A DYNAMIC STATE METABOLIC JOURNEY: FROM MASS SPECTROMETRY TO NETWORK ANALYSIS VIA ESTIMATION OF KINETIC PARAMETERS

by

Arockia Ranjitha Dhanasekaran

A dissertation submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Computer Science

Approved:

Bart C. Weimer
Co-Advisor

Donald H. Cooley
Co-Advisor

Nicholas S. Flann
Committee Member

Gregory J. Podgorski
Committee Member

John R. Stevens
Committee Member

Mark R. McLellan
Vice President for Research and Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2011
ABSTRACT

A Dynamic State Metabolic Journey: From Mass Spectrometry to Network Analysis via Estimation of Kinetic Parameters

by

Arockia Ranjitha Dhanasekaran, Doctor of Philosophy
Utah State University, 2011

Co-Advisor: Dr. Bart C. Weimer
Department: Computer Science

In the post-genomic era, there is a dire need for tools to perform metabolic analyses that include the structural, functional, and regulatory analysis of metabolic networks. This need arose because of the lag between the two phases of metabolic engineering, namely, synthesis and analysis. Molecular biological tools for synthesis like recombinant DNA technology and genetic engineering have advanced a lot farther than tools for systemic analysis. Consequently, bioinformatics is poised to play an important role in bridging the gap between the two phases of metabolic engineering, thereby accelerating the improvement of organisms by using predictive simulations that can be done in minutes rather than mutant constructions that require weeks to months.

In addition, metabolism occurs at a rapid speed compared to other cellular activities and has two states, dynamic state and steady state. Dynamic state analysis sheds more light on the mechanisms and regulation of metabolism than its steady state
counterpart. Currently, several in silico tools exist for steady-state analysis of metabolism, but tools for dynamic analysis are lacking. This research focused on simulating the dynamic state of metabolism for predictive analysis of the metabolic changes in an organism during metabolic engineering.

The goals of this research were accomplished by developing two software tools. Metabolome Searcher, a web-based high throughput tool, facilitates putative compound identification and metabolic pathway mapping of mass spectrometry data by applying genome-restriction. The second tool, DynaFlux, uses these compound identifications along with time course data obtained from a mass spectrometer in conjunction with the pathways of interest to simulate and estimate dynamic-state metabolic flux, as well as to analyze the network properties. The features available in DynaFlux are: 1) derivation of the metabolic reconstructions from Pathway Tools software for the simulation; 2) automated building of the mathematical model of the metabolic network; 3) estimation of the kinetic parameters, $K_R$, $v$, $V_{maxf}$, $V_{maxr}$, and $K_{dy}$, using hybrid-mutation random-restart hill climbing search; 4) perturbation studies of enzyme activities; 5) enumeration of feasible routes between two metabolites; 6) determination of the minimal enzyme set and dispensable enzyme set; 7) imputation of missing metabolite data; and 8) visualization of the network.

(213 pages)
The term “metabolism” refers to the chemical processes occurring in a living organism to convert the food consumed into the energy needed to maintain a living state. Metabolism consists of two states, namely, a dynamic state and a steady state. In the dynamic state, the rate of chemical conversion of a substance is proportional to the amount of substance available, whereas in the steady state this rate is constant and independent of the amount of substance present (Chapter 4 Figure 1). Like all other fields of engineering, metabolic engineering involves the analysis and synthesis of metabolism. Molecular biological tools for synthesis have advanced far ahead of the tools for analysis.

Bioinformatics is an inter-disciplinary field that applies computer science and information technology to biology. Bioinformatics has stepped in to bridge the gap between analysis and synthesis in metabolic engineering. There are several bioinformatics tools available to perform analysis of metabolism at steady state because in this state, the complex kinetics can be ignored due to the fact that the rate of a reaction is constant. However, studying metabolism at dynamic state gives more information about the processes and kinetics involved. Once the kinetics of a process is known, adjusting the parameters that will affect the rate of metabolite conversion becomes possible. Thus, a reaction can be modified to be faster or slower or be blocked based on the needs of the given situation in the process of metabolic engineering.

The focus of this doctoral research was on analysis of dynamic state metabolism. The goals were achieved by developing two software tools: 1) Metabolome Searcher and
2) DynaFlux. These two tools aid in filling the gap between the analysis and synthesis phases of metabolic engineering.
DEDICATION

I dedicate this work to my older brother, Antony Dhanasekaran, and my younger brother, Francis Dhanasekaran, both of whom I lost to adrenoleukodystrophy. They and their disease served as a motivation for me to take up this doctoral research in bioinformatics.

Arockia Ranjitha Dhanasekaran
ACKNOWLEDGMENTS

I place on record my sincere and heartfelt thanks to the distinguished personalities referred to below, who contributed to this dissertation with their valuable guidance and support that enabled me to make this study purposeful, meaningful, and successful.

I shall ever remain grateful to Dr. Bart C. Weimer, my advisor, who was kind enough to induct me into this work. The choice of topic for my doctoral research was born out of his expertise in the field of food microbiology and bioinformatics. He has been a great mentor of mine from the initial to the final stages of this project. His understanding and encouraging guidance provided a good foundation for the present dissertation. But for him, this dissertation would not have been possible.

I wish to extend my sincere appreciation to Dr. Donald H. Cooley for his kind support all the way through my Ph.D program. I am grateful to Dr. Nicholas S. Flann for his valuable suggestions, Dr. Gregory J. Podgorski for his guidance and moral support during this work, and Dr. John R. Stevens for his help in the area of statistics and statistical applications.

I owe my gratefulness to Dr. Daniel Watson (Interim Department Head, Department of Computer Science), who extended his help when it was most required. It is a pleasure to thank Dr. Timothy Doyle (Research Associate Professor, Department of Physics) and Dr. Alvan Hengge (Department Head, Department of Chemistry and Biochemistry) for their valuable direction.
I derive pleasure in thanking Dr. Balasubramanian Ganesan, Sweta Rajan, Prerak Desai, and other lab-mates in Dr. Weimer’s Lab, all of whom were ready to lend a helping hand whenever required.

I would like to thank Myra Cook, Thesis/Dissertation Coordinator, Department of Computer Science, for her assistance with refining the dissertation composition and converting it into a finished document.

I am much indebted to my dad, V. G. Dhanasekaran, and my mom, Inbamayam Dhanasekaran, for their unfailing love and unflinching support all the way through the years I have been away from home at Utah State. Their support and inspiration have contributed significantly to all that I have achieved.

Syed M. Sayem has been my philosopher and guide right from the time I set my foot in Utah State. He has been a great source of comfort and encouragement. He made his support available to me in a number of ways. I owe my deepest gratitude to him.

My sincere thanks are due to Mark Hargreaves and Carolyn Farrell who through their love and affection did all that they could to make me feel at home during the years I have spent at Utah State.

I also thank the Almighty for having blessed me with a sound state of body and mind so that I could complete this doctoral research.

Arockia Ranjitha Dhanasekaran
### CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>ABSTRACT ....................................................................................................................... iii</td>
<td></td>
</tr>
<tr>
<td>PUBLIC ABSTRACT</td>
<td>PUBLIC ABSTRACT .........................................................................................................v</td>
<td></td>
</tr>
<tr>
<td>DEDICATION</td>
<td>DEDICATION .................................................................................................................. vii</td>
<td></td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>ACKNOWLEDGMENTS ............................................................................................... viii</td>
<td></td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>LIST OF TABLES ............................................................................................................ xii</td>
<td></td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>LIST OF FIGURES ......................................................................................................... xiii</td>
<td></td>
</tr>
<tr>
<td>CHAPTER</td>
<td>CHAPTER</td>
<td></td>
</tr>
<tr>
<td>I.</td>
<td>INTRODUCTION .................................................................................................1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Problem Statement .................................................................................................1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Brief Overview .........................................................................................................1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Significance of Research ........................................................................................3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Research Objectives ...............................................................................................4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Outline Structure .................................................................................................5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>References ............................................................................................................7</td>
<td></td>
</tr>
<tr>
<td>II.</td>
<td>LITERATURE REVIEW .........................................................................................10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Metabolism ...............................................................................................................10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Metabolic Flux .........................................................................................................10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Need for Computational Tools/Bioinformatics ......................................................12</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Metabolic Engineering ............................................................................................14</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Analysis Step of Metabolic Engineering ..............................................................17</td>
<td></td>
</tr>
<tr>
<td></td>
<td>References ............................................................................................................34</td>
<td></td>
</tr>
<tr>
<td>III.</td>
<td>METABOLOME SEARCHER: A HIGH THROUGHPUT TOOL FOR METABOLITE IDENTIFICATION AND METABOLIC PATHWAY</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>Abstract ....................................................................................................................55</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Background ...............................................................................................................56</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Implementation ........................................................................................................62</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MS Data Validation ...............................................................................................67</td>
<td></td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tools for metabolic network reconstruction</td>
<td>47</td>
</tr>
<tr>
<td>2</td>
<td>Computational tools for metabolic flux analysis</td>
<td>48</td>
</tr>
<tr>
<td>3</td>
<td>Computational tools that aid in the kinetic analysis of metabolic networks</td>
<td>49</td>
</tr>
<tr>
<td>4</td>
<td>Metabolite distribution by molecular mass across metabolic encyclopedias</td>
<td>81</td>
</tr>
<tr>
<td>5</td>
<td>Organism-specific and general metabolic reference databases available for the Metabolome Searcher</td>
<td>82</td>
</tr>
<tr>
<td>6</td>
<td>Basic biochemical transformations and differential equations</td>
<td>122</td>
</tr>
<tr>
<td>7</td>
<td>Different representations of various reaction types</td>
<td>123</td>
</tr>
<tr>
<td>8</td>
<td>Different factors depicting the performance of the variants of hill-climbing</td>
<td>124</td>
</tr>
<tr>
<td>9</td>
<td>Kinetic parameters $V_{\text{max}f}$ and $V_{\text{max}r}$ of all reactions in the three pathways of Salmonella</td>
<td>125</td>
</tr>
<tr>
<td>10</td>
<td>Kinetic parameters $V_{\text{max}f}$ and $V_{\text{max}r}$ of all reactions in the three pathways of Salmonella + Caco-2</td>
<td>126</td>
</tr>
<tr>
<td>11</td>
<td>Kinetic parameter $K_{\text{dy}}$ of all reactions in the three pathways of Salmonella and Salmonella + Caco-2</td>
<td>127</td>
</tr>
<tr>
<td>12</td>
<td>Imputation of missing value of isocitrate at the third time point</td>
<td>166</td>
</tr>
<tr>
<td>13</td>
<td>Imputation of missing data</td>
<td>167</td>
</tr>
<tr>
<td>14</td>
<td>Feasible paths between succinate and glutamate</td>
<td>168</td>
</tr>
<tr>
<td>15</td>
<td>Feasible paths between citrate and malate</td>
<td>169</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Diagrammatic representation of iterative model building</td>
<td>50</td>
</tr>
<tr>
<td>2</td>
<td>Metabolic engineering cycle</td>
<td>51</td>
</tr>
<tr>
<td>3</td>
<td>Four intracellular variables and their interactions</td>
<td>52</td>
</tr>
<tr>
<td>4</td>
<td>Genotype to Phenotype I: Metabolic network reconstruction</td>
<td>53</td>
</tr>
<tr>
<td>5</td>
<td>Genotype to Phenotype II: Metabolic Flux Analysis</td>
<td>54</td>
</tr>
<tr>
<td>6</td>
<td>Metabolome Searcher user interface screenshot</td>
<td>83</td>
</tr>
<tr>
<td>7</td>
<td>(A) Metabolome Searcher workflow and (B) the flowchart of the search operations to depict how the compounds are matched</td>
<td>84</td>
</tr>
<tr>
<td>8</td>
<td>Screenshot of Metabolome Searcher’s output</td>
<td>85</td>
</tr>
<tr>
<td>9</td>
<td>Comparison of results from chemical vs metabolic databases with the monoisotopic mass of isocitrate (192.027±0.001 Da) as the query</td>
<td>86</td>
</tr>
<tr>
<td>10</td>
<td>Pathway assembly of Metabolome Searcher output with heat maps of LC/MS data from the compounds file and the pathways file (tryptophan biosynthesis and asparagine biosynthesis I)</td>
<td>87</td>
</tr>
<tr>
<td>11</td>
<td>Two states of metabolism: Dynamic State vs Steady State</td>
<td>128</td>
</tr>
<tr>
<td>12</td>
<td>Different representations of a biochemical reaction</td>
<td>129</td>
</tr>
<tr>
<td>13</td>
<td>Three metabolic pathways used in the study</td>
<td>130</td>
</tr>
<tr>
<td>14</td>
<td>Plot of the different hill-climbing variants</td>
<td>131</td>
</tr>
<tr>
<td>15</td>
<td>Experimental data vs Model data for the various metabolites present in the three metabolic pathways</td>
<td>132</td>
</tr>
<tr>
<td>16</td>
<td>The three metabolic pathways overlaid with the K_R values of Salmonella</td>
<td>133</td>
</tr>
<tr>
<td>17</td>
<td>The three metabolic pathways overlaid with the K_R values of Salmonella &amp; Caco-2</td>
<td>134</td>
</tr>
<tr>
<td>Page</td>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------</td>
<td>--------------------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>18</td>
<td>DynaFlux Architecture</td>
<td>170</td>
</tr>
<tr>
<td>19</td>
<td>User Interface of DynaFlux</td>
<td>171</td>
</tr>
<tr>
<td>20</td>
<td>Three pathways: Aspartate biosynthesis, TCA cycle and glyoxylate cycle</td>
<td>172</td>
</tr>
<tr>
<td>21</td>
<td>Metabolic flux estimates overlaid on the network</td>
<td>173</td>
</tr>
<tr>
<td>22</td>
<td>Feasible routes between succinate and glutamate</td>
<td>174</td>
</tr>
<tr>
<td>23</td>
<td>Feasible routes between citrate and malate</td>
<td>175</td>
</tr>
<tr>
<td>24</td>
<td>Metabolic network overlaid with fluxes from normal, knocked-out and amplified aspartate transaminase</td>
<td>176</td>
</tr>
<tr>
<td>25</td>
<td>Geometric representation of a bi-bi reaction</td>
<td>192</td>
</tr>
<tr>
<td>26</td>
<td>Patterns containing one less line than basic figure</td>
<td>193</td>
</tr>
</tbody>
</table>
CHAPTER I
INTRODUCTION

Problem Statement
Metabolism is one of the most complex cellular processes. Analysis of metabolism facilitates rational modification of an organism to enhance its metabolic capabilities. Currently, most mathematical methods and computational tools analyze metabolism at steady state, i.e., when the metabolite pools are constant. The problem with steady state analysis is that it provides only a static snapshot of the metabolic behavior, limits the scope of understanding, is accurate only in very specific conditions, and lacks predictive capabilities in unexplored genetic or environmental conditions. In order to gain a thorough understanding of metabolism, the analysis needs to be done in the dynamic state (when the metabolite pools are changing) of metabolism. Dynamic state analysis sheds light on the actual metabolic behavior of an organism. This research focuses on dynamic state analysis of metabolism by first identifying the metabolites present in the organism during a time course under a given condition and estimating the network flux using the identified metabolites.

Brief Overview
Metabolic engineering is initiated with an extensive bioinformatics analysis and theoretical synthesis in order to obtain an optimized metabolic pathway to produce the desired effect [1-4]. Analysis of metabolism encompasses metabolic pathway analysis, flux analysis, and control analysis [5], thereby mapping the genotype to the metabolic phenotype of the organism. Thus metabolic analysis is comprised of:
1. Determining the structure of the metabolic network using metabolic reconstruction from the sequenced genome [6, 7]

2. Identifying the metabolites and pathways of interest under a given condition using genome restrictions

3. Quantifying the metabolic fluxes through the metabolic network [8]

4. Identifying enzymic control methods and genetic regulatory mechanisms [8]

The technologies used for synthesis, such as molecular biology, recombinant DNA technology, and genetic engineering have advanced greatly in recent years. Unfortunately, tools for metabolic analysis have advanced at a slower pace. Consequently, there is a gap between the two phases of metabolic engineering, namely analysis and synthesis. Due to this lag, progress in improving organisms using metabolic engineering has been slow.

Bioinformatics is a boon that fills this gap. Computational tools to help analyze the metabolism of an organism in order to better understand the metabolic capabilities present based on the genome are just beginning to be widely used and made available with a sufficient database of networks so that they are useful to biologists. While there are several computational tools that can be employed for analysis of metabolic networks, most of them can be used for steady state analysis only [9-11], which limits their usefulness. In order to reduce the complexity, steady state analysis includes the stoichiometry of the metabolic network and few constraints to model metabolism [12] but ignores kinetic parameters and regulatory information that are mostly unknown [5, 13, 14]. To get an accurate picture of the metabolic network flux, there is an increasing
demand to examine metabolism during the dynamic state [8, 15-17]. Dynamic state analysis is achieved by including kinetic and regulatory information, which further allows forecasting the effects due to system perturbation. This research focuses on analysis of metabolism at dynamic state, using bioinformatics to create two tools: 1) to identify the metabolites present in an organism under a certain condition using genome restrictions, and 2) to estimate metabolic network flux and to perform a few network analyses.

**Significance of the Research**

Analysis of metabolism at steady state provides only a static snapshot of the metabolic activities and does not reflect on genetic or environmental changes. To overcome this limitation of steady state analysis, kinetic parameters and regulatory information has to be included in the metabolic modeling to shed light on the dynamic behavior of metabolism [18]. The software tools that have integrated kinetic parameters and simulated the dynamic behavior of an organism are GEPASI [16], MIST [19] and SCAMP [20]. The major drawback with these tools is that the user has to provide the kinetic mechanisms and constants as inputs but these inputs are mostly unknown or difficult to measure. Owing to the network complexity, shortage of computational resources, and lack of technical tools for model development, not much work was done in this area of metabolic analysis.

