -norm regularization in the invFBA formulation, “valid” \mathbf{c} ’s lie on the surface of the unit ball in \mathbb{R}^n . **(Right):** Illustration of the geometric intuition behind ℓ_2 -regularized FBA. The FBA feasible polyhedron can be partitioned using bisector lines. In this 2-dimensional example, the polyhedron is partitioned by 4 bisectors into 4 (colored) polytopes. For example, if the observed flux distribution \mathbf{x} lies in the yellow polytope then \mathbf{c}_1 is the corresponding objective function. If, however, the observed flux distribution lies in the bisector between the blue and the yellow region, invFBA will yield either \mathbf{c}_1 or \mathbf{c}_4 58

4.3 **(Left):** A simple simulated metabolic network model with a single metabolite A and three reactions. **(Right):** We plot the FBA polyhedron on the x_1 - x_2 plane. The red dot indicated a given metabolic flux distribution. The invFBA algorithm yields $\mathbf{c} = (1/3, 2/3)$ 62

- 4·6 Metabolite secretion flux (dashed line), inferred from experimental data through the TEAM approach, and maximum coefficient of the pyruvate secretion flux in the objective function (full line), as predicted by invFBA (through OVA), are plotted as a function of time. Metabolite and gene expression data come from time-dependent measurements performed during batch aerobic growth of the bacterium *S. oneidensis* on lactate. The secreted metabolites are pyruvate in Fig. 4·6a, glycolate in Fig. 4·6b, and acetate in Fig. 4·6c. Positive fluxes reflect secretion of the metabolite in question, while negative fluxes reflect uptake. 75
- 4·7 (A) Experimental measurements (by (Harcombe et al., 2013)) of acetate excretion and glucose uptake for the ancestral (red star) and evolved (blue and red dots) *E. coli* strains from Lenskis long-term evolutionary experiment. The red and blue colors are used here to highlight two distinct metabolic regimes that different strains seem to cluster around. (B) A projection (onto a two dimensional subspace) of the set of objective functions compatible with experimentally measured fluxes. The graph is obtained through a two-dimensional version of OVA: for each possible value of the growth flux coefficient of the objective function ($c_{Biomass}$), one can find the minimal and maximal value of the objective function coefficient for the respiratory flux ($c_{Respiration}$), obtaining areas that correspond to objective functions compatible with the measured fluxes. Such regions can be computed for the ancestral and all evolved *E. coli* strains. The strains corresponding to the different metabolic regimes (blue and red dots in (A)) map onto different regions in the space of objectives, labeled with similar colors. 78

4.8	Simulation flowchart for the inference of a nonparametric objective function.	82
4.9	Performance evaluation of the nonparametric objective function. . . .	83
5.1	The structure of the universal stoichiometric matrix \mathbf{S} . Block \mathbf{S}^u represents the set of uptake reactions used to absorb nutrients from the environment. Blocks $[\mathbf{S}^{e1}; \mathbf{S}^{e2}]$ represent the set of exchange reactions between external and internal metabolites. \mathbf{S}^i represents the set of internal reactions among internal metabolites.	87
5.2	The structure of the stoichiometric matrix for the whole community $\mathbf{S}^c \in \mathbb{R}^{M_c \times N_c}$	88
5.3	The structure of the artificial community with two different organisms. Both organisms 1 and 2 uptake the same external metabolite (EX) from the environment. Organism 2 produces Y only via $R1$ and obtains Z from organism 1. Organism 1 produces Z only via $R2$ and obtains Y from organism 2. Organisms 1 and 2 have the same biomass growth reaction which involves X , Y and Z . Organisms 1 and 2 cannot grow without each other.	92
5.4	The universal stoichiometric matrix \mathbf{S} . The rows of \mathbf{S} include three external metabolites (EX, EY, EZ) and three internal metabolites (X, Y, Z). The columns of \mathbf{S} include one uptake reaction, three exchange reactions, and three internal reactions.	93

5.5	The community stoichiometric matrix \mathbf{S}^c is shown in the top figure. The binary vector \mathbf{t} denotes the ground truth, consistent with Fig. 5.4. The optimal flux distribution \mathbf{x}^* is shown in the bottom. To distinguish the internal metabolites in the two organisms, we use $X1, Y1$ and $Z1$ to denote the internal metabolites in organism 1, and $X2, Y2$ and $Z2$ for the internal metabolites in organism 2.	93
5.6	The results of identifying \mathbf{t}^* and the corresponding flux vector \mathbf{x}^* . They are consistent with the ground truth in Fig. 5.5.	95
5.7	The universal stoichiometric matrix \mathbf{S} and the corresponding flux vector by maximizing the biomass growth rate for an individual bacterial model from (Covert et al., 2001) under the current growth conditions.	96
5.8	Visualization of the outcome of our algorithm, applied to a model of generic bacterial central carbon metabolism from (Covert et al., 2001). In this case, the algorithm is applied to two organisms that have the potential to contain all 12 internal reactions found in the initial network. Yet, the algorithm identifies two distinct species with fewer internal reactions that perform different tasks, but can collectively achieve all the metabolic functions necessary for growth.	98

List of Abbreviations

ALB	Albumin
ATP	Adenosine triphosphate
BFGS	Broyden Fletcher Goldfarb Shanno
EHRs	Electronic Health Records
EKF	Extended Kalman Filter
FBA	Flux Balance Analysis
ICU	Intense Care Unit
INR	International Normalized Ratio
InvFBA	Inverse Flux Balance Analysis
LASSO	Least Absolute Shrinkage and Selection Operator
LTI	Linear Time Invariant
LV	Lotka Volterra
MILP	Mixed Integer Linear Programming
MPP	Maximum Power Principle
MRAC	Model Reference Adaptive Control
MRC	Model Reference Control
NRSME	Normalized Root Mean Square Error
PLT	Platelet Count
PTT	Partial Thromboplastin Time
\mathbb{R}^2	the Real plane
RKHS	Reproducing Kernel Hilbert Space
RRT	Resource Ratio Theory
RSME	Root Mean Square Error
SGOT	Serum Glutamic Oxaloacetic Transaminase
SGPT	Serum Glutamic Pyruvic Transaminase.
TBILI	Total Bilirubin

Chapter 1

Introduction

1.1 Overview

Optimization and control theories are essential to any problems involving decision making and resource allocation. The primary goal of decision making is to choose the best solution among various alternatives. The measure of goodness is determined by an objective function (Chong and Zak, 2013). Control theory is an interdisciplinary branch of engineering and has found many application in several domains. It deals with the behavior of dynamic systems with inputs, and how their behavior is regulated by feedback. The most common objective is to control a system so that its output follows a desired signal. The optimization and control techniques have been widely applied to the fields of computer science, management science, biology, and many others.

In recent years, the optimization and control theories have received more and more attention, primarily because of the increasing availability of data and the rapid progress in computer technology. Due to the advantages of optimization and control theories, computational algorithms become more efficient and deeper insights behind the “big data” are revealed. In this thesis, we focus on the novel applications of optimization and control theories in biomedical systems, ranging from cells to microbial communities, to the human body. In the biomedical field, new modeling techniques and quantitative methods can to improve healthcare efficiency, interpret the metabolic mechanisms and design synthetic organisms, which has the potential to bring great

benefits to society.

1.2 Literature Review

In the past decades, researchers have never stopped exploring the field of biomedical systems by conducting experiments, analyzing data, and developing new models. Optimization and control theories have very important roles to play in this area of research. One of the motivations for new research is that the US health care system is considered costly and highly inefficient. While there is no broad agreement on the potential solutions, the meaningful use of Electronic Health Records (EHRs) is seen as a key to improving efficiency. The health care system, however, is not well equipped to deal with the impending deluge of personalized health-related EHRs and medical devices. EHRs and the digitization of data from medical devices have the potential to aid prevention, automate decision making, and facilitate treatments. Motivated by these, researchers focus on leveraging the EHRs to develop models so that they can be used for improving healthcare efficiency. For example, (Edrich et al., 2011b) and (Edrich et al., 2011a) propose a simple steady-state linear model to describe the effect of bivalirudin in cardiac surgical patients. In (Dai et al., 2015), optimization and supervised machine learning theory are applied to predict the hospitalization due to cardiac diseases using actual EHRs data. In (Ludwick and Doucette, 2009), several mathematical methods are introduced to analyze and to model the clinical data. These advanced clinical data analysis methods improve the efficiency of the health care system and bring significant benefits to both hospitals and patients.

At the micro biology level, optimization and control methods not only explain current adaptations of biological systems, but also help to predict new designs that may yet evolve. Linear programming is used to analyze metabolic networks and to predict the internal fluxes of an individual cell in (Kauffman et al., 2003). In (Sauer

et al., 1998), a non-convex optimization method is applied to infer the cells' objective functions under certain growth conditions. In (Segre, 2008), a non-linear optimization approach is adopted to develop the constraints based metabolic regulation model and to identify the corresponding regulation effect matrix. (Heinrich and Schuster, 1998) leverages the ideas of dynamic systems in control theory to model the dynamic metabolic system and analyze its controllability and optimality. (Heinrich and Rapoport, 1974) utilizes the dynamic system to describe a linear steady-state treatment of enzymatic chains. (Newsholme and Crabtree, 1979) leverages the properties of extreme points in linear optimization theory to find and to interpret the extreme pathways in the metabolic network. With these advanced models and methods, the structures of the metabolic systems can be described unambiguously, and predictions about their behaviors can be made in a systematic way.

Based on the well-developed models of the individual cells, researchers extended their attention to the synthetic ecology which involves design, construction and interpretation of microbial communities. Despite the growing availability of experimental data for a diverse range of complex natural microbial communities, the full characterization and understanding of these communities is still challenging (Zomorodi and Segrè, 2016). To solve this problem, efficient computational techniques and mathematical modeling tools are required. These models are critical in interpreting ecological and evolutionary questions (e.g., quantifying the inter-species interaction, environment factors on the cooperation, coexistence of cooperators and cheaters in the communities, etc.). The most important role of mathematical modeling is that we are able to rationally design and simulate synthetic consortia for desired applications. Two of the most important models in theoretical ecology are Resource Ratio Theory (RRT) and the Maximum Power Principle (MPP). RRT (Miller et al., 2005) has been primarily used to model competition and has been extended to account for cooperative

interactions as well. MPP (Miller et al., 2005) relies on the assumption that all biological systems self-organize to increase power. A traditional way to model the dynamics of microbial communities rooted in theoretical ecology is the use of coupled differential equations describing the temporal evolution of microbial species abundances. The most widely-used such model is the Lotka-Volterra (LV) model (Wangersky, 1978). The complex balancing of benefits and costs associated with inter-species interactions in microbial communities can also be effectively addressed by using game theory and evolutionary game approaches (Schuster et al., 2008). Flux Balance Analysis (FBA) (Harcombe et al., 2013) is a widely used approach for studying biochemical networks, and in particular the genome-scale metabolic network reconstructions. The problem of predicting a cell's chemical reaction fluxes is formulated as the linear optimization problem of maximizing a cellular objective (e.g., growth) subject to constraints capturing stoichiometry mass balances of the metabolic network and bounds that reflect the composition of the growth medium.

1.3 Our Contributions

In this dissertation, we first describe an automated decision making system that can be used to control medication dosing in the ICU. In this work, we develop an automated bivalirudin infusion system targeted for cardiac surgical patients in the ICU. To achieve this goal, we develop two approaches that predict the effect of bivalirudin. The first approach is model-free and utilizes regularized regression. The second approach is model-based and leverages more detailed biomedical knowledge. After obtaining the model and its best fit parameters, we develop three adaptive dosage controllers that regulate bivalirudin's effect to desired levels under different cases. Our algorithms are validated by using the actual clinical data.

In the second part of the thesis, we focus on inferring the cell's performance-based

on the experimentally observed fluxes under certain growth conditions. Towards this goal, we develop an *inverse Flux Balance Analysis (invFBA)* method which is a novel convex optimization based framework. The convexity of the invFBA approach matches the structure of the (forward) FBA and ensures scalability to large networks and optimality of the solution. Within this framework, we present three different forms of objective functions: linear, quadratic, and non-parametric. We show that in all cases, the inverse problem is tractable and can be solved efficiently. We also provide several numerical examples to evaluate the performance of invFBA and interpret the results from both biological and mathematical aspects.

In the third part, we extend our scope from the individual cells to microbial ecosystems. We develop a novel *Mixed Integer Linear Programming (MILP)* based approach to optimally allocate the metabolic functions among organisms in a microbial ecosystem. We test the method on both a toy artificial community and a community composed of two bacterial organisms with a simplified core metabolism. In both cases, the method helps us identify the individual metabolic network topology and elucidate the interaction between species in the microbial community. It offers a new platform for the rational design of organisms and communities towards future synthetic ecology applications.

1.4 Thesis Organization

The remainder of this thesis is organized as follows.

- Chapter 2 introduces the automatic decision system of bivalirudin. It describes a model-free regularized regression method first and then presents the dynamic system model and the model-based parameter identification method. The extended Kalman filter is used to achieve model parameter adaptation.
- Chapter 3 presents our novel convex optimization based method –Inverse Flux

Balance Analysis (InvFBA)– which is used to infer the objective function of a cell under certain growth conditions. We apply the invFBA to different data sets and interpret them from different aspects.

- Chapter 4 provides a mixed integer linear programming-based approach to optimally allocate metabolic reactions among organisms in a microbial ecosystem.
- Chapter 5 includes concluding remarks and introduces an outline of future work.

Notation: We use bold letters to denote vectors and matrices; vectors are denoted by lower case letters and matrices by upper case letters. Vectors are assumed to be column vectors unless explicitly stated otherwise. For economy of space we write $\mathbf{x} = (x_1, \dots, x_n)$ for the n -dimensional column vector $\mathbf{x} \in \mathbb{R}^n$. Prime denotes transpose, $\|\cdot\|$ denotes the Euclidean norm, $\mathbf{1}$ (resp., $\mathbf{0}$) denotes a vector or matrix with all components set to one (resp., zero), and $diag(\mathbf{x})$ denotes a diagonal matrix with the main diagonal being the vector \mathbf{x} and all other off-diagonal elements being zero.

Chapter 2

Predicting and Evaluating the Effect of Bivalirudin in Cardiac Surgical Patients

2.1 Background

Bivalirudin antagonizes the effect of thrombin in the blood clotting cascade, thereby preventing complications from blood clotting. It is currently FDA-approved for short term anticoagulation of patients undergoing cardiac catheterization to prevent complications due to undesired blood clots (Bittl et al., 2001; Lincoff et al., 2003; Stone et al., 2006). Bivalirudin is infused as a “blood thinner” in patients who have or are suspected of having blood clots or risk of blood clotting and who have a contraindication to heparin. It is infused continuously, and is eliminated via the kidney and by plasma protease-metabolism. It affects the coagulation parameters *Partial Thromboplastin Time (PTT)* and the *International Normalized Ratio (INR)* in a dose-dependent fashion. Both measure the ability of the blood to clot but while PTT is measured in seconds, INR is a dimensionless number. Clinically, *PTT* can be seen as the key effect of bivalirudin and the level of *PTT* highly depends on the dosage of bivalirudin and patients’ renal and liver conditions.

As a rarely used drug, bivalirudin is used more frequently in the *Intensive Care Unit (ICU)* but residents adjusting the infusion rate may have little experience, resulting in overdosing or underdosing. Adequate anticoagulation is necessary to avoid the risk of clot formation, but overshooting increases the risk of bleeding. There is

considerable inter- and intra-individual variability in the response to bivalirudin; it is challenging to titrate the drug. Currently, empiric titration of bivalirudin according to clinical experience or a simple nomogram is used to achieve desired anticoagulation (Kiser et al., 2011). For this reason, a mathematical model that predicts the PTT based on the past infusion rates of bivalirudin following dose adjustment would be useful to guide optimal therapy.

In earlier work ((Edrich et al., 2011b) and (Edrich et al., 2011a)), an one-state linear system model to describe the effect of bivalirudin in patients has been proposed. The models were designed using Matlab/simulink (Mathworks, Natick, MA) and default parameter identification procedures. Motivated by this work, in the present chapter we develop two new methods to predict PTT values based not only on past bivalirudin infusion rates but also on a host of patient-specific physiological variables that characterize coagulation, renal, and liver function. The results we obtain substantially improve accuracy compared with the earlier work.

Our first method is model-free, in the sense that a specific model does not need to be constructed in advance, and leverages regularized and kernelized regression. With uniform sampled data, standard time series analysis methods (e.g., ARX, ARMAX (Hannan and Deistler, 1988)) could have been a viable alternative. In our problem, we encounter highly non-uniform sampled data which challenges the standard methods, hence our use of regularized regression. Our method is purely data-driven and requires no explicit model to explain how bivalirudin affects PTT. It is flexible enough to use several samples of bivalirudin infusion rates from the immediate past in order to predict current PTT values. Since we use a rich set of predictors, we devise a regularization approach that can eliminate unnecessary predictors and regress on a reduced predictor set so as to avoid overfitting.

Our second method develops a more complex explicit dynamic state space system

model than the one developed in ((Edrich et al., 2011b) and (Edrich et al., 2011a)). This new model takes the elimination of bivalirudin by the kidneys and the liver into account. We identify model parameters by formulating a nonlinear optimization problem that minimizes the ℓ_2 norm of prediction error over a training set of measurements. As we mentioned before, we only have highly non-uniform sampled real data. Furthermore, the dosage of bivalirudin given to patients should be carefully titrated to ensure patient safety. As a result, we can not observe PTT values in response to arbitrary dosage. This suggests that canonical state-space system identification techniques (e.g., adaptive system identification (Ioannou and Kosmatopoulos, 2006), subspace state space system identification method (Ljung, 1987)) are not applicable.

The nonlinear optimization problem we formulate is solved by leveraging quasi-Newton methods. The dynamic system model we obtain performs only somewhat worse than the model-free approach, even though it uses a shorter history of past measurements. Building on this model, we develop an adaptive on-line algorithm based on the extended Kalman filter that can adapt the model parameters to individual patients. The algorithm starts from population-wide optimal parameters and, as it observes inputs and outputs, modifies model parameter values to better fit an individual patient. This adaptive model outperforms the model-free method in terms of average prediction error, despite using a shorter history of past measurements. This confirms the empirically known considerable individual variability in the response to bivalirudin. Further, with the adaptive model, the per-patient error variability is substantially reduced compared to the population-wide dynamic system model and it is on par with the corresponding error variance of the model-free method.

The approaches we put forth in this work are general and can be applicable to a host of related problems. As ICUs and hospital wards accelerate the digitization of patient records, tremendous opportunities arise for automated and mathematically

rigorous patient monitoring and medication dosing. It is in such a framework that the methods we develop can become useful.

The remainder of the chapter is organized as follows. Section 2.2 presents the model-free regularized regression method. Section 2.3 presents the dynamic system model. Section 2.4 develops the extended Kalman filter to achieve model parameter adaptation. Finally, concluding remarks appear in Section 2.5.

2.2 Regularized Regression

In this section we present our first “black box” approach. The idea is to use regression with appropriate function regularizers that can help avoid overfitting. We start with some preliminary material on regularization and then describe the features we use and the results we obtained.

2.2.1 Preliminaries

For completeness and to establish our notation we review standard material on regularized regression from (Zou and Hastie, 2005) and (Hastie et al., 2009). Let $\mathbf{x} \in \mathbb{R}^p$ denote the vector of features or predictors and $y \in \mathbb{R}$ the output or response. Suppose we are given a training set (\mathbf{x}_i, y_i) , $i = 1, \dots, N$, and we are interested in selecting a function $f : \mathbb{R}^p \rightarrow \mathbb{R}$ from some space \mathcal{H} so that it approximately holds $y_i \approx f(\mathbf{x}_i)$ over the training set. To that end, we can adopt a loss function $L(y, f(\mathbf{x}))$ that quantifies the quality of the fitting and solve the optimization problem

$$\min_{f \in \mathcal{H}} \left\{ \sum_{i=1}^N L(y_i, f(\mathbf{x}_i)) + \lambda G(f) \right\}, \quad (2.1)$$

where $G(f)$ is a regularization term that penalizes functions from \mathcal{H} which are not smooth enough, and λ is a parameter that determines the contribution of the penalty term. The purpose of the regularization term is to induce smoother functions that

do not have too many changes, and thus help avoid overfitting. The parameter λ can be selected by cross-validation as we will explain later.

An important special case concerns functions that live in a so called *Reproducing Kernel Hilbert Space (RKHS)* \mathcal{H}_K . For some (positive semi-definite) *kernel function* $K(\mathbf{x}, \mathbf{z})$, $\mathbf{x}, \mathbf{z} \in \mathbb{R}^p$, written as

$$K(\mathbf{x}, \mathbf{z}) = \sum_{j=1}^{\infty} \gamma_j \phi_j(\mathbf{x}) \phi_j(\mathbf{z}),$$

where $\phi_j(\cdot)$ s are basis functions which can be any popular functions (e.g., polynomial, Gaussian, cosine and so on) and the scalars $\gamma_j \geq 0$ satisfy $\sum_{j=1}^{\infty} \gamma_j^2 < \infty$, the functions $f \in \mathcal{H}_K$ can be written as

$$f(\mathbf{x}) = \sum_{j=1}^{\infty} c_j \phi_j(\mathbf{x}).$$

The regression problem (2.1) can now be written as

$$\min_{f \in \mathcal{H}_K} \left\{ \sum_{i=1}^N L(y_i, f(\mathbf{x}_i)) + \lambda \|f\|_{\mathcal{H}_K}^2 \right\}, \quad (2.2)$$

where the regularizer is the norm in \mathcal{H}_K defined as $\|f\|_{\mathcal{H}_K}^2 = \sum_{j=1}^{\infty} c_j^2 / \gamma_j$. According to the Representer theorem (Schölkopf et al., 2001), the solution to this problem has the form

$$f(\mathbf{x}) = \sum_{i=1}^N \alpha_i K(\mathbf{x}, \mathbf{x}_i), \quad (2.3)$$

and, in light of this form, finding the parameters α_i becomes as easy as solving a finite-dimensional optimization problem. If in fact one assumes a nice structure for the loss function, e.g., squared loss $L(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$, then there are very efficient algorithms for finding the α_i . In the case of squared loss, all we need to do is solve a simple convex unconstrained quadratic optimization problem which admits a closed-form solution.

The choice of the kernel is important but very much dependent on the particular

application one has to tackle. In our problem, a *polynomial kernel* achieves lower prediction error than alternatives. A polynomial kernel is of the form $K(\mathbf{x}, \mathbf{z}) = (\mathbf{x}'\mathbf{z} + 1)^d$, for $\mathbf{x}, \mathbf{z} \in \mathbb{R}^p$ and some integer d , which is the sum of all monomials $\prod_{i=1}^p x_i^{d_i^x} z_i^{d_i^z}$ with $d_i^x, d_i^z = 0, 1, \dots, d$ such that $\sum_{i=1}^p d_i^x \leq d$ and $\sum_{i=1}^p d_i^z \leq d$.

2.2.2 The Predictors

In our problem, we have clinical data from 233 patients. The key quantity (response) we would like to predict is the *Partial Thromboplastin Time (PTT)* $y_i(t)$ of each patient i at each time t . As predictors we include 11 key physiological variables sampled over M consecutive time instants $t > t - \tau_1 > \dots > t - \tau_{M-1}$ where τ_m for $m = 1, \dots, M - 1$ denote the time lags between consecutive measurements. These time lags are included as the 12th predictor.

The 11 physiological variables are:

1. **Bival rate** (mg/kg/h): the weight-based Bivalirudin injection rate.
2. **GFR** (mL/min): the Glomerular filtration rate, reflecting the ability of the kidneys to eliminate bivalirudin. Decreased GFR would increase the serum level of bivalirudin and the PTT in an approximately linear fashion (Company, 2011).
3. **PTT** (s): last measured partial thromboplastin time.
4. **INR** (Unit-less): the last measured international normalized ratio which is a coagulation time that is distinct, but associated with the PTT. This increases as the serum level of bivalirudin increases.
5. **SGOT** (Units/L): the Serum Glutamic Oxaloacetic Transaminase, and
6. **SGPT** (Units/L): the Serum Glutamic Pyruvic Transaminase. Increasing SGOT and SGPT would reflect liver dysfunction and decreased production of clotting

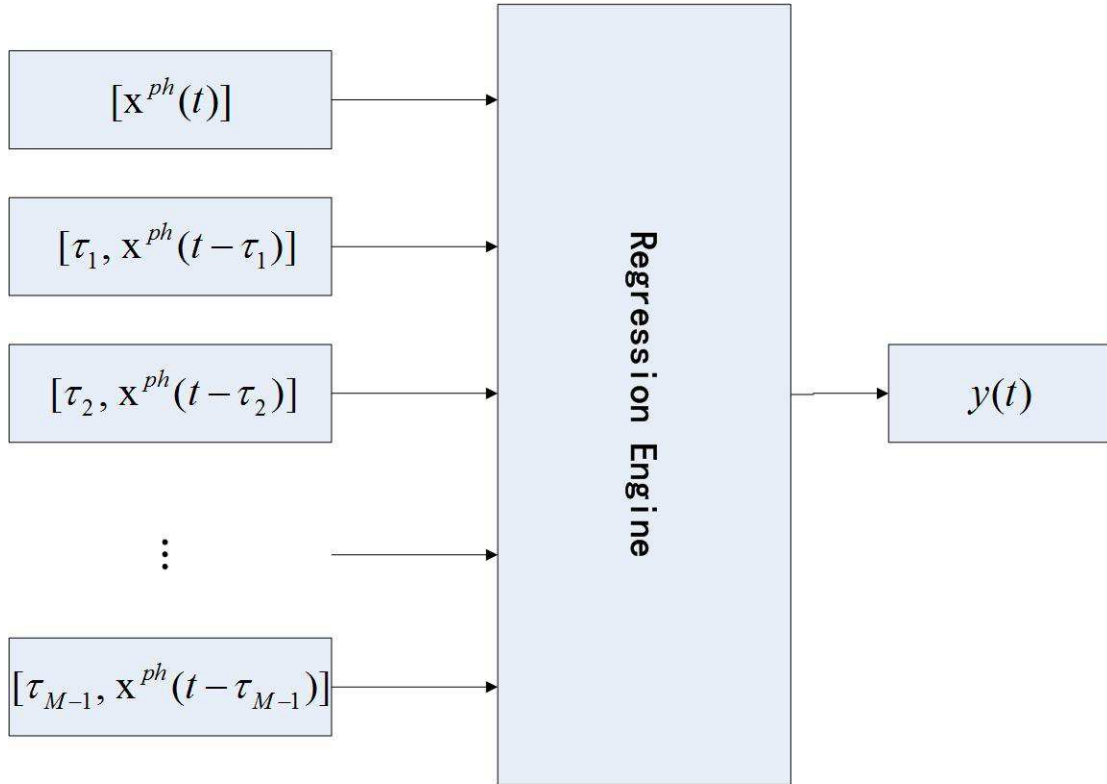


Figure 2.1: The inputs (predictors) and the output (response) of the regression engine.

factors, thus increasing PTT which, in turn increases with liver dysfunction. Since the liver produces clotting factors, liver dysfunction will increase the PTT.

7. **TBILI** (mg/dL): total bilirubin, a “waste product” normally eliminated by the liver. In liver dysfunction, this is positively associated with a rising PTT.
8. **ALB** (g/L): Albumin, which reduced in the setting of liver failure and therefore associated with a rising PTT.
9. **PLT** (K/mcL): Platelet count. Platelets help form blood clots with clotting factors from the liver. They are utilized when a clot is formed. A decrease-

ing platelet count can indicate ongoing clotting with consumption of clotting factors, thus, elevating the PTT and INR.

10. **HCT** (%): Hematocrit. HCT is a measure of the amount of red blood cells in the blood. When patients loose blood during operations and other non-operative bleeding, then fluids such as normal saline are provided to make up for the blood volume lost. This, however, lowers the HCT. At the same time, the added volume dilutes the clotting factors in the blood and causes PTT and INR to increase.
11. **FIB** (mg/dL): Fibrinogen. This protein helps produce clots and its decreased concentration may indicate that clotting is occurring. It follows that clotting factors are being depleted which will cause elevated PTT and INR.

Let now $\mathbf{x}_i^{ph}(t) \in \mathbb{R}^{11}$ denote the vector containing the 11 physiological variables presented above for patient i at time t . Note that $\mathbf{x}_i^{ph}(t)$ contains PTT and INR values at time $t - \tau_1$. To predict $y_i(t)$ we will use the predictor vector

$$\mathbf{x}_i(t) = (\mathbf{x}_i^{ph}(t), \tau_1, \mathbf{x}_i^{ph}(t - \tau_1), \dots, \tau_{M-1}, \mathbf{x}_i^{ph}(t - \tau_{M-1})), \quad (2.4)$$

that is, M consecutive measurements of $\mathbf{x}_i^{ph}(t)$ and the corresponding time lags which forms a $(12M - 1)$ -dimensional vector.