Dynamic metabolic modeling is more interesting than its static counterpart because these models are extrapolative in nature and can help predict the dynamic behavior of the organism in perturbation studies, as well as assist in estimating the metabolic behavior during unexamined conditions [21, 22]. Moreover, dynamic models
help theories be verified with actual behavior [16]. Dynamic state analysis provides information about the flux and inter-conversion of small molecules as opposed to steady state analysis that focuses on accumulation of biomass. Dynamic metabolic network modeling can explain properties of complex biological systems and also serve as a guide to experimentation [23]; however, the major challenge in this type of modeling is the estimation of a vast number of parameters. Recent high-throughput technologies have helped overcome the limitations of computational capability and accessibility to experimental data, thereby paving the way for dynamic state analysis of metabolism which in turn will play an important role in metabolic engineering.

The cost of experiments, needed skills, and laborious setup of experiments have restricted flux measurements to only a few laboratories [24]. High-throughput omics technologies like transcriptomics, proteomics, and metabolomics have made possible the systemic measurements of mRNA transcripts, proteins, and small metabolites, respectively. Bioinformatics has played an important role in profiling the fluxome of an organism using the omics data and has made flux analysis available to the scientific community.

**Research Objectives**

**Objective 1**
Identify metabolites present in the organism at a given condition and metabolic pathway mapping.
**Objective 2**
Estimate kinetic parameters that govern flux of a given pathway using metabolic reconstruction maps and quantitative metabolite data along with gene expression data.

**Objective 3**
Determine the most probable metabolic pathway route from a source metabolite to the desired product. Also, impute missing data in experimental measurements.

**Objective 4**
Validate the network using model perturbation including enzyme knock-out and amplification.

**Objective 5**
Develop a user-friendly interface for non-expert users that incorporates Objectives 2 through 5.

**Outline Structure**
Chapter 2 provides a comprehensive review of the literature covering the analysis of metabolic flux during steady state and dynamic state. In order to achieve the analysis of metabolic flux of an organism at dynamic state, the identification of metabolites that are present and active at the desired condition has to be completed. Chapter 3 presents a web application, Metabolome Searcher, which uses mass spectrometry data, searches publicly available metabolic databases, and returns a list of possible compound identifications along with their putative metabolic pathways.

Chapter 4 deals with estimating kinetic parameters and was accomplished by first transforming the metabolic reconstruction map into a graph theory (node-edge) representation resembling the metabolic pathway under study. This served as the
qualitative analysis of the data by acting as a representation check to be sure that the appropriate set of reactions was included. Next, a quantitative representation of the network was constructed using a mathematical definition by transforming the metabolic pathway into a set of coupled differential equations. Using the mathematical definition, the kinetic parameters were estimated and optimized using a novel artificial intelligence approach namely hybrid-mutation random-restart hill climbing.

Chapter 5 encompasses Objectives 2 through 5 and introduces the user-friendly dynamic state flux analysis tool, DynaFlux. Determining the shortest path and the most probable paths from source metabolite to desired product was accomplished using a graph theory approach by calculating the paths using the transformed graph theory representation. Also, a flux score is assigned to the most likely route(s) so that the user can rank or select the route of interest. The minimal enzyme set and the dispensable set was also enumerated.

Imputation of missing values of metabolites during unexplored or unexamined conditions was achieved using hybrid-mutation random-restart hill climbing algorithm in combination with statistical measures. Finally, the network model can be validated using perturbation methods such as knock-out and amplification of enzyme activity. Additionally, an automated visualization of the metabolic pathway under study is displayed with an overlay of the estimated flux.
References


24. Zamboni N, Fischer E, Sauer U: **FiatFlux—a software for metabolic flux analysis from 13C-glucose experiments.** *BMC Bioinformatics* 2005, **6**.
**CHAPTER II**

**LITERATURE REVIEW**

**Metabolism**

All living organisms need energy to grow, reproduce, and to meet their day-to-day nutrient requirements in order to maintain a living state [1]. *Metabolism* is the process by which organisms derive energy from the food they consume to perform these activities [2]. Metabolism comprises all the chemical reactions that are usually catalyzed by enzymes in an organism [3-5]. Metabolism is characterized by two opposing phases known as catabolism and anabolism. *Catabolism*, or degradation or destructive metabolism, is the phase during which an organism derives energy by breaking down large molecules. *Anabolism*, or biosynthesis or constructive metabolism, uses the energy released during catabolism, and builds complex molecules and structures for the survival of the organism [2, 4]. The complex series of consecutive, controlled, and enzyme-catalyzed chemical transformations constitutes a *metabolic pathway*. The coordinated interactions among various pathways form the *metabolic network* of the organism. Estimation of the metabolic network was largely impossible until the era of genome sequencing, which now provides a vast number of microbial genome sequences that serve as the metabolic blueprint.

**Metabolic Flux**

The metabolic intermediates formed due to chemical transformations between the initial reactant and the final product are known as metabolites [4]. The complete set of metabolites makes up the *metabolome* of an organism or a cell [6]. *Metabolomics* is the process for high throughput analysis to profile and to measure metabolite concentrations.
Commonly, mass spectrometry is used to identify and quantify metabolites [7]. Consequently, metabolomics can measure the entire chemical network of a cell simultaneously, but the rate at which the network runs is a key question in biology that is very difficult to answer. The rate at which a metabolite is processed through a metabolic pathway via the enzymatic and transport reactions is termed \textit{metabolic flux}. \textit{Fluxomics} is the study of the dynamic changes of molecules within a cell over time [8]. When the metabolic flux or metabolite pools in an organism remain constant, the metabolism is at pseudo or quasi \textit{steady state} [5], a state that only lasts briefly during normal life. Quantification of a system’s metabolic flux (i.e., the fluxome) determines the true metabolic behaviour of the cell [9, 10]. Measurements of flux also reveal the contributions of individual pathways (i.e., sub-networks) within the overall cellular systemic function [10]. However, a major challenge in biology is the inability to measure metabolic flux during metabolism. This is difficult due to the speed of the fluxome and the extensive interconnectedness of the metabolic network. Additionally, some enzymes have multiple activities with varying rates depending on the substrate concentration and type. This level of complexity is not readily “read” from the genomic blueprint, and so must be determined empirically. Consequently, until bioinformatics tools were developed to handle the large databases and complex interconnections in biological networks, fluxomics was slow to develop.

While scientists can measure specific reactions in a test tube to determine specific kinetic parameters, this approach is slow, very difficult, and expensive. With the invention of high throughput genome sequencing, determination of the metabolic
blueprint became less difficult to determine. Each biochemical step of metabolism is highly regulated by the cell and does not continue endlessly even if the substrate is available [11]. Flux describes the control that allows the breakdown or synthesis of only the required metabolites under different operating conditions. An organism maintains its metabolism at homeostasis even under different environmental conditions due to the control mechanisms involved in flux regulation. With the vast amount of genome sequences, it is now possible to compare the metabolic blueprint of each organism; however, still lacking is the ability to determine the flux regulation. Hence, biologists remain tied to traditional empirical experiments. High throughput methods are needed that enable biologists to predict the effect of changing conditions, enzyme content, and substrates that control flux.

**Need for Computational Tools/Bioinformatics**

In order to manipulate or improve the productivity of an organism, the metabolic capability of an organism needs to be analyzed first. Mathematical models and in silico simulations contribute significantly to the analysis of metabolism and its regulation [12, 13]. Mathematical models of metabolism help in understanding the details of the mechanisms involved using a mathematical representation [14, 15]. Simulation of metabolism helps in understanding, predicting responses of complex systems under unexamined conditions, and guiding the researcher to perform new experiments [16, 17]. Mathematical and computational tools help in modeling, designing, and predicting the properties of modified organisms [18]. These tools serve as a guide for experimental work [19, 20] by helping the researcher to devise new hypotheses [9, 16, 21]. In addition,
due to the simplification of the original process, these models serve as tools for hypotheses testing [22]. Mathematical models and in silico simulations also help in understanding the systemic functions and metabolic capabilities of an organism [10, 23, 24]. Along with the models, DNA microarrays, 2D gel electrophoresis, and mass spectrometry data help in the detailed analysis of cellular properties [22]. Technological advancement has triggered a recent surge in genome sequencing and molecular biological data, with over 800 genomes now available in the public domain. Unfortunately, techniques for automated and in silico metabolic analysis are still not on par with sequencing nor with molecular biology’s capability to create new genome sequences. Currently, there is a lack of tools to analyse biochemical data. Said situation demands the development of in silico models of cellular processes to understand the constituents, interactions, and systemic functions [10]. Systemic or “emergent” properties of the cell need to be analyzed in order to understand metabolism [25]. Bioinformatics helps to bridge this gap between these technologies through modeling and simulation.

Mathematical modeling and computer simulation of a biological process is iterative, with refinement in every step based on new information from laboratory experiments (Figure 1). Initially, the models depict the real scenario with much of the details avoided due to the complexity of nature [16]. Together modeling and experimentation can give a clearer picture of metabolism but may not provide a perfect representation [16].

Metabolic network modeling helps in predicting cellular behavior under changing environmental conditions with fixed genetic background, or fixed environmental
conditions with changing genetic background [22]. The influence of environmental conditions, development, and/or genetic background sheds light on metabolic functioning and regulation [6, 26].

Mathematical models for flux are useful for [12]:

1. Organizing disparate information into a coherent whole
2. Understanding the interactions between individual components
3. Discovering new strategies
4. Making corrections to conventional wisdom
5. Understanding qualitative features

**Metabolic Engineering**

Several industries including the chemical, food, pharmaceutical, and health care industries have been exploiting the metabolic capabilities of various organisms for many centuries to produce molecules that are valuable to society – such as antibiotics and anticancer compounds [27, 28]. Furthermore, biotechnology has helped to improve the capabilities of these organisms by increasing the productivity of the desired product [29]. In the past, classical crossovers of different strains have resulted in better strains with improved capabilities [28]. The classical approach involved several iterations of random mutagenesis followed by screening and selection to get an improved strain [19]. Metabolic engineering is a rational approach for strain improvement by employing targeted genetic changes. Recombinant DNA technologies are used to create metabolically engineered strains for strain improvement, which result in new metabolic networks that are not biochemically characterized [19, 30, 31].
Metabolic engineering has several definitions that have evolved over time. Bailey was the first to coin the definition for metabolic engineering in 1991, which states that “metabolic engineering is the improvement of cellular activities by manipulation of enzymatic, transport, and regulatory functions of the cell with the use of recombinant DNA technology” [32]. In 1993, Cameron and Tong defined metabolic engineering as “the purposeful modification of intermediary metabolism using recombinant DNA techniques” [18]. In recent years, it has become clear that as metabolic engineering evolved it emerged into a highly multidisciplinary field that draws support, tools, ideas, and methods from biochemistry, genetics, molecular biology, cell physiology, chemistry, chemical engineering, systems analysis, computer simulation, and increasingly bioinformatics [18, 19]. Metabolic engineering deals with identifying new pathways, thermodynamic feasibility, locating limiting branch points and enzymatic reactions in a network [33].

**Goals of metabolic engineering**

Metabolic engineering is applied for the following purposes:

1. To improve the production of chemicals already present in the host organism [18, 19, 34, 35]. For example, ethanol production in *Escherichia coli*.

2. To extend the substrate range for growth and product formation. Lactose, xylose, and cellulose utilization are a few examples [34, 35].

3. To add new catabolic activities for degradation of toxic chemicals [34]. For example, camphor and naphthalene in *Pseudomonas*.

4. To produce chemicals new to the host organism [18, 19, 34, 35]. Indigo production by *Escherichia coli* is an example.
5. For the production of heterologous peptides [19, 34].

6. For the improvement of cellular properties [18, 36].

7. For disease prevention or control [36].

**Steps in metabolic engineering**

Like all other fields of engineering, metabolic engineering involves analysis and synthesis, but metabolic engineering deals with metabolism that is a complex natural phenomenon that is difficult to comprehend and can change unexpectedly due to cellular interactions and DNA exchange. Metabolic engineering is an iterative process, as shown in Figure 2, that involves several cycles of analysis and synthesis to achieve the desired results [34, 35]. Steps in the cycle of metabolic engineering are [34, 37, 38]:

1. **Analysis:** This step involves a detailed characterization of the metabolic potential of the organism or cell by using analytical techniques to quantify flux and its control.

2. **Design:** This is an intermediate step that involves targeting of the next modification in the genome sequence.

3. **Synthesis:** The actual genetic modification to obtain the recombinant strain using molecular biological techniques is done in this step.

Based on the purpose for which the organism is being modified, one can start the cycle in the analysis or the synthesis step [34].

**Limitations**

There are a few limitations with respect to metabolic engineering. Even if metabolic engineering deals with directed genetic modifications targeted to bring about the desired results, in some cases there is the chance of getting unexpected or undesirable
results in addition to the desired change [9]. This may arise due to the regulatory mechanisms at the transcriptional, translational, or metabolic levels. These mechanisms are still not completely characterized. Interactions among biochemical reactions in a network must be understood before doing any kind of modification [36].

Another major limitation is that while there are tools for molecular biology available to bring about the genetic change, the analytical techniques to identify the optimal genetic modifications are still naïve [19]. Due to current workhorse technologies, there are lot of biological data [39], but tools to analyze and interpret this data are limited [37].

**Analysis Step of Metabolic Engineering**

This review focuses on the analysis step of metabolic engineering, with special emphasis on flux analysis and quantification. The analysis step in metabolic engineering deals with the initial metabolic characterization of the original strain in order to determine the necessary modifications to optimize the production of certain compounds. In the later rounds of the cycle, the modified strain is analyzed to check its performance against the original strain, assessing whether the desired results are achieved [34]. Therefore the analysis step has two different purposes [36]:

1. One purpose is to elucidate the metabolic network properties of the original strain to improve the yield in an existing process.

2. The second purpose is to evaluate the genetically engineered cells in the second and consecutive iterations so as to compare its performance with the original strain.
There are four important variables that contribute to the metabolic characterization of the organism or cell during analysis.

1. mRNA transcripts (genomics)
2. Proteins (proteomics)
3. Small metabolites (metabolomics)
4. Metabolic flux (fluxomics)

There is a high level of interaction between these variables, symbolically shown in Figure 3. These four intracellular variables constitute a system-wide description of the physiological state of an organism and are also useful for modeling cellular function [36]. Even though there are techniques to quantify these variables, there is a need for a framework to integrate these four variables and to provide a description for the interaction of genes [36, 40]. Metabolic engineering provides this framework for the integration of these four variables [36, 40]. Transcripts (mRNA), protein, metabolites, and flux give a comprehensive understanding of the systemic behaviour [6, 9]. The high throughput technologies used in genomics, proteomics, and metabolomics will help metabolic engineering in a significant manner to achieve its goals [34].

**Genome sequencing, post-genomic analysis and metabolic reconstruction**

To analyze the metabolic potentials of an organism, one needs a foundation: the metabolic network. Metabolic networks pave the way to understanding the mechanisms and adaptations of an organism to different conditions. Thus, for the analysis of metabolism or flux, a defined and established pathway is first required.
Due to high throughput technologies for DNA sequencing, used since the mid-1990s, the genomes of many organisms have been sequenced [41-43]. More than 800 completely sequenced genomes are available, out of which over 200 are microbial genomes [43, 44]. The rate at which new genomes are being sequenced far exceeds the experimental determination of metabolic pathway information present in each organism [45]. Consequently, there is a dire need to bridge the gap between sequencing genomes and assigning functions to the genes. Annotation of genes can be done experimentally or by prediction methods using homology [45]. Most of the genes encode gene products involved in metabolic activity [46, 47]. Once the genes have functions assigned, the metabolic pathways of the organism can be reconstructed [48]. Reconstruction of the metabolic network of an organism at the genomic level is vital for the analysis of the metabolic capabilities [49].

Once the genome of an organism is completely sequenced, bioinformatics tools like GLIMMER [50] aid in assigning the open reading frames (ORFs), while Pathway Tools [51] and KEGG [52] assist in reconstructing the metabolic map that complements the annotated genome. As shown in Figure 4, biochemical experiments can verify and help curate the draft of reconstructed metabolic networks from genome annotations. Biochemical data combined with genome annotation tend to give more complete predictions of metabolic networks. Physiological studies help fill in the gaps in metabolic reconstruction [10].

Currently, there are several automated tools that help reconstruct metabolic pathways from genomic data. These tools predict the pathways based on homology and
do not just predict textbook pathways. Pathway Tools and KEGG are the two most widely used software tools. There are several tools used for metabolic pathway reconstruction, a few of which are listed in Table 1.

**Pathway Tools – SRI International**

Pathway Tools consists of several specific organism databases and a general database (MetaCyc). There are two versions of the software – a desktop version and a web-based version. This software takes in the genomic sequence of an organism and reconstructs the metabolic pathway present in the organism by predicting pathways through comparison of the annotated genome with MetaCyc. The MetaCyc database consists of more than 1200 pathways from over 1600 different organisms [53]. The Pathway Tools software creates a pathway genome database (PGDB) for every organism, which contains information about the genes, proteins, metabolic network and genetic network of the organism. This software also aids in comparative genomics [51].

**KEGG - Kyoto Encyclopedia of Genes and Genomes**

KEGG is a general web-based pathway knowledge base that draws information from three internal databases – GENES, PATHWAYS and LIGAND. KEGG links genomic information with higher order information like metabolism, transport, and signal transduction [52].

Using one of the tools mentioned in Table 1, the metabolic network of an organism can be reconstructed using a bottom-up approach [3], i.e., from annotated genome to metabolic network. Once the foundation is laid, the properties of the network, such as the structure, function and regulation, need to be analyzed [54]. The underlying
mechanisms and properties of the network can be elucidated at the molecular level using the following techniques.

1. Detailed physiological studies [19]
2. Metabolic pathway analysis (MPA) [24]
3. Metabolic flux analysis (MFA) [19]
4. Metabolic control analysis (MCA) [19]
5. Thermodynamic analysis [19]
6. Kinetic modeling [19]/Mechanistic modeling/Dynamic simulation analysis

Dynamic analysis of metabolism has proven to be difficult [23], and so the problem has been simplified to be studied at steady state because the mechanistic details and kinetic parameters are mostly unknown [7, 26, 54]. There is no direct method to measure such parameters or know the details [26]. Therefore, the problem was simplified to include only the stoichiometry and only a few constraints to model metabolism [55].

**Stoichiometric modeling**

This section gives a brief introduction to the stoichiometric modeling that is used in many of the techniques for analyzing network properties. To facilitate the analysis of the network properties, a mathematical representation of the network is required [25]. In order to simplify the problem, an invariant property of the network is used for analysis. The stoichiometry of a network represents the structural or invariant property of the network because it does not change with variations in the environment [1] and also aids in the mathematical representation of the network. Stoichiometric modeling makes use of the available network topology or structure, and ignores unknown variables such as the
intracellular metabolite concentrations and kinetic parameters that are difficult to measure [55]. Also, stoichiometric modeling assumes quasi or pseudo steady state. Analogous to Kirchhoff’s first law for electrical circuits, the principle of mass balance states that the production and utilization of metabolites must balance [39, 46, 56], i.e., the formation fluxes of metabolites must be balanced by degradation fluxes. Applying the principle of mass balance, every reconstructed network can be mathematically represented as

\[
\frac{dx}{dt} = S.v
\]  

(Eq. 1)

where S is an m x n stoichiometric matrix with m corresponding to the number of metabolites and n is the number of reaction fluxes, x is the metabolite concentration vector and v is the flux vector [1]. In most cases, the number of metabolites is less than the number of reactions thus causing the system to be underdetermined [5]. Constraining the mass balance of the network to be at steady state, i.e., when there is no change in the metabolite concentration [1, 5, 7, 57], Equation 1 can be rewritten as

\[
0 = S.v
\]  

(Eq. 2)

This is the basic idea behind stoichiometric modeling which forms the basis for many of the analytical methods (Figure 5).

**Metabolic pathway analysis**

**Development in metabolic pathways**

Initially due to extensive biochemical research, individual reactions were well characterized. Later, the reactions that had shared metabolites were grouped into traditional pathways such as glycolysis, tricarboxylic acid cycle (TCA), and so on. The advancement in sequencing technology and reconstruction of metabolic networks has led to the creation of complete metabolic network models of an organism [58]. For the
The analysis of systemic properties of the network, it is advantageous to consider a reaction in its entirety rather than as a single independent entity [58]. Recently, there has been demand for a “network-based” definition of pathways. This kind of a definition would ease the mathematical analysis of the metabolic capabilities and bring out the true characteristics of the network. This demand arose because traditional pathways focussed only on subsets of the network and did not consider system-wide metabolic interactions. Traditional pathways cannot be ruled out completely because it carries its own advantages. However, network-based pathway analysis facilitates the evaluation of network properties [58], such as metabolic versatility, robustness [59] and optimal growth rates. Also in traditional pathways, control was concentrated in a single reaction step in the pathway, which was known as the rate-limiting step but network-based pathways recognize that control is distributed among many reaction steps in the pathway [36]. Thus, modification of metabolism can be brought about by considering the system as a whole and not individual reactions [35].

Metabolic pathways describe the topology and architecture of a network, which need to be understood so as to know the capabilities present in the organism. Metabolic pathway analysis is the identification and analysis of significant routes in metabolic networks [60-63]. Metabolic pathway analysis also provides information about the systemic or network-based invariant properties of the cell [55] and helps assess functional and structural properties of metabolic networks [60].

The goals of metabolic pathway analysis are:

1. To test the viability of reconstructed metabolic network [64].
2. To calculate inherent network properties [60, 61].

3. To analyze the functions of the pathways for metabolic engineering [62].

For pathway or structural analysis, the network topology is needed, which is known in most cases [54]. Along with the topology, the stoichiometry and the thermodynamic feasibility (i.e., the reversibility or irreversibility) of the reactions involved are also required to do the analysis [62, 63, 65]. Also, the analysis is done at steady state [64, 66].

**Extreme pathways - Mathematical approach to MPA**

The stoichiometric matrix is obtained from the metabolic network as discussed in the stoichiometric modeling section. Applying mass balance at steady state, the null space of the network is mathematically represented as

\[ S.v = 0 \]

By using linear algebra, this system of equations can be solved and the null space denotes the solution space for the above equation [55]. Since some reactions are reversible or bidirectional, the flux can be either positive or negative or zero. Due to the inequality involved, this system of equations is solved using convex analysis that results in the convex set, also known as the convex solution, flux cone or steady state flux space. In addition, the internal fluxes are considered to be non-negative by splitting the reversible reactions into forward and reverse reactions. This constrains the problem and narrows the solution down to the positive quadrant of the convex solution, as shown in Figure 5. The extreme rays or the generating vectors correspond to the unique set of extreme pathways that are present in the network. Any point in the cone can be represented as a non-negative linear combination of these extreme pathways. This set of extreme pathways can
be compared to a coordinate system used to represent a point in space [1]. The extreme pathway approach is a network-based pathway analysis technique [55]. This approach helps to compare different networks [60]. Thus, extreme pathway analysis decomposes the actual network into subnetworks and assists in performing the analysis [67].

**Development of metabolic pathway analysis**

The different approaches for metabolic pathway analysis that were developed over preceding years are listed below:

- Petri nets, a graph theory approach [68].
- Seressotis and Bailey used an artificial intelligence approach to construct a genetically independent pathway from a substrate to a product. Computational complexity limits this approach [61].
- Mavrovouniotis based the analysis on stoichiometric constraints combined with biochemical constraints [61]. However, in this method there was no mathematical proof to support the concepts.
- Stoichiometric modeling stepped in to fill the gap. The network is represented in a matrix form. The set of equations is then solved using linear algebra, and the null space of the matrix represents the solution [61]. The null space describes the capabilities of the genotype, i.e., all possible flux distributions in the network. However, the solution is non-unique, and there is no linear independence.
- To overcome the limitation of solving using linear algebra, convex analysis was employed to find the solution. The solution space is given by the basis vector and graphically represented by the intersection between the null space and the positive
orthant. The solution is restricted to the positive orthant because reversible reactions are split into forward and reverse reactions, and so the flux is always positive [61]. This is called the convex polyhedral cone. There are certain limitations to this approach as well. Dynamic system behaviour cannot be analyzed using this approach. Furthermore, no regulatory control mechanism is used in this approach.

**Metabolic flux analysis**

Metabolic flux analysis is the determination and study of metabolic flux, and is the heart of metabolic engineering [35, 36]. The analysis of flux sheds light on the metabolic pathways active at a given condition [46]. Metabolic flux analysis deals with the quantification of flux, understanding the interaction between pathways, and estimation of metabolic flux distribution under a given circumstance in the cell at a particular point in time [55, 69]. The aim of metabolic flux analysis is to understand and characterize flux and its control [5, 70]. Metabolic flux analysis encompasses all the quantitative approaches for experimentally measuring flux distributions [7]. However, the determination of flux is a very challenging task. Apart from being used for flux quantification, metabolic flux analysis is also used for [19]:

1. Identification of new routes or different pathways
2. Identification of rigid branch points
3. Quantification of maximal theoretical yield of a product
4. Quantification of unknown extracellular fluxes
Milestones in the development of metabolic flux analysis

1. Extracellular metabolite approach

In this approach, the pathway of interest was isolated from the rest of the metabolism, and the rate of change of an extracellular metabolite was used to estimate the pathway flux [35] especially the intracellular fluxes were obtained [36]. This approach was however limited to highly branched pathways not containing any cycles, which is not the case in many pathways. Also, external metabolite consumption or accumulation may not be linear over time [40].

2. Material balances and stoichiometry of intracellular reactions

To overcome the limitations of the extracellular metabolite approach, the stoichiometry of the intracellular reactions are included in the analysis. In addition, the fundamental law of mass conservation is applied to the metabolites [46]. In this approach, a set of measured fluxes (mainly extracellular fluxes) is used in combination with knowledge about the cell’s metabolism, which is represented by means of the general equation (Equation 2), to determine those fluxes that have not been measured [55]. Direct assessment of flux is difficult, so flux can be assessed using linear programming (LP) [46].

3. Elementary flux modes

The elementary flux mode approach is very similar to the extreme pathway approach discussed in the metabolic pathway analysis section. A flux-mode is the flux distribution of a network or subnetwork at steady state, and the word elementary refers to a non-decomposable subnetwork [14, 62]. Elementary flux
modes are minimal functional subnetworks that can be mathematically represented as the extreme rays of the convex pointed polyhedral cone [71].

There are two mathematical approaches to determine the elementary flux modes of a network. The first approach is similar to the Guass-Jordan method (Schuster’s algorithm) [63]. The other approach is the null space algorithm that uses stoichiometric modeling as discussed in the MPA section and solves the system of equations using convex analysis [66]. The elementary flux mode approach helps in the identification of enzyme subsets by null space analysis of the stoichiometric matrix. The solution of the convex analysis forms a convex polyhedral cone in the positive orthant of the flux space. The extreme rays that make up the convex cone constitute the elementary flux modes of the network.

The major applications of EFMs are as follows [71]:

- Identifies pathways, network flexibility and pathways with optimal yield
- Computes the minimal cut sets
- Detects thermodynamically infeasible cycles
- Suggests regulatory information
- Infers viability of mutants

The elementary flux mode approach predicts key aspects of a network, such as the functionality, robustness and gene regulation [54]. Elementary flux mode analysis can be applied to metabolic pathway analysis as well; thus, it is the link between metabolic pathway analysis and metabolic flux analysis [54]. In addition, the elementary flux mode is a superset of extreme pathways [5]. Elementary flux
modes and extreme pathways are two approaches for network-based pathway analysis [55].

Nevertheless, there are some limitations to this approach, too. There are many classical cofactors involved numerous pathways. To keep track of these cofactors in the stoichiometric matrix is an unmanageable task. This poses one of the major limitations to this approach. Further, this approach has less predictive power because the model lacks regulatory information [22].

4. *Isotopic labelling approach*

Due to the shortcomings of the stoichiometric approach, more constraints were required to be added to the model to depict real systems. So, information from isotopic labelling or carbon labelling experiments (CLE) was included in the stoichiometric model, and was called $^{13}$C MFA [69]. The measured isotopic label enrichment of isotopomers from the $^{13}$C experiments contributed an additional set of constraints to the model. Flux could be determined more precisely than the previous approaches by including the isotopic labelling information as an additional constraint [36]. The major shortcoming of this model is that it still only provides a static snapshot of metabolism [11, 19, 22].

*Non-mechanistic approach*

To overcome the problem of determining the kinetic parameters, nonmechanistic models have been built that account for the behaviour of the system with a limited number of parameters [26]. The power law representation such as the S-system is an example of nonmechanistic models used for analysis [72, 73]. Other approaches like
cybernetics[74, 75], neural networks [76, 77] or fuzzy logic [78, 79] also overcome the drawback of missing regulatory information. However, these approaches need a large amount of experimental data and provide less insight into the underlying mechanisms. The mixed integer linear programming (MILP) approach includes regulatory loops represented as discrete variables in the modeling [22].

**Flux balance analysis**

Flux balance analysis deals with the prediction of system behaviour or phenotype on the basis of some optimality hypothesis [55]. In other words, flux balance analysis is specifically concerned with deriving a feasible set of steady-state fluxes that optimizes a stated cellular objective, e.g., maximizing biomass production within a metabolic network subject to a set of constraints of conservation of mass [80]. Metabolic transients are rapid as compared to cell growth, so the analysis assumes quasi steady state [23]. The stoichiometry of all chemical reactions involved and the demands placed on the metabolic network [23] are required for flux balance analysis. Flux balance analysis applies stoichiometric modeling, metabolic demands, and a few parameters specific to the strain [46] to construct a model that is capable of predicting the phenotype or the flux distribution that is expressed under certain conditions [55]. Constraint-based [39] flux balance analysis involves solution of a set of linear algebraic equations [11] in a stoichiometric model. The equation $S.v=0$ is underdetermined, so external flux is also included to solve this equation [46]. In addition, the stoichiometry of the network constrains the solution obtained [23].
Flux balance analysis helps in quantifying metabolic fluxes [7], however there are certain limitations to this approach. The dynamics of metabolic behaviour cannot be assessed using flux balance analysis. Flux balance analysis narrows the phenotype instead of looking at the exact behaviour of metabolic networks, because the analysis lacks regulatory or kinetic information [11, 26, 39, 46]. Flux balance analysis provides a solution space representing the optimized metabolic capability rather than a single solution depicting the exact behaviour [25].

**Limitation of current tools**

There are several computational tools that have been built for flux analysis. Table 2 summarizes a few of the computational tools used for metabolic flux analysis. These tools have certain limitations including:

1. Most of the methods discussed for flux analysis can only be used for small networks. For large networks, the network must be broken into smaller components and then analysed with these methods.

2. Many of the tools are designed to do steady state analysis that reveal one particular phenotype, whereas dynamic models can reveal different states and the transition among those states [81].

3. The tools are either commercial or add-ons to commercial platforms that make accessing and utilizing these tools difficult.

4. The tools do not include kinetic parameters or control mechanism.
**Kinetic/mechanistic/dynamic analysis of metabolism**

Thus far, the review has discussed metabolic analysis at steady state which gives only a snapshot of the metabolic activities and does not reflect on genetic or environmental changes. Flux measurements or quantification at different operating conditions and in different mutants aid in understanding the underlying metabolism better than a snapshot that results from a single flux analysis at steady state at a given condition.

To overcome this drawback, kinetic properties and regulatory information must be added to the metabolic modeling to shed light on the dynamic behaviour of metabolism [82]. However, kinetic parameters that need to be estimated in order to make the dynamic models complete [10] are not easily measurable or maybe unknown [83]. This is also a difficult approach also because not all operating conditions can be experimentally analyzed. From its inception, this area of metabolic flux analysis has received very little attention. Kinetic (dynamic) analysis is the focus of this study.

Dynamic metabolic modeling is more interesting than its static counterpart because of the valuable information obtained that helps in understanding metabolism [40]. This kind of modeling is also considered to be predictive modeling because such models can predict the dynamic behaviour of the organism under unexamined conditions [5]. Further, dynamic models help in perturbation studies and assist in estimating the metabolic behaviour during environmental and genetic changes [11]. Moreover, dynamic models help theories be verified with actual behaviour [20]. Quantitative models mimic reality much better than qualitative models [16]. Kinetic models are able to predict scenarios that were not used while building the model [16]. Therefore, this kind of
analysis helps model dynamic scenarios and assist in understanding the quantitative behaviour [21]. A few of the software tools that have been employed for kinetic analysis are listed in Table 3.

One more requirement to obtain a successful tool to analyze metabolism is that genomic, proteomic, and metabolomic data should be collected at the same time points and under the same conditions, which requires a lot of effort and coordination among different laboratories [36].

**Modeling language**

Finally, there is a need for all models to agree upon a common language or information standard [81]. For this reason, eXtensible Markup Language (XML)-based modeling languages are used so that complex biological models built using one system can be exchanged with another tool [84, 85]. Systems Biology Markup Language (SBML) [84] is one such XML-based modeling language that is used to represent biochemical reaction systems including cell signaling pathways, metabolic pathways and gene regulation. Metabolic Flux Analysis Markup Language (MFAML) [85] is another XML-based modeling language which represents metabolic flux models.

In conclusion, metabolic engineering directly genetically modifies organisms for specific benefit to society. Traditionally, this is a slow process due to the lack of broad-scale genetic data to provide a rational and directed gene modification strategy. With the increasing availability of genome sequences and new genetic tools, greater progress is being made. The addition of in silico analysis provides new tools and new strategies that can further increase the pass of genetic modification of microbes. Bioinformatics will
pave the way for metabolic engineering to have a smooth transition between the analysis and synthesis of new metabolic routes in organisms, which in turn will lead to new strategies and constructs that produce organisms to fulfill our ever-increasing need for new products. Additionally, the analysis of the metabolic potentials of an organism during the dynamic state of metabolism plays an important role in metabolic engineering. The main focus of this research is to analyze and model the dynamic state of metabolism for ultimate production of small molecules to vaccines using microbial genetic engineering.