Our regression engine uses a polynomial kernel $K(\mathbf{x}, \mathbf{z}) = (\mathbf{x}'\mathbf{z} + 1)^d$ which can be written in terms of its (orthonormal) eigenfunctions, say as $K(\mathbf{x}, \mathbf{z}) = \sum_{j=1}^J \phi_j(\mathbf{x})\phi_j(\mathbf{z})$, where J is the number of eigenfunctions. For our polynomial kernel it holds $J = \binom{12M-1+d}{d}$. The response function takes the form $f(\mathbf{x}) = \sum_{j=1}^J c_j \phi_j(\mathbf{x})$. We write $\boldsymbol{\phi}(\mathbf{x}) = (\phi_1(\mathbf{x}), \dots, \phi_J(\mathbf{x}))$. We formulate a slightly different problem than (2.2) in that we use the so called elastic net penalty (Zou and Hastie, 2005). More specifically,

we minimize over the coefficients $\mathbf{c} = (c_1, \dots, c_J)$ the expression

$$\sum_{i=1}^N \sum_{t=t_i} [y_i(t) - \mathbf{c}'\phi(\mathbf{x}_i(t))]^2 + \lambda_1 \sum_{j=1}^J c_j^2 + \lambda_2 \sum_{j=1}^J |c_j|, \quad (2.5)$$

where N henceforth represents the number of patients in the training set and the summation over t_i is over the time instances in which we have measurements for those patients. The difference with (2.2) is the addition of a penalty term equal to the ℓ_1 norm of \mathbf{c} . This is done to induce sparsity and eliminate features that are “redundant” and not helpful in predicting $y_i(t)$. In our experiments, we found that it was important to include such a sparsity-inducing penalty especially because we use multiple time instances from the past to predict the current PTT value. Notice that the objective function in (2.5) is convex, thus, this problem can be easily solved using standard optimization techniques. The parameters λ_1 and λ_2 can be tuned using cross-validation as we will describe next. We also note that before solving (2.5) we standardized the predictors to have zero mean and unit variance.

2.2.3 Results

We randomly split the data into 3 sets of equal size. The first two sets are used for training and the third set for testing. To select optimal values for the parameters λ_1 and λ_2 in (2.5) we use leave-one-out 6-fold cross-validation applied to the training set (the first two sets). More specifically, we subdivide the training set into 6 equal parts. For a discretized collection of (λ_1, λ_2) we train (i.e., solve (2.5)) on the first 5 parts and use the 6th part to compute the prediction error (i.e., the first term in (2.5)). This is done for each $\frac{1}{6}$ th of the data left out of training and used only for validation and we then compute the average of the corresponding 6 prediction errors. We use this average value to select the best (λ_1, λ_2) . With values of (λ_1, λ_2) now fixed we resolve (2.5) on the whole training set and determine which c_j 's are (close to) zero.

We eliminate the corresponding features and then solve once more a simple regression problem (i.e., a problem with just the first term in (2.5)) on the whole training set using only the features that were not eliminated. This yields a final vector \mathbf{c} and a prediction

$$\hat{y}_i(t) = \mathbf{c}'\boldsymbol{\phi}(\mathbf{x}_i(t)), \quad (2.6)$$

for each patient i and time t . The reason for performing a simple regression with the non-eliminated features is that (2.5) naturally biases coefficients c_j towards smaller values.

For performance evaluation, we use two performance metrics. The first is the *Root Mean Square Error (RSME)*, which for patient i is defined as

$$\text{RMSE}_i = \sqrt{\frac{1}{T_i} \sum_{t=t_i^1}^{t_i^{T_i}} (\hat{y}_i(t) - y_i(t))^2}, \quad (2.7)$$

where $t_i^1, \dots, t_i^{T_i}$ are the time instants at which we make a PTT prediction for patient i .¹ We define RMSE for the whole population of patients as the average per patient RMSE, i.e., $\text{RMSE} = \frac{1}{N_t} \sum_{i=1}^{N_t} \text{RMSE}_i$, where N_t is the number of patients in the test set. We also define σ_{RMSE} to be the standard deviation of the RMSE_i values, which captures the variability of RMSE_i from RMSE.

To capture a notion of “relative” error we also compute the *Normalized Root Mean Square Error (NRMSE)* defined for each patient i as

$$\text{NRMSE}_i = \sqrt{\frac{1}{T_i} \sum_{t=t_i^1}^{t_i^{T_i}} [(\hat{y}_i(t) - y_i(t))/y_i(t)]^2}. \quad (2.8)$$

As with the RMSE, we define the population-wide NRMSE as the average of NRMSE_i over the patients and σ_{NRMSE} as the standard deviation of the NRMSE_i values.

Starting with the prediction in (2.6), we explored two instances of polynomial

¹We note that since we use a history of $M - 1$ measurements for the prediction at time t , we only make predictions starting from the M th available measurement for each patient.

kernels corresponding to $d = 1$ and $d = 2$ and also varied the length M which determines how far back in time we go to define the feature vector. We used again 6-fold cross-validation over the training set and plot the results in Figures 2·2 for kernel parameters $d = 1$ and $d = 2$, respectively. From the figures it follows that for each case there is an optimal value for M , which is $M = 5$ for $d = 1$ and $M = 4$ for $d = 2$. This make intuitive sense. As we increase M , the performance initially improves since the predictors provide more information but after some value of M the performance deteriorates due to overfitting.

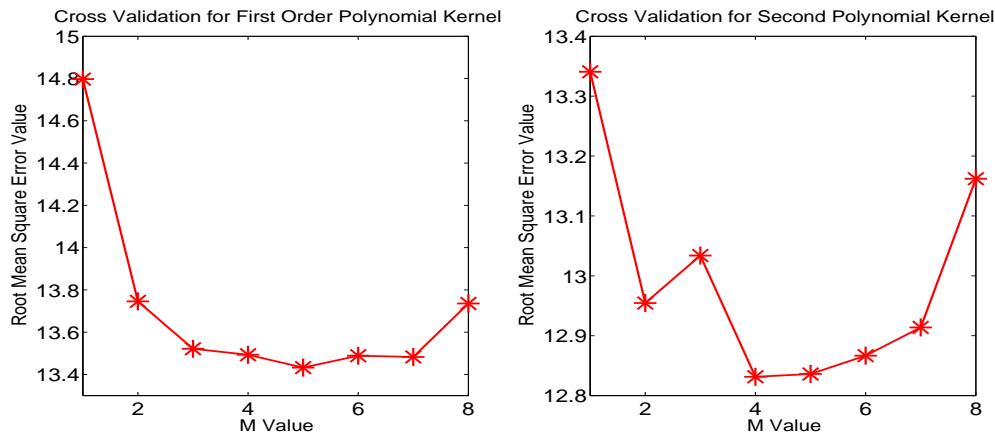


Figure 2·2: Performance of the regularized polynomial regression for $d = 1$ and $d = 2$ for different numbers M of time series inputs in the test set.

Finally, and for these two kernels and corresponding optimal M values, we evaluate the performance of the prediction given by (2.6) on the test set, that is, the $\frac{1}{3}$ rd of the data isolated from the training set in the beginning. Table 2.1 reports the results. It follows that the linear kernel performs better. We should note that the quadratic kernel achieves slightly less RMSE than the linear kernel on the training set (cf. Figs. 2·2) but apparently it overfits and the performance is not as good in the test set.

In summary, a key advantage of the regression method we presented in this section

Table 2.1: Performance of the regularized polynomial regression (test set).

	$d = 1$	$d = 2$
RMSE	11.54	18.69
σ_{NRMSE}	4.04	5.37
NRMSE	21.44%	36.58%
σ_{NRMSE}	6.33%	9.82%

is that it does not require any understanding of the mechanism by which bivalirudin affects PTT values for patients. The next section explores whether a specific model can provide better performance.

2.3 An Explicit Dynamic System Model

This section introduces a *Multiple Input Single Output (MISO)* dynamic system model that attempts to explicitly account for the way bivalirudin affects PTT values in patients.

2.3.1 The Model

The dynamic system model is shown in Fig. 2-3. It seeks to represent how bivalirudin acts in a single generic patient and uses as inputs the exact same physiological variables used as predictors in Section 2.2.3. In the absence of established quantitative models to relate most of the input variables to PTT, a simple linear model was assumed. Moreover, most of these variables change slowly over time and were measured infrequently (e.g., once per day) in the data we analyzed. As a result, they were modeled without a dynamic component.

To make things more precise, in this dynamic system, there are 11 inputs which are denoted by $u_i(t)$, $i = 1, \dots, 11$, and correspond to the physiological variables of Section 2.2. We note that for ease of notation, we will suppress the patient identifier in

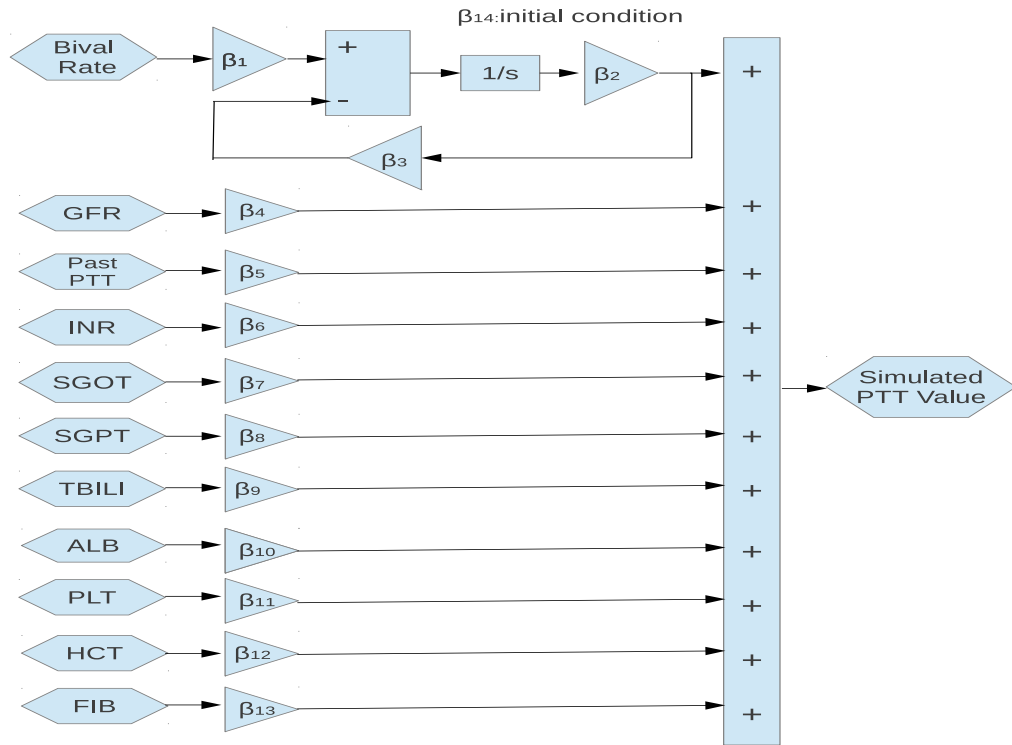


Figure 2-3: The single-state linear model includes a gain, $1/\beta_3$, representing the elimination time constant of bivalirudin from the body. The constant β_2 provides for the translation from serum concentration to the site-effect (PTT).

our generic model description. The input $u_1(t)$, in particular, denotes the bivalirudin infusion rate, and the remaining inputs correspond to the physiological variables 2–11 detailed earlier. These inputs capture the patient’s indicators of renal and liver function. There is only one output – the PTT value – which is denoted by $y(t)$. There is also a single state variable denoted by $x(t)$. The model has 14 unknown parameters: 13 of which correspond to the various gains and are denoted by β_i , $i = 1, \dots, 13$. The initial condition of the system is the 14th unknown parameter and is denoted by $x(0)$. We will refer to $\mathbf{z} = (\beta_1, \dots, \beta_{13}, x(0))$ as the parameter vector. In order to be consistent with the medical intuition and avoid the overfitting issues, upper and

lower bounds will be introduced for some of the parameters.

Let us denote $\mathbf{u}(t) = (u_1(t), \dots, u_{13}(t))$. The system dynamics can be expressed as follows:

$$\begin{aligned}\dot{x}(t) &= \mathbf{A}x(t) + \mathbf{B}\mathbf{u}(t), \\ y(t) &= \mathbf{C}x(t) + \mathbf{D}\mathbf{u}(t),\end{aligned}\tag{2.9}$$

where $\mathbf{A} = -\beta_3$, $\mathbf{B} = [\beta_1 \ 0 \ \dots \ 0]$, $\mathbf{C} = \beta_2$, and $\mathbf{D} = [0 \ \beta_4 \ \dots \ \beta_{13}]$. Clearly, this is a *Linear Time Invariant (LTI)* dynamic system. The challenge is that we only have sampled input, $\mathbf{u}(t)$, and observation values, $y(t)$, at certain times t for each patient. It is therefore needed to translate the continuous-time system dynamics to discrete-time dynamics.

Using a standard conversion from continuous to discrete time dynamics in LTI systems (see e.g., (Friedland, 2012)) we can write

$$x(t + \tau) = e^{\mathbf{A}\tau}x(t) + \int_t^{t+\tau} e^{\mathbf{A}(t+\tau-s)}\mathbf{B}\mathbf{u}(s)ds,\tag{2.10}$$

where in our case $e^{\tau\mathbf{A}} = e^{-\beta_3\tau}$. Assuming $\mathbf{u}(s) = \mathbf{u}(t)$ for $s \in [t, t + \tau]$ and after some algebra we arrive at the following discrete-time dynamics:

$$\begin{aligned}x(t + \tau) &= e^{-\beta_3\tau}x(t) + \frac{\beta_1}{\beta_3}(1 - e^{-\beta_3\tau})u_1(t), \\ y(t) &= \beta_2x(t) + \sum_{i=4}^{13} \beta_i u_{i-2}(t).\end{aligned}\tag{2.11}$$

These equations characterize a discrete-time LTI system for which we have a history of sampled input and output values. Next we describe how the training set can be used to identify the unknown parameters, namely, the initial condition $x(0)$ and the parameters β_i , $i = 1, \dots, 13$.

2.3.2 Parameter Identification

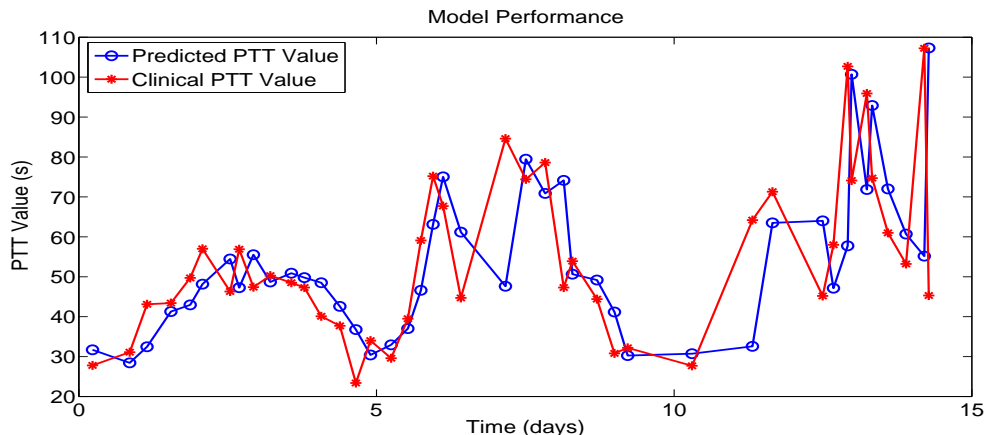


Figure 2-4: Illustrating the performance of the dynamic system model of Section 2.3 for a particular patient. The blue “o” represent predicted PTT values from our model and the red “*” represent actual measured values.

As we did in Section 2.2, we randomly split our data set of 233 post-cardiac surgical ICU patients into a training set corresponding to 2/3 of the total (155 patients) and a test set corresponding to 1/3 of the total (78 patients). We will use the former to identify the unknown system parameters and the latter to evaluate the performance of the resulting model.

Let us use a subscript j to denote the model primitives, i.e., the state $x_j(t)$, output $y_j(t)$, and inputs $\mathbf{u}_j(t)$ for each patient $j = 1, \dots, N$, where N denotes the number of patients in the training set. To distinguish between measurements of $y_j(t)$ and predictions based on the system dynamics (cf. (2.11)) we use $y_j(t)$ for the former and $\hat{y}_j(t)$ for the latter. Suppose for each patient j we have T_j measurements at times $t_j^1, \dots, t_j^{T_j}$, where we adopt the convention $t_j^0 = 0$ for all j . Using the discrete-time system dynamics from (2.11), we formulate the following nonlinear optimization

problem in order to identify the unknown system parameters:

$$\begin{aligned}
\min \quad & \sum_{j=1}^N \sum_{t=t_j^1}^{T_j} (\hat{y}_j(t) - y_j(t))^2 \\
\text{s.t.} \quad & x_j(t_j^n) = e^{-\beta_3(t_j^n - t_j^{n-1})} x_j(t_j^{n-1}) \\
& \quad + \frac{\beta_1}{\beta_3} (1 - e^{-\beta_3(t_j^n - t_j^{n-1})}) u_{j,1}(t_j^{n-1}), \\
& \quad \forall j = 1, \dots, N; n = 1, \dots, T_j, \\
& \hat{y}_j(t_j^n) = \beta_2 x_j(t_j^n) + \sum_{i=4}^{13} \beta_i u_{j,(i-2)}(t_j^n), \\
& \quad \forall j = 1, \dots, N; n = 1, \dots, T_j, \\
& x(0) \text{ is free,} \\
& \beta_m \leq 0, m = 1, 2, 3, 5, 7, 8, 9, \\
& \beta_n \geq 0, n = 4, 6, 10, 11, 12, 13,
\end{aligned} \tag{2.12}$$

where the decision variables are $x(0)(= x(t_j^0))$ for all j) and the parameters β_i , $i = 1, \dots, 13$. One can easily substitute the expressions from the constraints into the objective function and obtain a simple nonlinear optimization problem with the only remaining constraints being bounds on some decision values. Using counterexamples, it can be shown that the objective function obtained in this manner is not convex in the decision variables.

Although a lot of methods exist for nonlinear optimization problems, we used the *Broyden-Fletcher-Goldfarb-Shanno (BFGS) method* (Bertsekas and Bertsekas, 1999), which is considered the most effective general purpose quasi-Newton method. Quasi-Newton methods are gradient methods of the form

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \alpha^k \mathbf{d}^k, \quad \mathbf{d}^k = -\mathbf{D}^k \nabla f(\mathbf{z}^k),$$

where $f(\cdot)$ denotes the objective function, \mathbf{z}^k the decision variables at the k th iteration of the method, α^k is the stepsize at the k th iteration, and \mathbf{D}^k is a positive definite scaling matrix that scales the gradient at the k th iteration. Rather than determining \mathbf{D}^k by computing a Hessian and inverting it, which is computationally expensive, quasi-Newton methods recursively estimate the inverse of the Hessian by using successive iterates of \mathbf{z}^k and $\nabla f(\mathbf{z}^k)$.

We solved (2.12) on the training set using BFGS and obtained the optimal solution shown in Table 2.2. To avoid getting stuck at shallow local minima, which is possible in the absence of convexity, we used a multi-start approach, namely, we started BFGS from multiple randomly selected initial points and selected the best local minimum we obtained. Using the optimal parameters from Table 2.2, we evaluated the performance of the prediction on the test set and obtained the results shown in Table 2.3. It can be seen that the performance is somewhat worse than the one obtained with polynomial regression. This suggests that the simple model we devised in this section does a decent job of capturing the key effect of bivalirudin. It uses just a single state and, as it can be inferred from (2.11), the prediction at time t depends on the inputs $\mathbf{u}(t)$ at t , and the state-input pair, $x(t-\tau)$ and $\mathbf{u}(t-\tau)$ at the previous time instant $t-\tau$. In contrast, the best result with polynomial regression was obtained with a linear kernel but a history of inputs going back 4 consecutive time instants. One can of course devise a more complicated model that uses more memory, we elected not to pursue this direction in the interest of simplicity and taking into account that the potential difference in performance that is to be gained would not be big.

Table 2.2: Optimal parameters values.

index	name	value
1	k_dist	57.39
2	k_coag	59.74
3	1/T	60.00
4	kGFR	-2.65×10^{-7}
5	kINR	0.77
6	kPTT	3.13
7	kSGOT	4.66×10^{-7}
8	kSGPT	2.70×10^{-4}
9	kTBILI	0.21
10	kALB	-4.23×10^{-6}
11	kPLT	-7.59×10^{-8}
12	kHCT	-4.36×10^{-7}
13	kFIB	-4.16×10^{-8}
14	$x(0)$	0.61

To further illustrate how well the predictor matches the measured values, we plot

Table 2.3: Performance of the dynamic system model.

Metric	Value
RMSE	14.65
σ_{RMSE}	6.2
NRMSE	26.42%
σ_{NRMSE}	11.04%

in Fig. 2.4 predicted and actual PTT values over time for a particular randomly selected patient.

We note that the solution shown in Table 2.2 is obtained from a training set containing data from many patients and thus corresponds to optimal “population-wide” parameters. The signs of these parameters are consistent with medical intuition. It can be seen that some parameter values are quite small, yet, their presence improves the dynamic model performance; eliminating them will lead to worse performance. In addition to providing a simple model that explicitly models the effect of bivalirudin, the work in this section has additional benefits. As we will see next, having an explicit model allows us to adapt model parameters to better fit each individual patient.

2.4 An Adaptive Model: Extended Kalman Filter

In this section we focus on an arbitrary individual patient and seek a method to adapt the parameters of the model we proposed in Section 2.3 in order to better fit this particular patient. To that end, we view the model parameters as the “states” of a system and the output $y(t)$ as a nonlinear function of that state. We devise a recursive method to estimate the state. Due to the nonlinearity of $y(t)$ we use the *Extended Kalman Filter (EKF)* (see e.g., (Welch and Bishop, 2006)).

Let us denote the state of the system by $\mathbf{z} = (\beta_1, \dots, \beta_{13}, x(0))$, which are exactly the model parameters we want to estimate. We assume we have measurements of the

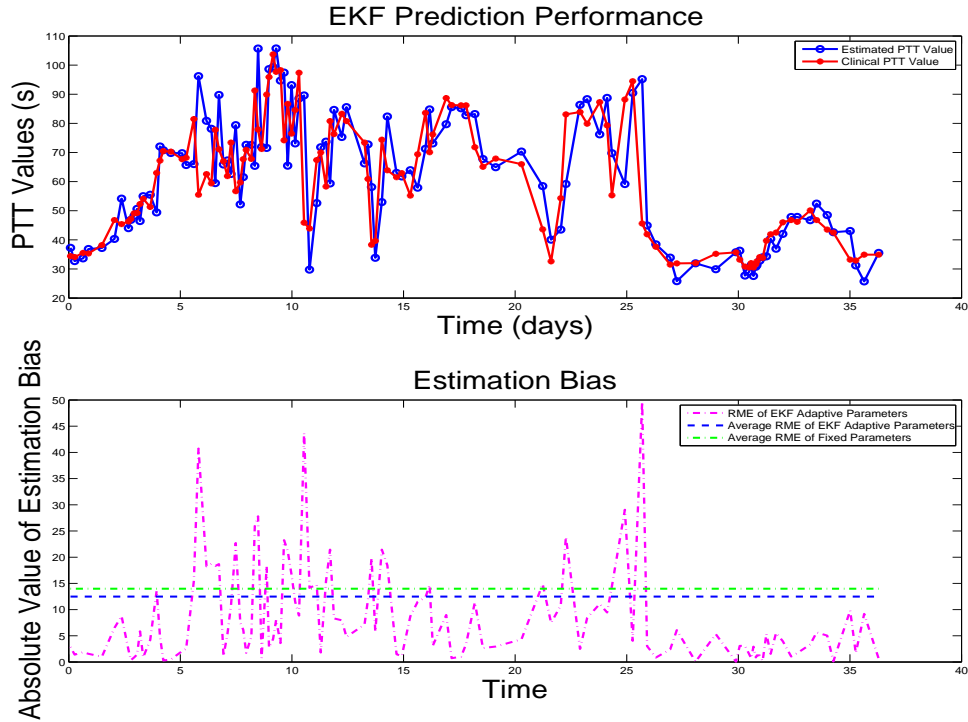


Figure 2-5: Illustrating the performance of the EKF algorithm for a particular patient. The blue “o” represent predicted PTT values from our model and the red “*” represent actual measured values. The top figure plots estimated and measured PTT and the bottom figure plots the running RMSE at each step. Additionally, blue dot line represents the average RME of EKF adaptive method while green dot line represents the fixed parameters solve by previous section. It is obvious that EKF adaptive methods fits the individual model better.

inputs $\mathbf{u}(t)$ and the PTT values $y(t)$ over many time instants. We will index these time instants by k , with $k = 0$ corresponding to $t = 0$ and $k = 1, 2, \dots, T$ corresponding to the time instants t^1, \dots, t^T at which we have measurements. (Notice we use the same notation as in Section 2.3 but suppress the index j used there to identify a patient.) We view the state \mathbf{z} as being invariant over time and not affected by noise, while the output y depends on \mathbf{z} but is subject to some noise due to both measurement noise

1. Initialization:

- (a) Set $\hat{\mathbf{z}}_{0|0} = (\beta_1, \dots, \beta_{13}, x(0))$ using the values obtained by solving the population-wide problem (2.12), and
- (b) set $\mathbf{P}_{0|0} = \mathbf{I}$.

2. Predict:

- (a) $\hat{\mathbf{z}}_{k+1|k} = \hat{\mathbf{z}}_{k|k}$;
- (b) $\mathbf{P}_{k+1|k} = \mathbf{A}\mathbf{P}_{k|k}\mathbf{A}'$.

3. Update:

- (a) $\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k}\mathbf{C}'(\mathbf{C}\mathbf{P}_{k+1|k}\mathbf{C}' + \sigma^2)^{-1}$;
 - (b) $\hat{\mathbf{z}}_{k+1|k+1} = \hat{\mathbf{z}}_{k+1|k} + \mathbf{K}_{k+1}(y_{k+1} - h(\hat{\mathbf{z}}_{k+1|k}))$;
 - (c) $\mathbf{P}_{k+1|k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{C})\mathbf{P}_{k+1|k}$.
-

Figure 2-6: The EKF algorithm for recursively estimating model parameters for an individual patient.

and model error. We can therefore write the (discrete) system dynamics as:

$$\begin{aligned}\mathbf{z}_k &= \mathbf{z}_{k-1}, \\ y_k &= h(\mathbf{z}_k) + \nu_k.\end{aligned}\tag{2.13}$$

In the above, $h(\cdot)$ is a known nonlinear function that expresses y_k as function of the parameter vector \mathbf{z}_k and $\mathbf{u}_k, \mathbf{u}_{k-1}, \dots, \mathbf{u}_0, x_0$ as specified by the dynamics in (2.11). The random variable ν_k represents the noise and we assume it is i.i.d. over time, zero mean Gaussian, with variance σ^2 , that is $\nu_k \sim N(0, \sigma^2)$ for all k .

Let $\mathbf{A} = \mathbf{I}$, and $\mathbf{C} = \nabla h'(\mathbf{z}_k)$. The EKF algorithm is given in Fig. 2-6, where “hat” denotes the estimate, \mathbf{P} the error covariance, and \mathbf{K} the Kalman gain.

To demonstrate the effect of this algorithm, we randomly selected a patient who has adequate sample data and applied the EKF algorithm of Fig. 2-6 (using $\sigma^2 =$

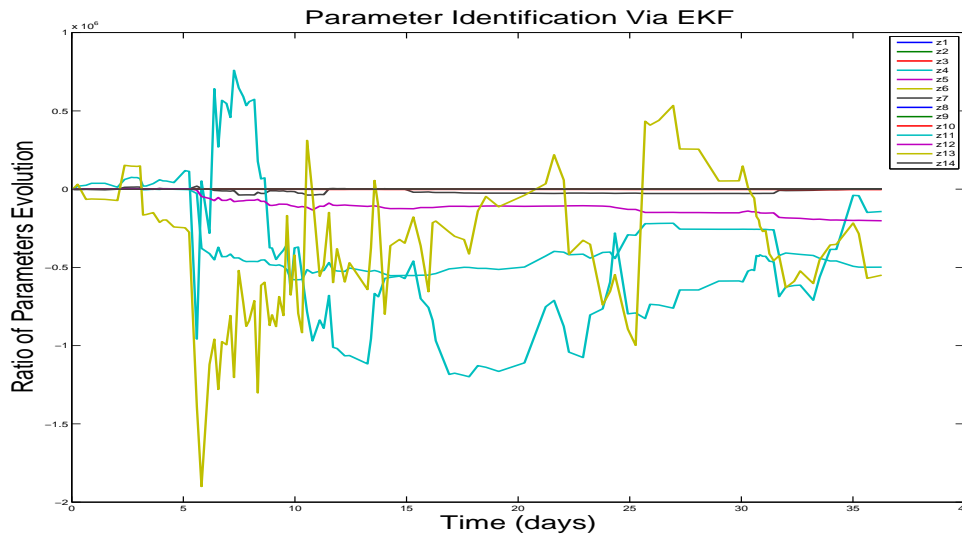


Figure 2-7: Illustrating the evolution of model parameter values $\mathbf{z} = (\beta_1, \dots, \beta_{13}, x(0))$ during the course of EKF algorithm.

0.006). The results are shown in Fig. 2-5. It is evident that after some initial steps, the algorithm “learns” better values for the model parameters than the ones in the population-wide model and produces better predictions for this particular patient. The model parameter values for the same patient during the course of the EKF algorithm are shown in Fig. 2-7. It can be seen that they do “adapt” over time from the initial population-wide values to values that are more appropriate for this patient.

To test the performance of the algorithm on a larger set of patients, we selected patients with enough samples; in particular, more than 60 data points. There are 19 out of 78 patients in our set with more than 60 data points. We applied the EKF (using again $\sigma^2 = 0.006$) with the optimal population wide parameter values as our initial point. By doing so, “warming process” of the EKF can be reduced significantly and we have a decent model even at the early steps; the latter being important for patient safety. In contrast, if we use an arbitrary initial point, the EKF takes at least 25 steps for the parameter values to stabilize. Moreover, the model error in these early steps becomes quite large which is unacceptable in an eventual use of our

system in clinical practice.