References


74. Varner J, Ramkrishna D: Metabolic engineering from a cybernetic perspective.


75. Varner J, Ramkrishna D: Metabolic engineering from a cybernetic Perspective.


<table>
<thead>
<tr>
<th>Tool</th>
<th>Last Updated</th>
<th>Features</th>
<th>Reference</th>
</tr>
</thead>
</table>
| Rahnuma              | 2009         | • Hypergraph approach  
                    • Depth first traversal to find the shortest path between source and product metabolite | [86]      |
| MetaRoute            | 2008         | • Cross-species comparison and dynamic extraction of local networks  
                    • Uses atom mapping rules and path weighting search to find relevant routes in a genome-wide metabolic network | [87]      |
| Pathway Analyst      | 2006         | • Pathway prediction using machine learning algorithm  
                    • Predicts the catalysts of reactions using the protein sequences  
                    • Organizes the predicted catalysts and reactions into previously defined metabolic pathways | [45, 88]  |
| Pathway Hunter Tool  | 2005         | • Uses breadth first search to find the shortest path that connects two metabolites using structural similarity | [89]      |
| Ma et al             | 2003         | • Uses a breadth first approach to find a shortest metabolic route from source to product  
                    • Network is represented as a directed graph | [90]      |
| BioMiner             | 2002         | • Represents, analyzes and visualizes metabolic pathways and networks based on sequence similarity  
                    • Can compare metabolic networks of different organisms | [91]      |
| What is there (WIT)  | 2000         | • Genome analysis system  
                    • Provides a system of tools to study ORFs based on similarities  
                    • ORF clustering | [92]      |
| Arita                | 2000         | • Uses a graph algorithm to reconstruct metabolic pathways at the atomic level of compounds  
                    • Uses a variant of k-shortest path algorithm | [3]       |
<table>
<thead>
<tr>
<th>Tool</th>
<th>Last Updated</th>
<th>Features</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CycSim</td>
<td>2009</td>
<td>Web-based tool for performing metabolic analysis using constraint based models (CBM) with a stoichiometric matrix of the whole cell and a set of reaction constraints</td>
<td>[93]</td>
</tr>
<tr>
<td>MetaFluxNet</td>
<td>2008</td>
<td>Integrated tool to create models</td>
<td>[24, 94]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Perform metabolic analysis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Compare models from different genomes</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dynamic visualization of flux results</td>
<td></td>
</tr>
<tr>
<td>Metatool</td>
<td>2006</td>
<td>Structural or topological analysis of metabolic pathways</td>
<td>[64, 95]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>New algorithm for detecting enzyme subsets</td>
<td></td>
</tr>
<tr>
<td>SNA</td>
<td>2006</td>
<td>Stoichiometric network analysis at steady state</td>
<td>[96]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Computes the generating and elementary flux vectors of the conversion cone using linear programming</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The elementary vectors gives a description of the possible steady state behaviours</td>
<td></td>
</tr>
<tr>
<td>YANA</td>
<td>2005</td>
<td>Software tool with a user-friendly interface</td>
<td>[97]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Uses an evolutionary algorithm to calculate, edit and compare elementary flux modes</td>
<td></td>
</tr>
<tr>
<td>FiatFlux</td>
<td>2005</td>
<td>Analytical determination of metabolic flux ratios from MS data</td>
<td>[98]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Estimate absolute in vivo fluxes using a $^{13}$C constrained flux balancing approach</td>
<td></td>
</tr>
<tr>
<td>FluxAnalyzer</td>
<td>2003</td>
<td>Analysis of flux ratio followed by estimation of net flux</td>
<td>[99]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>User-friendly interface</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Facilitates pathway and flux analysis at steady state</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides interactive flux map</td>
<td></td>
</tr>
<tr>
<td>DBSolve</td>
<td>2001</td>
<td>Workbench that helps the user to generate mathematical models of biological processes</td>
<td>[15]</td>
</tr>
</tbody>
</table>
Table 3 - Computational tools that aid in the kinetic analysis of metabolic networks

<table>
<thead>
<tr>
<th>Tool</th>
<th>Last Updated</th>
<th>Features</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gepasi</td>
<td>2001</td>
<td>• Define the topology and kinetics of the pathway</td>
<td>[20]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Assign values to the kinetic constants and initial concentrations needed for simulation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Select the output options</td>
<td></td>
</tr>
<tr>
<td>MIST</td>
<td>1994</td>
<td>• Dynamic simulations, stoichiometric modeling and metabolic control analysis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• User-friendly graphical interface</td>
<td>[100]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Define metabolic pathways of any complexity</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Output can be stored and retrieved</td>
<td></td>
</tr>
<tr>
<td>SCAMP</td>
<td>1993</td>
<td>• Define metabolic pathways consisting of any structure and kinetics</td>
<td>[101]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Build models directly from differential equations</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Provides access to flux, concentration and rate of change information</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Allows the detection of conserved cycles</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Provides access to all coefficients and elasticities of metabolic control analysis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Allows user-configurable output options</td>
<td></td>
</tr>
</tbody>
</table>
Figure 1 - Diagrammatic representation of iterative model building
Figure 2 - Metabolic engineering cycle
Figure 3 - Four intracellular variables and their interactions
Figure 4 - Genotype to Phenotype I: Metabolic network reconstruction
Figure 5 - Genotype to Phenotype II: Metabolic flux analysis
CHAPTER III
METABOLOME SEARCHER: A HIGH THROUGHPUT TOOL FOR METABOLITE IDENTIFICATION AND METABOLIC PATHWAY MAPPING¹

Abstract

Background
Mass spectrometric analysis of microbial metabolism provides a long list of possible compounds. Restricting the identification of the possible compounds to those produced by the specific organism would benefit the identification process. Currently, identification of mass spectrometry (MS) data is commonly done using empirically derived compound databases. Unfortunately, most databases contain relatively few compounds, leaving long lists of unidentified molecules. Incorporating genome-encoded metabolism enables MS output identification that may not be included in databases. Using an organism’s genome as a database restricts metabolite identification to only those compounds that the organism can produce.

Results
To address the challenge of metabolomic analysis from MS data, a web-based application to directly search genome-constructed metabolic databases was developed. The user query returns a genome-restricted list of possible compound identifications

¹ Co-authored by Arockia Ranjitha Dhanasekaran, Jon L. Pearson, Balasubramanian Ganesan, and Bart C. Weimer.
along with the putative metabolic pathways based on the name, formula, SMILES structure, and the compound mass as defined by the user. Multiple queries can be done simultaneously by submitting a text file created by the user or obtained from the MS analysis software. The user can also provide parameters specific to the experiment’s MS analysis conditions, such as mass deviation, adducts, and detection mode during the query so as to provide additional levels of evidence to produce the tentative identification. The query results are provided as an HTML page and downloadable text file of possible compounds that are restricted to a specific genome. Hyperlinks provided in the HTML file connect the user to the curated metabolic databases housed in ProCyc, a Pathway Tools platform, as well as the KEGG Pathway database for visualization and metabolic pathway analysis.

**Conclusions**

Metabolome Searcher, a web-based tool, facilitates putative compound identification of MS output based on genome-restricted metabolic capability. This enables researchers to rapidly extend the possible identifications of large data sets for metabolites that are not in compound databases. Putative compound names with their associated metabolic pathways from metabolomics data sets are returned to the user for additional biological interpretation and visualization. This novel approach enables compound identification by restricting the possible masses to those encoded in the genome.

**Background**

Bacterial metabolism impacts almost every aspect of our life. Microbial
metabolism was exploited by early human civilization to create fermented foods and beverages [1, 2]. The oldest known metabolically derived products from microbes include bread, cured meats, cheese, and beer [2-4]. Currently, metabolic engineering for the production of pharmaceuticals and bioactive compounds is giving way to the discovery of novel metabolic pathways for production of alternative fuels [5-7]. Burgeoning needs to produce novel antibiotics for disease treatment and health supplements, such as amino-sugars and vitamins, also represent the metabolic end products that are genome encoded of an organism [8-11].

The virulence of bacterial pathogens is closely linked to their metabolism during infection, which leads to metabolomic disease biomarkers that, in turn, pushes the boundaries of robust methods to quickly identify high throughput metabolomic data [12, 13]. Cumulatively, the unusual metabolic networks of organisms in ecological niches are renewing interests in metabolites that highlight the lack of high throughput analysis tools for rapid compound identification when the compound is not included in a database. Unfortunately, rapid, simultaneous identification of multiple metabolites is also lacking. However, if one considers an organism’s genome to be a database of possible metabolic pathways and metabolite production, customization of MS output analysis based on the specific organism becomes possible. Approaching the genome as a metabolite database is being done using metabolic reconstruction methods in KEGG and Pathway Tools.

The metabolism of an organism changes during growth, survival, and persistence via complex gene expression changes. In many cases, metabolism begins with the transport of chemically diverse molecules for integration into biologically functional
blocks. An organism’s metabolic capability can be envisaged as a highly interconnected network of enzymatic reactions that provide energy, intermediates for macromolecular biosynthesis, cellular signaling, regulation of stress, and control of oxidation/reduction to ensure growth or survival [14]. Highly tuned regulatory mechanisms to modulate the metabolic network via gene expression and enzyme attenuation are needed to quickly adapt to local environmental changes. Evolution of genetic control and gene acquisition are critical to ensure the organism’s survival in the near- and long-term [15]. Adaptation and genetic evolution results in new metabolic nodes in the interconnected network that modifies the intermediate and end product metabolism [14, 16]. Of recent interest, metabolic engineering is largely dependent on understanding the metabolic network to regulate production of specific low molecular weight end products that often accumulate.

Low molecular weight metabolites, usually <1,000 Da (small molecules; Table 4), including sugars, lipids, fatty acids, amino acids, nucleotides, vitamins, and co-factors are typically the targets of metabolomics, which have bioactivity and lead to biomarker profiles (www.metacyc.org; [17]). An organism’s metabolic demands are met by catabolism of complex macromolecules to the constituent small molecules (e.g., polysaccharides to sugars) or digestion of the molecules themselves (e.g., vitamins and amino acids) to end products. The products of catabolism are reassembled through anabolic pathways into macromolecules of the organism to derive energy, oxidation/reduction regulation, pH control, and to maintain membrane potential that fuels transport functions. During growth, catabolic and anabolic processes are regulated both genetically and biochemically to maintain a balance between growth and survival [18,
All of these activities are encoded in the genome, which provides an inherent genetic database of the possible metabolic compounds that an organism can produce during changing growth conditions.

Metabolomics aspires to identify all the metabolites produced by an organism [20, 21]. However, large data sets, limited identification databases, and limited MS parameters to differentiate small molecules are stumbling blocks for metabolomic analysis, which in turn limits the subsequent bioinformatic analysis and construction of biologically informative models [16, 21]. Currently, NMR is of limited use for high throughput small molecule identification due to the lack of sensitivity and limited throughput, but is useful to elucidate the structures of unique metabolites [22, 23]. However, NMR is very useful to track the metabolic fate of a small molecule with isotope labels, which provides information for a handful of metabolites once the entire compound list is narrowed to a specific set of metabolic intermediates [18]. Other post-separation detection techniques like photometric, electrochemical, and fluorescent detection are actively used to identify specific metabolites at a substantially reduced analytical scale, but the need to identify the set of compounds produced is overwhelmingly changing the goals of metabolite analysis [24-26]. Conversely in addition to metabolic tracking, MS analysis estimates the masses of hundreds to thousands of small molecules within minutes and provides information on their relative levels in the sample [27-29], making it very useful for high throughput metabolome analysis. Unfortunately, it lacks specific information as to the identity of the small molecules, which highlights the need to have curated databases for compound
One approach to overcome the need to identify important molecules uses principal component analysis (PCA) to find changes in response to a specific treatment. From MS data acquisition, this produces a reduced list of small molecules that are tagged as biomarkers [30, 31]. Often the diagnostic peak is an unknown compound that is difficult to identify. Subsequently, more complex chemical analysis is used to determine the elemental composition of these biomarkers, which requires additional time, expertise, and often multiple instrumentation capabilities [32, 33]. Biomarker identities are subsequently validated by standard compound injection to produce a compound library [34]. While this statistics-driven analytical approach favors method development for MS, it ignores the underlying biochemistry and the importance of relatively minor changes of small molecules. This can lead to misinterpretation of the biological impact of new small molecule production. This is especially prevalent for key metabolite classes like hormones, vitamins, and enzyme co-factors in which small changes regulate large scale proteomic and metabolic fluctuations [35]. One way to overcome this limitation is to use tools that include all possible putative compounds generated directly from matched compound identities prior to statistical analysis. Subsequently, a significant list of putative compounds can be used for metabolic mapping to facilitate biological identity by linking compound identities to metabolic pathways and routes. Feist et al. [36] review the reconstruction approach with specific attention to metabolite identification.

Unfortunately, metabolite identification from hundreds to thousands of masses by searching a large compound database is a slow process that is ill-defined relative to the
specific search criteria of providing confident compounds’ assignments. GC-MS analysis often identifies compounds by comparing the MS spectra with large, well-established compound libraries (www.nist.gov). Such compound libraries for LC-MS analysis are available for only a small set of masses and are tightly linked to the LC conditions. Large compound databases such as Pubchem (http://pubchem.ncbi.nlm.nih.gov) and Chemspider (http://www.chemspider.com) allow searches of single masses and other query types, but they do not allow queries from large lists of masses or connect putative compounds to metabolic pathways. However, as the query list expands as it does in metabolome data sets, data analysis using single queries becomes unrealistic for a timely and accurate analysis.

Querying large compound databases that contain millions of non-biological molecules can impede a researcher’s ability to overlay a metabolic context onto metabolomic data. Biologists are producing data at rates that outstrip the ability of analysts to examine the data set to uncover the biological importance. To keep pace with metabolome analysis, high throughput bioinformatic tools that bring compound identity and pathway relevance together to the biologist are crucial. This can be accomplished with: a) automated searches of metabolic databases to retrieve putative compound identification, b) large scale queries that are performed seamlessly with MS output, c) provide users the flexibility of using multiple query types, and d) map query results to metabolic pathways to allow data analysis in a biological context.

The availability of over 1,000 annotated microbial genome sequences enables bioinformatic reconstruction (biocyc.org) of an organism’s metabolic capability via the
genome, thus providing a broad network of metabolism that can be used to predict small molecule production [27, 28]. Consequently, recent efforts have focused on uncovering the metabolic networks in many different biological systems [19, 37]. Genome reconstructions of the metabolic pathways coupled to analytical methods, such as liquid chromatography (LC), gas chromatography (GC) and capillary electrophoresis with nuclear magnetic resonance spectroscopy (NMR) and mass spectrometry (MS) provide new ways to leverage genomic sequence to provide putative compound identification quickly [27, 38].

This study employs a user-friendly web-based application called Metabolome Searcher to retrieve a list of small molecule identifications based on chemical formula, SMILES structure, and the monoisotopic mass was created using an organism’s genome as a putative compound database. While single queries can be directly entered, multiple queries with one or more query types can also be done using a text file containing the query list. One or more reference databases can be selected from the list against which the queries are performed. The output connects small molecules in a sample to metabolic databases via embedded links to specific metabolic pathways. The Metabolome Searcher’s output allows researchers using metabolome data from different technologies to group the compound identifications into metabolic information so as to uncover the relevant biological function using multiple chemical criteria.

**Implementation**

**Metabolic reference database creation**

A Metabolic Reference Database (MRDB) of an organism is a flat file (tab-delimited, plain text file) that initially contains only the compound name, molecular
formula, molecular weight, SMILES structure, and the respective pathways for all compounds extracted from the Pathway Tools Pathway/Genome Database (PGDB) of that organism. The script to create the MRDB communicates with Pathway Tools [17] via the PerlCyc module (v1.1; www.arabidopsis.org/biocyc/perlcyc). The same approach was used to create an additional non-redundant database using Metacyc [17] and KEGG [39] (Table 5). The reference monoisotopic masses of individual elements were obtained from a publicly available compilation (Scientific Instrument Services, Inc., Ringoes, NJ; www.sisweb.com). Using the monoisotopic masses of individual elements, the monoisotopic masses of all compounds in the MRDB in their charged and neutral states were calculated based on their formulae. The MRDB was then modified to include the calculated monoisotopic masses, which is queried for compound identification and pathway mapping via the Metabolome Searcher’s web interface.

**Query input**

The Metabolome Searcher allows the user to enter a single query by typing the name, formula, molecular/monoisotopic mass or SMILES structure, or multiple queries by uploading a query list within a file (Figure 6). This file contains masses and intensities of compounds as a tab-delimited text file. For mass searches, whether from a single entry or a file, the user selects the type (molecular weight or monoisotopic mass). Most MS systems contain software that enables data export to a text or an Excel file [40]. We used the QTof system (QTof Premier, Waters, MA) with MarkerLynx software for marker identification and analysis to test this approach. A MarkerLynx-derived text file was used without modification for the Metabolome Searcher query by submitting the file under the
“MarkerLynx file” input on the interface (Figure 6). Alternately, analysis of output from other MS systems can be done using the “text file” option (Figure 6). While using the text file option, query values of any type, whether masses or specific compound names or a mixture of query types, were listed in the first column of the query file. Any headers, empty lines, and non-query values in the first column were removed prior to submission of data as a text file for matching. For both the file options, other information like statistics, marker quality, peak areas, peak heights, and concentrations across experiments and replicates were still retained in the file.

**Compound identification for MS analysis**

For compound identification from monoisotopic masses, the user specifies the acceptable deviation from the theoretical masses (ppm or Da, under “Mass deviation”; Figure 6), the ionization mode (positive or negative, under “Electrospray mode”; Figure 6), the maximum number of charges (0-5; under the “Number of proton charge states”; Figure 6), and adducts (mass or formula; optional; under “Adduct or Deduct molecule” and “Maximum number of adducts/deducts”; Figure 6). The deviation value allows the software to obtain matches for queried masses within an acceptable range to narrow or expand the putative identification list. Acceptable mass deviation values may be experimentally determined or obtained from the literature based on a particular instrument and operating conditions [41].

Typically during MS analysis, the molecules are detected by prior ionization with or by removal of protons (positive and negative mode, respectively) [35]. The MS settings are optimized to mainly produce singly charged ions. However, a molecule may
still carry multiple charges depending on the MS settings [42]. The user can verify the charge state of compounds contained in the input list to recalibrate the MS settings by selecting different charge states during multiple search sessions.