Table 2.4 reports average results from the EKF algorithm applied to the patients in the reduced test set of 19 patients described in the previous paragraph. RMSE and NRMSE are computed on a per-patient basis and then averaged over these patients. By testing the same subset of the original test set with EKF and the dynamic model, we notice that NRMSE improves by 22.23% (4.73% in 21.28%), which is significant. More details are shown in Table 2.4. This confirms the significant individual patient variability in response to bivalirudin which has been only empirically observed.

Further more, the EKF algorithm leads to improved performance even in comparison to the model-free method of Section 2.2 which, as we have discussed, uses a longer history of past measurements. In particular, NRMSE improves by 23% while the variance stays about the same.

Table 2.4: EKF vs. dynamic model.

Metric	EKF	Dynamic
RMSE	8.61	11.36
σ_{RMSE}	3.28	3.29
NRMSE	16.55%	21.28%
σ_{NRMSE}	6.90%	5.8%

2.5 Conclusions

We have developed two main approaches to predict the effect of bivalirudin in cardiac surgical patients. The first approach is *model-free* and leverages regularized regression. We find that a linear kernel performs best and that the corresponding set of predictors uses a collection of physiological variables characterizing bivalirudin infusion rate, several coagulation indicators, and indicators of renal and liver function sampled over a set of four time instances before the time at which a PTT prediction is sought.

Namely, this model-free method uses a history of four prior feature vectors to make a PTT prediction.

Our second approach is *model-based* and constructs a specific model that captures how bivalirudin affects PTT values. The model uses a shorter history of prior feature vectors than the model-free approach in order to arrive at a prediction. Model parameter identification is done by solving a nonlinear optimization problem over a training set. The model-based approach produces somewhat worse performance than the model-free one, which is understandable given the shorter history used.

The model-based approach, however, enables the development of an adaptive extended Kalman filtering algorithm that can adapt model parameters to individual patients. This approach produces best average performance and a variance which is almost identical to the model-free method. Compared to the population-wide optimal dynamic model, NRMSE is reduced by 22.23%. This shows that patient-specific models have significant advantages over population-wide models.

The mathematical models and prediction approaches described in this study may provide a better reference to guide the optimal therapy in cardiac patients in need of bivalirudin. In addition, such mathematical ideas and methods may be useful to test medication dosing strategies and may provide a mathematical mechanism for development and testing of nomograms. In the next chapter, we will introduce three different adaptive control laws to obtain the optimal dosage.

Chapter 3

Adaptive Control of Bivalirudin in the Cardiac Intensive Care

3.1 Introduction

In previous chapter, we developed two methods for predicting future PTT values given past infusion rates and the patient's renal and liver function characteristics. We proposed an explicit dynamic system model which was shown to produce quite accurate results when tested against actual patient data. In this chapter, we pursue what we view as the natural next step. Leveraging the dynamic system model from the previous chapter, we seek to synthesize controllers that can regulate the infusion rate to drive PTT within a desirable range. Other methodologies such as expert systems have also been used for controlling some drugs (Held and Roy, 1995). We develop two types of control laws. First, assuming that a dynamic system model that can predict PTT given dosage is completely characterized, we develop a *Model Reference Control (MRC)* law. Model parameters, however, may be viewed as not known with certainty, which is due to modeling errors and inter- or intra-individual variability. To overcome this problem, we develop an indirect *Model Reference Adaptive Control (MRAC)* law that identifies the model parameters first and then adapts the controller in real-time. Furthermore, we develop a direct Model Reference Adaptive Control law that adapts the controller directly without estimating model parameters first, which is more efficient. For each case, we present analytical and numerical evidence showing

that the controllers do drive PTT to the desirable range. Our numerical validation is in fact done using actual patient data from the Brigham and Women’s Hospital – a large hospital in the Boston area.

The remainder of the chapter is organized as follows. Section 3.2 presents the dynamic system model that predicts the effect of bivalirudin given dosage and patient physiological information. Section 3.3 presents the proposed control schemes; Section 3.3.1 develops the MRC law whereas Section 3.3.2 develops the indirect MRAC law based on the patient model but with unknown parameters. Section 3.3.3 develops the direct MRAC, which is more efficient in adapting the controller. Finally, concluding remarks appear in Section 3.4.

3.2 Dynamic System Model Formulation

3.2.1 The Model

This section presents a *Multiple Input Single Output (MISO)* dynamic system model that attempts to explicitly account for the way bivalirudin affects PTT in patients. The model was developed and validated in Chapter 2, Section 2.3; it is presented here briefly to establish the notation and to set the stage for the control schemes of Section 3.3.

The key quantity (response) we would like to predict is the PTT at each time t . The dynamic model structure is shown in Fig. 2-3. There are 11 inputs which are denoted by $u_i(t)$, $i = 1, \dots, 11$ and correspond to important physiological variables used as predictors. More specifically, inputs $u_1(t), \dots, u_{11}(t)$ respectively correspond to:

1. **Bival rate** (mg/kg/h): the weight-based bivalirudin injection rate.
2. **GFR** (mL/min): the glomerular filtration rate.

3. **PTT**(s): last measured PTT value.
4. **INR**(unit-less): last measured INR value.
5. **SGOT** (Units/L): the Serum Glutamic Oxaloacetic Transaminase.
6. **SGPT** (Units/L): the Serum Glutamic Pyruvic Transaminase.
7. **TBILI** (mg/dL): total bilirubin.
8. **ALB** (g/L): Albumin.
9. **PLT** (K/mcL): Platelet count.
10. **HCT** (%): Hematocrit.
11. **FIB** (mg/dL): Fibrinogen.

More detailed description of these physiological variables can be found in Chapter 2, Section 2.3.

The model of Fig. 2-3 has a single output –the PTT value– which is denoted by $y(t)$. There is also a single state variable denoted by $x(t)$. Overall there are 14 unknown parameters: 13 of which correspond to the various gains and are denoted by β_i , $i = 1, \dots, 13$. The initial condition of the system is the 14th unknown parameter and is denoted by $x(0)$ (β_{14}). The system dynamics are:

$$\begin{aligned}\dot{x}(t) &= \mathbf{A}x(t) + \mathbf{B}u(t), \\ y(t) &= \mathbf{C}x(t) + \mathbf{D}u(t),\end{aligned}\tag{3.1}$$

where $\mathbf{A} = -\beta_3$, $\mathbf{B} = [\beta_1 \ 0 \ \dots \ 0]$, $\mathbf{C} = \beta_2$, and $\mathbf{D} = [0 \ \beta_4 \ \dots \ \beta_{13}]$. Clearly, this is a *Linear Time Invariant (LTI)* dynamic system. The challenge is that we do not know the model parameters and we only have non-uniform sampled inputs $\mathbf{u}(t)$, and clinical observation values $y(t)$ at certain times t for each patient. It is therefore necessary

to translate the continuous-time system dynamics to discrete-time dynamics before proceeding with parameter identification.

3.2.2 Parameter Identification

Given the highly non-uniform sampled data, two methods were introduced to identify model parameters in Chapter 2, Section 2.3.2. First, after converting to discrete-time dynamics, we formulated the parameter identification problem as the nonlinear optimization problem of minimizing some metric of fitness to a training set of sampled data. This yielded a population-wide model in the sense that its parameters produced the best fit with the sampled data. Furthermore, and to accommodate variability across patients, we used a recursive estimation method (Extended Kalman Filter) to estimate the parameter values that best fit a given individual patient in real-time.

3.3 Bivalirudin Control Scheme

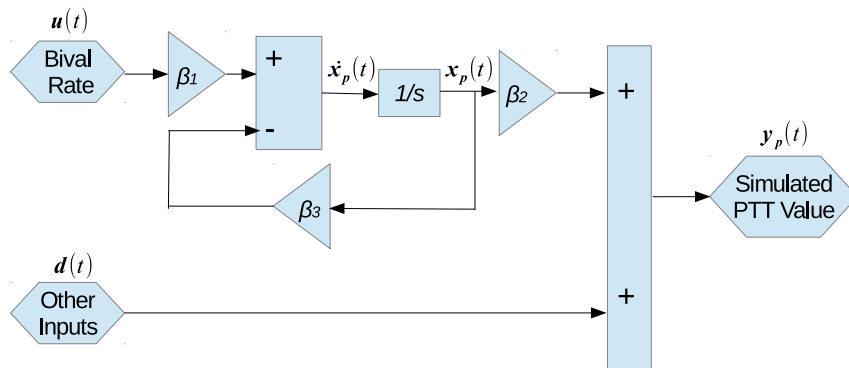


Figure 3-1: In this dynamic model, the bivalirudin infusion rate $u(t)$ is the only controllable input. $d(t)$ is the linear combination of the rest of the inputs.

We now turn to our goal of devising a proper controller to keep the PTT value in the range of 60s-80s. According to clinical experience, this range is safe and optimal

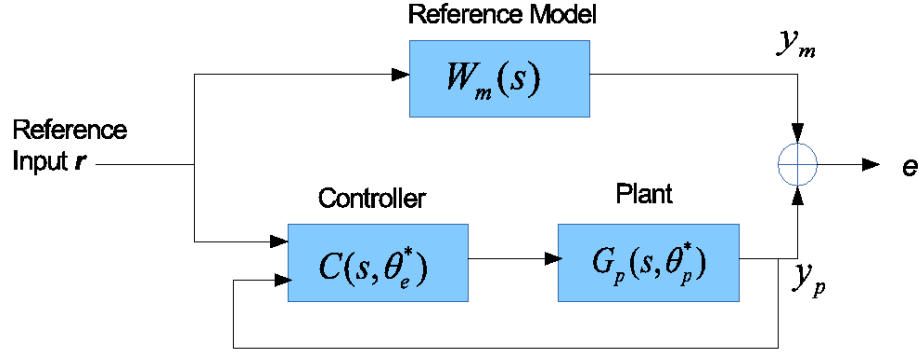


Figure 3.2: Model Reference Adaptive Control (MRAC) structure.

for cardiac surgery patients. For the system we have defined in Eq. (3.1), note that the only controllable part is the bivalirudin infusion rate. The rest of the inputs are indicators of patients' liver (SGOT, SGPT, TBILI, ALB) and renal function (GFR) and include some metrics related to the blood (PLT, HCT, FIB). Arguably, these variables are not immediately affected by the drug but change over a longer time-scale than the one we focus on for controlling PTT through the infusion of bivalirudin. Therefore, we consider them as non-controllable and aggregate them in a variable $d(t)$ which is their linear combination with the appropriate gains $\beta_4, \dots, \beta_{13}$ (see Fig. 3.1).

Ideally, we want to design a reference model which can generate sufficient but safe PTT values driven by a reference input signal. Based on the output of the reference model, we want to drive our system to perform similarly to the reference model by a proper control signal. Motivated by this, we adopt the so called continuous-time MRAC scheme. Fig. 3.2 shows the general structure of this control law. $W_m(s)$ denotes an ideal reference transfer function that can generate the desired reference output signal. The controllable system is represented by $G_p(s, \theta_p^*)$, where θ_p^* is a parameter vector. The objective is to design a controller $C(s, \theta_e^*)$, parameterized by θ_e^* , to generate the proper control signals that can drive the controllable system to track the reference output values.

Our first controller is an MRC law that is designed assuming that the system parameters $\boldsymbol{\theta}_p^*$ are known.

3.3.1 Model Reference Control (MRC)

By observing the system in Fig. 3-1, we can rewrite the dynamics of a particular patient as

$$\dot{x}_p(t) = -\beta_3 x_p(t) + \beta_1 u(t), \quad (3.2)$$

$$y_p(t) = \beta_2 x_p(t) + d(t), \quad (3.3)$$

where we use $u(t)$, $x_p(t)$, $y_p(t)$ to denote the input signal (bivalirudin infusion rate), the state variable, and the output signal (PTT), respectively, and where $d(t) = \sum_{i=2}^{11} \beta_{i+2} u_i(t)$. Clinically, since the renal/liver functions and blood metrics of patients do not vary much within a certain period, we do not need to measure these physiological variables continuously and we assume that they are constant within the sample interval. By observing the clinical data, we find that $d(t)$ is a step-wise signal. Therefore, we assume that the first order derivative of $d(t)$ ($\dot{d}(t)$) is 0 within the sample interval. By taking the derivative on both sides of (3.3), using (3.2) to substitute for $\dot{x}_p(t)$, and using (3.3) to eliminate $\dot{x}_p(t)$, we obtain:

$$\dot{y}_p(t) = -\beta_3 y_p(t) + \beta_1 \beta_2 u(t) + \beta_3 d(t). \quad (3.4)$$

In the frequency domain, we have

$$Y_p(s) = \frac{\beta_1 \beta_2 U(s) + \beta_3 D(s)}{s + \beta_3},$$

where the system output $y_p(t)$ ($Y_p(s)$), the input $u(t)$ ($U(s)$), and $d(t)$ ($D(s)$) can be observed. Hence, in our setting, the system transfer function is $G_p(s, \boldsymbol{\theta}_p^*) = Y_p(s)/U(s)$ and it is parameterized by β_1 , β_2 and β_3 . In this case, $\boldsymbol{\theta}_p^* = (\beta_1, \beta_2,$

β_3).

Next, we design a reference transfer function $W_m(s)$. We take $W_m(s)$ to be a first-order LTI system driven by a reference signal $r(t)$:

$$W_m(s) = \frac{Y_m(s)}{R(s)} = \frac{b_m}{s + a_m},$$

which is equivalent to

$$\begin{aligned} \dot{y}_m(t) &= -a_m y_m(t) + b_m r(t), \quad \text{or} \\ Y_m(s) &= \frac{b_m}{s + a_m} R(s), \end{aligned} \quad (3.5)$$

for any bounded piecewise continuous signal $r(t)$, where $a_m > 0$, $b_m \neq 0$ are known. We assume that a_m , b_m , and $r(t)$ are chosen so that $y_m(t)$ represents the desired output signal. This reference system is capable of driving the *PPT* value to the desired level with a short transition time, which is significant in our medical application.

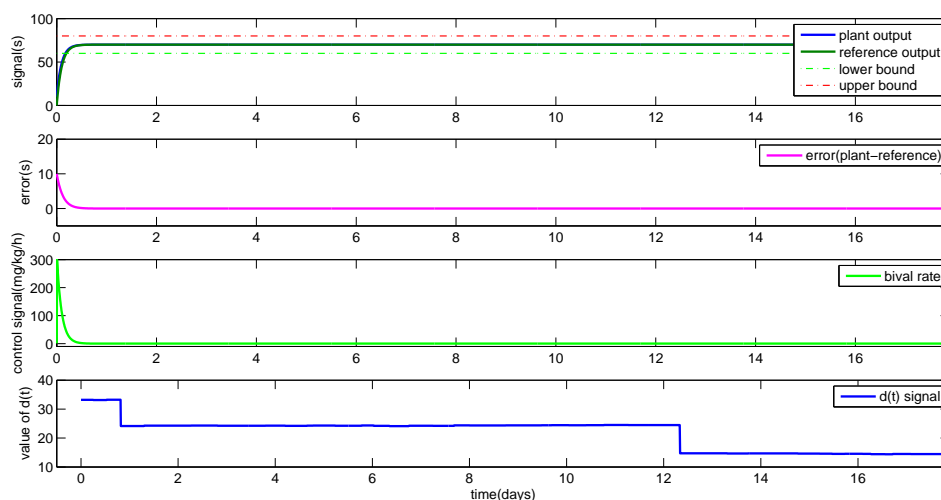


Figure 3-3: The effect of the MRC law derived for and applied to one randomly selected patient.

Before introducing the MRC law, we start with two definitions and a theorem.

Definition 1. A state \mathbf{x}_e is said to be an equilibrium state of the system $\dot{\mathbf{x}} = f(t, \mathbf{x})$, $\mathbf{x}(t_0) = \mathbf{x}_0$, where $\mathbf{x} \in \mathbb{R}^n$, $f : \mathcal{T} \times B(r) \rightarrow \mathbb{R}^n$, $\mathcal{T} = [t_0, \infty)$, $B(r) = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\| < r\}$, if $f(t, \mathbf{x}_e) \equiv 0 \forall t \geq t_0$. We assume that f is such that for every $\mathbf{x}_0 \in B(r)$ and every $t_0 \in [0, \infty)$, the system possesses one and only one solution $\mathbf{x}(t; t_0, \mathbf{x}_0)$.

Definition 2. A equilibrium state \mathbf{x}_e is exponentially stable if there exists an $\alpha > 0$ and for every $\epsilon > 0$ there exists $\delta(\epsilon) > 0$, such that $\|\mathbf{x}(t; t_0, \mathbf{x}_0) - \mathbf{x}_e\| \leq \epsilon e^{-\alpha(t-t_0)}$, $\forall t \geq t_0$ whenever $\|\mathbf{x}_0 - \mathbf{x}_e\| < \delta(\epsilon)$.

Theorem 3.3.1. If we choose $a_m > 0$, $b_m \neq 0$, and $r(t) = C_r$ (constant), the reference model equilibrium state $y_{me} = \frac{b_m C_r}{a_m}$ is exponentially stable.

Proof. The solution to Eq. (3.5) is

$$y_m(t) = \Phi(t, t_0)y_m(t_0) + \int_{t_0}^t \Phi(t, \tau)b_m r(\tau)d\tau,$$

where $\Phi(t, \tau) = e^{-a_m(t-\tau)}$ is the state transition function in this problem. Since $r(t) = C_r$, which is a constant, the solution to (3.5) can be written as

$$y_m(t; t_0, y_m(t_0)) = e^{-a_m(t-t_0)} \left(y_m(t_0) - \frac{b_m C_r}{a_m} \right) + \frac{b_m C_r}{a_m}. \quad (3.6)$$

Equation (3.6) indicates that as $y_m(t; t_0, y_m(t_0)) \rightarrow \frac{b_m C_r}{a_m}$ which is a constant, as $t \rightarrow \infty$. In addition, using Definition 1, it can be easily verified that $y_{me} = \frac{b_m C_r}{a_m}$ is the equilibrium state of our reference system. Furthermore, $|y_m(t; t_0, y_m(t_0)) - y_{me}| = |e^{-a_m(t-t_0)}(y_m(t_0) - y_{me})| = |y_m(t_0) - y_{me}|e^{-a_m(t-t_0)}$, $\forall t \geq t_0$. Therefore, by Definition 2, it follows that the reference model equilibrium state y_{me} is exponentially stable. \square

We will now design a proper controller $u(t)$ such that all signals in the closed-loop system are bounded and the system output $y_p(t)$ tracks the reference model output $y_m(t)$. The control law should be chosen so that the closed-loop plant transfer function from the input $r(t)$ to the output $y_p(t)$ is equal to the reference model transfer function. Motivated by this, we propose the control law

$$C(s, \boldsymbol{\theta}_e^*) = U(s) = -k_1^* Y_P(s) + k_2^* R(s) - k_3^* D(s),$$

or equivalently, in the time domain

$$u^*(t) = -k_1^* y_p(t) + k_2^* r(t) - k_3^* d(t), \quad (3.7)$$

where k_1^* , k_2^* , k_3^* are controller coefficients chosen so that

$$\frac{Y_p(s)}{R(s)} = \frac{b_m}{s + a_m} = \frac{Y_m(s)}{R(s)}. \quad (3.8)$$

Eq. (3.8) is satisfied, if we select

$$k_1^* = -\frac{1}{\beta_1 \beta_2} (\beta_3 - a_m), \quad k_2^* = \frac{b_m}{\beta_1 \beta_2}, \quad k_3^* = \frac{\beta_3}{\beta_1 \beta_2},$$

which yields

$$u(t) = \frac{1}{\beta_1 \beta_2} (\beta_3 - a_m) y_p(t) + \frac{b_m}{\beta_1 \beta_2} r(t) - \frac{\beta_3}{\beta_1 \beta_2} d(t), \quad (3.9)$$

provided of course that $\beta_1, \beta_2, \beta_3 \neq 0$, i.e., the system is controllable. Such a transfer function matching guarantees that $y_p(t) = y_m(t)$, $\forall t \geq t_0$, when $y_p(t_0) = y_m(t_0)$, or $|y_p(t) - y_m(t)| \rightarrow 0$ exponentially fast when $y_m(t_0) \neq y_p(t_0)$ for any bounded reference signal $r(t)$. We also note that, depending on the parameters of some patients, this law may yield a negative control signal which can not be implemented in practice (corresponds to “extraction of bivalirudin” from the patient). In such a case, we need to set a lower threshold of zero for the control signal. The final MRC control signal becomes $\max\{0, u(t)\}$ with $u(t)$ defined as in (3.9).

We test the performance of the MRC on the data set we described in Section 3.2.2. We only use the test set (1/3 of the total) for testing since the remaining training set was used for model parameter identification. As mentioned before, $d(t)$ (the linear combination of physiological variables at time t) is a step-wise signal over time. By applying the parameter identification method outlined in Section 3.2.2, we obtained both population-wide parameter values and individual model parameter values.

We tested the MRC control law on a subset of patients and the results were qualitatively the same in each case. We report results from a randomly selected patient who has identified model parameters and available input data. To that end, we set the reference parameters as $a_m = 10$, $b_m = 700$, $r(t) = 1$. Choosing these values keeps the reference PTT value to be 70s, which is in the middle of the desirable range. We note that these parameter values are simply an example and physicians have the freedom of selecting alternative values depending on the stable value and response time they wish to achieve.

The effect of the MRC law (3.9) on this randomly selected patient is shown in Fig. 3-3. It can be seen that driven by inputs generated by the MRC law, the system output quickly converges to the reference output (top figure). The tracking error ($e(t) = y_p(t) - y_m(t)$) quickly converges to zero and remains at zero (second figure). We also obtain the control signal which corresponds to the bivalirudin infusion rate (third figure). The MRC control law we introduced is robust to the uncontrollable signal $d(t)$ (bottom figure). Although $d(t)$ changes over time, the control signal can adapt and drive the system to track the reference signal closely.

We next evaluate the performance of the MRC scheme on all patients in the test set we described earlier. For performance evaluation, we use three performance metrics. The first one is the Root Mean Square Error (RMSE), which for patient i is defined as

$$\text{RMSE}_i = \sqrt{\frac{1}{T_i} \sum_{t=t_i^1}^{t_i^{T_i}} (y_p^i(t) - y_m^i(t))^2},$$

where $t_i^1, \dots, t_i^{T_i}$ are the time instants at which we adapt the controller for patient i . We define RMSE for the whole population of patients as the average per patient RMSE, i.e., $\text{RMSE} = \frac{1}{N_t} \sum_{i=1}^{N_t} \text{RMSE}_i$, where N_t is the number of patients in the test set. We also define σ_{E_i} to be the standard deviation of the errors $e^i(t) = y_p^i(t) - y_m^i(t)$,

$t = t_i^1, \dots, t_i^{T_i}$, of patient i . Similarly, we define σ_E as the average standard deviation of σ_{Ei} 's, i.e., $\sigma_E = \frac{1}{N_t} \sum_{i=1}^{N_t} \sigma_{Ei}$.

To capture a notion of “relative” error, we also compute the Normalized Root Mean Square Error (NRMSE) defined for each patient i as

$$\text{NRMSE}_i = \sqrt{\frac{1}{T_i} \sum_{t=t_i^1}^{t_i^{T_i}} [(y_p^i(t) - y_m^i(t))/y_m^i(t)]^2}.$$

As with the RMSE, we define the population-wide NRMSE as the average of NRMSE_i 's over the patients. Similarly, we also define σ_{NEi} as the standard deviation of the normalized tracking errors $e^i(t) = (y_p^i(t) - y_m^i(t))/y_m^i(t)$ for patient i , and σ_{NE} as the average of σ_{NEi} 's over the patients.

Furthermore, to illustrate the percentage of PTT outliers which are outside clinically safe bounds, i.e., not in the interval $[y_m(t) - 10, y_m(t) + 10]$ during the transient and not in the interval $[60s, 80s]$ in steady-state, the Risk Percentage (RP) for patient i is defined as

$$\text{RP}_i = \frac{N_i^{\text{risk}}}{T_i},$$

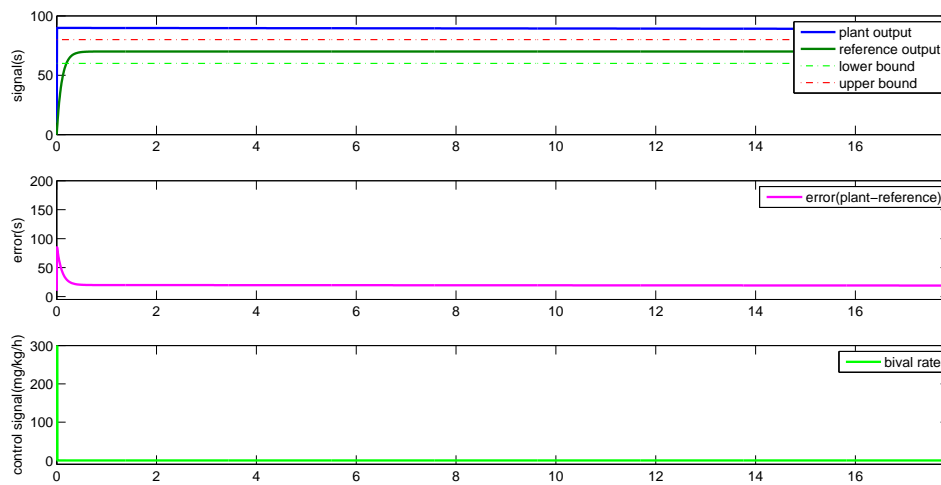
where N_i^{risk} is the number of time instants $t_i^1, \dots, t_i^{T_i}$ at which the PTT value of patient i is outside the safe bounds. Then, RP is defined as the average of RP_i 's over patients. We note that this metric is from a clinical perspective the most important in assessing the efficacy of our methods. Table 3.1 reports the performance of the MRC law for the patients in the test set.

In summary, in the case that model parameters are known, the MRC law tracks the reference signal quite well as demonstrated by the low RMSE and NRMSE. The RP value is zero, which completely assures clinical safety. The corresponding standard deviations for RMSE and NRMSE are small as well.

Table 3.1: Performance of the Model Reference Control (test set).

Metric	Value
RMSE	0.84
σ_E	0.82
NRMSE	1.20%
σ_{NE}	1.17%
RP	0%

3.3.2 Indirect Model Reference Adaptive Control (MRAC)

**Figure 3.4:** The MRC law derived for one patient but applied to another patient.

As we mentioned in the Introduction, there is significant patient variability in the response to bivalirudin. We have already established in section 2.4, that adapting model parameters to individual patients leads to improved performance. This suggests that the model structure is largely accurate but model parameters of an individual patient can deviate from population-wide parameter values.

To better assess the effect of this variability, we test the performance of the MRC law derived using parameter values of a specific patient when applied to another

Table 3.2: Performance of the Inaccurate Model Reference Control (test set).

Metric	Value
RMSE	9.93
σ_E	3.98
NRMSE	14.18%
σ_{NE}	5.68%
RP	61.60%

patient with different model parameters. Fig. 3-4 plots the MRC law performance for such a case. The top figure shows that there exists a large gap between the reference output signal and the system output signal. In addition, the system output is outside the safe range.

We also tested the MRC law derived using parameter values of a specific patient against all patients in the test set. Table 3.2 reports the results. We note that RMSE, NRMSE, and RP are substantially higher (and clinically unacceptable) than those in Table 3.1. The difference of course is due to the fact the results of Table 3.1 are obtained when using the MRC law with the correct model parameters for each patient, whereas Table 3.2 results apply the MRC law with parameters of some patient to another patient. The small values of σ_E and σ_{NE} in Table 3.2 indicate that performance of the MRC law when used with the wrong parameters is consistently poor. This situation should obviously be avoided because overdosing or underdosing is very dangerous for the patients.

To address this important issue, we next develop a method that first estimates the individual model parameters, and then adopts the MRC law we introduced using a certainty equivalence principle (Ioannou and Kosmatopoulos, 2006). Such a control scheme is called indirect *Model Reference Adaptive Control (MRAC)* law.

By adding and subtracting $-a_m y_p(t)$ to (3.4), we can obtain the *State-Space Para-*

metric Model (SSPM):

$$\dot{y}_p(t) = -a_m y_p(t) + (a_m - \beta_3) y_p(t) + \beta_1 \beta_2 u(t) + \beta_3 d(t). \quad (3.10)$$

Based on (3.10), the series-parallel estimation model (Ioannou and Sun, 2012) is given by:

$$\dot{\hat{y}}_p(t) = -a_m \hat{y}_p(t) + (a_m - \hat{\beta}_3(t)) y_p(t) + \hat{\beta}_1(t) \hat{\beta}_2(t) u(t) + \hat{\beta}_3(t) d(t), \quad (3.11)$$

where $\hat{y}_p(t)$ is an estimated value of $y_p(t)$, and $\hat{\beta}_1(t)$, $\hat{\beta}_2(t)$, $\hat{\beta}_3(t)$ are estimates of the system parameters β_1 , β_2 , and β_3 at time t . Note that in (3.11), $y_p(t)$ is treated as an input available for measurement. By using the certainty equivalence principle (cf. (3.9)), we take the control scheme structure to be:

$$u(t) = -k_1(t) y_p(t) + k_2(t) r(t) - k_3(t) d(t), \quad (3.12)$$

where

$$k_1(t) = \frac{a_m - \hat{\beta}_3(t)}{\hat{\beta}_1(t) \hat{\beta}_2(t)}, \quad k_2(t) = \frac{b_m}{\hat{\beta}_1(t) \hat{\beta}_2(t)}, \quad k_3(t) = \frac{\hat{\beta}_3(t)}{\hat{\beta}_1(t) \hat{\beta}_2(t)}.$$

In this problem, we will estimate the product of $\beta_1(t)$ and $\beta_2(t)$ instead of estimating them separately. The model estimation error is $e(t) = y_p(t) - \hat{y}_p(t)$ which implies:

$$\begin{aligned} \dot{e}(t) &= \dot{y}_p(t) - \dot{\hat{y}}_p(t) \\ &= -a_m e(t) + \tilde{\beta}_3(t) (y_p(t) - d(t)) - \tilde{\beta}_{12}(t) u(t), \end{aligned} \quad (3.13)$$

where

$$\tilde{\beta}_3(t) = \hat{\beta}_3(t) - \beta_3, \quad \tilde{\beta}_{12}(t) = \hat{\beta}_1(t) \hat{\beta}_2(t) - \beta_1 \beta_2. \quad (3.14)$$

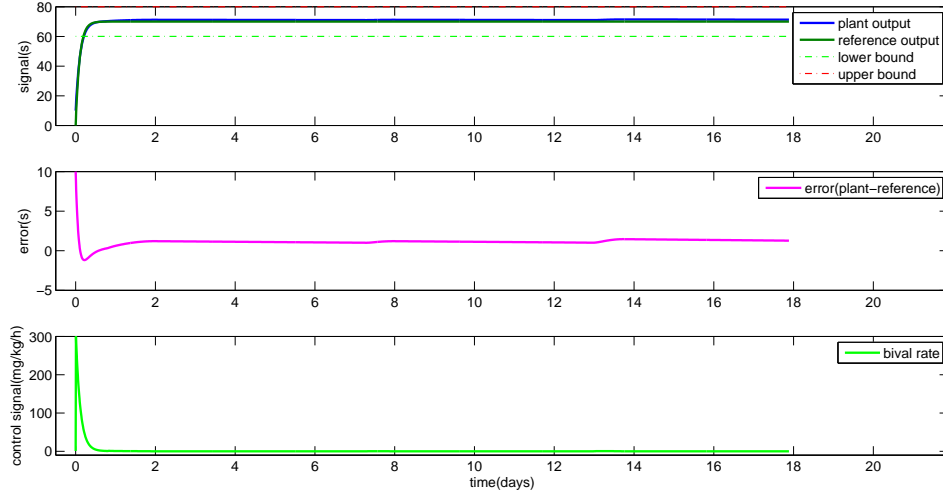


Figure 3-5: The performance of the indirect MRAC law.

We now choose a Lyapunov-like function

$$V(t) = \frac{1}{2}e(t)^2 + \frac{1}{2\gamma_1}\tilde{\beta}_3(t)^2 + \frac{1}{2\gamma_2}\tilde{\beta}_{12}(t)^2, \quad (3.15)$$

which, with $\gamma_1, \gamma_2 > 0$, is non-negative for all t . By taking the derivative on both sides of (3.15) we obtain

$$\begin{aligned} \dot{V}(t) &= e(t)\dot{e}(t) + \frac{1}{\gamma_1}\tilde{\beta}_3(t)\dot{\tilde{\beta}}_3(t) + \frac{1}{\gamma_2}\tilde{\beta}_{12}(t)\dot{\tilde{\beta}}_{12}(t) \\ &= -a_m(e(t))^2 + \tilde{\beta}_3(t) \left[e(t)(y_p(t) - d(t)) + \frac{1}{\gamma_1}\dot{\tilde{\beta}}_3(t) \right] \\ &\quad + \tilde{\beta}_{12}(t) \left[\frac{1}{\gamma_2}\dot{\tilde{\beta}}_{12}(t) - e(t)u(t) \right]. \end{aligned} \quad (3.16)$$

Then, choosing

$$\dot{\tilde{\beta}}_3(t) = -\gamma_1(y_p(t) - d(t))e(t) \quad \text{and} \quad \dot{\tilde{\beta}}_{12}(t) = \gamma_2 e(t)u(t)$$

leads to $\dot{V}(t) = -a_m e(t)^2 \leq 0$. In addition, since $\beta_1\beta_2$ and β_3 are constants, (3.14) implies $\dot{\hat{\beta}}_3(t) = \dot{\tilde{\beta}}_3(t)$ and $\dot{\hat{\beta}}_{12}(t) = d(\hat{\beta}_1(t)\hat{\beta}_2(t))/dt$. It follows that we could estimate

the model parameters by:

$$\begin{aligned}\widehat{\beta}_3(t + \Delta t) &= \widehat{\beta}_3(t) + \dot{\widehat{\beta}}_3(t)\Delta t, \\ \widehat{\beta}_{12}(t + \Delta t) &= \widehat{\beta}_{12}(t) + \frac{d(\widehat{\beta}_1(t)\widehat{\beta}_2(t))}{dt}\Delta t,\end{aligned}\tag{3.17}$$

for small Δt . Then, we can adapt the controller coefficients recursively and control the system in real-time by using (cf. (3.12))

$$u^*(t) = -\frac{a_m - \widehat{\beta}_3(t)}{\widehat{\beta}_{12}(t)}y_p(t) + \frac{b_m}{\widehat{\beta}_{12}(t)}r(t) - \frac{\widehat{\beta}_3(t)}{\widehat{\beta}_{12}(t)}d(t).\tag{3.18}$$

Similarly, as we did earlier for the MRC law, we will use the control $\max\{0, u^*(t)\}$ to avoid negative values.

Theorem 3.3.2. *Under the control law (3.18), the tracking error converges to 0 as $t \rightarrow \infty$.*

Proof. By choosing such control law, $\dot{V}(t) = -a_m e(t)^2 \leq 0, \forall t > t_0$. Since $V(t)$ is bounded from below and non-increasing, it converges to a constant. This implies that $-a_m \int_{t_0}^{\infty} e^2(t)dt = V(\infty) - V(t_0)$ is bounded, which in turn implies that $e(t) \rightarrow 0$ as $t \rightarrow \infty$ according to Barbalat's lemma (Popov and Georgescu, 1973). It also follows that $\dot{\widehat{\beta}}_3(t), \dot{\widehat{\beta}}_{12}(t) \rightarrow 0$ as $t \rightarrow \infty$. \square

One key flaw of the adaptive control law (3.18) is that the boundness of control signal $u(t)$ can not be established unless we show that $k_1(t), k_2(t), k_3(t)$ are all bounded. However, such a control law may generate estimates of $\beta_1\beta_2$ arbitrarily close or even equal to zero, which leads to the uncontrollability of the estimated model and unboundness of $u(t)$. To avoid this issue, we propose a modification to the control law (3.18). One method is to modify the adaptive law for $\widehat{\beta}_{12}(t)$ so that adaptation takes place in a subset of \mathbb{R} which does not include the zero element. We need to use the a priori knowledge of $\beta_1 \geq \beta_1^{lb} > 0$ and $\beta_2 \geq \beta_2^{lb} > 0$ to do the projection:

$$\dot{\widehat{\beta}}_3(t) = -\gamma_1(y_p(t) - d(t))e(t),$$

$$\dot{\tilde{\beta}}_{12}(t) = \begin{cases} \gamma_2 e(t)u(t), & \text{if } |\tilde{\beta}_{12}(t)| > \beta_1^{lb} \beta_2^{lb}, \\ & \text{or } |\tilde{\beta}_{12}(t)| = \beta_1^{lb} \beta_2^{lb} \\ & \text{and } e(t)u(t)\text{sgn}(\tilde{\beta}_{12}(t)) \geq 0, \\ 0, & \text{otherwise.} \end{cases} \quad (3.19)$$

After modifying the adaptive control law, the time derivative of the Lyapunov function becomes:

$$\dot{V}(t) = \begin{cases} -a_m(e(t))^2, & \text{if } |\tilde{\beta}_{12}(t)| > \beta_1^{lb} \beta_2^{lb}, \\ & \text{or } |\tilde{\beta}_{12}(t)| = \beta_1^{lb} \beta_2^{lb} \\ & \text{and } e(t)u(t)\text{sgn}(\tilde{\beta}_{12}(t)) \geq 0, \\ -a_m(e(t))^2 \\ + \tilde{\beta}_{12}(t)e(t)u(t), & \text{if } |\tilde{\beta}_{12}(t)| = \beta_1^{lb} \beta_2^{lb} \\ & \text{and } e(t)u(t)\text{sgn}(\tilde{\beta}_{12}(t)) < 0. \end{cases}$$

Therefore, $\dot{V}(t) \leq -a_m e^2(t) \leq 0, \forall t \geq t_0$.

Using a similar argument as before, it can be shown that by using this modified parameter estimation law (3.19), the tracking error converges to zero driven by a bounded control signal. Additionally, we have shown (cf. Thm. 3.3.1) that the reference output response is exponentially stable, and it follows that the system output can be driven to the stable state exponentially fast.

We next test the indirect MRAC law using the patient data. The parameter values of the reference model are the same as in Section 3.3.1. We choose the population-wide parameter values $\beta_3^* = 7.9 \times 10^{-4}$, and $\beta_1^* \beta_2^* = 4.22$ as initial values of $\hat{\beta}_3(t)$ and $\hat{\beta}_{12}(t)$, respectively. The MRAC adapts based on these estimates in real-time. We also set $\gamma_1 = \gamma_2 = 5 \times 10^{-4}$. The trajectory of the system under the indirect MRAC is shown in Fig. 3.5.

Fig. 3-5 indicates that the system output quickly converges to the reference output and it remains within the desired range (top figure). The tracking error oscillates around zero (middle figure), but it is not as smooth as in Fig. 3-3. This is due to the fact that the indirect MRAC takes some time to estimate the system parameters first and then adapts the controller coefficients. Similarly, we can also obtain the bivalirudin infusion rate (bottom figure). Notice that although $d(t)$ changes over time, the control signal can drive the system to track the reference output signal well. The performance of indirect MRAC for all patients in the test set is reported in Table 3.3.

Table 3.3: Performance of the Indirect Model Reference Adaptive Control (test set).

Metric	Value
RMSE	5.52
σ_E	2.00
NRMSE	7.88%
σ_{NE}	2.86%
RP	2.30%

We note that in this case, the values of our three performance metrics are all higher than those in the Table 3.1. This is again explained by the fact that adaptation of the indirect MRAC law is not very fast due to the parameter estimation step. Yet, RMSE, NRMSE and RP values are now significantly less than those in Table 3.2 and they could be considered acceptable in a clinical setting.

3.3.3 Direct Model Reference Adaptive Control MRAC

In this section, we focus on designing the direct MRAC without estimating the model parameters first. Because the individual model parameters are unknown, we can not apply the MRC law directly. Based on the structure of MRC law, we propose an

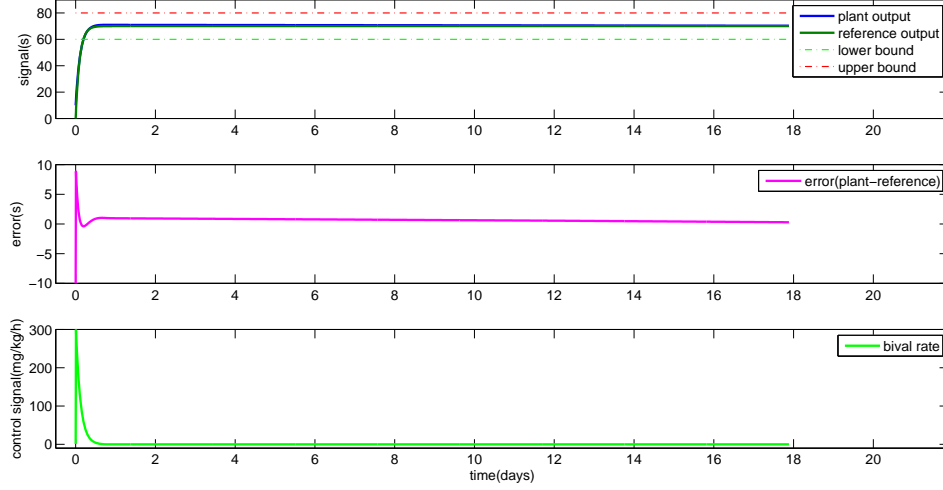


Figure 3-6: The performance of direct MRAC law.

adaptive control law with similar structure:

$$u(t) = -\hat{k}_1(t)y_p(t) + \hat{k}_2(t)r(t) - \hat{k}_3(t)d(t),$$

where $\hat{k}_1(t)$, $\hat{k}_2(t)$, and $\hat{k}_3(t)$ are the estimates of MRC controller coefficients k_1^* , k_2^* , and k_3^* at time t , respectively. We will devise a control law that estimates these coefficients directly.

Consider the error derivative:

$$\begin{aligned} \dot{e}(t) &= \dot{y}_p(t) - \dot{y}_m(t) \\ &= -a_m e(t) + \beta_1 \beta_2 [-\tilde{k}_1(t)y_p(t) + \tilde{k}_2(t)r(t) - \tilde{k}_3(t)d(t)], \end{aligned}$$

where $\tilde{k}_i(t) = \hat{k}_i(t) - k_i^*$, $i = 1, 2, 3$. It follows that $\dot{\hat{k}}_i(t) = \dot{\tilde{k}}_i(t)$, $i = 1, 2, 3$.

To design the controller, consider the Lyapunov-like function:

$$V(t) = \frac{e(t)^2}{2} + \beta_1 \beta_2 \left[\frac{\tilde{k}_1(t)^2}{2\gamma_1} + \frac{\tilde{k}_2(t)^2}{2\gamma_2} + \frac{\tilde{k}_3(t)^2}{2\gamma_3} \right].$$

By taking the derivative, we obtain:

$$\begin{aligned}\dot{V}(t) &= \dot{e}(t)e(t) + \beta_1\beta_2 \left[\frac{\tilde{k}_1(t)\dot{\tilde{k}}_1(t)}{\gamma_1} + \frac{\tilde{k}_2(t)\dot{\tilde{k}}_2(t)}{\gamma_2} + \frac{\tilde{k}_3(t)\dot{\tilde{k}}_3(t)}{\gamma_3} \right] \\ &= -a_m e^2(t) + \beta_1\beta_2 \left[\frac{\tilde{k}_1(t)}{\gamma_1} (\dot{\tilde{k}}_1(t) - e(t)\gamma_1 y_p(t)) \right. \\ &\quad \left. + \frac{\tilde{k}_2(t)}{\gamma_2} (\dot{\tilde{k}}_2(t) + e(t)\gamma_2 r(t)) + \frac{\tilde{k}_3(t)}{\gamma_3} (\dot{\tilde{k}}_3(t) - e(t)\gamma_3 d(t)) \right],\end{aligned}$$

where we have the a priori knowledge that $\beta_1\beta_2 > 0$.

Choosing

$$\begin{aligned}\dot{\tilde{k}}_1(t) &= \gamma_1 e(t) y_p(t), \\ \dot{\tilde{k}}_2(t) &= -\gamma_2 e(t) r(t), \\ \dot{\tilde{k}}_3(t) &= \gamma_3 e(t) d(t),\end{aligned}$$

leads to $\dot{V}(t) = -a_m e(t)^2 \leq 0$. Furthermore, the controller coefficients can be adapted by

$$\widehat{k}_i(t + \Delta t) = \widehat{k}_i(t) + \dot{\widehat{k}}_i(t)\Delta t, \quad i = 1, 2, 3,$$

and our direct MRAC controller becomes

$$u(t) = -\widehat{k}_1(t)y_p(t) + \widehat{k}_2(t)r(t) - \widehat{k}_3(t)d(t). \quad (3.20)$$

As with the previous two controllers, we will use $\max\{0, u(t)\}$ to avoid negative controls.

We establish the following result; we omit the proof because it is similar to the proof of Theorem 3.3.2.

Theorem 3.3.3. *Under the control law (3.20), the tracking error converges to 0 as $t \rightarrow \infty$.*

We know that the reference model output is exponentially stable and the tracking

error converges to zero as t increases, which establishes that the system output can be driven to track the reference output well. To avoid overdosing and underdosing risks that may occur in the time it takes the MRAC to converge to the appropriate controller parameters, we can use as initial estimates of these coefficients the MRC coefficients, namely $\hat{k}_1(0) = 2.37$, $\hat{k}_2(0) = 165.75$, and $\hat{k}_3(0) = 1.87 \times 10^{-4}$, obtained from a system model parametrized with population-wide parameters.

We applied such a direct MRAC law to the same patient used for generating Fig. 3.5. The result is shown in Fig. 3.6 (using $\gamma_1 = \gamma_2 = \gamma_3 = 0.001$). We note that driven by the direct MRAC law, the system output quickly converges to the reference output. Although the system output oscillates around the reference signal, it remains within the desired range (top figure). The tracking error also converges to zero (middle figure). We also plot the bivalirudin infusion rate (bottom figure). Compared to the indirect MRAC, the direct MRAC has similar performance on controlling the PTT. However, the direct MRAC avoids parameter estimation and estimates controller parameters directly.

Table 3.4: Performance of the Direct Model Reference Adaptive Control (test set).

Metric	Value
RMSE	0.80
σ_E	0.80
NRMSE	1.15%
σ_{NE}	1.14%
RP	0.09%

Table 3.5 reports the performances of all three adaptive control laws, including the direct MRAC for the patients in the test set. Compared to the results from the indirect MRAC, the direct MRAC achieves lower values for all performance metrics, i.e., RMSE, NRMSE, σ_E , σ_{NE} and RP. Notice the rather significant decrease of the

Table 3.5: Performances of the three control laws (test set).

Metric	MRC	Indirect MRAC	Direct MRAC
RMSE	0.84	5.52	0.80
σ_E	0.82	2.00	0.80
NRMSE	1.20%	7.88%	1.15%
σ_{NE}	1.17%	2.68%	1.14%
RP	0%	2.30%	0.09%

RP value, which, as we argued, is clinically a top priority. This table validates the fact that the direct MRAC is a more efficient and safer control scheme than the indirect MRAC.

3.4 Conclusions

Based on a specific dynamic system model of bivalirudin acting in cardiac surgical patients, we developed two methods for synthesizing a controller to regulate the bivalirudin infusion rate and induce a PTT within a desirable range. The first method assumes that the model parameters are available and develops a control law that tracks a physician specified reference output signal. Our second method considers patients for which past clinical records are sparse and accurate model parameters are not readily available. It develops an indirect control scheme (indirect MRAC) that first estimates the model parameters and then adapts the corresponding controller based on these estimates. Alternatively, a direct control scheme (direct MRAC) that adapts the controller without estimating the model parameters first is also developed. Testing of these schemes against actual patient data from a hospital, shows that the direct MRAC is more efficient than the indirect version.

The methods we developed can be seen as key steps towards automation of dosage decisions in a hospital setting, which can help eliminate errors and neutralize the inexperience of residents who are currently responsible for these decisions.

Chapter 4

Learning Cellular Objectives from Fluxes by Inverse Optimization

4.1 Introduction

In this chapter, we switch our scope from the human to the cellular metabolism. Specifically, we focus on the genome-scale flux balance models of metabolism. These models provide testable predictions of all metabolic rates in an organism, by assuming that the cell is optimizing a metabolic goal known as the objective function. In practice, however, reaction fluxes of the cells under specific growth conditions are available to be measured with ^{13}C technology (Zamboni et al., 2009), but the primal goals of cells are not necessarily known. Understanding the metabolic network structure can elucidate the cellular metabolic control mechanisms and infer important information regarding an organism's evolution.

A metabolic network is used to describe the process of thousands of enzymatic reactions used to convert nutrients into metabolites and energy. Organisms have different optimal performances (e.g., maximizing growth rate, ATP generation) under a range of growth conditions (Edwards et al., 2001; Sauer et al., 1998; Papoutsakis, 1984). *Flux Balance Analysis (FBA)* (Kauffman et al., 2003) has emerged as one of the most important methodologies to analyze the metabolic network in steady-state.

FBA formulates the problem of predicting the metabolic reaction fluxes as a *linear programming* problem. Let $\mathbf{S} \in \mathbb{R}^{m \times n}$ denote the stoichiometric matrix (expressing

mass balance) where m is the number of metabolites and n the number of metabolic fluxes, $\mathbf{x} \in \mathbb{R}^n$ the vector of metabolic fluxes (internal and external), and \mathbf{v}_{lb} , \mathbf{v}_{ub} lower and upper bounds on the metabolic fluxes implied by the composition of the growth medium. External fluxes represent the transfer rates between metabolites which exist inside and outside of the cell. Internal fluxes represent the reactions rates among metabolites inside of the cell. The FBA problem is formulated as:

$$\begin{aligned} Z_{opt} &= \max_{\mathbf{x}} \mathbf{c}'\mathbf{x} && (4.1) \\ \text{s.t. } \mathbf{S}\mathbf{x} &= \mathbf{0}, \\ \mathbf{v}_{lb} &\leq \mathbf{x} \leq \mathbf{v}_{ub}, \end{aligned}$$

where $\mathbf{0}$ is the vector of all zeroes, \mathbf{c} the vector of coefficients expressing the cellular objective (e.g., biomass), and we use Z_{opt} to denote the optimal objective value. In addition, we will use \mathcal{F} to denote the feasible set. Fig. 4-1 illustrates the polyhedral feasible set of the FBA problem and possible scenarios on where the optimal metabolic flux distribution, say \mathbf{x}^* , may lie.

In most applications of FBA, maximizing the biomass growth rate is assumed to be the unique objective function. Yet, our knowledge of the optimal performances (objective functions) that govern the fluxes distributions is quite limited and alternatives to biomass have been proposed. In (Schuetz et al., 2012), it is suggested that other than maximizing the biomass growth rate, cells also try to minimize the effort to adjust to a new environment. Mathematically, this performance can be formulated as a quadratic objective function. In (Harcombe et al., 2013), comparing ancestral and evolved strains of *E. coli* indicated that evolved strains migrate away from predicted optimal fluxes obtained by FBA with a biomass growth rate as an objective function. Experimentally, intra-cellular fluxes are available to be measured by the ^{13}C -based flux analysis technique. In this chapter, we seek to infer the right objective

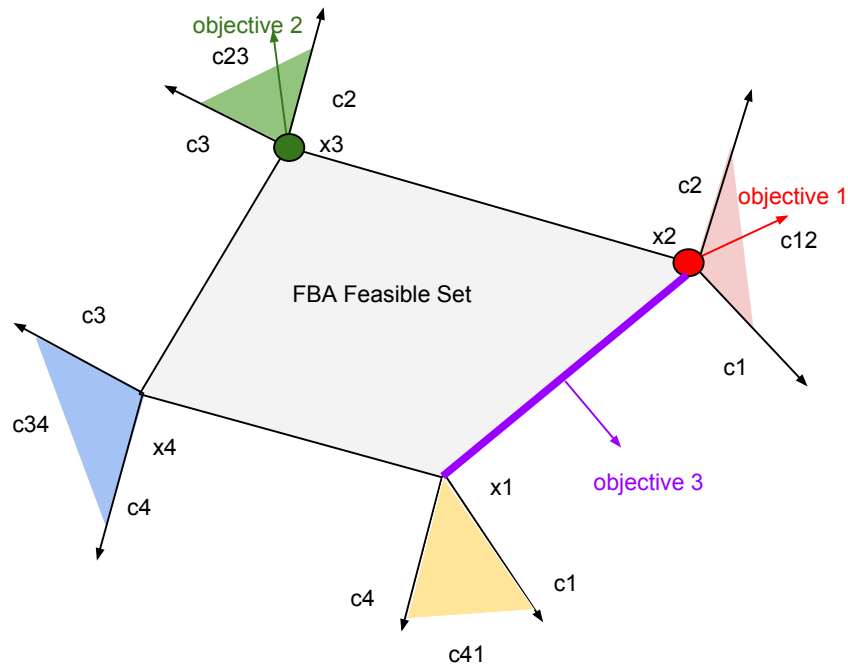


Figure 4-1: The feasible set of the FBA problem is a polyhedron – the intersection of points satisfying the linear constraints of (4.1). \mathbf{x}^j , $j = 1, \dots, 4$, denote extreme points of the feasible set. Depending on the objective function coefficient \mathbf{c} , one of these extreme points is optimal. In some cases, a particular \mathbf{c} can lead to multiple extreme points being optimal (purple case), in which case the whole edge/facet containing them is optimal. Notice, that for each optimal extreme point there are many \mathbf{c} 's that could have led to its optimality. These “valid” \mathbf{c} 's belong to a cone (cf. the four colored cones drawn).

function of a cell based on available measurements of internal and external fluxes under a certain growth condition.

Towards the same goal, some FBA-based methods have been proposed. (Burgard and Maranas, 2003) formulated the objective inference problem as a least squares problem with a bi-linear feasible region. This is a non-convex formulation which does not guarantee global optimality. In (Gianchandani et al., 2008), a bi-level nonlinear optimization formulation was proposed. In this nonlinear optimization, the computation cost is substantial when the size of problem becomes large, and global optimality

cannot be guaranteed either. In (Knorr et al., 2007), a Bayesian-based selection of metabolic objective functions was proposed. This is a probabilistic approach, which is unable to yield any deterministic solution and biological insights. In this chapter, leveraging a more general framework developed in (Bertsimas et al., 2013), we develop a novel convex optimization-based framework called InvFBA to infer the cellular objective function from the experimentally measured fluxes. In our InvFBA formulation, the linearity of constraints and the convexity of the objective function provide global optimality guarantees and an efficient algorithm (polynomial time). We provide both the theoretical underpinnings and the biological insight behind the InvFBA approach. In essence, our approach characterizes a family of objective functions consistent with the given measurements. To recover a single and biologically meaningful objective function we propose a proper regularization technique.

This chapter is organized as follows. In Section 4.2, based on the FBA framework, we develop three invFBA methods to infer linear, quadratic and non-parametric cellular objective functions respectively. We also provide a rigorous proof and geometric understanding of InvFBA methods. In Section 4.3, we test our invFBA methods on different data sets and assess their performances from different aspects.

4.2 Inverse Process of FBA

Solving (4.1) in practice, often leads to multiple optimal solutions. Thus, to select a unique optimal solution from the optimal solution set, a second optimization problem is used. In particular, one minimizes the l_1 -norm of the metabolic flux vector subject to the constraints of (4.1) and one constraint that guarantees that the same objective

value is achieved (Kauffman et al., 2003). The formulation is:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \sum_{i=1}^n |x_i| \\ \text{s.t.} \quad & \mathbf{c}'\mathbf{x} = Z_{opt}, \\ & \mathbf{S}\mathbf{x} = \mathbf{0}, \\ & \mathbf{v}_{lb} \leq \mathbf{x} \leq \mathbf{v}_{ub}. \end{aligned} \tag{4.2}$$

The optimal solution \mathbf{x}^* to problem (4.2) is the theoretical prediction of the flux distribution inside the cell. FBA has been widely used to predict the flux distributions of cells under different growth conditions.

We now introduce an *inverse* procedure of FBA – InvFBA. As explained earlier, the main motivation for invFBA is that the cellular objective, which is often assumed to be biomass maximization, may not always be valid especially at “transition” times when the composition of the growth medium changes. Given that experimentally measured metabolic flux distributions become more accessible, it becomes interesting to leverage that information to infer the cellular objective. We will discuss three different forms of the objective functions: linear, quadratic, and non-parametric.

4.2.1 Inverse FBA with Linear Objective Function

We assume that the FBA objective function is of the form $Z(\mathbf{x}) = \mathbf{c}'\mathbf{x}$ but \mathbf{c} is unknown. We have, however, access to N measured metabolic flux distribution vectors ($\mathbf{x}_i, i = 1, \dots, N$). Due to potential measurement noise or other forms of error, \mathbf{x}_i are feasible solutions of (4.1) but not necessarily optimal. We define a notion of approximate optimality.

Definition 4.2.1. *The feasible solution $\mathbf{x}_i \in \mathcal{F}$ is an ϵ_i -approximate optimal solution to FBA problem (4.1) if*

$$\mathbf{c}'\mathbf{x}^* - \mathbf{c}'\mathbf{x}_i = \epsilon_i, \quad \forall \mathbf{x} \in \mathcal{F}, \tag{4.3}$$

where \mathbf{x}^* is an optimal solution of problem (4.1).

In the above definition, $\epsilon_i \geq 0$ denotes the suboptimality gap. The next theorem characterizes the space of objective vectors \mathbf{c} corresponding to \mathbf{x}_i .