Positively charged ionic species, such as sodium (Na⁺) and potassium (K⁺), or negative species, such as chloride (Cl⁻) and formate (HCOO⁻), are also used during ionization due to their abundance in a sample. The addition of ionic species or adducts during ionization shifts the observed monoisotopic mass from that of the intact molecule plus/minus a proton [42]. These adducts can be specified either as individual elements or as partial functional groups in the “Adduct or Deduct molecule” textbox (Figure 6). Similar to adducts, if the user wishes to specify fragments lost during ionization or fragmentation the “Deduct” option can be selected. The user can also provide more than one adduct or deduct in the textbox simultaneously and specify the number of maximum possible adducts or fragments (“Maximum number of adducts/deducts” option).

**Database selection**

MRDBs that contain metabolites from different PGDBs or the KEGG database along with calculated monoisotopic masses are used for the queries. MRDBs are included for user selection from the ones listed on the interface (Table 5) wherein the user can select single or multiple MRDBs for searching (Figure 6). If the user intends to query known metabolic pathways in an organism, the organism-specific MRDBs are provided for more specific and narrow options of possible compounds due to the known annotated pathways. However, if the intent is to discover new pathways unknown in a particular system, but identified in other organisms, or if an organism without a preconstructed
MRDB is being studied, the user can select a genotypically related organism’s MRDB or the MetaCyc MRDB for matching. A user-generated PGDB can also be incorporated as an MRDB using the scripts defined above prior to the user defined query. The MRDBs were created in a flat file format to reduce complexity in processing and data handling such that newer MRDBs for other organisms can be created in a consistent format and readily incorporated as per the user’s need. Pathway Tools was selected as the main metabolic database platform to create MRDBs and link back to PGDBs due to its interactive features and user-level flexibility for metabolic database development and curation of whole genome PGDBs [17], while queries of an MRDB for the KEGG database [43] are also supported.

**Database searching**

Once a text query has been submitted, the Metabolome Searcher determines whether a text input is the name of a compound, its chemical formula or its SMILES structure independent of any specifications. After the query is classified into the specific type, information of the corresponding type in the MRDB is used for matching (i.e., names-to-names, formulae-to-formulae, and masses-to-masses) (Figure 7). All matches obtained within the parameters specified for searches are provided in the output files for viewing and analysis.

**Output generation**

After entering a single query or uploading a query file and specifying the MRDBs along with other MS analysis parameters, the user submits the query. The queries are matched against the MRDBs, and the output files are created. Query parameters are
printed at the top of all the output files to ensure that the parameters submitted by the user were used for searching the database (Figure 8A).

Three different output files are provided as the result of the analysis, one HTML and two text files. The two text files are embedded as links at the top of the HTML page (Figure 8A) that the user can download. One text file (“compounds file”) lists only the matched compounds without any metabolic pathway information, while the other (“pathways file”) repeats each compound’s data by all the pathways that it belongs to as a metabolite.

All scripts were written in Perl (v5.8.6; www.perl.org). The scripts and the metabolic reference databases for Metabolome Searcher are hosted in an Apple XGrid computational cluster (Panther OS 10.3.9) at the Western Dairy Center at Utah State University as well as University of California, Davis. Web pages for data input and output were created using Perl CGI.

**MS Data Validation**

**Chemical standards preparation**

All compounds used were purchased from Sigma-Aldrich (St. Louis, MO). A chemically defined medium described previously by Ganesan et al. [18] was prepared as a complex mixture for testing Metabolome Searcher’s performance. The major components of this medium are 20 amino acids, sodium chloride, citrate, phosphate, 3-(N-morpholino)propane sulfonic acid (MOPS), vitamin solution (containing 15 different compounds), and glucose. Individual standard solutions of selected amino acids, glucose, citrate, and MOPS were also used for molecule identification.
**Mass spectrometry**

Separation and analysis of standard compound mixtures were done at the mass spectrometry facility in the CIB. The samples were separated by liquid chromatography (2795 LC system; Waters) prior to introduction by electrospray into the mass spectrometer (QTof Premier; Waters) as described by Mortishire-Smith et al. [44]. Briefly, the separation was done for 10 min using a linear gradient of water:acetonitrile from 0-95% using a Symmetry C18 column (Waters). After introduction into the MS by electrospray, the molecules were detected using both positive and negative electrospray conditions, with calibrated settings recommended by the manufacturer. The QTof instrument was operated in W mode throughout MS analysis. For both positive and negative electrospray analysis, the conditions were: desolvation temperature of 250°C, source temperature of 120°C, cone voltage of 40 V, and collision energy of 4 eV. Data acquisition was performed for a mass range of 50-1,000 Da. After acquisition, the data were centroided [44] using 1 ng/µl leucine-enkephalin infused at 10 µl/min as a reference, with an m/z of 556.2771 in positive mode and m/z of 554.25 in negative mode. In order to subtract background from the LC column and sample matrix, HPLC-grade water (Thermo Fisher Scientific Inc., Waltham, MA) was injected into the MS as a negative control. All samples were analyzed in technical duplicates.

Peak detection, intensity extraction, and normalization were performed using MarkerLynx software (Waters) to obtain monoisotopic masses and molecule retention times. In this study, only the monoisotopic masses of the markers were used for database searches. The Metabolome Searcher does not support any data analysis of the concentrations or relative measures of compound levels obtained from MarkerLynx.
Results and Discussion

Metabolomic assessment provides a list of compounds that facilitates the estimation of metabolic flux through both single pathways and networks [45, 46]. Metabolome analysis enables determination of abiotic conditions and genetic regulation of metabolic networks. To achieve these purposes, a tool that rapidly determines the compound identity, pathways, and metabolic networks was needed [47, 48]. The tool accepts queries from common data types and facilitates data integration from independent sources into a unified compound identification and pathway-mapping scheme. To our knowledge, this is the first tool of this type to be developed. The Metabolome Searcher addresses these purposes by receiving input from the user, querying the user-selected metabolic reference database(s), and displaying the generated output for further biological interpretation (Figure 8).

Of the Metabolome Searcher’s outputs, the compounds file is useful when the user plans to conduct compound classification, data clustering, principal component analysis, analysis of variance, or graphical visualization. The pathways file allows the users to sort the data by pathways and facilitates analysis of the matches by pathways to enable interpretation of metabolic flux and pathway connections to determine if a compound is an intermediate or an end product. The main feature of the HTML output is that it lists and links compounds to all metabolic pathways in which the metabolite is involved (Figure 8). These links help the user understand the role of that particular metabolite in the organism’s metabolic network. The user can click on any one of these links that will navigate them to the PGDBs curated and hosted at ProCyc (www.usu.edu/westcent/procyc). The user need not repeat queries on the Metabolome Searcher as the
HTML file contains the links to the pathways associated with the returned putative compound IDs.

**Verification**

For names, formulae, and SMILES structures, any partial matches will also be detected and listed. For example, a query of “glucose” against the MetaCyc database identifies D-glucose and an additional 52 hits (data not shown) that also include alpha-methyl-glucose, NDP-Glucoses, as well as all other molecules that contain the substring “glucose” in the name. String matching offers the user the ability to obtain partial matches and allows additional control over the query specificity and flexibility for unknown pathways. In most cases, if the specific MetaCyc compound names are used, the results will be restricted to one hit.

Compound identification from LC-MS or NMR spectrometry data has proven to be a challenge to biologists because the compound databases are limited, especially with respect to the compounds that a specific organism can produce. Based on the user selection of MRDB(s) in Metabolome Searcher, the number of hits is refined and is metabolically relevant to the organism under study, thus providing a basis for biological conclusions to be drawn. As an example of the convenience provided by Metabolome Searcher, we initially queried the MetaCyc MRDB with the monoisotopic mass of isocitrate as the search query and used the results for further narrowing the hits by querying organism-specific MRDBs. These results were compared to those hits obtained by querying the monoisotopic mass of isocitrate using Chemspider (Figure 9). The ChemSpider query returned 118 possible compound identifications that included non-
biological compounds and required extensive analysis outside the query system to derive possible identifications, whereas querying the MetaCyc MRDB provided hits that included ten compounds with similar monoisotopic masses to that of isocitrate. Each genome (i.e., organism) further reduced the hits to two to five compounds that reflected the genetic differences in metabolism, all of which were related to citrate. Combining genome restriction with the MS compound list refined the possible identification list to a low number of compounds that was reasonable for empirical confirmation.

The interface and search function were verified by accessing the database search function, using known exact masses and a data set generated from a known mixture of compounds (i.e., a chemically defined bacterial growth medium) from LC-MS output. The resulting markers exported into a MarkerLynx format text file were used to query the compound identification using Metabolome Searcher. All the main ingredients of the growth medium represented in the MetaCyc MRDB were detected during the search. MOPS, a buffering salt, was used as a negative control for the chemical challenge, which was done by excluding it from the MetaCyc MRDB. Interestingly, after excluding MOPS, some of the query masses also matched multiple metabolites, many of which were isomeric forms of the metabolites being tested. This allowed further restriction of identification to narrower ranges of mass deviation to obtain better accuracy. However, in nearly 90% of compounds identified, the number of hits was limited to less than five metabolites, thus aiding the directed development of protocols for further compound identification. This approach enabled detection of common starting substrates for metabolism and verified that if the compound was in the database Metabolome Searcher
found it.

**Uses of metabolome searcher**

An example demonstration of the Metabolome Searcher for microbial metabolomics was performed by collecting metabolomics profiles for both sterile chemically defined media and spent media collected after inoculation with the bacterium *Lactococcus lactis* IL1403 for 16 h. Metabolomics profiles were collected by LC-MS analysis in both positive and negative electrospray modes for the same samples, and the masses obtained from MarkerLynx were queried against the *L. lactis* IL1403 MRDB (Table 5). After overlaying the compound identifications, we quickly inferred changes in compound classes, such as amino acids (Figure 10), by sorting the compounds file, or pathways file that changed during growth of *L. lactis* IL1403 (Figure 10). This example demonstrated that Metabolome Searcher performed the intended search and enabled the biological meaning to rapidly assign the identified compounds using constructed databases from metabolic reconstruction maps.

**Conclusions**

The Metabolome Searcher provides an automated tool to identify metabolites from MS analyses from metabolic reconstruction of specific genomes. This approach couples long lists of masses to specific genomic-based metabolites for identification and subsequent visualization via metabolic pathways. The tool is flexible so that queries can use many types of data that include names, molecular formulae, or SMILES structures, and monoisotopic masses that are entered singly or in bulk as a text file. The matches to queries are then presented as results along with other input parameters that the user
included in the query and the pathways in which the matched metabolites are involved. The versatility of accepted query types and the provision of pathways mapped to queries are unique to the Metabolome Searcher. The Metabolome Searcher’s utility and flexibility facilitates rapid advances from metabolomics to biological comprehension.

References


Table 4 - Metabolite distribution by molecular mass across metabolic encyclopedias

<table>
<thead>
<tr>
<th>Molecular weight range (Da)</th>
<th>Number of compounds in Metacyc</th>
<th>Number of compounds in KEGG</th>
</tr>
</thead>
<tbody>
<tr>
<td>50-99</td>
<td>254</td>
<td>267</td>
</tr>
<tr>
<td>100-199</td>
<td>1,731</td>
<td>2,402</td>
</tr>
<tr>
<td>200-299</td>
<td>1,050</td>
<td>2,614</td>
</tr>
<tr>
<td>300-399</td>
<td>842</td>
<td>2,575</td>
</tr>
<tr>
<td>400-499</td>
<td>461</td>
<td>1,150</td>
</tr>
<tr>
<td>500-599</td>
<td>311</td>
<td>719</td>
</tr>
<tr>
<td>600-699</td>
<td>169</td>
<td>337</td>
</tr>
<tr>
<td>700-799</td>
<td>199</td>
<td>201</td>
</tr>
<tr>
<td>800-899</td>
<td>198</td>
<td>232</td>
</tr>
<tr>
<td>900-999</td>
<td>133</td>
<td>160</td>
</tr>
<tr>
<td>&gt;1,000</td>
<td>119</td>
<td>224</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>5,467</strong></td>
<td><strong>10,881</strong></td>
</tr>
</tbody>
</table>
Table 5 - Organism-specific and general metabolic reference databases available for the Metabolome Searcher

<table>
<thead>
<tr>
<th>Organism</th>
<th>ProCyc Database</th>
<th>Metabolic reference database</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Escherichia coli</em> K12</td>
<td>EcoCyc</td>
<td><em>E. coli</em> K12</td>
</tr>
<tr>
<td><em>Escherichia coli</em> O157:H7</td>
<td>Ecoo157Cyc</td>
<td><em>E. coli</em> O157:H7</td>
</tr>
<tr>
<td><em>Homosapiens</em></td>
<td>HumanCyc</td>
<td><em>Homo sapiens</em></td>
</tr>
<tr>
<td>KEGG Compounds</td>
<td></td>
<td>KEGG Compounds</td>
</tr>
<tr>
<td><em>Lactococcus lactis</em> ssp. lactis IL1403</td>
<td>LlactisCyc&lt;sup&gt;1&lt;/sup&gt;</td>
<td><em>L. lactis</em> ssp. lactis IL1403</td>
</tr>
<tr>
<td><em>Lactococcus lactis</em> ssp. cremoris SK11</td>
<td>LaccremoCyc&lt;sup&gt;1&lt;/sup&gt;</td>
<td><em>L. lactis</em> ssp. cremoris SK11</td>
</tr>
<tr>
<td><em>Lactobacillus acidophilus</em> NCFM</td>
<td>LbacidCyc&lt;sup&gt;1&lt;/sup&gt;</td>
<td><em>Lb. acidophilus</em> NCFM</td>
</tr>
<tr>
<td><em>Lactobacillus johnsonii</em> NCC 533</td>
<td>LbjohnCyc&lt;sup&gt;1&lt;/sup&gt;</td>
<td><em>Lb. johnsonii</em> NCC 533</td>
</tr>
<tr>
<td><em>Lactobacillus plantarum</em> WCFS1</td>
<td>LbplanCyc&lt;sup&gt;1&lt;/sup&gt;</td>
<td><em>Lb. plantarum</em> WCFS1</td>
</tr>
<tr>
<td><em>Listeria monocytogenes</em> EGDe</td>
<td>LmonoCyc&lt;sup&gt;1&lt;/sup&gt;</td>
<td><em>Listeria</em> monocytogenes EGDe</td>
</tr>
<tr>
<td><em>Mycobacterium bovis</em> AF2122/97</td>
<td>MbovisCyc&lt;sup&gt;1&lt;/sup&gt;</td>
<td><em>M. bovis</em> AF2122/97</td>
</tr>
<tr>
<td>MetaCyc</td>
<td>MetaCyc</td>
<td>MetaCyc</td>
</tr>
<tr>
<td><em>Staphylococcus aureus</em> Mu50</td>
<td>SaureusCyc&lt;sup&gt;1&lt;/sup&gt;</td>
<td><em>S. aureus</em> ssp. aureus Mu50</td>
</tr>
<tr>
<td><em>Saccharomyces cerevisiae</em> S288C</td>
<td>YeastCyc&lt;sup&gt;2&lt;/sup&gt;</td>
<td><em>S. cerevisiae</em> S288C</td>
</tr>
<tr>
<td><em>Salmonella enterica</em> servoTyphimurium LT2</td>
<td>Styp99287Cyc&lt;sup&gt;3&lt;/sup&gt;</td>
<td><em>Salmonella typhimurium LT2</em></td>
</tr>
</tbody>
</table>

<sup>1</sup>PGDBs reconstructed, curated and hosted in ProCyc
<sup>2</sup>Obtained from the Yeast genome database
<sup>3</sup>Downloaded from the Pathway Tools registry of PGDBs
Metabolome Database Searcher

Search for all pathways containing compounds of interest.

Enter a COMPOUND by name, chemical formula, smiles structure, or molecular weight or monoisotopic mass (e.g. 60, 60.1, 60-70 for a range)

If mass value, check query type: Monoisotopicmass Molecularweight

Electrospray Mode: Positive Negative

Number of proton charge states:

Mass Deviation: 25 ppm Da

(Enter a single value or a range for mass deviation. For example, 100 or 0-100. Maximum value should be 500 ppm or 0.05 Da)

*Adduct or Deduct Molecule: Adduct Deduct

(Input a list of comma separated atoms or groups or their monoisotopic mass. For example: Na,Cl,NH2,CH3,69,012)

*Maximum number of adducts/deducts:

*Upload a text file containing a list of compounds:

(Choose File) no file selected

*Upload a MARKERLYNX file:

(Choose File) no file selected

-----------------------------------------------

B. linens ATCC 9174 E. coli K12 E. coli O157:H7 Homo sapiens KEGG_Compounds L. lactis IL1403 L. lactis subsp. cremoris SK11 Select organism(s)

Submit

Fields with * are OPTIONAL; all other fields need to be filled

Please turn off auto-entry options in your browser prior to using the searcher

Figure 6 - Metabolome Searcher user interface screenshot
Figure 7 - (A) Metabolome Searcher workflow and (B) the flowchart of the search operations to depict how the compounds are matched
Figure 8 - Screenshot of Metabolome Searcher’s output
(A) The top portion of the HTML results page and (B) the body of the HTML file
demonstrate that sections containing queries, matches, compound and pathway links, and
other data and information are provided with in the output
Figure 9 - Comparison of results from chemical vs metabolic databases with the monoisotopic mass of isocitrate (192.027±0.001 Da) as the query

Hits to an encyclopedia of genes (MetaCyc), *E. coli* (EcoCyc), *Listeria monocytogenes* (LmonCyc), *Lactococcus lactis* IL1403 (LlactisCyc), and *Lactobacillus johnsonii* (LbjohnCyc) databases were used to demonstrate multiple genome-restrictions using Metabolome Searcher.
Figure 10 - Pathway assembly of Metabolome Searcher output with heat maps of LC/MS data from the compounds file and the pathways file (tryptophan biosynthesis and asparagine biosynthesis I)

This output is obtained by clicking on the pathway link for “asparagine biosynthesis I” inside the HTML file that brings up the pathway page at ProCyc. The heat maps are color-coded with green being 0, black the median value, and red the highest value within that data set. The heatmap number of 1 was at the initial time and number 2 was 24 hours later, which demonstrates the change in metabolites, from the MS list and visualized with the pathway and concentration.
CHAPTER IV

SIMULATION AND ESTIMATION OF KINETIC PARAMETERS OF A METABOLIC NETWORK AT DYNAMIC STATE

Abstract

Background

Metabolism is one of the most complex cellular processes by which an organism maintains a living state. High-throughput sequencing technology has resulted in completely sequenced genome for many organisms. The metabolic network present in an organism can be effectively reconstructed from the annotated genome using software tools like Pathway Tools and Kyoto Encyclopedia of Genes and Genomes. However, understanding the function and regulation of the metabolism of living organisms is a major challenge. Metabolism has two states: dynamic and steady-state. Steady-state analysis of metabolism is straightforward and has received much attention in the recent past but reveals only a static snapshot of the metabolic network. Dynamic analysis sheds more light on the transient behavior of the network and helps predict changes in metabolism due to genetic or environmental perturbations. The focus of this research was on dynamic analysis of metabolic networks using novel variants of random-restart hill-climbing.