Theorem 4.2.1. *Suppose the FBA problem (4.1) is feasible. Then, the solution \mathbf{x}_i is an ϵ_i -approximate optimal solution to problem (4.1) if and only if there exist \mathbf{p}^i , \mathbf{q}_1^i and \mathbf{q}_2^i satisfying*

$$\begin{aligned} \mathbf{p}^{i'} \mathbf{S} - \mathbf{q}_1^{i'} + \mathbf{q}_2^{i'} &= \mathbf{c}', \\ \mathbf{q}_2^{i'} \mathbf{v}_{ub} - \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \epsilon_i &= \mathbf{c}' \mathbf{x}_i, \\ \mathbf{q}_1, \mathbf{q}_2 &\geq \mathbf{0}. \end{aligned} \tag{4.4}$$

Proof. First suppose that \mathbf{x}_i is an ϵ_i -approximate optimal solution. Then, from Eq. (4.3), it follows

$$\mathbf{c}' \mathbf{x}_i + \epsilon_i = \max_{\mathbf{x} \in \mathcal{F}} \mathbf{c}' \mathbf{x}, \quad \forall \mathbf{x} \in \mathcal{F}.$$

The right hand side is the optimal value of a linear programming problem over \mathbf{x} . Since $\mathcal{F} \neq \emptyset$, strong duality holds, which implies that there exists dual variables \mathbf{p}^i , \mathbf{q}_1^i and \mathbf{q}_2^i corresponding to the constraints $\mathbf{S}\mathbf{x} = \mathbf{0}$, $\mathbf{x} \geq \mathbf{v}_{lb}$, and $\mathbf{x} \leq \mathbf{v}_{ub}$, respectively, and attain the optimum in the dual problem of (4.1), which is

$$\begin{aligned} \min_{\mathbf{p}, \mathbf{q}_1, \mathbf{q}_2} \quad & \mathbf{q}_2' \mathbf{v}_{ub} - \mathbf{q}_1' \mathbf{v}_{lb} \\ \text{s.t.} \quad & \mathbf{p}' \mathbf{S} - \mathbf{q}_1' + \mathbf{q}_2' = \mathbf{c}', \\ & \mathbf{q}_1, \mathbf{q}_2 \geq \mathbf{0}. \end{aligned} \tag{4.5}$$

Because of the strong duality, we have $\max_{\mathbf{x} \in \mathcal{F}} \mathbf{c}' \mathbf{x}$ equals the optimal value of (4.5). By substituting this duality constraint and rearranging all equations above we establish the result. The reverse direction of the proof is analogous and easy to be completed. \square

Notice that the equations in Thm 4.2.1 describe a set of \mathbf{c} 's that lead to the near-optimality of \mathbf{x}_i . In particular, they describe a *cone* \mathcal{C} , i.e., a set that contains all non-negative multiples of its elements. This conclusion is quite intuitive since it suffices to determine the objective coefficient vector up to a non-negative multiplicative constant. It is now clear that FBA does not lead to unique inference of the cellular objective from measured fluxes.

Collecting the constraints in (4.4) for all measurements $i = 1, \dots, N$ and minimizing the total suboptimality gap of the measured flux distribution vectors we obtain

the optimization problem:

$$\begin{aligned}
 \min_{\epsilon_i, \mathbf{p}^i, \mathbf{q}_1^i, \mathbf{q}_2^i, \mathbf{c}} \quad & \sum_{i=1}^N \epsilon_i \\
 \text{s.t.} \quad & \mathbf{p}^{i'} \mathbf{S} - \mathbf{q}_1^{i'} + \mathbf{q}_2^{i'} = \mathbf{c}', \quad \forall i, \\
 & \mathbf{q}_2^{i'} \mathbf{v}_{ub} - \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \epsilon_i = \mathbf{c}' \mathbf{x}_i, \quad \forall i, \\
 & \mathbf{q}_1, \mathbf{q}_2 \geq \mathbf{0}, \quad \forall i, \\
 & \epsilon_i \geq 0, \quad \forall i.
 \end{aligned} \tag{4.6}$$

The problem above has the trivial solution $\mathbf{c} = \mathbf{0}$ and this is reasonable given our observation that we can only determine the objective coefficient vector up to a non-negative multiplicative constant. It follows, that we need to introduce some form of *regularization* to restrict \mathbf{c} to non-trivial choices.

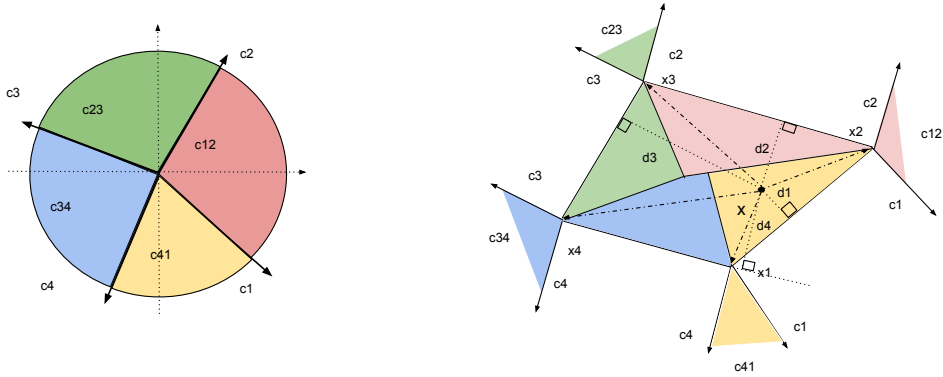


Figure 4-2: (Left): The union of these cones is all of \mathbb{R}^n ($n = 2$, in the figure). If we use an ℓ_2 -norm regularization in the invFBA formulation, “valid” \mathbf{c} ’s lie on the surface of the unit ball in \mathbb{R}^n . **(Right):** Illustration of the geometric intuition behind ℓ_2 -regularized FBA. The FBA feasible polyhedron can be partitioned using bisector lines. In this 2-dimensional example, the polyhedron is partitioned by 4 bisectors into 4 (colored) polytopes. For example, if the observed flux distribution \mathbf{x} lies in the yellow polytope then \mathbf{c}_1 is the corresponding objective function. If, however, the observed flux distribution lies in the bisector between the blue and the yellow region, invFBA will yield either \mathbf{c}_1 or \mathbf{c}_4 .

One of the possible regularizations is to add to the formulation in (4.6) the ℓ_2 -norm equality constraint $\|\mathbf{c}\|^2 = 1$. In this case, the objective function coefficient vector \mathbf{c} lies on the surface of the unit ball in \mathbb{R}^n (cf. Fig. 4-2). To gain more

geometric insight into the proposed ℓ_2 -regularized invFBA, consider the case of a single measured metabolic flux vector, say \mathbf{x} (i.e., $N = 1$). Solving problem (4.6) amounts to minimizing $\mathbf{c}'(\mathbf{x}^j - \mathbf{x})$ over all \mathbf{c} and all extreme points \mathbf{x}^j of the FBA polyhedron (feasible set of (4.1)). We have

$$\min_j \mathbf{c}'(\mathbf{x}^j - \mathbf{x}) = \min_j \|\mathbf{c}\| \|\mathbf{x}^j - \mathbf{x}\| \cos \alpha, \quad (4.7)$$

where α is the angle between \mathbf{c} and $\mathbf{x}^j - \mathbf{x}$ and $\|\mathbf{x}^j - \mathbf{x}\| \cos \alpha$ is the projection of $\mathbf{x}^j - \mathbf{x}$ onto \mathbf{c} . Thus, and since $\|\mathbf{c}\| = 1$, minimizing (4.7) over \mathbf{c} is equivalent to projecting \mathbf{x} on all facets of the FBA polyhedron and selecting the \mathbf{c} that is perpendicular to the closest facet. As an example, in Fig. 4-2 (right), we compare the distances d_j , $j = 1 \dots, 4$, between \mathbf{x} and the four facets, which yields d_1 as the minimum and sets the corresponding optimal objective coefficient vector to \mathbf{c}_1 .

We have already discussed that FBA is used in practice with an additional sparsity-inducing ℓ_1 -norm regularization (cf. (4.2)). This motivates the use of a similar regularization in the context of invFBA. An ℓ_1 -norm is appealing because one would essentially be looking for the sparse \mathbf{c} vectors rendering a given set of measured metabolic flux distributions FBA-optimal. An ℓ_1 -norm constraint can also be seen as a relaxation of the combinatorial problem that minimizes the number of nonzero elements of \mathbf{c} in (4.6). To make the objective coefficient vector \mathbf{c} non-zero and biologically interpretable, we will add a normalization constraint and look for \mathbf{c} 's such that $\sum_{j=1}^n c_j = 1$. This leads to the formulation:

$$\begin{aligned} Z_{opt}^I = \min_{\epsilon_i, \mathbf{p}^i, \mathbf{q}_1^i, \mathbf{q}_2^i, \mathbf{c}} & \sum_{i=1}^N \epsilon_i \\ \text{s.t.} & \sum_{j=1}^n c_j = 1, \\ & \mathbf{p}^{i'} \mathbf{S} - \mathbf{q}_1^{i'} + \mathbf{q}_2^{i'} = \mathbf{c}', \quad \forall i, \\ & \mathbf{q}_2^{i'} \mathbf{v}_{ub} - \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \epsilon_i = \mathbf{c}' \mathbf{x}_i, \quad \forall i, \\ & \mathbf{q}_1, \mathbf{q}_2 \geq \mathbf{0}, \quad \forall i, \\ & \epsilon_i \geq 0, \quad \forall i, \end{aligned} \quad (4.8)$$

where Z_{opt}^I denotes the optimal value.

Again motivated by the 2nd step in (4.2) of FBA's application in practice, we also propose a subsequent step whose purpose is to minimize the ℓ_1 -norm of $\mathbf{c} = (c_1, \dots, c_n)$ vectors that solve (4.8):

$$\begin{aligned}
\min_{\epsilon_i, \mathbf{p}^i, \mathbf{q}_1^i, \mathbf{q}_2^i, \mathbf{c}} \quad & \sum_{i=1}^n |c_i| \\
\text{s.t.} \quad & \sum_{i=1}^N \epsilon_i = Z_{opt}^I, \\
& \sum_{j=1}^n c_j = 1 \\
& \mathbf{p}^{i'} \mathbf{S} - \mathbf{q}_1^{i'} + \mathbf{q}_2^{i'} = \mathbf{c}', \quad \forall i, \\
& \mathbf{q}_2^{i'} \mathbf{v}_{ub} - \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \epsilon_i = \mathbf{c}' \mathbf{x}_i, \quad \forall i, \\
& \mathbf{q}_1, \mathbf{q}_2 \geq \mathbf{0}, \quad \forall i, \\
& \epsilon_i \geq 0, \quad \forall i.
\end{aligned} \tag{4.9}$$

Part of the optimal solution of (4.9) is a sparse \mathbf{c} vector that renders the given set of measured metabolic flux distributions $\mathbf{x}_1, \dots, \mathbf{x}_N$ near-optimal in the FBA optimization (4.1). One can then interpret non-zero elements of \mathbf{c} as corresponding to important metabolic fluxes that are critical in the FBA optimization context and provide a minimal description of the cellular objective function. In the sequel, when we refer to the *invFBA algorithm* we will mean the 2-step procedure of solving problems (4.8) and (4.9). Due to the ℓ_1 regularization and the fact that several measured metabolic flux vectors are used, the \mathbf{c} resulting from invFBA may not be perpendicular to one of the hyperplanes defining the FBA polytope; it can in fact be interior to the cone \mathcal{C} containing all valid \mathbf{c} 's.

Problems (4.9) is a linear programming problem minimizing the ℓ_1 norm of the vector \mathbf{c} . It can be viewed as a convex relaxation of a problem with identical constraints, which minimizes the ℓ_0 norm of \mathbf{c} (i.e., the number of non-zero elements in \mathbf{c}). To pursue more sparse objective functions, an integer programming problem is introduced to minimize the ℓ_0 -norm of \mathbf{c} . This problem is formulated as:

$$\begin{aligned}
\min_{\epsilon_i, \mathbf{p}^i, \mathbf{q}_1^i, \mathbf{q}_2^i, \mathbf{c}, \mathbf{z}} \quad & \sum_{j=1}^n z_j \\
\text{s.t.} \quad & \sum_{i=1}^N \epsilon_i = Z_{opt}^I, \\
& \sum_{j=1}^n c_j = 1 \\
& \mathbf{p}^{i'} \mathbf{S} - \mathbf{q}_1^{i'} + \mathbf{q}_2^{i'} = \mathbf{c}', \quad \forall i, \\
& \mathbf{q}_2^{i'} \mathbf{v}_{ub} - \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \epsilon_i = \mathbf{c}' \mathbf{x}_i, \quad \forall i, \\
& z_j \geq \frac{|c_j|}{L}, \quad z_j \in \{0, 1\}, \quad \forall j, \\
& \mathbf{q}_1, \mathbf{q}_2 \geq \mathbf{0}, \quad \forall i, \\
& \epsilon_i \geq 0, \quad \forall i.
\end{aligned} \tag{4.10}$$

In (4.10), the binary variable z_j is the indicator of whether c_j is nonzero (then $z_j = 1$) or not (then, $z_j = 0$). The constant L is a large number that together with the integrality constraint $z_j \in \{0, 1\}$ forces z_j to be 1 when c_j is nonzero. Formulation (4.10) is an integer programming problem and formulation (4.9) is a convex relaxation of (4.10). Of course, it is much more computationally expensive to solve (4.10) compared to solving (4.9). We can, however, use the solution of (4.9) as a feasible solution for (4.10), which can substantially speed up the solution time of (4.10) (using the value of the (4.9) solution in a branch-and-bound algorithm for (4.10)).

An important observation is that both problems (4.8) and (4.9) that comprise our invFBA algorithm are linear programming problems. This is important because a global optimal solution is guaranteed (as opposed to earlier approaches as in (Burgard and Maranas, 2003) where one deals with a non-convex problem). Moreover, very efficient polynomial-time algorithms exist for solving such problems. It is interesting that the complexity of the invFBA algorithm is exactly the same as that of FBA – both are linear programming problems. This is not in general true for inverse optimization problems (Bertsimas et al., 2013). An alternative LASSO-like invFBA method is introduced in Appendix A.

To illustrate the use of the invFBA algorithm (4.8) and (4.9), we apply it to a simple metabolic network shown in Fig. 4-3 (Left). The stoichiometric matrix is $\mathbf{S} = [1, 2, -1]$ and we impose the constraint $x_3 \leq 2$. Thus, the FBA polyhedron

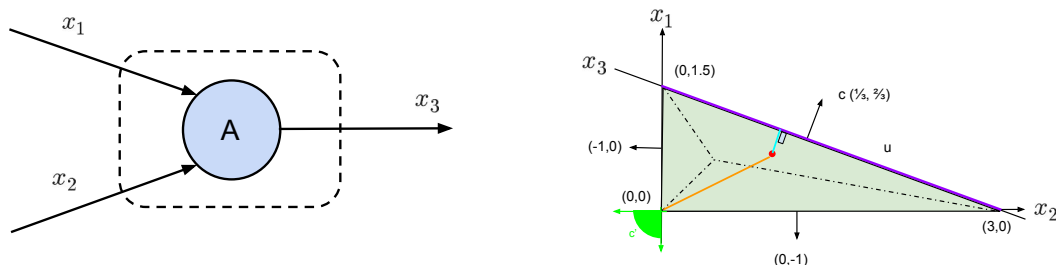


Figure 4.3: (Left): A simple simulated metabolic network model with a single metabolite A and three reactions. (Right): We plot the FBA polyhedron on the x_1 - x_2 plane. The red dot indicated a given metabolic flux distribution. The invFBA algorithm yields $\mathbf{c} = (1/3, 2/3)$.

is $\{\mathbf{x} \mid \mathbf{S}\mathbf{x} = \mathbf{0}, \mathbf{x} \geq \mathbf{0}, x_3 \leq 2\}$. We eliminate x_3 and plot in Fig. 4.3 (Right) the polyhedron in the x_1 - x_2 plane. The red dot shown in the figure identifies a given metabolic flux distribution. The invFBA algorithm applied by using this point as the measured flux vector yields $\mathbf{c} = (1/3, 2/3)$ which is the vector perpendicular to the closest hyperplane. We note that if instead we seek the closest extreme point of the FBA polytope to the red dot (as in (Burgard and Maranas, 2003; Gianchandani et al., 2008)) then we would obtain the extreme point $(x_1, x_2) = (0, 0)$ and a \mathbf{c} that renders this point optimal (i.e., a \mathbf{c} in the green cone shown in the figure). It is clear from the figure that such a cellular objective coefficient may not be the most appropriate.

4.2.2 Linear Objective Variability Analysis

To analyze the variability of each element in the objective function within the optimal solution space, we developed a method called Objective Variability Analysis (OVA). OVA is a heuristic optimization method used to compute the upper and lower bound of the elements in the objective function. The first step of OVA is to solve the linear optimization problem given in problem (4.9).

By solving this problem, we can obtain an optimal solution for \mathbf{c} and the optimal objective value $\sum_{i=1}^N \epsilon_i = Z_{opt}^I$. Let \mathfrak{R} be the set of reaction indices which are important in the FBA problem. To find the possible range for each c_r , $r \in \mathfrak{R}$, we solve (4.11)

$$\begin{aligned}
\min_{\epsilon_i, \mathbf{p}^i, \mathbf{q}_1^i, \mathbf{q}_2^i, \mathbf{c}} \quad & c_r + |\mathbf{c}| \\
\text{s.t.} \quad & \sum_{i=1}^N \epsilon_i = Z_{opt}^I, \\
& \sum_{j=1}^n c_j = 1 \\
& \mathbf{p}^{i'} \mathbf{S} - \mathbf{q}_1^{i'} + \mathbf{q}_2^{i'} = \mathbf{c}', \quad \forall i, \\
& \mathbf{q}_2^{i'} \mathbf{v}_{ub} - \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \epsilon_i = \mathbf{c}' \mathbf{x}_i, \quad \forall i, \\
& \mathbf{q}_1, \mathbf{q}_2 \geq \mathbf{0}, \quad \forall i, \\
& \epsilon_i \geq 0, \quad \forall i,
\end{aligned} \tag{4.11}$$

and an identical problem with the only difference being that we maximize $c_r - |\mathbf{c}|$ (instead of minimizing) so as to find the largest possible value of c_r and maintain a small ℓ_1 norm of \mathbf{c} . Solving these problems yields upper and lower bounds on each element in the objective function.

To apply OVA in practice, some extra constraints on \mathbf{c} should be added to (4.11). Consider reactions represented in the flux vectors \mathbf{x}_i which have a flux equal to zero for all i . For these reactions, the corresponding elements c_j in \mathbf{c} can take arbitrary feasible values because of the term $\mathbf{c}' \mathbf{x}_i$ in (4.11). For this reason, it is meaningless to run OVA on these c_j . To that end, we set $c_j = 0$ for all those indices. In the later case study involving simulated *E. coli* data, since the flux distributions are very sparse, we applied this technique and focused on the non-trivial reactions only.

4.2.3 Inverse FBA with Quadratic Objective Function

As we have discussed in the Introduction, non-linear objectives have been proposed for FBA, e.g., (Schuetz et al., 2012) which argued for a trade-off between optimality under current growth conditions and minimal adjustment to past recent growth conditions.

This reasoning motivates the following quadratic-FBA problem:

$$\begin{aligned} \min \quad & \mathbf{c}'\mathbf{x} + \beta\|\mathbf{x} - \hat{\mathbf{x}}\|^2 \\ \text{s.t.} \quad & \mathbf{S}\mathbf{x} = \mathbf{0}, \\ & \mathbf{v}_{lb} \leq \mathbf{x} \leq \mathbf{v}_{ub}, \end{aligned} \tag{4.12}$$

where β is some tunable and non-negative constant and $\hat{\mathbf{x}}$ the previously observed flux distribution. We use function $Z(\mathbf{x}) = \beta\mathbf{x}'\mathbf{x} + (\mathbf{c} - 2\beta\hat{\mathbf{x}})'\mathbf{x}$ to represent the quadratic objective function (ignoring a constant). A solution \mathbf{x}^* is optimal in problem (4.12) if and only if it satisfies the optimality conditions

$$\nabla Z(\mathbf{x}^*)'(\mathbf{x} - \mathbf{x}^*) \geq 0, \quad \forall \mathbf{x} \in \mathcal{F}. \tag{4.13}$$

Similar to Definition 4.2.1, we provide a notion of approximate optimality.

Definition 4.2.2. *The feasible solution $\mathbf{x}_i \in \mathcal{F}$ is an ϵ_i -approximate optimal solution to FBA problem (4.12) if*

$$\nabla Z(\mathbf{x}_i)'(\mathbf{x} - \mathbf{x}_i) \geq -\epsilon_i, \quad \forall \mathbf{x} \in \mathcal{F}. \tag{4.14}$$

The following theorem characterizes the feasible space of unknown parameters \mathbf{c} and β corresponding to \mathbf{x}_i .

Theorem 4.2.2. *Suppose \mathcal{F} is non-empty. Then, the solution \mathbf{x}_i is an ϵ_i -optimal solution to problem (4.12) if and only if there exist \mathbf{p}^i , \mathbf{q}_1^i , and \mathbf{q}_2^i satisfying*

$$\begin{aligned} [\mathbf{c}' + 2\beta(\mathbf{x}_i - \hat{\mathbf{x}})']\mathbf{x}_i - \epsilon_i &\leq \mathbf{q}_1^{i'}\mathbf{v}_{lb} - \mathbf{q}_2^{i'}\mathbf{v}_{ub}, \\ \mathbf{S}'\mathbf{p}^i + \mathbf{q}_1^i - \mathbf{q}_2^i &= \mathbf{c} + 2\beta(\mathbf{x}_i - \hat{\mathbf{x}}), \\ \mathbf{q}_1^i, \mathbf{q}_2^i &\geq \mathbf{0}. \end{aligned} \tag{4.15}$$

Proof. First suppose that \mathbf{x}_i is an ϵ_i -approximate optimal solution to (4.12). Then from equation (4.13), that we have

$$\nabla Z(\mathbf{x}_i)'(\mathbf{x} - \mathbf{x}_i) \geq -\epsilon_i, \quad \forall \mathbf{x} \in \mathcal{F},$$

which is equivalent to

$$\nabla Z(\mathbf{x}_i)'\mathbf{x}_i - \epsilon_i \leq \min_{\mathbf{x} \in \mathcal{F}} \nabla Z(\mathbf{x}_i)'\mathbf{x}.$$

The right hand side is the optimal value of a linear programming problem whose dual is

$$\begin{aligned} \max_{\mathbf{p}^i, \mathbf{q}_1^i, \mathbf{q}_2^i} \quad & \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \mathbf{q}_2^{i'} \mathbf{v}_{ub} \\ \text{s.t.} \quad & \mathbf{c} + 2\beta(\mathbf{x}_i - \hat{\mathbf{x}}) = \mathbf{S}'\mathbf{p}^i + \mathbf{q}_1^i - \mathbf{q}_2^i, \\ & \mathbf{q}_1^i, \mathbf{q}_2^i \geq \mathbf{0}, \end{aligned} \quad (4.16)$$

where \mathbf{p}^i , \mathbf{q}_1^i and \mathbf{q}_2^i are dual variables. Since $\mathcal{F} \neq \emptyset$, strong duality holds which implies that $\min_{\mathbf{x} \in \mathcal{F}} \nabla Z(\mathbf{x}_i)' \mathbf{x}$ is equal to the optimal value of (4.16). This proves the result. The reverse direction of the proof is analogous and easy to be completed. \square

Given Thm. 4.2.2 we formulate the inverse optimization problem as

$$\begin{aligned} \min_{\mathbf{p}^i, \mathbf{q}_1^i, \epsilon_i, \mathbf{q}_2^i, \mathbf{c}, \beta} \quad & \sum_{i=1}^N \epsilon_i + \lambda \sum_{j=1}^n |c_j| \\ \text{s.t.} \quad & \mathbf{S}'\mathbf{p}^i + \mathbf{q}_1^i - \mathbf{q}_2^i = \mathbf{c} + 2\beta(\mathbf{x}_i - \hat{\mathbf{x}}), \quad \forall i, \\ & [\mathbf{c}' + 2\beta(\mathbf{x}_i - \hat{\mathbf{x}})'] \mathbf{x}_i - \epsilon_i \leq \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \mathbf{q}_2^{i'} \mathbf{v}_{ub}, \quad \forall i, \\ & \mathbf{q}_1^i, \mathbf{q}_2^i \geq \mathbf{0}, \quad \forall i, \quad \beta \geq 0, \\ & \sum_{j=1}^n c_j = -1. \end{aligned} \quad (4.17)$$

Similar to problem (A.1), we have again added the regularization term $\lambda \sum_{j=1}^n |c_j|$ to induce a sparse vector \mathbf{c} . Notice that (4.17) is a linear programming problem.

4.2.4 Inverse FBA with Non-parametric Objective Function

In practice, the activities of cells can be extremely complicated. Even though linear or quadratic objective functions are able to describe these activities in most cases, we next attempt to explore more general forms of objective functions that can improve FBA modeling. To that end, we leverage ideas from machine learning and use kernel methods to infer non-parametric forms of the objective function. As before, we will use $Z(\mathbf{x})$ to denote the convex objective and consider an FBA-like problem of

$$\min_{\mathbf{x} \in \mathcal{F}} Z(\mathbf{x}). \quad (4.18)$$

Kernel methods are commonly used for feature extraction, often in the context of support vector machines or principal components analysis. A more thorough treat-

ment of kernel methods can be found in (Evgeniou et al., 2000). Consider the FBA convex objective function $Z(\mathbf{x})$ and let $\mathbf{z}(\mathbf{x}) = \nabla Z(\mathbf{x}) = (z_1(\mathbf{x}), \dots, z_n(\mathbf{x}))$. We can let each $z_i \in \mathcal{H}$, $i = 1, \dots, n$, where \mathcal{H} is a Reproducing Kernel Hilbert Space (RKHS) (Hastie et al., 2009) with an associated positive semi-definite kernel function $k : \mathcal{F} \times \mathcal{F} \rightarrow \mathbb{R}$. Different choices for the kernel are possible including linear, polynomial, or a radial basis function.

According to the representer Theorem (Schölkopf et al., 2001), scalar functions $f \in \mathcal{H}$ can be written as linear combinations of kernel functions as $f(\mathbf{x}) = \sum_{i=1}^N \alpha_i k(\mathbf{x}_i, \mathbf{x})$. Let g another element of \mathcal{H} , written as $g(\mathbf{x}) = \sum_{j=1}^N \beta_j k(\mathbf{x}_j, \mathbf{x})$. Then we can define an inner product over \mathcal{H} as $f, g \in \mathcal{H}$ such that

$$\langle f, g \rangle_{\mathcal{H}} = \sum_{i=1}^N \sum_{j=1}^N \alpha_i \beta_j k(\mathbf{x}_i, \mathbf{x}_j), \quad (4.19)$$

and a norm by $\|f\|_{\mathcal{H}} = \sqrt{\langle f, f \rangle_{\mathcal{H}}}$. The norm $\|f\|_{\mathcal{H}}$ is a measure of the “smoothness” of the function $f(\cdot)$. Thus, functions with small norm are “smooth,” meaning that they do not change rapidly in a small neighborhood of their argument.

As in earlier subsections, assume now we have access to N measured flux vectors \mathbf{x}_i , $i = 1, \dots, N$. Each \mathbf{x}_i is an ϵ_i -approximate optimal solution to the FBA problem with objective function $Z(\mathbf{x})$, where the notion of approximate optimality is the same as in Definition 4.2.2. We can represent each partial derivative $z_i(\mathbf{x})$ of $Z(\mathbf{x})$ as an element of \mathcal{H} and express it as a linear combination of kernel functions as in $z_i(\mathbf{x}; \boldsymbol{\alpha}_i) = \sum_{j=1}^N \alpha_{i,j} k(\mathbf{x}_j, \mathbf{x})$ for some appropriate coefficient vector $\boldsymbol{\alpha}_i = (\alpha_{i,1}, \dots, \alpha_{i,N})$. Note that $\mathbf{z}(\mathbf{x}_i; \boldsymbol{\alpha}) = \boldsymbol{\alpha} \mathbf{K} \mathbf{e}_i$ where $\boldsymbol{\alpha} \in \mathbb{R}^{n \times N}$ is the matrix whose rows are the $\boldsymbol{\alpha}_i$'s, $\mathbf{K} = (k(\mathbf{x}_j, \mathbf{x}_i))_{i,j=1}^N$ and \mathbf{e}_i is the i th unit vector.

The following theorem characterizes the set of objective function gradients that are consistent with our measurements.

Theorem 4.2.3. *Suppose \mathcal{F} is non-empty. Then, the solution \mathbf{x}_i is an ϵ_i -optimal*

solution to problem (4.18) if and only if there exist $\boldsymbol{\alpha}$, \mathbf{p}^i , \mathbf{q}_1^i , and \mathbf{q}_2^i satisfying

$$\begin{aligned} (\boldsymbol{\alpha}\mathbf{K}\mathbf{e}_i)' \mathbf{x}_i - \epsilon_i &\leq \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \mathbf{q}_2^{i'} \mathbf{v}_{ub}, \\ \boldsymbol{\alpha}\mathbf{K}\mathbf{e}_i &= \mathbf{S}' \mathbf{p}^i + \mathbf{q}_1^i - \mathbf{q}_2^i, \\ \mathbf{q}_1^i, \mathbf{q}_2^i &\geq \mathbf{0}. \end{aligned} \quad (4.20)$$

Proof. The proof is similar to that of Theorem 4.2.2. The only difference is that $\mathbf{z}(\mathbf{x}_i; \boldsymbol{\alpha}) = \boldsymbol{\alpha}\mathbf{K}\mathbf{e}_i$. \square

Biologically, we rarely expect cellular objectives to change rapidly within a small time interval. Therefore, the smoothness of the objective function is a reasonable assumption. We want, therefore, to find the objective function which leads to a small suboptimality gap and a small norm. This leads to the following formulation:

$$\min_{\{z_i \in \mathcal{H}, \epsilon_i \geq 0, \forall i\}} \sum_{i=1}^N \epsilon_i + \lambda \sum_{i=1}^n \|z_i\|_{\mathcal{H}}^2, \quad (4.21)$$

where $\lambda > 0$ a tunable constant used to trade-off between the minimal optimality gap and the smoothness of the objective function. Since $z_j(\mathbf{x}_i; \boldsymbol{\alpha}) = \mathbf{e}_j' \boldsymbol{\alpha}\mathbf{K}\mathbf{e}_i$, it follows that $\|z_j\|_{\mathcal{H}}^2 = \mathbf{e}_j' \boldsymbol{\alpha}\mathbf{K}\boldsymbol{\alpha}' \mathbf{e}_j$. This yields the problem:

$$\begin{aligned} \min_{\epsilon_i, \mathbf{p}^i, \mathbf{q}_1^i, \mathbf{q}_2^i, \boldsymbol{\alpha}} \quad & \sum_{i=1}^N \epsilon_i + \lambda \sum_{j=1}^n \mathbf{e}_j' \boldsymbol{\alpha}\mathbf{K}\boldsymbol{\alpha}' \mathbf{e}_j \\ \text{s.t.} \quad & \boldsymbol{\alpha}\mathbf{K}\mathbf{e}_i = \mathbf{S}' \mathbf{p}^i + \mathbf{q}_1^i - \mathbf{q}_2^i, \quad \forall i, \\ & (\boldsymbol{\alpha}\mathbf{K}\mathbf{e}_i)' \mathbf{x}_i - \epsilon_i \leq \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \mathbf{q}_2^{i'} \mathbf{v}_{ub}, \quad \forall i, \\ & \mathbf{q}_1^i, \mathbf{q}_2^i \geq \mathbf{0}, \quad \forall i, \\ & \epsilon_i \geq 0, \quad \forall i, \\ & \sum_{j=1}^n \sum_{i=1}^N \alpha_{j,i} = 1, \end{aligned} \quad (4.22)$$

where the last constraint is introduced for normalization purposes. By solving problem (4.22), we can obtain an optimal $\boldsymbol{\alpha}^*$ giving rise to a gradient function $\mathbf{z}(\mathbf{x}; \boldsymbol{\alpha}^*)$. Objective functions with the same gradient have the same optimal solution(s) for the forward FBA problem. It can be easily seen that problem (4.22) is a quadratic programming problem which guarantees tractability and global optimality.

4.3 Numerical Examples

In this section, we will present several numerical examples to illustrate the performance of our invFBA approaches from different aspects.

4.3.1 Inference of Linear Objective Functions in Simulated *E. coli* Fluxes

The objective function in FBA (Fig. 4.4) is encoded by a vector \mathbf{c} , whose elements represent the extent to which individual fluxes tend to be maximized or minimized in the resource allocation problem that the cell tries to solve. Mathematically, the linear combination of fluxes being maximized or minimized is expressed in the form $\sum_{j=1}^n c_j x_j$, where n is the total number of reactions in the model. The problem addressed by invFBA is to infer, from measurements of the fluxes under a given condition, the vector \mathbf{c} that best represents its objective. Most FBA calculations include only one nonzero element in \mathbf{c} , corresponding to the biomass production flux. In our invFBA approach, we want to assume that more complex \mathbf{c} vectors may better capture the objective function implied by experimentally measured fluxes.

Before applying invFBA to experimental measurements of a cell’s metabolism whose underlying objective is unknown, we first tested invFBA on in silico fluxes simulated by FBA with a known objective. Using the iJO1366 metabolic model for *E. coli* (Orth et al., 2011), we simulated growth in a standard minimal medium under three different carbon source limitations: glucose, succinate, and glycerol. In all cases, the objective function was chosen such that FBA maximizes the biomass reaction flux. We next used the output flux vectors predicted by FBA (which we will refer to as the “observed fluxes”) as an input to invFBA. The invFBA algorithm tries to infer possible objective functions that could yield the observed fluxes as solutions in FBA. Our standard formulation of invFBA works in two steps: the first step identifies a set of objective functions compatible with the observed fluxes; the second step narrows

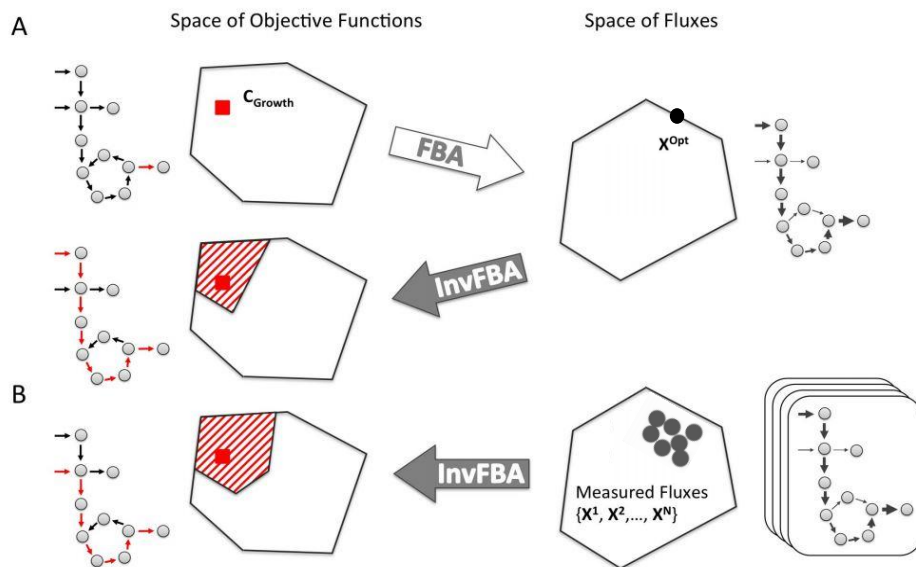


Figure 4-4: This diagram illustrates concisely the flow of information for invFBA calculations in this work. The right part of the figure displays schematic representations of the set of metabolic fluxes. Each flux vector can also be visualized on a metabolic chart (right-most part of the figure), where gray arrows of different thicknesses indicate different intensities of reaction fluxes throughout a network. The left part of the figure displays instead the space of metabolic objectives. Coefficients of the objective function can also be visualized on a metabolic chart (left-most part of the figure), with red arrows representing non-zero components of the objective. (A) FBA uses a given objective function (here c_{growth}) to predict a set of fluxes (X^{Opt}), or multiple equivalent sets of fluxes (not shown). From one FBA solution one can use InvFBA to infer possible objective functions. The solution is not necessarily unique, though the space of possible solutions can be rigorously characterized, and contains the original objective function. (B) InvFBA can be applied to multiple (noisy) experimental measurements of fluxes, leading, as in the test case of panel A, to a space of possible objective functions.

down this set to a putative sparse objective, with a minimal ℓ_1 norm. The third step is alternatively used to find the sparsest objective (which has a minimal number of nonzero elements in the objective function) if needed.

Table 4.1: InvFBA results from ℓ_1 and ℓ_0 norm regularization of simulated *E. coli* data under different growth conditions.

Growth condition		ℓ_1 norm regularization	ℓ_0 norm regularization
Glucose minimal medium	# of non-zero in \mathbf{c} elements	15	1
	maximal reaction name	biomass growth	biomass growth
	The maximal value	0.7676	1
Succinate minimal medium	# of non-zero in \mathbf{c} elements	1	1
	maximal reaction name	biomass growth	biomass growth
	The maximal value	1	1
Glycerol minimal medium	# of non-zero in \mathbf{c} elements	3	1
	maximal reaction name	biomass growth	biomass growth
	The maximal value	0.7065	1

Upon applying invFBA to the FBA-generated observed fluxes, we found that the algorithm correctly recovered maximization of the biomass flux in all three conditions (inferred coefficients are shown in Table 4.1). One immediate question is whether this solution is unique. In order to explore the spectrum of possible equivalent invFBA solutions, we extended to invFBA the method of flux variability analysis often used in classical FBA calculations (Orth et al., 2010). In this case, we wanted to characterize the possible range for each possible element in the objective function vector \mathbf{c} . The OVA we introduced before determines the full range of values each coefficient of the objective function can assume while being consistent with optimality. By running OVA to these test cases, we found that while invFBA yielded maximization of biomass as a solution under all conditions, alternative objective functions were equally compatible with the observed fluxes under the different conditions. For instance, under succinate-limited medium, an equivalent objective function is the maximization of succinate uptake. While surprising at first, this result is intuitive considering that,

to maximize growth, the cell needs to maximize uptake of its limiting nutrient. This simple example already points out an important aspect of FBA and its inverse problem, as rigorously addressed by invFBA: while the inverse algorithm rules out a large subset of objectives whose optimization could not possibly lead to the observed fluxes, different \mathbf{c} vectors may still, when used in FBA, yield the same observed fluxes. Note that if two such equivalent objectives were used in the forward FBA problem, it is not guaranteed they will produce the same fluxes, due to the existence of alternative optimal solutions in FBA itself. Yet, any \mathbf{c} inferred by invFBA will produce a flux distribution lying on the facet of the FBA polyhedron, which contains all optimal flux distributions. While the above analysis was focused on testing the capacity of invFBA to recover growth maximization as the underlying objective, one may wonder whether the algorithm could similarly recover alternative objectives. Towards this goal, we generated FBA-predicted fluxes using maximization of ATP synthase flux and minimization of glucose uptake for a fixed growth rate as alternative objectives. As shown in Tables 4.2 and 4.3, the sparse invFBA algorithm consistently recovered the correct objective function.

4.3.2 Recovering Objectives and Fluxes From Noisy Data

Unlike fluxes predicted by FBA simulations, experimentally observed fluxes will likely contain some noise that may mask the compatibility with different optimality criteria. For example, while any FBA flux vector predicted through the maximization of the biomass flux will have precisely the maximal possible growth flux value, experimentally measured fluxes, even if close to a growth optimum, will likely fall within an area around it. In order to simulate this process, and test invFBA under noisy flux measurements, we implemented our inverse algorithm under increasing levels of noise, and tested our capacity to recover the correct objective. In particular, we wanted to add noise to the optimal solution of FBA while keeping noisy fluxes in the feasible so-

Table 4.2: InvFBA results from ℓ_1 and ℓ_0 regularization of simulated *E. coli* data under different growth conditions (Maximization of ATP synthase).

Growth condition		ℓ_1 norm regularization	ℓ_0 norm regularization
Glucose minimal medium	# of non-zero elements in c	3	1
	Reaction of maximal coefficient	ATP synthase	ATP synthase
	The maximal value	0.8333	1
Succinate minimal medium	# of non-zero elements in c	1	1
	Reaction of maximal coefficient	ATP synthase	ATP synthase
	The maximal value	1	1
Glycerol minimal medium	# of non-zero elements in c	2	1
	Reaction of maximal coefficient	ATP synthase	ATP synthase
	The maximal value	0.8333	1

Table 4.3: InvFBA results from ℓ_1 and ℓ_0 regularization of simulated *E. coli* data under glucose minimal medium (Minimization of glucose uptake).

Growth condition		ℓ_1 norm regularization	ℓ_0 norm regularization
Glucose minimal medium	# of non-zero elements in c	1	1
	Reaction of maximal coefficient	D-Glucose exchange	D-Glucose exchange
	The maximal value	1	1

lution space (i.e., such that all reactions are in steady state and mass-balanced). This can be achieved by running an additional FBA-like optimization that samples random points close (within a given radius σ^2) to a previously computed FBA optimum (see Appendix B).

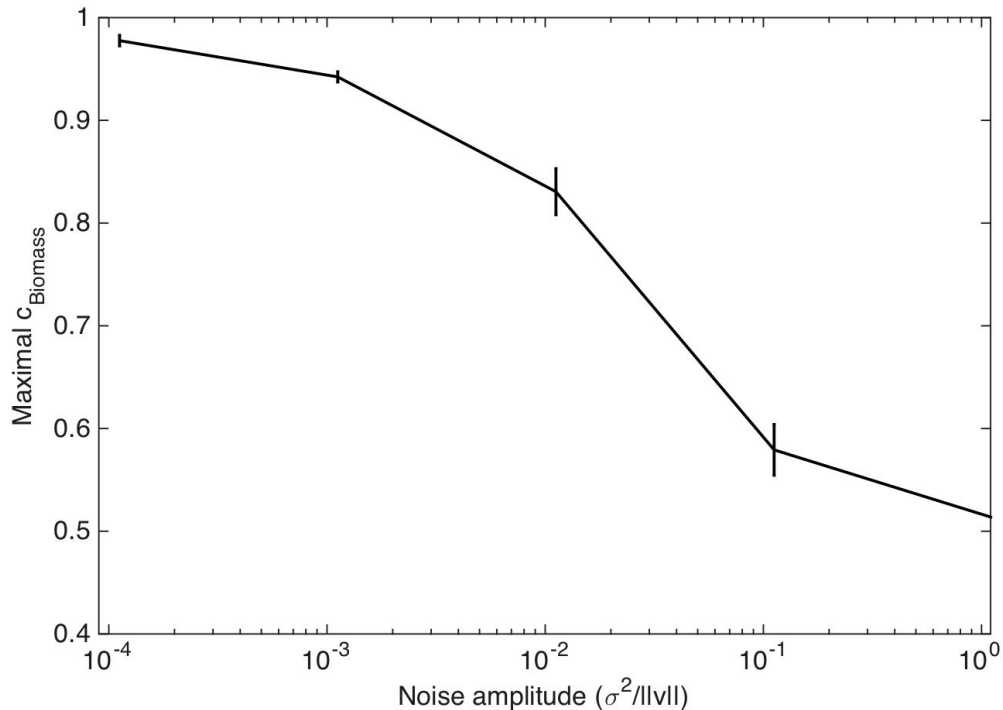


Figure 4-5: The maximal value of the biomass coefficient $c_{biomass}$, found by invFBA and subsequent Objective Variability Analysis (OVA), is plotted as a function of the level of noise (σ^2) in FBA-simulated flux data for *E. coli*. These FBA-simulated fluxes are produced using maximization of biomass production as the objective function. Thus, a value of $c_{biomass}$ close to unity in OVA indicates that invFBA recovers the original objective. As the levels of noise increases, however, the capacity of recovering the original objective is highly reduced.

As shown in Fig. 4-5, as the noise approaches zero, invFBA solutions converge to having as main component the growth maximization objective. As the magnitude of the noise increases, the maximum possible value for the biomass reaction component

of the objective decays further and further away from unity, with a major downshift at the point where the noise level is between 1% and 10% relative to the flux norm. At that point, the correlation is quickly lost, meaning that the information carried by the noisy fluxes is not informative of the original objective any more.

4.3.3 Inference of a Linear Objective Function with Time-dependent Fluxes

After testing the performance of invFBA on exact or noisy flux distributions simulated by FBA, we took a first step towards employing invFBA for the analysis of experimental data. In particular, we applied invFBA to genome-scale metabolic fluxes inferred from a combination of experimentally measured gene expression data and stoichiometric modeling. In particular, researchers inferred putative time-series flux vectors from time-series gene expression data at different stages of growth by *Shewanella oneidensis* under aerobic, carbon-limited conditions (Collins et al., 2012). The method used for that analysis (temporal expression-based analysis of metabolism, TEAM (Collins et al., 2012)), extended a prior approach (Blazier and Papin, 2012) by penalizing the cost of maintaining flux through a reaction with low gene expression. In TEAM, in contrast to prior methods, the penalty, different for each gene, was estimated based on a large compendium of gene expression data. Like many other FBA-gene expression integration methods, TEAM does not use a biologically motivated pre-assumed objective function, but rather maximizes consistency with measured gene expression data. Thus, fluxes inferred through TEAM correspond to the outcome of a heuristic approach for the interpretation of expression data in terms of metabolic fluxes, but do not assume any prior knowledge on the metabolic objective of the cell. We should emphasize that, as described recently (Machado and Herrgård, 2014), integration of gene expression data to help predict fluxes is still problematic, partly due to the nontrivial relationship between mRNA and protein levels (Taniguchi et al., 2010). However,

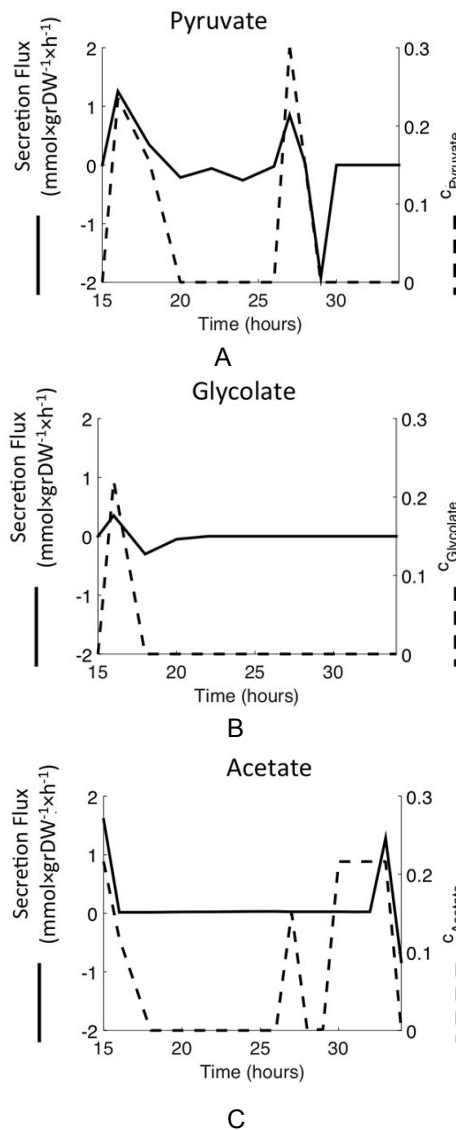


Figure 4-6: Metabolite secretion flux (dashed line), inferred from experimental data through the TEAM approach, and maximum coefficient of the pyruvate secretion flux in the objective function (full line), as predicted by invFBA (through OVA), are plotted as a function of time. Metabolite and gene expression data come from time-dependent measurements performed during batch aerobic growth of the bacterium *S. oneidensis* on lactate. The secreted metabolites are pyruvate in Fig. 4-6a, glycolate in Fig. 4-6b, and acetate in Fig. 4-6c. Positive fluxes reflect secretion of the metabolite in question, while negative fluxes reflect uptake.

in the context of the current work, the TEAM-inferred dataset gives the unique opportunity of obtaining putative objectives from genome-scale fluxes that reflect the metabolic effort of the bacterium as it undergoes changes throughout batch growth. As in the previous case of model-generated fluxes, the inverse problem admits a large space of possible solutions, i.e., maximally sparse objective functions that could give rise to the observed fluxes. Rather than providing specific arbitrary choices of objectives within the possible range, we report the outcome of OVA, as described above. Among all possible components of the identified objectives, we highlight the ones that can be compared directly with nontrivial experimental flux measurements, e.g., pyruvate secretion or uptake. The scope of OVA, or, more precisely, the reactions it can include in the objective function, was accordingly confined to exchange reactions. As seen in Fig. 4-6a, the largest pyruvate secretion component of the objective function (as computed by OVA) at different time points recapitulates the experimentally detected accumulation of pyruvate in the external medium, previously hypothesized to be the outcome of overflow metabolism (Vemuri et al., 2006). The same trend holds for glycolate (Fig. 4-6b) and acetate (Fig. 4-6c), although invFBA predicts optimization of acetate secretion at several time points leading up to the renewed secretion of acetate at 33 h. Applied to genome-scale fluxes obtained at each sample along the growth curve, the integer-programming variant of invFBA identified biomass production as the objective function at all time points. Optimization of biomass production agrees with these flux distributions originating from a growing *S. oneidensis* culture (Collins et al., 2012). These results lend confidence to the capacity of invFBA and OVA to correctly capture essential features of flux datasets. At the same time it highlights the importance of being cautious in the interpretation of objective functions, as a large component of the objective (e.g., pyruvate secretion) cannot be necessarily ascribed to a specifically evolved metabolic trait, and may rather be the outcome of

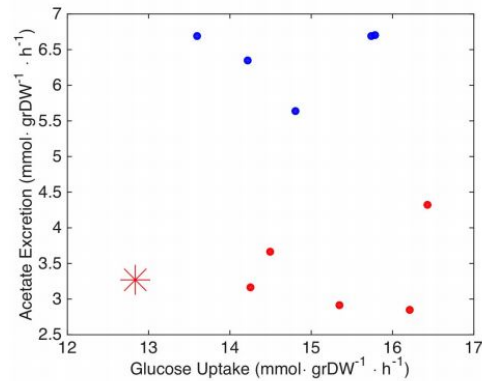
undesirable overflow metabolism.

4.3.4 The Inference of Linear Objective Functions in *E. coli* Strains that Underwent Long-term Evolutionary Experiments

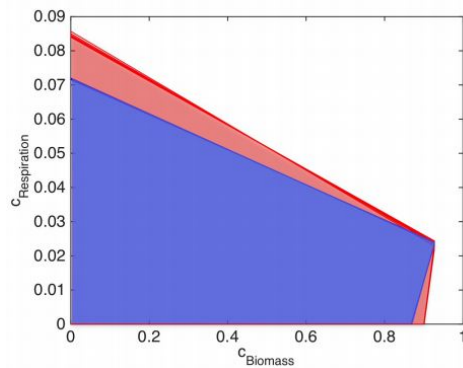
The most interesting application of invFBA is the inference of objective functions for microbial species and environments for which direct flux measurements are available and important questions on adaptation and optimality are at stake. An excellent example of this scenario is the availability of recently measured metabolic flux ratios (Harcombe et al., 2013) for some of the *E. coli* strains that underwent long-term experimental evolution in the Lenski Lab. These strains were evolved for 50,000 generations in glucose minimal medium, leading to important observations and discoveries on how adaptation works (Beg et al., 2012), (Vemuri et al., 2006), (Maddamsetti et al., 2015), and (Wiser et al., 2013). The reported flux ratios can be converted to flux vectors compatible with the stoichiometric constraints (see Appendix C).

The previous FBA analysis of metabolic activity in these strains has suggested that objective functions other than standard biomass flux maximization may best describe their evolutionary trajectory (Harcombe et al., 2013). Such analysis, however, only assessed the capacity of a small set of specific objective functions to lead to correct fluxes. Using invFBA, it is possible to reanalyze these flux data in an unbiased way, and characterize the space of objective functions compatible with the observations. A particularly striking feature of the flux data was the fact that six of the strains (five evolved and the ancestral) show comparatively low levels of acetate secretion (or, equivalently, high levels of glucose oxidation), as illustrated in Fig. 4-7A.

Upon applying invFBA to the complete set of measured flux data, we found, as in the aforementioned case studies, an infinite set of possible solutions, i.e., a convex polyhedral set of objective functions. Each of these objective functions if used in FBA could give the observed fluxes as an optimal solution (one of many alterna-



A



B

Figure 4.7: (A) Experimental measurements (by (Harcombe et al., 2013)) of acetate excretion and glucose uptake for the ancestral (red star) and evolved (blue and red dots) *E. coli* strains from Lenski's long-term evolutionary experiment. The red and blue colors are used here to highlight two distinct metabolic regimes that different strains seem to cluster around. (B) A projection (onto a two-dimensional subspace) of the set of objective functions compatible with experimentally measured fluxes. The graph is obtained through a two-dimensional version of OVA: for each possible value of the growth flux coefficient of the objective function ($c_{Biomass}$), one can find the minimal and maximal value of the objective function coefficient for the respiratory flux ($c_{Respiration}$), obtaining areas that correspond to objective functions compatible with the measured fluxes. Such regions can be computed for the ancestral and all evolved *E. coli* strains. The strains corresponding to the different metabolic regimes (blue and red dots in (A)) map onto different regions in the space of objectives, labeled with similar colors.

tive optima). Interestingly, the objective function consisting of only maximization of growth flux is not part of any of these sets (neither for the ancestral, nor for the evolved). Using OVA, one can find the maximal possible contribution of the growth flux in the objective function. While OVA provides ranges for the contributions to the objective by individual fluxes, it does not give any information on correlations and tradeoffs between the different flux components in the objective function. Visualization of the whole space of possible objective functions identified by invFBA is possible only upon reducing somehow the dimensionality. As illustrated in Fig. 4.7B, this can be achieved by projecting the space of possible optimal objectives onto a two-dimensional plane whose components are two specified biologically interesting fluxes. Upon visualizing this space in the plane of growth vs. respiration flux, again two sets of strains readily emerge, corresponding to the low- and high- respiration strains shown in Fig. 4.7B. For the same objective coefficient in the growth reaction, the low-acetate secreting strains have a lower maximal coefficient for respiration in the objective. This means that, despite the freedom of choice of objectives compatible with the experimental data, the signatures of how different fluxes may have adapted is still readable from the specific boundaries of the space of feasible objectives. Our analysis captures the dichotomy observed at the level of acetate and glucose transport fluxes, and suggests that the low acetate-excretion strains may be interpreted as having a higher maximization of respiration.

4.3.5 The Inference of a Quadratic Objective Function with Time Series Fluxes

We next test invFBA on metabolic fluxes simulated by TEAM (Collins et al., 2012) data, assuming the objective function is quadratic in this case. TEAM avoids assuming objective functions in favor of maximizing consistency with measured gene expression data.

In this experiment, we assume that the objective function is quadratic. As we mentioned in problem (4.12), we assume that the cell attempts to achieve its optimal performance while keeping the least distance from the previously observed state. The stoichiometric matrix $\mathbf{S} \in \mathbb{R}^{713 \times 870}$ is available and 15 flux vector samples were observed sequentially during the cell’s growth period. We applied our invFBA approach (4.17) on these 15 observed flux vectors sequentially to recover the parameters in the objective functions. The partial parameter values are shown in Table 4.4.

Table 4.4: Inferred parameter values.

	Reaction name	value	β value
t=1	'B-E'	0.7175	
t=2	'N-A-D-G-E'	-0.4051	1.01E-12
t=3	'B-E'	-0.7615	4.73E-07
t=4	'B-E'	-0.2788	1.59E-11
t=5	'N-A-D-G-E'	-0.4350	2.50E-12
t=6	'B-E'	-0.9205	1.21E-10
t=7	'B-E'	-0.9198	6.27E-10
t=8	'B-E'	-0.9094	3.71E-10
t=9	'B-E'	-0.7733	1.54E-07
t=10	'B-E'	-0.7226	4.98E-07
t=11	'B-E'	-0.9072	6.90E-11
t=12	'B-E'	-0.9194	1.03E-10
t=13	'B-E'	-0.7175	3.12E-07
t=14	'B-E'	-0.7871	7.79E-07
t=15	'B-E'	-0.7640	7.18E-07

More specifically, we applied the invFBA method (4.8) and (4.9) on the first sample and the InvFBA method (4.17) on the remaining 14 samples. In the second and the third columns, we report the reaction names of the largest elements in \mathbf{c} and their values. ‘N-A-D-G-E’ represents the reaction ‘N-Acetyl-D-glucosamine exchange’ and ‘B-E’ represents the reaction ‘Biomass Equation’. In the fourth column, we list the values of β in problem (4.12) for different samples. We note that the dominant objec-

tive function still corresponds to maximizing the biomass reaction flux. When $t = 2$ and $t = 4$, the objective function becomes equivalent to maximizing the N-Acetyl-D-glucosamine exchange rate. While surprising at first, it is intuitive considering that to maximize growth rate it suffices to maximize its limited nutrient. We also observed that the values of β are very small, which suggests that the effect of the quadratic term in the objective function is minimal. The observations are consistent with the TEAM simulation that measured samples during the cells growth period.

4.3.6 The Inference of a Kernelized Objective Function on *E. coli* Data

The *E. coli* model iAF1260 (Feist et al., 2007) has 3 different sizes: small size (24 metabolites and 32 reactions), midsize (65 metabolites and 84 reactions) and large size (1039 metabolites and 2077 reactions). In this example, we test invFBA method (4.22) on midsize model. To test the ability of invFBA to recover a non-parametric objective function and to evaluate the performance conveniently, we formulate the forward FBA problem as:

$$\min_{\mathbf{x} \in \mathcal{F}} Z(\mathbf{x}) = \min_{\mathbf{x} \in \mathcal{F}} (-x_{bio} + \mathbf{x}'\mathbf{\Sigma}\mathbf{x}), \quad (4.23)$$

where the objective function is to maximize the biomass growth rate and to minimize the weighted least square fluxes. Biologically, the second term aims at reducing internal flux circles. In problem (4.23), it is easy to compute the gradient of the objective function which is $\nabla Z(\mathbf{x}_i) = -\mathbf{e}_{bio} + 2\mathbf{\Sigma}\mathbf{x}$, where \mathbf{e}_{bio} represents the unit vector with 1 corresponding to the biomass growth reaction flux. The simulation process to infer the non-parametric objective function is shown in Fig. 4-8. In this simulation, we choose $\mathbf{\Sigma} = 10 \times \mathbf{I}$, and we simulate 40 ($N = 40$) noisy but feasible samples which we use as inputs to invFBA method (4.22). In this simulation process, a more complicated objective function is chosen than the one in (4.23). To validate

the performance, we compute the Pearson correlation coefficient between the actual gradient vector $\nabla Z(\mathbf{x}_i)$ and the inferred gradient vector $\nabla \hat{Z}(\mathbf{x}_i)$ at $\mathbf{x}_i, i = 1, \dots, 40$.

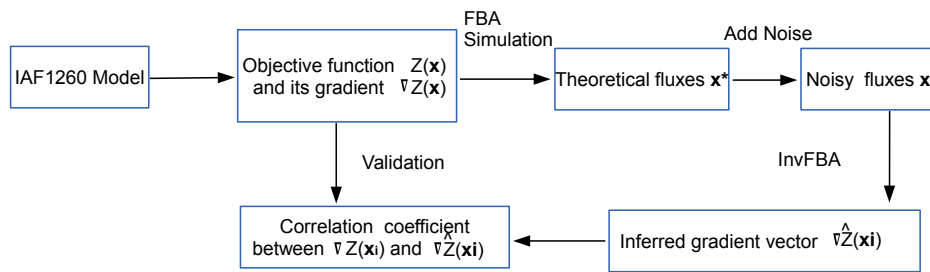


Figure 4-8: Simulation flowchart for the inference of a nonparametric objective function.