Results

In this work, the metabolic network of an organism was mathematically represented as a system of coupled ordinary differential equations. Four different variants of the random-restart hill-climbing search algorithm were employed in this study,
namely, random-mutation, directed-mutation, hybrid-mutation and alternate-mutation. The latter three are novel variants that were introduced in this work. Among the four variants, random-restart hybrid-mutation performed the best in estimating parameters that contributed to the time course experimental data. Integration of metabolomics and genomics data into the model was a highlight of this work. Four kinetic parameters that can describe the flux at dynamic state, namely, $K_R$, $V_{max_f}$, $V_{max_r}$ and $K_{dy}$, were defined and estimated. Metabolomics and genomics data from Salmonella typhimurium (pathogen) and Salmonella typhimurium in Caco-2 (host cell invaded by the pathogen) were employed in this study to analyze the shifts in metabolic flux in three basal pathways, namely, aspartate biosynthesis, glyoxylate cycle and tricarboxylic acid cycle. Kinetic parameters of the two experiments were successfully estimated and analyzed to diagnose the differences in the dynamic behavior of the metabolic pathway under the two given conditions.

**Conclusions**

Dynamic state metabolic flux analysis describes the behavior of the network and also serves as a guide to predict alterations in flux during unexplored conditions. The hybrid-mutation hill-climbing optimization approach that was employed to find the best-fit against the experimental data was successful in estimating kinetic parameters that represent flux at dynamic state. A proof of concept was used to validate the model and estimated kinetic parameters. This work on dynamic state analysis of metabolism is extensible to other metabolic pathways of any complexity present in any organism.
Background

Metabolism

Recent developments in high throughput biotechnology have paved the way for systemic analysis of an organism and in turn have resulted in enormous amounts of omics data. Consequently, there is a dire need for tools to analyze and integrate these large data sets to understand the various cellular mechanisms and their interactions that take place within an organism [1, 2]. Metabolism is one of the cellular processes that has been receiving increased focus during the past decade. Metabolism is the process by which organisms obtain energy from the food they intake to perform activities so as to maintain a living state [3]. The goal of metabolic analysis is to aid metabolic engineering in comprehending and improving the metabolic potentials of an organism to yield a desired product [4, 5]. Metabolic analysis also sheds light on metabolic changes due to genetic or environmental perturbations, which assist in diagnosing metabolic disorders and thereby the discovery of novel drug targets [6].

Metabolism occurs at a rapid rate when compared to other cellular processes and comprises of chemical reactions which usually are catalyzed by enzymes [7-9]. The process of metabolism consists of two phases or states classified based on the rate of product formation that varies with the substrate concentration. In an enzyme-catalyzed reaction, at lower substrate concentrations the rate rapidly increases when the substrate concentration increases, but at higher concentrations the rate reaches a maximum and then begins to level off. Based on this fact, the process of metabolism is demarcated into two states: pre-steady state and steady state. Pre-steady state is also referred to as the transient state or dynamic state. Figure 11 shows the two states of metabolism: dynamic
state preceding the steady state. During the dynamic state, the initial reaction velocity is proportional to the substrate concentration, whereas in the steady state region of the curve the initial reaction velocity remains constant independent of the substrate concentration [9].

An organism maintains its metabolism at homeostasis even with changing environmental conditions. Every reaction in the metabolic network of a cell is highly regulated and does not continue indefinitely even if conditions are favorable [10]. The phenomenon of homeostasis through regulation can be explained by characterizing the metabolic flux or fluxome of the cell. Metabolic flux profiling can be done considering the network to be at steady state or dynamic state.

**Reconstruction of metabolic network**

The genome of an organism dictates the metabolic capabilities present in the organism. Advancement in genome sequencing has resulted in a growing number of completely sequenced genomes. However, the experimental realization of the metabolic capabilities present in an organism is not on par with genome sequencing. Hence, the solution to address this lag is to derive the metabolic network information from the annotated genome sequence of the organism [11]. Bioinformatic tools like Pathway Tools and Kyoto Encyclopedia of Genes and Genomes have bridged the gap between the genome sequence and the metabolic network of an organism [12-15]. These tools are used for the reconstruction of the metabolic network from the annotated genome using a bottom up approach [7]. Once the metabolic network has been reconstructed from the annotated genome of an organism, the stage is set for metabolic analysis. Analysis of the
metabolic network includes the study of the structure, function, and regulatory mechanism of the network [16].

**Steady-state analysis**

A mathematical representation of the metabolic network is required to initiate the process of metabolic analysis. The stoichiometry of the network is an invariant property of the network since it is unaffected by external perturbations [17]. Stoichiometric modeling takes into consideration only the network topology and ignores other variant properties like kinetic parameters and intracellular metabolite concentrations [18]. Thus, stoichiometric modeling forms the core concept in steady state analysis of metabolic networks and is based on the principle of mass balance. This principle states that the formation fluxes must be balanced by degradation fluxes in the network and is mathematically represented as

$$\frac{dx}{dt} = S \cdot v$$  (1)

where $S$ is an $m \times n$ stoichiometric matrix with $m$ corresponding to the number of metabolites and $n$ is the number of reaction fluxes, $x$ is the metabolite concentration vector and $v$ is the flux vector [17]. When the network is at steady state, there is no change in metabolite concentrations and thus Equation 1 is reduced to

$$0 = S \cdot v$$  (2)

Stoichiometric modeling forms the basis for most of the analytical methods used for metabolic analysis at steady state. Metabolic pathway analysis, metabolic flux analysis, and flux balance analysis use stoichiometric modeling as the foundation to assess functional properties and quantify metabolic flux of the network [19-21].
To begin with, steady state metabolic analysis explored the qualitative features of the network. Metabolic pathway analysis deals with the identification and analysis of significant routes in the network referred to as extreme pathways [20-23]. In addition, metabolic pathway analysis aids in testing the feasibility of reconstructed metabolic networks [24], calculating inherent properties of the network [20] and also to analyze the functions of the pathway for metabolic engineering [22]. The network topology, stoichiometry and the thermodynamic feasibility of the reactions are the properties of the network that are used in metabolic pathway analysis [22, 23]. Convex analysis is the mathematical approach used for solving the system of equations represented by Equation 2. The solution is the basis vector represented by the positive orthant and is called the convex polyhedral cone [21]. The extreme rays or the generating vectors of the cone constitute the set of extreme pathways present in the network.

The goals of metabolic flux analysis (MFA) are quantification of flux and to understand the interactions between pathways via flux control [25]. MFA also deals with identification of new routes, rigid branch points, and quantification of maximum theoretical yield of a product [26]. Initially, only the extracellular metabolite concentrations were used to calculate flux, but this approach was limited to pathways without cycles [27]. To overcome this limitation, the stoichiometry of the network and principle of mass balance were also included along with the extracellular metabolite concentrations to determine unknown fluxes [18]. In the $^{13}$C MFA approach, the measured isotopic label enrichments from isotopic labelling or carbon labelling experiments were also included as additional constraints to the stoichiometric model. The
additional constraints facilitated the flux estimations to be more precise than the previous approaches [25]. MFA also includes the enumeration of the set of elementary flux modes of the network. Elementary flux modes are similar to extreme pathways in metabolic pathway analysis. An elementary flux mode is the flux distribution through a nondecomposable subnetwork [28]. Two mathematical approaches can be used to determine the elementary flux modes: Schuster’s algorithm [23] and convex analysis [29]. The extreme rays of the convex cone represent the elementary flux modes of the network in convex analysis. Elementary flux modes help in predicting the functionality, robustness, and gene regulation in a network [16].

Flux balance analysis is another quantitative approach employed in metabolic analysis, and is used for obtaining the actual flux distribution within the feasible set of steady-state fluxes using a stated cellular objective [30]. The commonly used objective functions are maximization of biomass, minimizing ATP production, minimizing nutrient intake, and maximizing metabolite production. Stoichiometric modeling is utilized in flux balance analysis along with the metabolic demands placed on the network, as well as a few parameters specific to the organism [19]. Flux balance analysis predicts the optimal flux distribution (phenotype) in the network under certain conditions (adjustable and non-adjustable constraints) [18].

Metabolism is a complex phenomenon taking place in a living cell, and capturing such a complex process in a mathematical representation is not an easy task. In all the mathematical representations discussed thus far, the major assumption is that the system is at steady state. Steady state analysis comes with its own limitations. Steady state
implies that the metabolite pools are constant, and thus kinetic parameters are not included in these methods. Also, stoichiometric modeling and $^{13}$C MFA does not include regulatory control mechanisms. Ignoring kinetic parameters and regulatory mechanisms simplifies the mathematical representation, but analyzing the system at steady state provides only a static snapshot of the metabolic activities. The predictive nature is less in these models owing to the lack of regulatory mechanisms [31]. Metabolic pathways consist not only of metabolites but also many classical cofactors. To keep track of all the cofactors in the stoichiometric matrix becomes an overwhelming task. Stoichiometric modeling fails in situations involving parallel metabolic pathways and metabolic cycles not coupled to measurable fluxes [25].

**Dynamic state analysis**

Even though steady state analysis of a metabolic network has proven to be a success, the analysis only provides limited insight into the metabolism of an organism and is unable to determine the dynamic behavior of the network. Therefore, kinetic parameters and regulatory information must be included in the modeling to better understand the dynamic behavior of metabolism of an organism [32]. Dynamic metabolic modeling is extrapolative in nature because these models can help predict the dynamic behavior of the organism during unexamined conditions [8].

Dynamic state analysis provides information about the flux and inter-conversion of small molecules as opposed to steady state analysis that focuses on accumulation of biomass. Small molecules are metabolites like sugars, lipids, fatty acids, amino acids, nucleotides, vitamins, and cofactors with molecular weight lower than 800 Daltons.
Dynamic metabolic network modeling can explain properties of complex biological systems and also serve as a guide to experimentation [33]; however, the major challenge in this type of modeling is the estimation of a vast number of parameters.

Several optimization methods have been employed to simulate a metabolic network at dynamic state and to estimate the parameters involved in the model. Linear programming or linear optimization is a widely used optimization method because it can handle a large number of model parameters. For the purposes of employing this approach, the metabolic network was represented as an S-system and solved [34]. However, the objective function employed in this approach is linear, and thus linear programming cannot be used for biochemical systems that are non-linear in nature.

Non-linear optimization methods are ideal to estimate parameters in a biochemical system. Random search, steepest descent, Newton-type minimization using Lanczos method, gradient projection method using limited memory BFGS matrix, Monte Carlo simulated annealing with exponential cooling, evolutionary programming, genetic algorithms, multi-start hill-climbing, tensor method, Levenberg-Marquardt method, Hooke and Jeeves direct search method were the non-linear optimization methods that were implemented in Gepasi, a software written in C++ to estimate parameters of metabolic networks [35]. A numerical integration of a set of ordinary differential equations representing the metabolic network using an explicit fourth-order Runge-Kutta algorithm was employed in the Metabolic Interactive Simulation Tool (MIST) [36]. SCAMP is another tool that was used for simulation of metabolic and chemical network [37]. Monte Carlo simulation using the Gillespie algorithm was also used for kinetic
modeling of metabolic networks [38]. In addition to defining the structure of the network and kinetics of the reaction, numerical values of kinetic constants and initial concentrations of metabolites were required as input for optimization in all of the above tools. The input requirement posed a major problem since these kinetic constants are difficult to be measured experimentally.

There are four important variables that contribute to the metabolic characterization of an organism or cell during analysis, namely, mRNA transcripts, proteins, small molecules, and metabolic flux. A major constraint in building a successful kinetic model to analyze metabolism is that genomic, proteomic, and metabolomic data are to be measured under similar conditions over a course of time. In the past, this would have required a lot of effort and coordination among different laboratories [39]. However, due to recent advances in molecular biological techniques, omics data for an organism can be determined in parallel under the same conditions, thus in turn resulting in the availability of real biological data as inputs for simulation. This overcomes the limitation of metabolic modeling with hypothetical data as was done in the past. In addition, computational power was another requisite to satisfy the demands of developing and analyzing a genome-scale metabolic model. Modern-day advancement in computational capability, such as high-performance computing, has made genome-scale metabolic modeling feasible [40]. Also, the availability of several technical computing softwares have further aided in analyzing metabolism at dynamic state.
**Solving the inverse problem**

An inverse problem is one in which a system’s experimental measurements (effects) are used to characterize parameters (causes) that govern the system, which cannot be directly observed. Conversely, in a forward (direct/normal) problem, the solution is to predict the behavior of the system based on the complete description of the parameters that define the system [41]. This study involves the estimation of kinetic rate constants of the metabolic network given the experimental time course data and thus is classified to be an inverse problem. Solving this inverse problem using optimization is the best way to overcome the curse of dimensionality due to the complexity of the metabolic network [42].

**Methods**

**Mathematical representation of metabolic networks**

A biochemical network is a set of coupled reactions each consisting of reactants, products and the enzyme that catalyzes the reaction. The conventional representation of a biochemical reaction is shown in Figure 12a. As a first step towards metabolic analysis, the biochemical network must be represented in a conceptual form that encapsulates the architecture and topology of the network. This representation will enable the qualitative analysis of the network and aid in the visualization of the network. In this study, the biochemical network is mapped onto a directed hypergraph. A hypergraph is an extension of a graph wherein a generalized edge (hyperedge) connects more than two vertices (Figure 12b) [43]. A hypergraph representation is preferred over a graph representation because a hypergraph better encompasses the details of the biological process and thus serves as an excellent mathematical representation of biological networks [44]. In the
hypergraph representation of metabolic networks, the vertices symbolize metabolites and the transformations of metabolites (reactions) are represented by hyperedges.

For the purposes of quantitative analysis, the biochemical network is transformed into a set of coupled differential equations. This mathematical representation aids in the estimation of dynamic flux through the network. Typically, a biochemical reaction takes place in a series of elementary steps resulting in intermediates before the actual product of the reaction is formed. The sequence of intermediate steps of a reaction sums up to form the reaction mechanism. Every intermediate step of a reaction is considered to be reversible. For example, the intermediate steps of the hypothetical reaction in Figure 12a following an ordered sequential mechanism is given by Equation 3. In this case, the mechanism is also referred to as the ordered bi-bi mechanism because the reaction converts two reactants into two products.

\[
\begin{align*}
\text{E} + \text{A} &\rightleftharpoons \text{EA} \\
& \quad \quad k_{12} \\
\text{EA} + \text{B} &\rightleftharpoons \text{EAB} \\
& \quad \quad k_{23} \\
\text{EAB} &\rightleftharpoons \text{C} + \text{ED} \\
& \quad \quad k_{34} \\
\text{ED} &\rightleftharpoons \text{D} + \text{E} \\
& \quad \quad k_{41}
\end{align*}
\]
In Equation 3, E is the enzyme, A and B are the substrates, C and D are the products, $k_{12}$, $k_{23}$, $k_{34}$ and $k_{41}$ are the forward rate constants, and $k_{21}$, $k_{32}$, $k_{43}$ and $k_{14}$ are the reverse rate constants. Every metabolite or intermediate involved in the reaction is either formed or consumed during the course of the reaction. The formation and consumption of reaction intermediates can be mathematically represented using ordinary differential equations (ODE). Table 6 lists the basic biochemical transformations that were considered in this study and the corresponding differential equations. The set of ODEs of all the intermediates of every reaction in the metabolic network forms the set of coupled differential equations representing the network. In order to generate the set of coupled differential equations, the conventional representation (Figure 12a) is transformed into a symbolic representation (Figure 12c). Table 7 displays the three different representations for various network architectures that were considered in the study.

**Optimization methods**

*Random-mutation hill-climbing*

In optimization problems, the emphasis is on finding the goal state, and the path to the goal is irrelevant. So a local search algorithm such as a hill-climbing search can be employed to find the goal state. Even though the hill-climbing algorithm makes fast progress towards a solution, it looks only at immediate neighbors to make an uphill move and does not think of what is coming next. This might result in the search getting stuck in a local maximum even when a goal state exists. To overcome the problem of local maxima, this study began the search for optimum using random-restart hill-climbing, a variant of the hill-climbing algorithm. Random-restart hill-climbing comprises of several
trials of a hill-climbing search with randomly generated initial states. In random-mutation hill-climbing, one of the variables representing the current state is chosen at random and mutated. Random mutation of the state variable is repeated until it results in an uphill move. The random-restart random-mutation hill-climbing algorithm that was employed in this study is:

1. Assign uniformly distributed random numbers to the model parameters. Run the simulation and evaluate the cost function. Call this set of model parameters as the best-evaluated.