We plot the Pearson correlation coefficients for each sample point in Fig. 4-9. In this simulation, we tested RBF kernel, polynomial kernel and linear kernel. The linear kernel has the best performance. We can see that the inferred gradient vector is highly consistent with the theoretical gradient vector at those sample points.

4.4 Conclusions

Motivated by the need to understand how cells organize their metabolic networks, we have developed an inverse optimization framework to recover cellular objective functions from observed metabolic reactions fluxes. Within this framework, we introduced three different methods corresponding to a linear objective function, a quadratic objective function, and a non-parametric objective function, respectively. In each of these cases, we provided a tractable formulation of the inverse problem and numerical examples to validate our method.

If the objective function is linear or quadratic, our invFBA method results in a linear programming problem. Simulation results illustrate that invFBA can suc-

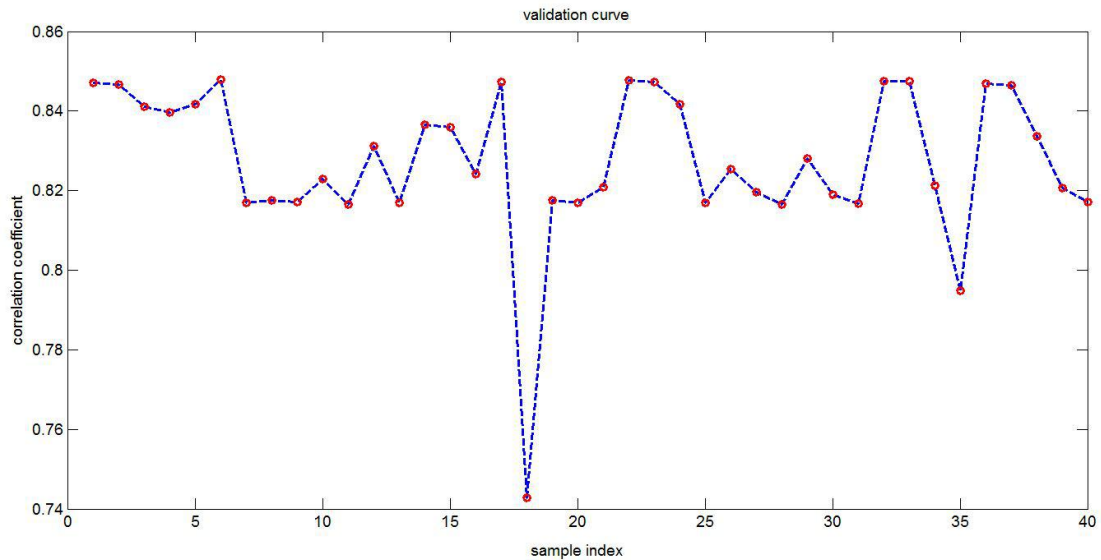


Figure 4-9: Performance evaluation of the nonparametric objective function.

cessfully recover the cellular objective functions and provide biological insight. If the objective function is too complicated to describe parametrically, we developed a non-parametric approach to infer its gradient. To that end we leveraged kernel methods. The corresponding inverse problem becomes a convex quadratic programming problem.

The proposed inverse optimization framework offers a way to infer cellular objective functions from experimental measurements and increase confidence in silico predictions.

Chapter 5

Optimal Allocation of Metabolic Functions among Organisms in a Microbial Ecosystem

Cellular metabolism is comprised of a complex network of biochemical reactions, which collectively provide every living cell with energy currency and building blocks for maintenance and reproduction. Microbial organisms can display a broad diversity of metabolic functions, which combine the ability to endogenously produce specific molecules from precursors, with a vast array of specialized transport capabilities. In microbial communities, each microbial species may secrete into the environment metabolites that other organisms may be able to use as precursors, potentially giving rise to cross-feeding interactions. Furthermore, different organisms may compete for environmentally available common resources. These metabolic interdependencies form complex and poorly understood ecological networks, likely to play a key role in the dynamics of microbial communities. In parallel to trying to understand the interactions and dynamics in natural communities, there has been rising interest in the possibility of designing engineered communities for a number of applications, including human health (Cho and Blaser, 2012), waste water treatment (Daims et al., 2006) and production of biofuels (Shong et al., 2012).

Despite the growing availability of experimental data for a diverse range of complex natural microbial communities, the full characterization and understanding of these communities is still a challenge (Zomorodi and Segrè, 2016). To solve this prob-

lem, efficient computational techniques and mathematical modeling tools are required. In recent years, there has been an incredible development of techniques specifically aimed at trying to predict computationally the metabolic activity of microbial systems and communities. In particular, one can view genome-scale and ecosystem-level metabolism as a resource allocation problem. Under a steady-state approximation, this problem can be efficiently addressed using linear programming, as done by *Flux Balance Analysis (FBA)* (Edwards et al., 2001). In the FBA formulation for a single species, one typically uses linear optimization to identify the set of fluxes that maximize a cellular objective (e.g., growth) subject to constraints capturing mass balance of each metabolite and bounds that reflect the composition of the growth medium.

In this chapter, inspired by previous work on ecosystem-level FBA (Zomorodi and Segrè, 2016; Khandelwal et al., 2013; Klitgord and Segre, 2010; Stolyar et al., 2007; Zomorodi and Maranas, 2012), we focus on metabolic resource allocation in a microbial community, and set the stage for asking questions about how different metabolic reactions may be partitioned between different species in order to achieve a certain community-level goal. We formulate the problem as a *Mixed Integer Linear Programming (MILP)* problem inspired by FBA and our recently proposed Inverse FBA method (Zhao et al., 2015).

The remainder of the chapter is organized as follows. In Section 5.1, we define the problem mathematically and formulate it as an MILP problem. In Section 5.2, we test our method on an artificial microbial community containing two different toy species and assess the method’s effectiveness. In Section 5.3, we apply our method to a community of two simplified bacteria and evaluate their ability to co-exist. Finally, concluding remarks and discussions appear in Section 5.4.

5.1 Problem Formulation

5.1.1 Community-level Flux Balance Analysis

We introduce a “universal stoichiometric matrix,” denoted by \mathbf{S} , which expresses mass balance for all possible reactions in a microbial community irrespective of the organism they belong to. Specifically, $\mathbf{S} \in \mathbb{R}^{M \times N}$ where $M = M_e + M_i$ represents the number of distinct metabolites and $N = N_u + N_e + N_i$ represents the number of distinct reactions. The M distinct metabolites consist of two types: M_e external and M_i internal metabolites. The external metabolites exist in the shared extracellular environment of all organisms and the internal metabolites are intracellular. There are 3 different types of reactions: N_u uptake reactions, N_e exchange reactions and N_i internal reactions. The availability of nutrients (external metabolites) from the environment is encoded in the uptake reactions. With exchange reactions, organisms transport these metabolites between their intracellular compartment and the extracellular environment. Internal reactions take place among internal metabolites.

Assuming that the community consists of K different species, we use \mathbf{S}^k , $k = 1, \dots, K$, to denote the (individual) stoichiometric matrix of species k . This is the matrix that expresses mass balance constraints for all reactions utilized by the metabolic network of that species. The columns in \mathbf{S}^k are a subset of the columns in \mathbf{S} . The matrices \mathbf{S}^k may have identical columns (reactions) if more than one species use the same reaction. Without loss of generality, the structure of \mathbf{S} is shown in Fig. 5.1. Given the matrix \mathbf{S} and the number of species K , our goal is to design the individual stoichiometric matrices \mathbf{S}^k that render each species and the community viable, that is, satisfying appropriate optimality criteria.

In order to formulate the design problem, we first reformulate the universal stoichiometric matrix \mathbf{S} to construct putative stoichiometric matrices for each species in the community (Klitgord and Segre, 2010). In particular, we construct a community

		N_u	N_e	N_i
		Uptake Reactions	Exchange Reactions	Internal Reactions
M_e	{	\mathbf{S}^u	\mathbf{S}^{e1}	$\mathbf{0}$
M_i	{	$\mathbf{0}$	\mathbf{S}^{e2}	\mathbf{S}^i
		External Metabolites		Internal Metabolites

Figure 5.1: The structure of the universal stoichiometric matrix \mathbf{S} . Block \mathbf{S}^u represents the set of uptake reactions used to absorb nutrients from the environment. Blocks $[\mathbf{S}^{e1}; \mathbf{S}^{e2}]$ represent the set of exchange reactions between external and internal metabolites. \mathbf{S}^i represents the set of internal reactions among internal metabolites.

stoichiometric matrix \mathbf{S}^c whose structure is shown in Fig. 5.2. The block matrices \mathbf{S}^u , \mathbf{S}^{e1} , and \mathbf{S}^{e2} in \mathbf{S}^c are consistent with those in \mathbf{S} . Organisms in the community share the same nutrients and uptake reactions. Because there are K organisms in the community, we replicate the block $[\mathbf{S}^{e2}, \mathbf{S}^i]$ that includes exchange reactions and internal reactions K times and diagonally arrange them in \mathbf{S}^c . This arrangement represents the process according to which organisms absorb the nutrients via the same uptake reactions and then allocate them to individual organisms in the community.

After obtaining the internal metabolites via the exchange reactions, internal reactions take place inside each organism. This construction leads to a community stoichiometric matrix $\mathbf{S}^c \in \mathbb{R}^{M_c \times N_c}$, where $M_c = M_e + KM_i$ and $N_c = N_u + K(N_e + N_i)$.

Notice that \mathbf{S}^c has one block column for uptake reactions (N_u columns) and K block columns (of dimension $N_e + N_i$), one for each organism, including all exchange and internal reactions.

To capture design choices, we introduce a binary putative vector $\mathbf{t} = (t_1, \dots, t_{N_c})$, where $t_i \in \{0, 1\}$ is a binary variable, indicating whether the i -th reaction is included or not in the corresponding organism (cf. Fig. 5.2). With \mathbf{t} and \mathbf{S}^c available, we can partition \mathbf{S}^c to K individual matrices, \mathbf{S}^k , by removing columns j with $t_j = 0$.

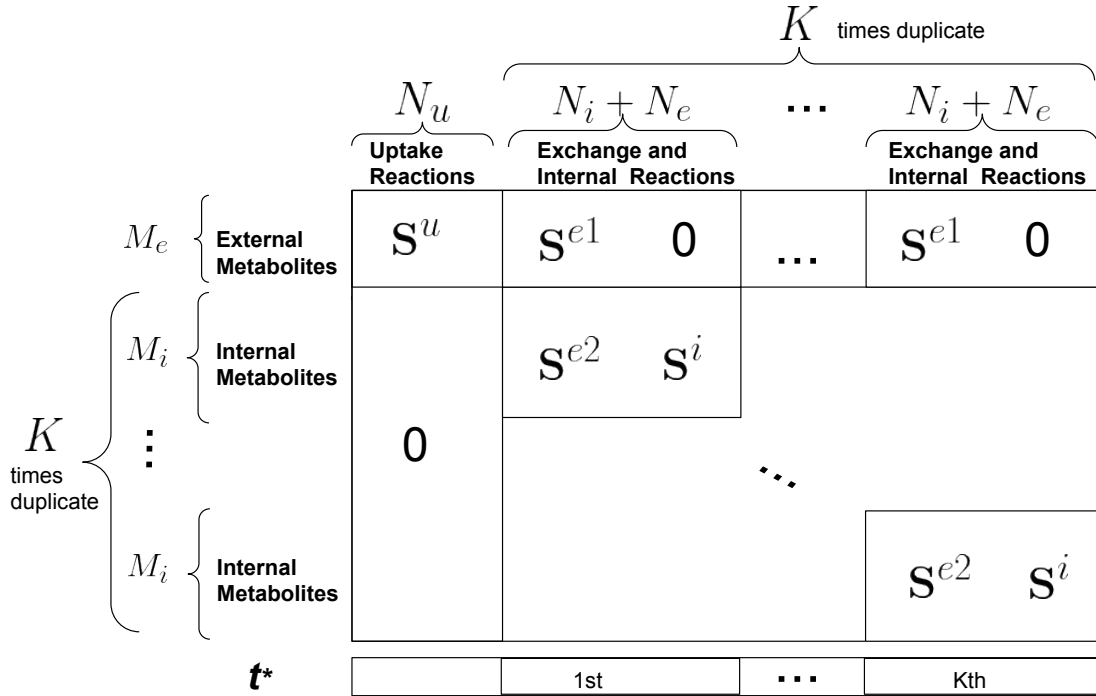


Figure 5.2: The structure of the stoichiometric matrix for the whole community $\mathbf{S}^c \in \mathbb{R}^{M_e \times N_c}$.

If \mathbf{S}^c and \mathbf{t} are known, we can formulate the FBA problem to predict the flux distribution vector in the entire community as:

$$\begin{aligned} \max_{\mathbf{x}} \quad & \mathbf{c}'\mathbf{x} \\ \text{s.t.} \quad & \mathbf{S}^c\mathbf{x} = \mathbf{0}, \\ & \text{diag}(\mathbf{t})\mathbf{x}_{lb} \leq \mathbf{x} \leq \text{diag}(\mathbf{t})\mathbf{x}_{ub}, \end{aligned} \quad (5.1)$$

where $\mathbf{c} \in \mathbb{R}^{N_c}$ corresponds to the community objective (e.g., maximizing the total

biomass growth rate, maximizing the total uptake rate of a limited nutrient, etc. (Sauer et al., 1998)) and \mathbf{x}_{ub} and \mathbf{x}_{lb} are upper and lower bounds on fluxes. The constraints $diag(\mathbf{t})\mathbf{x}_{lb} \leq \mathbf{x} \leq diag(\mathbf{t})\mathbf{x}_{ub}$ guarantee that the flux of a reaction that is not included in an organism is set to 0. Problem (5.1) is a linear programming problem and it can be solved efficiently. The optimal solution of (5.1), denoted by \mathbf{x}^* , represents the flux vector for the entire community.

We now write another linear programming problem which, as we will see, reduces to the dual of (5.1) when all the non-active (with a corresponding $t_j = 0$) reactions j and their fluxes x_j are removed from (5.1).

$$\begin{aligned} \min_{\mathbf{p}, \mathbf{q}_1, \mathbf{q}_2} \quad & \mathbf{q}'_2 \mathbf{x}_{ub} - \mathbf{q}'_1 \mathbf{x}_{lb} \\ \text{s.t.} \quad & \mathbf{S}' \mathbf{p} - \mathbf{q}_1 + \mathbf{q}_2 - \mathbf{c} - L(\mathbf{1} - \mathbf{t}) \leq \mathbf{0}, \\ & \mathbf{S}' \mathbf{p} - \mathbf{q}_1 + \mathbf{q}_2 - \mathbf{c} + L(\mathbf{1} - \mathbf{t}) \geq \mathbf{0}, \\ & \mathbf{q}_1, \mathbf{q}_2 \geq \mathbf{0}, \\ & \mathbf{q}_1, \mathbf{q}_2 \leq L\mathbf{t}, \end{aligned} \tag{5.2}$$

where \mathbf{p} , \mathbf{q}_1 and \mathbf{q}_2 are dual variables and L is a sufficiently large constant. The role of L is to affect the dual feasibility constraints. The primal variable x_i in (5.1) corresponds to the i th dual constraint in $-L(\mathbf{1} - \mathbf{t}) \leq \mathbf{S}' \mathbf{p} - \mathbf{q}_1 + \mathbf{q}_2 - \mathbf{c} \leq L(\mathbf{1} - \mathbf{t})$. For the reactions included in the corresponding organism we have $t_i = 1$ for i in some index set \mathcal{I} . For these $i \in \mathcal{I}$, the corresponding dual constraint becomes $[\mathbf{S}' \mathbf{p}]_i - q_{1i} + q_{2i} - c_i = 0$. For those reactions j , however, which are not included in the corresponding organism ($t_j = 0$, $j \notin \mathcal{I}$), the corresponding dual constraint becomes $-L \leq [\mathbf{S}' \mathbf{p}]_j - q_{1j} + q_{2j} - c_j \leq L$, which is trivially satisfied for a large enough L . Similarly, the dual variables \mathbf{q}_1 and \mathbf{q}_2 correspond to the lower and upper bounds on \mathbf{x} in (5.1). If x_i is non-zero ($t_i = 1$), q_{1i} and q_{2i} are both non-negative and can take arbitrarily large values bounded by the large constant L . If, however, x_j is set to zero ($t_j = 0$), then q_{1j} and q_{2j} are set to 0 as well to avoid an unbounded objective value in (5.2).

5.1.2 Community Stoichiometric Partitioning

In practice, the putative vector \mathbf{t} is unavailable. In fact, identifying \mathbf{t} is our primary goal in this chapter. Based on the analysis of the community-level FBA and the corresponding dual problem, the feasible set of \mathbf{t} , denoted by \mathcal{F} can be represented by the following constraints:

$$\begin{aligned}
\mathbf{S}^c \mathbf{x} &= \mathbf{0}, \\
\mathbf{x} &\geq \text{diag}(\mathbf{t}) \mathbf{x}_{lb}, \\
\mathbf{x} &\leq \text{diag}(\mathbf{t}) \mathbf{x}_{ub}, \\
\mathbf{S}' \mathbf{p} - \mathbf{q}_1 + \mathbf{q}_2 - \mathbf{c} - L(\mathbf{1} - \mathbf{t}) &\leq \mathbf{0}, \\
\mathbf{S}' \mathbf{p} - \mathbf{q}_1 + \mathbf{q}_2 - \mathbf{c} + L(\mathbf{1} - \mathbf{t}) &\geq \mathbf{0}, \\
\mathbf{q}_1, \mathbf{q}_2 &\geq \mathbf{0}, \\
\mathbf{q}_1, \mathbf{q}_2 &\leq L\mathbf{t}, \\
t_i &\in \{0, 1\}, \\
\mathbf{c}' \mathbf{x} &= \mathbf{q}'_2 \mathbf{x}_{ub} - \mathbf{q}'_1 \mathbf{x}_{lb}.
\end{aligned} \tag{5.3}$$

These constraints capture primal feasibility, dual feasibility and strong duality conditions, respectively. Other than \mathcal{F} , additional regularization constraints on \mathbf{t} are necessary to obtain a biologically meaningful solution. Specifically, we impose

$$\mathbf{t}_{min} \leq \mathbf{R}\mathbf{t} \leq \mathbf{t}_{max}, \tag{5.4}$$

where $\mathbf{R} \in \mathbb{R}^{M_r \times N_c}$ is a regularization matrix. The number of rows M_r depends on the number of regularization constraints we wish to introduce. Such constraints can, for example, impose upper and lower bounds on the number of reactions active in each organism, and/or enforce upper and lower bounds on the number of repeated reactions in different organisms (hence, controlling for community diversity and robustness). The regularization constraints enable us to partition the community (\mathbf{S}^c) into individual species (\mathbf{S}^k) that are biologically meaningful.

The problem of identifying \mathbf{t} can now be formulated as the following *Mixed Integer*

Linear Programming (MILP) problem:

$$\begin{aligned} \max_{\mathbf{x}, \mathbf{p}, \mathbf{q}_1, \mathbf{q}_2, \mathbf{t}} \quad & f(\mathbf{x}, \mathbf{p}, \mathbf{q}_1, \mathbf{q}_2, \mathbf{t}) \\ \text{s.t.} \quad & \mathbf{x}, \mathbf{p}, \mathbf{q}_1, \mathbf{q}_2, \mathbf{t} \in \mathcal{F}, \\ & \mathbf{t}_{min} \leq \mathbf{R}\mathbf{t} \leq \mathbf{t}_{max}, \end{aligned} \tag{5.5}$$

where $f(\cdot)$ is a global linear objective function. Relevant objective functions could include maximization of the total uptake rate of some nutrients or maximization of the total biomass growth rate. By solving problem (5.5), we can obtain the putative vector \mathbf{t}^* and the flux distribution \mathbf{x}^* for the entire community. Then, we can partition \mathbf{S}^c into a set of \mathbf{S}^k stoichiometric matrices, one for each species $k = 1, \dots, K$, based on \mathbf{t}^* .

5.2 An Artificial Toy Model

In this section, we will demonstrate the use of our method with a very simple microbial community toy model consisting of two different organisms (Klitgord and Segre, 2010). In particular, we first analyze the community-level FBA for this model and we then validate our method by comparing the reaction partition we obtain with ground truth.

5.2.1 Model Description

The metabolic networks of the two individual species are depicted in Fig. 5-3. The two species share the same environment, the same metabolites (X, Y, Z), the same uptake secretion properties, and the same usage of precursors to produce biomass. They differ, however, in their internal metabolic reactions ($R1$ and $R2$). The arrows in Fig. 5-3 represent reactions. The boxes represent distinct (intracellular) metabolic compartments. This level of compartmentalization is usually sufficient to appropriately model individual species.

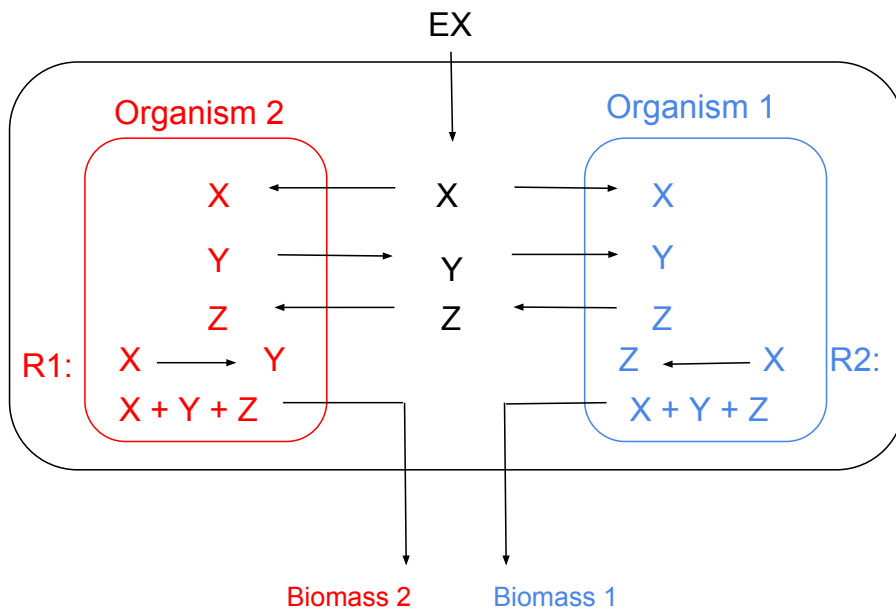


Figure 5-3: The structure of the artificial community with two different organisms. Both organisms 1 and 2 uptake the same external metabolite (EX) from the environment. Organism 2 produces Y only via $R1$ and obtains Z from organism 1. Organism 1 produces Z only via $R2$ and obtains Y from organism 2. Organisms 1 and 2 have the same biomass growth reaction which involves X , Y and Z . Organisms 1 and 2 cannot grow without each other.

5.2.2 FBA at the Community Level

The universal stoichiometric matrix \mathbf{S} of this model is shown in Fig. 5-4. $\mathbf{S} \in \mathbb{R}^{6 \times 7}$ is like a library which contains all possible reactions in this community. Following the structure in Fig. 5-2, we construct the community stoichiometric matrix $\mathbf{S}^c \in \mathbb{R}^{9 \times 13}$ shown in Fig. 5-5. The combination of \mathbf{S}^c and \mathbf{t} mathematically describes the true topology of the community metabolic network.

With \mathbf{S}^c , \mathbf{t} , \mathbf{x}_{lb} and \mathbf{x}_{ub} available, we solve the community-level FBA problem (5.1). The objective is to maximize the sum of biomass growth rates of organisms 1

		Uptaking reaction	Exchange reactions			Internal reactions		
		X_u	X_{ex}	Y_{ex}	Z_{ex}	R1	R2	Biomass
External Metabolites	EX	-1	-1	0	0	0	0	0
	EY		0	-1	0	0	0	0
	EZ		0	0	-1	0	0	0
Internal Metabolites	X		1	0	0	-1	-1	-1
	Y		0	1	0	1	0	-1
	Z		0	0	1	0	1	-1

Figure 5-4: The universal stoichiometric matrix \mathbf{S} . The rows of \mathbf{S} include three external metabolites (EX, EY, EZ) and three internal metabolites (X, Y, Z). The columns of \mathbf{S} include one uptake reaction, three exchange reactions, and three internal reactions.

	X_u	X_{ex1}	Y_{ex1}	Z_{ex1}	R11	R21	Biomass1	X_{ex2}	Y_{ex2}	Z_{ex2}	R12	R22	Biomass2
EX	-1	1	0	0	0	0	0	1	0	0	0	0	0
EY	0	1	0	0	0	0	0	0	1	0	0	0	0
EZ	0	0	1	0	0	0	0	0	0	1	0	0	0
X1		-1	0	0	-1	-1	-1						
Y1		0	-1	0	1	0	-1						
Z1		0	0	-1	0	1	-1						
X2								-1	0	0	-1	-1	-1
Y2								0	-1	0	1	0	-1
Z2								0	0	-1	0	1	-1
\mathbf{t}	1	1	1	1	0	1	1	1	1	1	1	1	0
Ground Truth													
x_{lb}	-12	-6	-6	-6	0	0	1	-6	-6	-6	0	0	1
x_{ub}	0	6	6	6	0	5	5	6	5	5	5	5	0
\mathbf{x}^*	-12	-6	-2	2	0	4	2	-6	2	-2	4	0	2
maximize the sum of biomass growth rates													

Figure 5-5: The community stoichiometric matrix \mathbf{S}^c is shown in the top figure. The binary vector \mathbf{t} denotes the ground truth, consistent with Fig. 5-4. The optimal flux distribution \mathbf{x}^* is shown in the bottom. To distinguish the internal metabolites in the two organisms, we use $X1, Y1$ and $Z1$ to denote the internal metabolites in organism 1, and $X2, Y2$ and $Z2$ for the internal metabolites in organism 2.

and 2 (mathematically $x_{b1} + x_{b2}$). In the rest of this chapter, we use abbreviations bi to denote the biomass reactions and fluxes of organism i . The predicted flux distribution \mathbf{x}^* in steady-state is shown in Fig. 5-5. Both organisms 1 and 2 take the external metabolite EX with a rate equal to 6. Organism 2 transports the external metabolite EX to the internal metabolite X with a rate of 6 and generates Y with a rate of 4. Organism 1 transports EX to the internal metabolite X with a rate of 6 and generates Z with a rate of 4. At the same time, Y is transported from organism 2 to

1 through EY with a rate of 2 and Z is transported from organism 1 to 2 through EZ with a rate of 2. With these reactions taking place simultaneously, both organisms can grow with a rate of 2. We note that organism 2 needs to feed organism 1 with metabolite Y and organism 1 needs to feed organism 2 with metabolite Z for both of them to grow. It is clear that neither organism can grow on its own. These fluxes obtained from FBA provide us with the ground truth to validate our method in the next subsection.

5.2.3 Identification of the Putative Vector \mathbf{t}

In the previous subsection, we assume the availability of \mathbf{t} for solving the community-level FBA problem. Next, we assume that the metabolic network structure of each individual organism is unknown and the putative vector \mathbf{t} is not available. We would like to identify \mathbf{t} based on available \mathbf{S} , \mathbf{x}_{lb} , \mathbf{x}_{ub} and the regularization constraints on \mathbf{t} . We apply the formulation (5.5) to obtain \mathbf{t}^* . In this case, we impose the regularization constraints on \mathbf{t} described in the sequel.

First, we enable the biomass reactions for each organism, which implies (5.6), and set the lower bounds of x_{b1} and x_{b2} to 1.

$$t_{b1} = 1, \quad t_{b2} = 1. \quad (5.6)$$

The interactions between organisms 1 and 2 must produce EY and EZ in the growth medium. For this reason, we impose constraints (5.7).

$$t_{Y_{ex1}} = 1, \quad t_{Y_{ex2}} = 1, \quad t_{Z_{ex1}} = 1, \quad t_{Z_{ex2}} = 1. \quad (5.7)$$

We also impose in (5.8) bounds on the number of enabled reactions, where the lower bound is determined by (5.6) and (5.7) and the upper bound is determined by

the number of columns in \mathbf{S}^c .

$$6 \leq \sum_{i=1}^{13} t_i \leq 13. \quad (5.8)$$

As the global objective function in (5.5) we select $f(\mathbf{x}, \mathbf{p}, \mathbf{q}_1, \mathbf{q}_2, \mathbf{t}) = x_{b1} + x_{b2}$, which seeks to maximize the total biomass growth rate in this community. The solutions (\mathbf{t}^* and \mathbf{x}^*) are shown in Fig. 5-6. The inferred \mathbf{t}^* is identical with the ground truth and the flux distribution \mathbf{x}^* under steady-state is consistent with the flux vector in the bottom of Fig. 5-5. Therefore, our method can successfully identify the structure of each organism and predict the steady-state flux distribution for the entire community simultaneously.

	\mathbf{t}^*	\mathbf{x}^*
X_u	1	-12
X_ex1	1	-6
Y_ex1	1	2
Z_ex1	1	-2
R11	1	4
R21	0	0
Biomass1	1	2
X_ex2	1	-6
Y_ex2	1	-2
Z_ex2	1	2
R12	0	0
R22	1	4
Biomass2	1	2

Figure 5-6: The results of identifying \mathbf{t}^* and the corresponding flux vector \mathbf{x}^* . They are consistent with the ground truth in Fig. 5-5.

5.3 A Simplified Bacterial Community

5.3.1 Model Description

In this section, we apply our method to a community with two simplified bacterial networks. We start by assuming that two identical bacterial species are present in a community, but constrain the number of metabolic reactions available in each species.