2. Pick a model parameter at random and mutate. If the mutation leads to a better fit, then set best-evaluated to the resulting set of model parameters.

3. Repeat step 2 until no more mutations improve the cost function.

4. Repeat steps 1-3 with a new set of model parameters for several times.

5. Return the overall best-evaluated as the solution to the inverse problem.

Directed-mutation hill-climbing

Directed-mutation hill-climbing (DMHC) is a novel variant of hill-climbing algorithm that was introduced in this study to search the state space. As the name implies, in DMHC, the parameter to be mutated is not chosen randomly, rather the information about the model parameters representing the current state is first analyzed by the mutation operator. Based on the analysis, the mutation is then directed towards a parameter, which contributes to an uphill move. Directed-mutation is analogous to induced-mutation in the genetic material of an organism.
Hybrid-mutation hill-climbing algorithm

This variant of hill-climbing was also introduced in this study. The hybrid-mutation hill-climbing (HMHC) algorithm employs a sequence of DMHC followed by RMHC. Based on observation, DMHC begins to fail when the solution is in proximity. One probable reason for failure is that the step-size of the mutation might be large resulting in overshooting the maximum. At such a point in the landscape, when DMHC exceeds the allowable limit of consecutive failures, the indication is that the solution is close at hand, but DMHC cannot proceed. Subsequently, RMHC takes over and guides the search towards the solution. The complete random-restart HMHC algorithm is

1. Assign uniformly distributed random numbers to the model parameters. Run the simulation and evaluate the cost function. Call this set of model parameters as the best-evaluated.

2. Analyze the set of model parameters and choose the parameter for directed or induced mutation. If the mutation leads to a better fit, then set best-evaluated to the resulting set of model parameters.

3. Repeat step 2 until no more mutations improve the cost function.

4. Pick a model parameter at random and mutate. If the mutation leads to a better fit, set best-evaluated to the resulting set of model parameters.

5. Repeat step 4 until no more mutations improve the cost function.

6. Repeat steps 1-5 several times with a new set of model parameters.

7. Return the overall-best-evaluated as the solution to the inverse problem.
Alternate-mutation hill-climbing algorithm

One other variant of hill-climbing is the alternate-mutation hill-climbing (AMHC) that was also initiated in this study to avoid local minima obtained as a result of employing DMHC. In the DMHC algorithm, the search results in local minima when directed-mutation fails. In this situation, the algorithm uses random mutation to jump out of the local minima. Once the search is out of the local minima, the algorithm switches to directed-mutation. Every time directed-mutation fails, a single random mutation is opted to bring the search out of the local minima, which is then followed by directed-mutations, and so on. Thus, directed and random mutations alternate until a global minimum is reached.

Cost function

An objective function is used in any optimization problem employing a local search algorithm. When the objective function is a cost function, the aim is to minimize the value of the function. The goal of an optimization problem is to optimize the value of the cost function so as to determine the best state that represents the solution to the problem [45]. The cost function employed for all the optimization methods used in this work is the sum of squares of the residuals between the experimental data and the model data [46]. The cost function is given by Equation 4

$$\sum_{t=0}^{T} \sum_{i=1}^{N} \left[ C_{\text{Model}}^{(i)}(t) - C_{\text{Data}}^{(i)}(t) \right]^2$$  \hspace{1cm} (4)

where T represents the time points at which the value of the metabolites were measured, N represents the different metabolites present in the pathway, $C_{\text{Model}}^{(i)}$ is the value of the
metabolite generated by the model, and $C_{data}$ is the value of the metabolite measured experimentally. In this study, “value” can represent concentration, peak area, or any other measure depicting the quantity of metabolite that was measured over time. This gives the flexibility to estimate flux irrespective of the units of the data that were experimentally measured.

**Definition of kinetic parameters**

$K_R$, $V_{maxf}$, $V_{maxr}$, and $K_{dy}$ are the kinetic parameters estimated and analyzed in this research. $K_m$, the Michaelis constant, describes the affinity of substrate towards the enzyme, as well as the rate at which the enzyme-substrate complex is converted to product. The value of $K_m$ of a substrate is directly proportional to the rate of product formation and inversely proportional to the affinity of the substrate towards the enzyme. The important assumption associated with deriving the equation for $K_m$ is that the reaction is at steady state. Since the focus of this study is dynamic state metabolic analysis, the assumption for $K_m$ does not hold true in this study. For this reason, a constant $K_R$ derived from $K_m$ was used in this study.

Based on the definitions of $K_m$ for reactants of uni-uni and bi-bi reaction [47], a generalized formula was derived to quantify enzyme affinity and rate of product formation. This formula is applicable for reactions: 1) with any number of substrates and products, and 2) at steady state or dynamic state. The formula of the kinetic parameter was determined using Ranjitha’s Shortcut Method (Appendix A) in this study and is denoted by $K_R$. The definition for $K_R$ is

$$K_R = \frac{k_{DFF} \cdot k_{DLC}}{k_{AR} \cdot k_{DP}}$$ (5)
where \( k_{DFP} \) = forward rate constant representing dissociation of final product from complex

\( k_{DLC} = \) Sum of the forward and reverse rate constants representing dissociation of largest enzyme-substrate complex. Special case: For the first substrate in a reaction with more than one substrate, only the forward rate constant representing dissociation of product from largest complex

\( k_{AR} = \) Forward rate constant representing association of this reactant with enzyme to form complex

\( k_{DP} = \) Sum of the forward rate constants representing dissociation of product(s) from respective complex(es)

Similarly, using the definition of maximum velocity of a reaction, \( V_{max} \), for a uni-uni and bi-bi reaction \[47\], the maximum forward velocity (\( V_{maxf} \)) and the maximum reverse velocity (\( V_{maxr} \)) for a reaction with any number of reactants can be determined.

The definition for \( V_{max} \) obtained using Ranjitha’s Shortcut Method is

\[
V_{max} = \left[ \prod_{i=x}^{n} \frac{k_i}{\sum_{i=x}^{n} k_i} \right] \times e_0
\]

(6)

where \( k_i, i = x \) to \( n \) are the forward rate constants involved in release of the products of the reaction. Note: a special case is when the reaction consists of only one product. In such a scenario, the denominator is substituted with the numerical value ‘1’.

Ranjitha’s Shortcut Method for \( K_R \) and \( V_{max} \) was proved by using the King-Altman method of deriving steady-state velocity equations \[48\] and the proof can be found in the supplement. The last kinetic parameter is \( K_{dy} \) and is the ratio of the rate
constant favoring the formation of the metabolite to the rate constant favoring the consumption of the metabolite, and is similar to $K_e$ defined in [49]. For example from Equation 3, $K_{dy}$ for metabolite A is given by

$$K_{dyA} = \frac{k_{21}}{k_{12}}$$  

(7)

If $K_{dyA} > 1$, then formation of the metabolite is favored whereas $K_{dyA} < 1$ implies that consumption of the metabolite is favored. The constant $K_e$ defined in [49] is calculated when the system is at equilibrium, i.e., the concentrations of the reactants are not changing but $K_{dy}$ of the metabolite is a ratio determined when the system is at dynamic state and remains a constant throughout the time period considered for dynamic analysis. The overall $K_{dy}$ of the reaction is

$$K_{dy} = \frac{\prod_{i=1}^{n} K_{dyi}}{\prod_{j=1}^{m} K_{dyj}}$$  

(8)

where $n$ is the number of products and $m$ is the number of substrates present in the reaction. $K_{dy} > 1$ favors product formation, $K_{dy} < 1$ favors substrate formation, and $K_{dy} = 1$ implies that the formation of products and substrates is balanced.

**Results and Discussion**

Analysis of metabolism has recently received considerable attention, with the intention to aid metabolic engineering in producing mutants that have desirable metabolic potentials. Biological methods for metabolic analysis have not advanced much when compared to developments in molecular tools for synthesis. Thus, bioinformatics fills the lag and thereby plays an important role in the process of metabolic engineering to meet
the requirements of metabolic analysis. Computational modeling of metabolism has improved the rate at which a new mutant can be constructed, especially that of a microbe.

In order to completely understand the metabolism of a cell, the four cellular variables, mRNA transcripts, proteins, small molecules and flux, have to be measured [1]. Recent high-throughput technologies have made measuring the former three variables at the same time under similar conditions feasible. But experimentally measuring metabolic flux poses a tedious and costly endeavor. Mathematical models and computer simulation have made the estimations of flux using other measurable quantities possible. This study introduces a framework to integrate genomics and metabolomics time course data in order to estimate the fluxome of the organism. Regulatory flux balance analysis (RFBA) and probabilistic regulation of metabolism (PROM) are other methods that have integrated genomics and metabolomics, but the modeling focuses on steady state analysis so as to circumvent the use of kinetics in the model [6].

**Case study**

In this study, proof of concept is used as a way to validate the mathematical modeling and simulation-optimization. Metabolic capabilities vary in different organisms based on their energy requirements, available nutrients, environmental and genetic conditions. Yet, there are a few basal metabolic pathways that are universally present across different organisms. Aspartate biosynthesis, the tricarboxylic acid cycle, and the glyoxylate cycle are three basal metabolic pathways that were chosen for analysis in this work, and are shown in Figure 13. These pathways are well-characterized components of a metabolic network and are of importance to the treatments considered in this study.
Two sets of experimental data were used for simulation-optimization. One set of data was measured when Salmonella was grown alone and is considered as the control dataset. Salmonella in Caco-2 cells was the other set of data and is considered to be the treatment. The dataset consisted of time course metabolite data and gene expression data.

**Performance of optimization algorithms**

The model was fit to the experimental data using the different variants of hill-climbing. Random-restart hill-climbing with random-mutation, directed-mutation, hybrid-mutation, and alternate-mutation were the variants used in this analysis. In other studies, several optimization methods have been employed for searching the space of kinetic constants like random search, steepest descent, Newton-type minimization using Lanczos method, gradient projection method using limited memory BFGS matrix, Monte Carlo simulated annealing with exponential cooling, evolutionary programming, genetic algorithm, multistart hill-climbing, tensor method, Levenberg-Marquardt method, Hooke and Jeeves direct search method, fourth order Runge-Kutta algorithm, and Gillespie algorithm [35, 38]. This study began with random-restart hill-climbing with random mutation so as to prove that the fitting of the model against experimental values was accurate and then progressed towards employing different mutation methods in order to decrease the computational complexity.

Figure 14 shows the performance of these algorithms using a plot of the value of the cost function against the simulations. The slope of RMHC is lower than the other three variants employing directed-mutation, implying that directed-mutation variants approached the goal faster than the usual RMHC. The plots of all the variants involving
directed-mutation followed the same trend and also overlapped, validating the robustness of the method of directed-mutation. The different factors that were used to judge the algorithms, namely, the least squared difference value obtained using the cost function, time taken for reaching the optimum, number of simulations to reach the optimum, and the percentage of simulations that minimized the cost function, are listed in Table 8.

Considering the time taken by the algorithms, DMHC reached the minimum in comparatively less time, and the percentage of better simulations was also high. However, the value of the cost function projects that the algorithm reached a local minimum due to overshooting the next best move, most probably because of the step-size of the directed-mutation being large. The next best performer based on time taken and the percentage of better simulations was AMHC, but the cost function did not result in the least value among the variants. RMHC not only resulted in the greatest value for the cost function but also took the longest time to reach the value. HMHC generated the best value for the cost function, but the percentage of better simulations was the least. As shown in Figure 14, this can be attributed to the random-mutations that followed directed-mutations. However, the time taken to get the best fit was still better than RMHC. Thus, HMHC combined the goodness of both mutations to get the best fit between the experimental data and model data, and hence was chosen for solving the inverse problem. The number of simulations of the various optimization methods used in [42] are far greater than the number simulations of the methods used in this study. Also, the metabolic pathway considered in [42] was smaller than the one used in this work.
**Kinetic parameters**

There are twelve reactions present in the three pathways, and few of the reactions are present in two overlapping pathways, namely, tricarboxylic acid cycle and glyoxylate cycle. The three metabolic pathways consisted of twelve metabolites and eleven enzymes. Each reaction was modeled to follow sequential mechanism that resulted in a metabolic model comprising 54 intermediate kinetic constants. The kinetic constants were estimated in this study as opposed to other studies [35-37] that required the kinetic constants as input to model the metabolic pathway. These kinetic constants are difficult to measure experimentally. Thus, providing these constants as input posed a major challenge.

The kinetic parameters of the model estimated by the simulation-optimization approach were validated based on the extent to which the model data fits the experimental data, and thereby describes the system’s behavior. The best-fit of the model against the experimental data is shown in Figure 15. The x-axis is the time in minutes, and the y-axis represents the log2 of the peak area of the metabolite measured using mass spectrometry. The experimental data of the metabolite is the line with circular markers, and the line with square markers indicates the model data that were produced as the best-fit. Experimental data from the treatment of *Salmonella* with Caco-2 was used to produce these plots. Other researches that performed dynamic state analysis of metabolism have used only hypothetical data [42, 46] as opposed to real data that was used in this study.

The model resulted in a good fit for most of the metabolites, whereas the fit for glutamate and aspartate were not so good, indicating that the difference was due to external factors like regulatory information that were not accounted for in the metabolic
model under study [32]. However, the nonlinear behavior observed in the model data indicates the presence of some kind of regulation of the production or consumption of a metabolite as an emergent property of the model.

The kinetic parameters are intrinsic properties of the enzyme and the substrates involved in the reaction under certain conditions. In this work, a control and a treatment, Salmonella, and Salmonella with Caco-2, respectively, were considered; and the corresponding intermediate rate constants were estimated using HMHC. The kinetic parameters, $K_R$, $V_{maxf}$, $V_{maxr}$ and $K_{dy}$, were calculated for the control and for the treatment using the formula defined in the previous section, and for the estimated intermediate kinetic constants. Figures 6 and 7 display the estimated values of $K_R$ overlaid on the three pathways of Salmonella and Salmonella with Caco2, respectively. The values of the kinetic parameters $V_{maxf}$, $V_{maxr}$ and $K_{dy}$ are tabulated in Tables 4, 5, and 6, respectively. Validating these kinetic parameters is not trivial because of the inadequate information available about the model parameters in the literature and public databases. The intermediate kinetic constants cannot be measured from experiments or even calculated based on available information. Moreover, the parameters are contingent upon the in vivo or in vitro conditions under which they are experimentally measured when feasible [50]. For these reasons, the only way to validate these model parameters is by proof of concept, i.e., estimating the model parameters by approximately fitting the model to the experimental data. Thus, the study employs proof of concept to validate the model parameters and provides relative insight of the parameter values based on different conditions or enzyme activities in reactions within the same condition.
Figure 16 shows that oxaloacetate has the least value of $K_R$ for aspartate transaminase compared to citrate synthase or malate dehydrogenase, implying that oxaloacetate was actively taking part in aspartate biosynthesis. Aspartate transaminase continued using oxaloacetate longer than the other two enzymes, and thus resulted in the production of more aspartate and $\alpha$-ketoglutarate. Also, the $K_R$ for $\alpha$-ketoglutarate is high when compared to the other metabolites involved in aspartate biosynthesis, proving the fact that the reaction was proceeding in the forward direction and the reverse reaction was slow compared to the forward reaction. Supporting this information is the value of $V_{\text{maxf}}$ for aspartate transaminase from Table 9, which is very low compared to other reactions involving oxaloacetate and is also lower than $V_{\text{maxr}}$. The inference from the estimated kinetic parameters for this experiment is that the aspartate biosynthesis pathway was active among the three pathways that were considered in the study.

Similarly, succinate was used by succinate dehydrogenase and not by succinyl-CoA synthetase or isocitrate lyase. This conclusion is based on the $K_R$ values for succinate for the three enzymes. In addition, the reaction proceeded in the forward reaction to produce fumarate because $V_{\text{maxf}}$ is less than $V_{\text{maxr}}$ for the reaction catalyzed by succinate dehydrogenase. The glyoxylate cycle was actively producing glyoxylate that was converted into malate that then entered the TCA cycle; and the by-product of the glyoxylate cycle was succinate, which had also entered the TCA cycle.

The $K_R$ value of oxaloacetate for aspartate transaminase increased in Salmonella with Caco-2 as compared to Salmonella in isolation, indicating a decrease in $\alpha$-ketoglutarate production during the treatment in the time interval. Instead, oxaloacetate
was routed towards the TCA cycle due to the fact that $K_R$ of oxaloacetate for citrate synthase is the least at the branch point. Furthermore, the reaction proceeded in the forward direction producing citrate because $V_{\text{max}_f}$ is greater than $V_{\text{max}_r}$. Isocitrate was taking part in the TCA cycle, since $K_R$ of isocitrate for isocitrate dehydrogenase is the least at this branch point. The conclusion from the treatment is that the TCA cycle was active during the treatment in the particular time period.

In summary, different pathways were active in the control and treatment during the given time interval. There is a shift in the active route of metabolism when Caco-2 is added to Salmonella. Similar conclusions can be drawn using the estimated kinetic parameters for the various reactions present in different pathways at different environmental or genetic conditions.