The fully functional bacterial model is a generic bacterial core metabolic network from (Covert et al., 2001) which contains 7 external and 12 internal metabolites involved in 28 reactions, including 7 uptake reactions, 7 exchange reactions, and 14 internal reactions. The corresponding universal stoichiometric matrix $\mathbf{S} \in \mathbb{R}^{19 \times 28}$ is shown in Fig. 5-7. We apply FBA with the objective of maximizing the biomass growth rate of this single bacterial model and the corresponding flux distribution \mathbf{x}^* is shown at the bottom of Fig. 5-7. The maximum biomass growth rate is 3.63 under the current growth conditions.

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	
		uptake reactions							exchange reactions							internal reactions														
		IClear	IC2eaf	Ieaf	Ie2eaf	Ieaf	Ie2eaf	IO2eaf	Ic1	Ic2	Ic	Iv	Iv	Ih	Ioz	R1	R2a	R2b	R2	R4	R5a	R5b	R6	R7	R8a	R8b	R9a	R9b	Growth	BiomassOut
1	IClear	1	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	IC2eaf	0	1	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	Ieaf	0	0	1	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	Ie2eaf	0	0	0	1	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	Ieaf	0	0	0	0	1	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	Ie2eaf	0	0	0	0	0	1	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	IO2eaf	0	0	0	0	0	0	1	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	IC	0	0	0	0	0	0	0	1	1	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	IC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	-1	-1	-1	0	0	0	0	0	0	0	0	0	0	0
10	IC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	-1	0	-1	0.8	0.8	-1	-1	0	0	0	-1	0	
11	IC	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	IC	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	IC	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	IC	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	H	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	CCP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	ATP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	2	-2	0	0	0	0	0	0	0	0	0	0	0	0
18	NADH	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	-2	0	0	0	0	0	0	0	0	0	0	0	0
19	Biomass	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	\mathbf{x}^*	20.0000	30.0000	36.9558	40.5000	0.0000	0.0000	6.0000	20.0000	30.0000	0.0000	###	40.5000	0.0000	6.0000	30.0000	25.2634	0.0000	3.6276	3.6276	0.0000	0.0000	5.6592	17.5000	3.6276	0.0000	6.0000	3.6276	3.6276	

Figure 5-7: The universal stoichiometric matrix \mathbf{S} and the corresponding flux vector by maximizing the biomass growth rate for an individual bacterial model from (Covert et al., 2001) under the current growth conditions.

5.3.2 Community Analysis

In this test, we put two partially functional bacteria in the same growth medium. To analyze the relationship between the total biomass growth rate and the number of active internal reactions in each strain, we use the following objective function:

$$f(\mathbf{x}, \mathbf{p}, \mathbf{q}, \mathbf{t}) = x_{b1} + x_{b2} - \lambda \sum_{j=1}^2 \sum_{j \in \mathcal{R}_j} t_j, \quad (5.9)$$

where \mathcal{R}_j represents the set of internal reactions in strain j , $j = 1, 2$, and λ is a positive tunable constant used to capture the trade-off between total biomass growth rate and the sparsity of each individual bacterial strain. In this case, to guarantee the

growth of both strains 1 and 2, we set the lower bounds of the corresponding fluxes to 0.5. Then, we apply different values of λ and the corresponding results are shown in Table 5.1.

Table 5.1: Biomass growth rate vs. the number of active reactions in the community.

	$\lambda = 0$	$\lambda = 1$	$\lambda = 10$	$\lambda = 100$	$\lambda = 1000$
Total biomass growth rate	3.63	3.16	3.16	1.23	1.05
# of all reactions	34	36	33	31	29
N_i in strain 1	9	10	9	10	9
N_i in strain 2	9	9	9	6	8

When $\lambda = 0$, the total biomass growth rate achieves the maximum value (3.63), which is consistent with the FBA result in Fig. 5.7. The fully functional bacterial organism contains 12 internal reactions, however, there are only 9 internal reactions in each strain 1 and 2. This implies that not all internal reactions are required to maintain the maximum biomass growth rate. As λ increases, the total biomass growth rate decreases and the total number of reactions decreases as well. When $\lambda \geq 1000$, the total biomass growth rate achieves its lower bound (1.0), and the total number of reactions is reduced compared to $\lambda = 0$.

In order to better visualize the community of two bacterial strains (obtained with $\lambda = 0$) and the corresponding flux distribution, we use the network visualization software VisAnt (Hu et al., 2004) to draw Fig. 5.8. VisAnt has the unique capability of concurrently visualizing intracellular metabolism and ecological metabolite-mediated interaction networks. Note that none of the organism identified as a solution to this problem is able to produce all biomass components by itself. As seen in the figure, one of the two resulting networks (right) utilizes oxygen available from the external environment, and a molecule (F) secreted by the other organism to produce hydrogen through redox reactions. The other organism (left) utilizes multiple carbon sources, and the hydrogen transferred from the partner organism, to run most of the

5.4 Conclusions

Motivated by the need to interpret how organisms interact in a microbial community, we have developed an optimization framework to identify the individual metabolic network of each organism in the community and to predict the resulting flux distributions. We formulated the problem of allocating reactions to organisms as an MILP problem. We tested the method on both a toy artificial community and a community composed of two bacterial organisms with a simplified core metabolism. In both cases, the method helped us identify the individual metabolic network topologies and elucidate the interaction between species in the microbial community.

The proposed method provides a meaningful way to analyze and simulate the combinatorial complexity of metabolite-mediated interactions between multiple organisms in a microbial community. It also offers a new platform for the rational design of organisms and communities towards future synthetic ecology applications.

Chapter 6

Conclusions

We start this chapter by summarizing our current progress and contributions in Section 6.1. Then we discuss directions for future work in Section 6.2.

6.1 Summary

In this thesis, we have considered three biomedical problems involving optimization and control methods: the automated medication dosage control problem, the cellular objective inference problem and the metabolic function allocation problem.

In the problem of automating medication dosage, we have developed two main approaches to predict the effect of bivalirudin in cardiac surgical patients. The first approach is model-free and leverages regularized regression. The second approach is model-based and constructs a specific model that captures how bivalirudin affects PTT values. The mathematical models and prediction approaches provide a better reference to guide the optimal therapy in cardiac patients in need of bivalirudin.

Based on the specific dynamic system model of bivalirudin acting in cardiac surgical patients, we have developed two methods for synthesizing a controller to regulate the bivalirudin infusion rate and to induce the effect of bivalirudin within a desirable range. The first method assumes that the model parameters are available and develops a control law that tracks a physician-specified reference output signal. Our second method considers patients for which past clinical records are sparse and accurate model parameters are not readily available. We have developed an indirect

control scheme (indirect MRAC) that first estimates the model parameters and then adapts the corresponding controller based on these estimates. Alternatively, a direct control scheme (direct MRAC) that adapts the controller without estimating the model parameters first is also developed. Testing of these schemes against actual patient data from a hospital showed that the direct MRAC is more efficient than the indirect version. The methods we developed can be seen as key steps toward automation of dosage decisions in the hospital setting, which can help eliminate errors and neutralize the inexperience of residents who are currently responsible for these decisions. In addition, such mathematical ideas and methods may be useful to test medication dosing strategies and may provide a mathematical mechanism for development and testing of nomograms.

In the cellular objective inference problem, we have introduced a new theoretical framework that enables a more formal, efficient and systematic analysis of the possible objective functions. The framework is compatible to linear objectives, to quadratic objectives, and to non-parametric convex objectives. The invFBA approach we developed constitutes a specific instance of a broader, powerful, and still highly unexplored avenue for posing and solving inverse optimization problems. The relevance of our algorithm may extend beyond the realm of metabolic network modeling, for example to game-theoretic models of traffic equilibria in transportation problem and price-setting games in economics.

In the problem of allocating metabolic functionality to distinct organisms living together, which can be used to interpret how organisms interact in a microbial community, we have developed an optimization-based framework to identify the metabolic network structure of the individual organisms and to predict the flux distribution for the whole community. This problem has been formulated as a mixed integer linear programming problem. We tested the method on an artificial community and a com-

munity of two simplified bacteria. For both cases, the method helped us identify the individual network topology and understand interaction behaviors in the microbial community. The proposed method provides a meaningful way to analyze and simulate the combinatorial complexity of metabolite-mediated interactions between multiple organisms in a microbial community. It also offers a new platform for the rational design of organisms and communities towards future synthetic ecology applications.

6.2 Future Works

Continuing with the metabolic functionality allocation problem, we already proposed a Mixed Integer Linear Programming (MILP) based approach and tested it on two simple models. However, the real microbial community includes a large number of organisms. Therefore, we need to perform more assessments of the method's effectiveness and scalability before applying it to a more realistic microbial community model. The first necessary assessment is to evaluate our method on an artificial community which includes more than two organisms.

In addition to the large number of organisms in the real microbial community, the complicated metabolic network structure is another challenge for our current method. A full stoichiometric matrix of an organism may include more than thousands of metabolites and reactions. Because our current method is based on mixed integer linear programming, the computational task may become very heavy in the large scale case. For this reason, we should evaluate the performance (e.g., optimality, running time, etc.) of our method in the large scale case. Furthermore, an appropriate relaxation of the current method is required in order to ensure scalability to large networks.

Appendix A

LASSO Version of InvFBA

We have already discussed that FBA in practice is used with an additional sparsity-inducing ℓ_1 -norm regularization (cf. (4.2)). This motivates the use of a similar regularization in the context of InvFBA. Alternatively, a convenient way to combine the least optimality gap and sparsity of objective function is to formulate the invFBA problem as:

$$\begin{aligned}
 \min_{\epsilon_i, \mathbf{p}^i, \mathbf{q}_1^i, \mathbf{q}_2^i, \mathbf{c}} \quad & \sum_{i=1}^N \epsilon_i + \lambda \sum_{j=1}^n |c_j| & (\text{A.1}) \\
 \text{s.t.} \quad & \sum_{j=1}^n c_j = 1, \\
 & \mathbf{p}^{i'} \mathbf{S} - \mathbf{q}_1^{i'} + \mathbf{q}_2^{i'} = \mathbf{c}', \quad \forall i, \\
 & \mathbf{q}_2^{i'} \mathbf{v}_{ub} - \mathbf{q}_1^{i'} \mathbf{v}_{lb} - \epsilon_i = \mathbf{c}' \mathbf{x}_i, \quad \forall i, \\
 & \mathbf{q}_1, \mathbf{q}_2 \geq \mathbf{0}, \quad \forall i, \\
 & \epsilon_i \geq 0, \quad \forall i,
 \end{aligned}$$

where λ is some tunable non-negative constant. This formulation is reminiscent of the *LASSO*-type regression (Hastie et al., 2009).

One can interpret non-zero elements in optimal solutions \mathbf{c} of (A.1) as corresponding to important metabolic fluxes that are critical in the FBA optimization context and provide a minimal description of the cellular objective function. Notice that formulation (A.1) is a linear programming problem, just like the forward FBA problem, which is not in general true for inverse optimization problems (Bertsimas et al., 2013).

Appendix B

Noise Generation

In order to generate simulated feasible flux vectors around a defined point, containing a given amount of noise, we devised the following optimization problem:

$$\begin{aligned}
 \max_{\mathbf{x}_i} \quad & \mathbf{r}'\mathbf{x}_i && \text{(B.1)} \\
 \text{s.t.} \quad & \mathbf{S}\mathbf{x}_i = \mathbf{0}, \\
 & \mathbf{v}lb \leq \mathbf{x}_i \leq \mathbf{v}ub, \\
 & \|\mathbf{x}_i - \mathbf{x}^*\| \leq \sigma^2,
 \end{aligned}$$

where \mathbf{r} is a random objective function, \mathbf{x}_i is the noisy flux distribution, \mathbf{x}^* is the pre-computed optimal flux distribution, and σ^2 denotes the largest Euclidean distance between optimal flux and noisy flux. Changing the value of σ^2 yields different magnitudes of noise.

Appendix C

Inference of Fluxes from Experimentally Measured Branching Ratios

To apply the invFBA algorithm and infer the objective function in *E. coli* strains that underwent long-term evolutionary experiments (LTEE), we needed to convert the ^{13}C -labeling raw measurements of flux ratios and uptake/secretion rates into central carbon metabolism flux values.

The dataset we used is obtained from (Harcombe et al., 2013). This dataset includes measurements for one ancestral strain (Anc) and ten evolved strains (named, as in the original paper, $A+1, A+2, A+3, A+4, A+5, A-1, A-2, A-3, A-4, A5, A-6$). For each strain, six pathway branch ratios (Ser from glycolysis, PYR through ED pathway, upper bound of PEP through PPP, lower bound of PYR from MAL, OAA from PEP, PEP from OAA; see also (Zamboni et al., 2005)) and three external fluxes (glucose uptake rate, acetate excretion rate and growth rate) are available. All fluxes are part of a central carbon metabolism model for *E. coli* with stoichiometric matrix \mathbf{S} . In our formulation, we call R_s^i ($s = 1, \dots, 11; i = 1, \dots, 6$) the measured pathway branch ratio i of strain s and E_s^j ($s = 1, \dots, 11; j = 1, 2, 3$) the measured value of external flux j of strain s . Each flux ratio R_s^i can be expressed in terms of the flux vectors, appropriately weighted by two vectors $\mathbf{a}_i \in \mathbb{R}^n$ and $\mathbf{b}_i \in \mathbb{R}^n$:

$$\mathbf{a}'_i \mathbf{x} / \mathbf{b}'_i \mathbf{x} = R_s^i, \quad i = 1, \dots, 6. \quad (\text{C.1})$$

These equations can be reformulated as standard linear equations:

$$\mathbf{Ax} = \mathbf{0}, \quad (\text{C.2})$$

where the i th row of \mathbf{A} is $\mathbf{a}'_i - \mathbf{b}'_i R_s^i$, $i = 1, \dots, 6$. Translating the measured ratios and external fluxes to a feasible flux distribution for strain s is posed as the following optimization problem

$$\begin{aligned} \min \quad & \|\mathbf{Ax}_s\|^2 \\ \text{s.t.} \quad & \mathbf{Sx}_s = \mathbf{0}, \\ & E_s^j - \beta \cdot std_s^j \leq x_s^j \leq E_s^j + \beta \cdot std_s^j, \quad j = 1, 2, 3, \end{aligned} \quad (\text{C.3})$$

where β is a coefficient determining the feasible range (here, we set $\beta = 1$), and std_s^j is the standard deviation of the measurements of external flux j of strain s . The problem (C.3) is a standard quadratic programming problem yielding the flux distribution, which is the closest one to the measured pathway branch ratio and is consistent with the stoichiometry constraints and external flux measurements. The problem can be solved efficiently and global optimality can be guaranteed. The optimal solution \mathbf{x}_s of (C.3) is the feasible flux distribution for strain s and can be used to test our invFBA algorithm.

The problem (C.3) is a standard quadratic programming problem yielding the flux distribution, which is the closest one to the measured pathway branch ratio and is consistent with the stoichiometry constraints and external flux measurements. The problem can be solved efficiently and global optimality can be guaranteed. The optimal solution \mathbf{x}_s of (C.3) is the feasible flux distribution for strain s and can be used to test our invFBA algorithm.

References

- Beg, Q. K., Zampieri, M., Klitgord, N., Collins, S. B., Altafini, C., Serres, M. H., and Segre, D. (2012). Detection of transcriptional triggers in the dynamics of microbial growth: application to the respiratorily versatile bacterium *shewanella oneidensis*. *Nucleic acids research*, 40(15):7132–7149.
- Bertsekas, D. P. and Bertsekas, D. P. (1999). *Nonlinear programming*. Athena Scientific.
- Bertsimas, D., Gupta, V., and Paschalidis, I. C. (2013). Data-driven estimation in equilibrium using inverse optimization. *Mathematical Programming*, 153.2 (2015): 595-633.
- Bittl, J. A., Chaitman, B. R., Feit, F., Kimball, W., and Topol, E. J. (2001). Bivalirudin versus heparin during coronary angioplasty for unstable or postinfarction angina: final report reanalysis of the bivalirudin angioplasty study. *American heart journal*, 142(6):952–959.
- Blazier, A. S. and Papin, J. A. (2012). Integration of expression data in genome-scale metabolic network reconstructions. *Frontiers in Physiology*, 3(299).
- Burgard, A. P. and Maranas, C. D. (2003). Optimization-based framework for inferring and testing hypothesized metabolic objective functions. *Biotechnology and bioengineering*, 82(6):670–677.
- Cho, I. and Blaser, M. J. (2012). The human microbiome: at the interface of health and disease. *Nature Reviews Genetics*, 13(4):260–270.
- Chong, E. K. and Zak, S. H. (2013). *An introduction to optimization*. John Wiley & Sons.
- Collins, S. B., Reznik, E., and Segrè, D. (2012). Temporal expression-based analysis of metabolism. *PLoS computational biology*, 8(11):e1002781.
- The Medicines Company (2011). Angiomax (bivalirudin) Prescribing Information. <http://www.themedicinescompany.com/page/angiomax-us/1>.
- Covert, M. W., Schilling, C. H., and Palsson, B. (2001). Regulation of gene expression in flux balance models of metabolism. *Journal of theoretical biology*, 213(1):73–88.

- Dai, W., Brisimi, T. S., Adams, W. G., Mela, T., Saligrama, V., and Paschalidis, I. C. (2015). Prediction of hospitalization due to heart diseases by supervised learning methods. *International journal of medical informatics*, 84(3):189–197.
- Daims, H., Taylor, M. W., and Wagner, M. (2006). Wastewater treatment: a model system for microbial ecology. *Trends in biotechnology*, 24(11):483–489.
- Edrich, T., Frendl, G., and Paschalidis, Y. (2011a). 711: Tachyphylaxis in post-cardiac surgical patients receiving bivalirudin—a retrospective dynamic study using a pkpd model. *Critical Care Medicine*, 39(12):198.
- Edrich, T., Frendl, G., Rawn, J. D., and Paschalidis, Y. (2011b). Modeling the effects of bivalirudin in cardiac surgical patients. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 120–123, doi: 10.1109/IEMBS.2011.6089911.
- Edwards, J. S., Ibarra, R. U., and Palsson, B. O. (2001). In silico predictions of escherichia coli metabolic capabilities are consistent with experimental data. *Nature biotechnology*, 19(2):125–130.
- Evgeniou, T., Pontil, M., and Poggio, T. (2000). Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13(1):1–50.
- Feist, A. M., Henry, C. S., Reed, J. L., Krummenacker, M., Joyce, A. R., Karp, P. D., Broadbelt, L. J., Hatzimanikatis, V., and Palsson, B. O. (2007). A genome-scale metabolic reconstruction for escherichia coli k-12 mg1655 that accounts for 1260 orfs and thermodynamic information. *Molecular systems biology*, 3(1):121.
- Friedland, B. (2012). *Control system design: an introduction to state-space methods*. Courier Corporation.
- Gianchandani, E. P., Oberhardt, M. A., Burgard, A. P., Maranas, C. D., and Papin, J. A. (2008). Predicting biological system objectives de novo from internal state measurements. *BMC bioinformatics*, 9(1):43.
- Hannan, E. J. and Deistler, M. (1988). *The statistical theory of linear systems*. New York: Wiley.
- Harcombe, W. R., Delaney, N. F., Leiby, N., Klitgord, N., and Marx, C. J. (2013). The ability of flux balance analysis to predict evolution of central metabolism scales with the initial distance to the optimum. *PLoS computational biology*, 9(6):e1003091.
- Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J., and Tibshirani, R. (2009). *The elements of statistical learning*, volume 2. Springer.

- Heinrich, R. and Rapoport, T. A. (1974). A linear steady-state treatment of enzymatic chains. *European Journal of Biochemistry*, 42(1):97–105.
- Heinrich, R. and Schuster, S. (1998). The modelling of metabolic systems. structure, control and optimality. *Biosystems*, 47(1):61–77.
- Held, C. and Roy, R. J. (1995). Multiple drug hemodynamic control by means of a supervisory-fuzzy rule-based adaptive control system: validation on a model. *IEEE Transactions on Biomedical Engineering*, 42(4):371–385.
- Hu, Z., Mellor, J., Wu, J., and DeLisi, C. (2004). Visant: an online visualization and analysis tool for biological interaction data. *BMC bioinformatics*, 5(1):1.
- Ioannou, P. A. and Kosmatopoulos, E. B. (2006). *Adaptive Control*. Wiley Online Library.
- Ioannou, P. A. and Sun, J. (2012). *Robust adaptive control*. Courier Corporation.
- Kauffman, K. J., Prakash, P., and Edwards, J. S. (2003). Advances in flux balance analysis. *Current opinion in biotechnology*, 14(5):491–496.
- Khandelwal, R. A., Olivier, B. G., Röling, W. F., Teusink, B., and Bruggeman, F. J. (2013). Community flux balance analysis for microbial consortia at balanced growth. *PLoS One*, 8(5):e64567.
- Kiser, T. H., Mann, A. M., Trujillo, T. C., and Hassell, K. L. (2011). Evaluation of empiric versus nomogram-based direct thrombin inhibitor management in patients with suspected heparin-induced thrombocytopenia. *American journal of hematology*, 86(3):267–272.
- Klitgord, N. and Segre, D. (2010). Environments that induce synthetic microbial ecosystems. *PLoS Computational Biology*, 6(11):e1001002.
- Knorr, A. L., Jain, R., and Srivastava, R. (2007). Bayesian-based selection of metabolic objective functions. *Bioinformatics*, 23(3):351–357.
- Lincoff, A. M., Bittl, J. A., Harrington, R. A., Feit, F., Kleiman, N. S., Jackman, J. D., Sarembock, I. J., Cohen, D. J., Spriggs, D., Ebrahimi, R., et al. (2003). Bivalirudin and provisional glycoprotein iib/iiia blockade compared with heparin and planned glycoprotein iib/iiia blockade during percutaneous coronary intervention. *JAMA: the journal of the American Medical Association*, 289(7):853–863.
- Ljung, L. (1987). *System identification: theory for the user*. Englewood Cliffs, NJ: Prentice-Hall.

- Ludwick, D. A. and Doucette, J. (2009). Adopting electronic medical records in primary care: lessons learned from health information systems implementation experience in seven countries. *International journal of medical informatics*, 78(1):22–31.
- Machado, D. and Herrgård, M. (2014). Systematic evaluation of methods for integration of transcriptomic data into constraint-based models of metabolism. *PLoS Comput Biol*, 10(4):e1003580.
- Maddamsetti, R., Lenski, R. E., and Barrick, J. E. (2015). Adaptation, clonal interference, and frequency-dependent interactions in a long-term evolution experiment with escherichia coli. *Genetics*, 200(2):619–631.
- Miller, T. E., Burns, J. H., Munguia, P., Walters, E. L., Kneitel, J. M., Richards, P. M., Mouquet, N., and Buckley, H. L. (2005). A critical review of twenty years use of the resource-ratio theory. *The American Naturalist*, 165(4):439–448.
- Newsholme, E. and Crabtree, B. (1979). Theoretical principles in the approaches to control of metabolic pathways and their application to glycolysis in muscle. *Journal of molecular and cellular cardiology*, 11(9):839–856.
- Orth, J. D., Conrad, T. M., Na, J., Lerman, J. A., Nam, H., Feist, A. M., and Palsson, B. Ø. (2011). A comprehensive genome-scale reconstruction of escherichia coli metabolism2011. *Molecular systems biology*, 7(1):535.doi: 10.1038/msb.2011.65.
- Orth, J. D., Thiele, I., and Palsson, B. Ø. (2010). What is flux balance analysis? *Nature biotechnology*, 28(3):245–248.
- Papoutsakis, E. T. (1984). Equations and calculations for fermentations of butyric acid bacteria. *Biotechnology and bioengineering*, 26(2):174–187.
- Popov, V. M. and Georgescu, R. (1973). *Hyperstability of control systems*. Springer-Verlag New York, Inc.
- Sauer, U., Cameron, D. C., and Bailey, J. E. (1998). Metabolic capacity of bacillus subtilis for the production of purine nucleosides, riboflavin, and folic acid. *Biotechnology and bioengineering*, 59(2):227–238.
- Schölkopf, B., Herbrich, R., and Smola, A. J. (2001). A generalized representer theorem. In *Computational learning theory*, pages 416–426. Springer.
- Schuetz, R., Zamboni, N., Zampieri, M., Heinemann, M., and Sauer, U. (2012). Multidimensional optimality of microbial metabolism. *Science*, 336(6081):601–604.
- Schuster, S., Kreft, J.-U., Schroeter, A., and Pfeiffer, T. (2008). Use of game-theoretical methods in biochemistry and biophysics. *Journal of biological physics*, 34(1-2):1–17.

- Segre, D. (2008). Optimal metabolic regulation using a constraint-based model. *Genome Informatics*, 20:299.
- Shong, J., Diaz, M. R. J., and Collins, C. H. (2012). Towards synthetic microbial consortia for bioprocessing. *Current Opinion in Biotechnology*, 23(5):798–802.
- Stolyar, S., Van Dien, S., Hillesland, K. L., Pinel, N., Lie, T. J., Leigh, J. A., and Stahl, D. A. (2007). Metabolic modeling of a mutualistic microbial community. *Molecular systems biology*, 3(1):92.
- Stone, G. W., McLaurin, B. T., Cox, D. A., Bertrand, M. E., Lincoff, A. M., Moses, J. W., White, H. D., Pocock, S. J., Ware, J. H., Feit, F., et al. (2006). Bivalirudin for patients with acute coronary syndromes. *New England Journal of Medicine*, 355(21):2203–2216.
- Taniguchi, Y., Choi, P. J., Li, G.-W., Chen, H., Babu, M., Hearn, J., Emili, A., and Xie, X. S. (2010). Quantifying e. coli proteome and transcriptome with single-molecule sensitivity in single cells. *Science*, 329(5991):533–538.
- Vemuri, G., Altman, E., Sangurdekar, D., Khodursky, A., and Eiteman, M. (2006). Overflow metabolism in escherichia coli during steady-state growth: transcriptional regulation and effect of the redox ratio. *Applied and environmental microbiology*, 72(5):3653–3661.
- Wangersky, P. J. (1978). Lotka-volterra population models. *Annual Review of Ecology and Systematics*, 9:189–218.
- Welch, G. and Bishop, G. (2006). An introduction to the Kalman filter. Department of computer science, university of North Carolina. http://www.cs.unc.edu/~tracker/media/pdf/SIGGRAPH2001_CoursePack_08.pdf
- Wiser, M. J., Ribbeck, N., and Lenski, R. E. (2013). Long-term dynamics of adaptation in asexual populations. *Science*, 342(6164):1364–1367.
- Zamboni, N., Fendt, S.-M., Rühl, M., and Sauer, U. (2009). ¹³C-based metabolic flux analysis. *Nature protocols*, 4(6):878–892.
- Zamboni, N., Fischer, E., and Sauer, U. (2005). Fiatflux—a software for metabolic flux analysis from ¹³C-glucose experiments. *BMC bioinformatics*, 6(1):209.
- Zhao, Q., Stettner, A., Reznik, E., Segrè, D., and Paschalidis, I. C. (2015). Learning cellular objectives from fluxes by inverse optimization. *In 2015 54th IEEE Conference on Decision and Control (CDC)*, Pp. 1271-1276. IEEE
- Zomorodi, A. R. and Maranas, C. D. (2012). Optcom: a multi-level optimization framework for the metabolic modeling and analysis of microbial communities. *PLoS Computational Biology*, 8(2):e1002363.

- Zomorodi, A. R. and Segrè, D. (2016). Synthetic ecology of microbes: Mathematical models and applications. *Journal of molecular biology*, 428(5 Pt B):837-861. doi: 10.1016/j.jmb.2015.10.019
- Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320.

CURRICULUM VITAE

Qi Zhao

Qi Zhao is a Ph.D. candidate in the Division of Systems Engineering at Boston University. After obtaining a B.S. in Electrical Engineering and Automation (2010) from the University of Science and Technology, Beijing, he received an M.S. (2012) in Electrical Engineering from Boston University. His research interests lie in the fields of systems and control, optimization, operations research, computational biology, and bioinformatics.

Publications:

- Q. Zhao, T. Edrich, and I. C. Paschalidis, A Predictive Model for the Anticoagulant Bivalirudin Administered to Cardiac Surgical Patients, the 52nd IEEE Conference on Decision and Control, Florence, Italy, December 2013, Pp. 121-126.
- Q. Zhao, T. Edrich, and I. C. Paschalidis, Predicting and Evaluating the Effect of Bivalirudin in Cardiac Surgical Patients, IEEE Transactions on Biomedical Engineering, 2014, Feb; 61(2):435-43.
- Q. Zhao, T. Edrich, and I. C. Paschalidis, Adaptive Control of Bivalirudin in the Cardiac Intensive Care Unit, 19th World Congress of the International Federation of Automatic Control, Cape Town, South Africa, August 2014, Pp. 8427-8432.
- Q. Zhao, T. Edrich, and I. C. Paschalidis, Adaptive Control of Bivalirudin in the Cardiac Intensive Care Unit, IEEE Transactions on Biomedical Engineering, 2015 Feb; 62(2):638-47.
- Q. Zhao, A. Stettner, D. Segre and I. C. Paschalidis, Revisiting Flux Balance Analysis: Learning cellular Objectives from Fluxes, the 53rd IEEE Conference on Decision and Control, Osaka, Japan, December 2015, Pp. 1271-1276.
- Q. Zhao, A. Stettner, D. Segre and I. C. Paschalidis, What Is A Cell Optimal For? Characterizing the Space of Objective Functions from Experimentally Observed Fluxes. (Submitted to Genome Biology.)
- Q. Zhao, D. Segre and I. C. Paschalidis, Optimal Allocation of Metabolic Functions among Organisms in a Microbial Ecosystem. (Submitted to 2016 IEEE Conference on Decision and Control.)