**Time complexity**

Consider a set of $n$ reactions. On an average, every reaction consists of four reactants (metabolites and enzymes). There are two steps involved in the simulation: to generate the differential equations, and estimate the kinetic parameters. The mathematical representation of the metabolic network consists of one differential equation per reactant to produce the set of differential equations. Therefore, the algorithm to generate the set of differential equations is $O(n)$, i.e., the algorithm runs in linear time. This is applicable when the set of reactions are not coupled (connected) in the pathway. Typically, in a metabolic network, the reactions are connected and the running time becomes $O(\log n)$, i.e., the set of coupled differential equations are produced by the algorithm in logarithmic time [51].
An optimization approach is used to estimate the kinetic constants. An optimization problem deals with finding the optimum, in this case the minimum of the cost function. Usually, a cost function is non-linear, and the constraints to the function are also non-linear. Due to the non-linearity of the cost function, there could be several minima. In many cases, the local minimum will be sufficient and can be considered to approximate the global minimum. Hard combinatorial problems dealing with optimization are considered to be NP-complete [42].

**Conclusions**

Analysis of metabolism at dynamic state provides more information about the behavior of the system than its steady state counterpart. Steady state analysis can successfully project the possible flux distributions, optimal flux distribution using an objective function, and extreme pathways present in a metabolic network. The major drawback of steady state analysis is that the model lacks predictive power and can provide only a snapshot. In the past, metabolic models were restricted to steady state due to the limitations of computational capability and accessibility to experimental data. Recent high-throughput technologies have helped overcome these limitations and have paved the way for dynamic state analysis of metabolism which in turn can play an important role in metabolic engineering.

The focus of this study was dynamic flux analysis for characterizing the behavior of the metabolic system. The model was validated using metabolomics and genomic data of Salmonella typhimurium measured under different experimental conditions. Estimating kinetic parameters from experimental time course data is categorized as an
inverse problem. The metabolic network was represented as a set of coupled differential equations, and optimization was used to solve the inverse problem. The highlight of this work was the integration of metabolomics and genomics data for modeling the behavior of the system. Three variants of random-restart hill-climbing algorithm were introduced and employed along with random-mutation hill-climbing to optimize the cost function in order to find the best fit for the experimental data. Hybrid-mutation hill-climbing was the variant that out-performed the other approaches. Thus, kinetic parameters of the metabolic model at dynamic state were successfully estimated and validated based on proof of concept, since the actual parameter values for dynamic state are difficult to be measured experimentally.

References


40. Henry CS, Xia F, Stevens R: Application of high-performance computing to
the reconstruction, analysis, and optimization of genome-scale metabolic

41. Velho HFDC, Ramos FM, Chalhoub ES, Stephany S, Carvalho JoC, De Sousa
FL: Inverse problems in space science and technology. Inverse Problems in

42. Mendes P, Kell D: Non-linear optimization of biochemical pathways:
Applications to metabolic engineering and parameter estimation.

43. Guedes ALP, Markenzon L: Directed hypergraph planarity. Pesquisa


46. Henning PA, Moffitt RA, Allegood JC, Wang E, Merrill Jr. AH, Wang MD:
Computationally predicting rate constants in pathway models. In: IEEE
Engineering in Medicine and Biology 27th Annual Conference: September 1-4
2005; Shanghai, China; 2005.

47. Cornish-Bowden A: Fundamentals of enzyme kinetics, 2nd edn. London:


Table 6 - Basic biochemical transformations and differential equations

<table>
<thead>
<tr>
<th>Type</th>
<th>Representation</th>
<th>Description</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$y \rightarrow x^k$</td>
<td>$x$ direct production by $y$</td>
<td>$\frac{d}{dt}[x] = k[y]$</td>
</tr>
<tr>
<td>2</td>
<td>$x \rightarrow y^k$</td>
<td>$x$ direct consumption by $y$</td>
<td>$\frac{d}{dt}[x] = -k[y]$</td>
</tr>
<tr>
<td>3</td>
<td>$y + z \rightarrow x^k$</td>
<td>$x$ combined production by $y$ &amp; $z$</td>
<td>$\frac{d}{dt}[x] = k[y][z]$</td>
</tr>
<tr>
<td>4</td>
<td>$x \rightarrow y + z^k$</td>
<td>$x$ combined consumption by $y$ &amp; $z$</td>
<td>$\frac{d}{dt}[x] = -k[y][z]$</td>
</tr>
</tbody>
</table>
### Table 7 - Different representations of various reaction types

<table>
<thead>
<tr>
<th>Reaction type</th>
<th>Network Representation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>i. Linear</strong></td>
<td><strong>Biochemical</strong></td>
</tr>
<tr>
<td></td>
<td>Oxaloacetate + GltA + Acetyl-CoA -&gt; Citrate + CoenzymeA</td>
</tr>
<tr>
<td></td>
<td>i. Linear</td>
</tr>
<tr>
<td></td>
<td>citrate + acnB -&gt; cis-aconitate</td>
</tr>
<tr>
<td></td>
<td>cis-aconitate + acnA -&gt; isocitrate</td>
</tr>
<tr>
<td></td>
<td>iia. Single</td>
</tr>
<tr>
<td></td>
<td>iib. Double</td>
</tr>
<tr>
<td></td>
<td>iib. Double</td>
</tr>
<tr>
<td></td>
<td>iic. Triple</td>
</tr>
<tr>
<td></td>
<td>iic. Triple</td>
</tr>
<tr>
<td></td>
<td>iic. Triple</td>
</tr>
<tr>
<td></td>
<td>iii. Cyclic</td>
</tr>
<tr>
<td></td>
<td>iii. Cyclic</td>
</tr>
<tr>
<td></td>
<td>iii. Cyclic</td>
</tr>
<tr>
<td></td>
<td>iii. Cyclic</td>
</tr>
<tr>
<td></td>
<td>iii. Cyclic</td>
</tr>
</tbody>
</table>
Table 8 - Different factors depicting the performance of the variants of hill-climbing

<table>
<thead>
<tr>
<th>Mutation Method</th>
<th>Cost Function</th>
<th>Time (sec)</th>
<th>Simulations</th>
<th>Better Simulations</th>
<th>% Better Simulations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>230.249</td>
<td>9726</td>
<td>7731</td>
<td>3048</td>
<td>39.43</td>
</tr>
<tr>
<td>Directed</td>
<td>76.098</td>
<td>836</td>
<td>3638</td>
<td>2262</td>
<td>62.18</td>
</tr>
<tr>
<td>Hybrid</td>
<td>8.0837</td>
<td>4600</td>
<td>13713</td>
<td>4123</td>
<td>30.07</td>
</tr>
<tr>
<td>Alternate</td>
<td>23.6257</td>
<td>1642</td>
<td>4882</td>
<td>2524</td>
<td>51.70</td>
</tr>
</tbody>
</table>
Table 9 - Kinetic parameters $V_{\text{maxf}}$ and $V_{\text{maxr}}$ of all reactions in the three pathways of Salmonella

<table>
<thead>
<tr>
<th>Reaction</th>
<th>Pathway</th>
<th>Enzyme</th>
<th>$V_{\text{maxf}}$</th>
<th>$V_{\text{maxr}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Aspartate Biosynthesis</td>
<td>Aspartate transaminase</td>
<td>39.7</td>
<td>71.3</td>
</tr>
<tr>
<td>2</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>aconitase B (citrate-&gt;aconitate)</td>
<td>5001.3</td>
<td>22.0</td>
</tr>
<tr>
<td>3</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>citrate synthase</td>
<td>2165.1</td>
<td>1391.6</td>
</tr>
<tr>
<td>4</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>aconitase B (aconitate-&gt;isocit)</td>
<td>4728.8</td>
<td>904.3</td>
</tr>
<tr>
<td>5</td>
<td>Glyoxylate Cycle</td>
<td>isocitrate lyase</td>
<td>1421.5</td>
<td>360.9</td>
</tr>
<tr>
<td>6</td>
<td>Glyoxylate Cycle</td>
<td>malate synthase G</td>
<td>386.3</td>
<td>411.2</td>
</tr>
<tr>
<td>7</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>malate dehydrogenase</td>
<td>2443.6</td>
<td>714.8</td>
</tr>
<tr>
<td>8</td>
<td>TCA cycle</td>
<td>succinate dehydrogenase</td>
<td>1877.9</td>
<td>2647.1</td>
</tr>
<tr>
<td>9</td>
<td>TCA cycle</td>
<td>isocitrate dehydrogenase</td>
<td>3961.9</td>
<td>2364.0</td>
</tr>
<tr>
<td>10</td>
<td>TCA cycle</td>
<td>2-oxoglutarate dehydrogenase</td>
<td>1652.9</td>
<td>784.0</td>
</tr>
<tr>
<td>11</td>
<td>TCA cycle</td>
<td>succinyl-CoA synthetase</td>
<td>1405.0</td>
<td>3430.6</td>
</tr>
<tr>
<td>12</td>
<td>TCA cycle</td>
<td>fumarase A</td>
<td>1622.8</td>
<td>775.3</td>
</tr>
<tr>
<td>Reaction</td>
<td>Pathway</td>
<td>Enzyme</td>
<td>$V_{\text{max}}$</td>
<td>$V_{\text{maxr}}$</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------</td>
<td>---------------------------------------------</td>
<td>------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>1</td>
<td>Aspartate Biosynthesis</td>
<td>Aspartate transaminase</td>
<td>19259.6</td>
<td>8651.0</td>
</tr>
<tr>
<td>2</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>aconitase B (citrate-&gt;aconitate)</td>
<td>2077.7</td>
<td>1326.2</td>
</tr>
<tr>
<td>3</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>citrate synthase</td>
<td>259.6</td>
<td>3095.0</td>
</tr>
<tr>
<td>4</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>aconitase B (aconitate-&gt;isocit)</td>
<td>1011.6</td>
<td>9111.5</td>
</tr>
<tr>
<td>5</td>
<td>Glyoxylate Cycle</td>
<td>isocitrate lyase</td>
<td>101654.0</td>
<td>64.9</td>
</tr>
<tr>
<td>6</td>
<td>Glyoxylate Cycle</td>
<td>malate synthase G</td>
<td>668.7</td>
<td>1041.0</td>
</tr>
<tr>
<td>7</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>malate dehydrogenase</td>
<td>1293.6</td>
<td>8728.2</td>
</tr>
<tr>
<td>8</td>
<td>TCA cycle</td>
<td>succinate dehydrogenase</td>
<td>1189.8</td>
<td>1149.6</td>
</tr>
<tr>
<td>9</td>
<td>TCA cycle</td>
<td>isocitrate dehydrogenase</td>
<td>2376.4</td>
<td>2561.5</td>
</tr>
<tr>
<td>10</td>
<td>TCA cycle</td>
<td>2-oxoglutarate dehydrogenase</td>
<td>2276.1</td>
<td>1131.2</td>
</tr>
<tr>
<td>11</td>
<td>TCA cycle</td>
<td>succinyl-CoA synthetase</td>
<td>2634.3</td>
<td>723.9</td>
</tr>
<tr>
<td>12</td>
<td>TCA cycle</td>
<td>fumarase A</td>
<td>802.1</td>
<td>64659.8</td>
</tr>
</tbody>
</table>
Table 11 - Kinetic parameter $K_{dy}$ of all reactions in the three pathways of Salmonella and Salmonella + Caco-2

<table>
<thead>
<tr>
<th>Reaction</th>
<th>Pathway</th>
<th>Enzyme</th>
<th>Salmonella</th>
<th>Salmonella + Caco2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Aspartate Biosynthesis TCA cycle &amp; Glyoxylate Cycle</td>
<td>Aspartate transaminase</td>
<td>0.8</td>
<td>59082.6</td>
</tr>
<tr>
<td>2</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>aconitase B (citrate-&gt;aconitate)</td>
<td>421.5</td>
<td>0.3</td>
</tr>
<tr>
<td>3</td>
<td>TCA cycle &amp; Glyoxylate Cycle</td>
<td>citrate synthase</td>
<td>1.2</td>
<td>0.5</td>
</tr>
<tr>
<td>4</td>
<td>Glyoxylate Cycle</td>
<td>aconitase B (aconitate-&gt;isocit)</td>
<td>8.3</td>
<td>0.1</td>
</tr>
<tr>
<td>5</td>
<td>Glyoxylate Cycle</td>
<td>isocitrate lyase</td>
<td>130733.0</td>
<td>4.4*10^12</td>
</tr>
<tr>
<td>6</td>
<td>Glyoxylate Cycle</td>
<td>malate synthase G</td>
<td>3.3</td>
<td>1.7</td>
</tr>
<tr>
<td>7</td>
<td>Glyoxylate Cycle</td>
<td>malate dehydrogenase</td>
<td>5.1</td>
<td>0.4</td>
</tr>
<tr>
<td>8</td>
<td>TCA cycle</td>
<td>succinate dehydrogenase isocitrate dehydrogenase</td>
<td>1.3</td>
<td>0.4</td>
</tr>
<tr>
<td>9</td>
<td>TCA cycle</td>
<td>2-oxoglutarate dehydrogenase succinyl-CoA synthetase</td>
<td>1.6</td>
<td>0.6</td>
</tr>
<tr>
<td>10</td>
<td>TCA cycle</td>
<td>succinyl-CoA synthetase</td>
<td>7.3</td>
<td>0.1</td>
</tr>
<tr>
<td>11</td>
<td>TCA cycle</td>
<td>fumarase A</td>
<td>1.1</td>
<td>0.3</td>
</tr>
<tr>
<td>12</td>
<td>TCA cycle</td>
<td>fumarase A</td>
<td>1.4</td>
<td>0.0</td>
</tr>
</tbody>
</table>
Figure 11 - Two states of metabolism: Dynamic State vs Steady State
Figure 12 - Different representations of a biochemical reaction
(a) Conventional, (b) Hypergraph and (c) Symbolic. E – enzyme, A & B – substrates, C & D – products
Figure 13 - Three metabolic pathways used in the study
1) Aspartate biosynthesis, 2) Tricarboxylic acid cycle and 3) Glyoxylate cycle
Figure 14 - Plot of the different hill-climbing variants
Figure 15 - Experimental data vs Model data for the various metabolites present in the three metabolic pathways
The line with circular markers represents experimental data and the line with square markers denotes model data. RSS is the residual sum of squares value for the metabolite.
Figure 16 - The three metabolic pathways overlaid with the $K_R$ values of Salmonella
Figure 17 - The three metabolic pathways overlaid with the $K_R$ values of Salmonella & Caco-2
CHAPTER V

DYNAFLUX: A TOOL FOR DYNAMIC STATE FLUX

ESTIMATION AND NETWORK ANALYSES

Abstract

Background
A major challenge in the post-genomic era is to understand the metabolism of an organism. The four cellular variables, namely, mRNA transcripts, proteins, small molecules, and metabolic flux aid in the comprehensive characterization of the metabolic capabilities present in an organism. mRNA transcripts, proteins, and small molecules can be directly measured using high-throughput omics technologies like transcriptomics, proteomics, and metabolomics, respectively. However, intracellular fluxes cannot be directly measured and thus have to be inferred from measurable data. The cost of experiments, needed skills, and laborious setup of experiments have restricted flux measurements to only a few laboratories. Bioinformatics plays an important role in profiling the fluxome of an organism and has made flux analysis available to the scientific community. Several bioinformatic tools have been developed for steady state analysis of flux, but there is a shortage of dynamic state flux tools. Dynamic state analysis of metabolic flux sheds more light on the transient behavior of metabolism and serves to predict metabolic behavior in unexplored conditions.

Results
A user-friendly software tool, DynaFlux, was developed to make flux estimation available to non-expert users. The tool was built using Wolfram Mathematica® 7.0 and
the Integrated Development Environment Wolfram Workbench™ 2.0. The features available in the tool include: 1) deriving metabolic reconstructions for the simulation from Pathway Tools; 2) automated building of the mathematical model of the metabolic network; 3) parameter estimation using hybrid-mutation random-restart hill climbing algorithm; 4) perturbation studies of enzyme activities; 5) enumeration of feasible routes between two metabolites 6) minimal enzyme set; 7) imputation of missing metabolite data and 8) visualization of the network.

**Conclusions**

DynaFlux aids in the estimation of fluxes through the metabolic network and analysis of the network at dynamic state. Estimation of flux at dynamic state helps in understanding the transient behavior of the network and facilitates prediction of metabolic phenotypes at unexamined genetic or environmental conditions. In order to detour the metabolite towards the product of interest, enumeration of feasible routes along with the fluxes assists in decision-making at critical branch points. Perturbation studies help achieve the goals of metabolic engineering by knocking out or amplifying the enzymes to increase the production of the useful metabolite. Finally, imputation of missing metabolite data circumvents the need to conduct another tedious experiment.

**Background**

**Metabolic engineering**

Metabolic engineering is a rational approach to improve organisms using recombinant DNA technology to restructure metabolic networks for redirecting biochemical flux in order to enhance metabolic capabilities. Prior to metabolic
engineering, classical crossover of different strains was the rule of thumb to improve the metabolic capabilities [1]. Even though classical crossovers resulted in better strains with improved capabilities, the process was slow and involved random mutagenesis [2]. Conversely, metabolic engineering employs targeted genetic modifications, which is considered to be a more rational technique for strain improvement.

The field of metabolic engineering has grown to be highly multidisciplinary in nature, and derives support, tools, ideas, and methods from biochemistry, genetics, molecular biology, cell physiology, chemistry, chemical engineering, systems analysis, computer simulation, and bioinformatics [2, 3]. Metabolic engineering is an iterative process that consists of two steps: analysis and synthesis [4, 5]. Genetic modifications are done in the synthesis step to produce the desired strain with enhanced metabolic qualities using molecular biological techniques like genetic engineering and recombinant DNA technology. Analysis is the step wherein the metabolic capabilities present in the organism are characterized using analytical techniques [4, 6, 7].

To achieve the goals of metabolic engineering, there is a need to map the genotype of an organism to the metabolic phenotype. As a first step towards phenotypic mapping, the metabolic potential present in an organism has to be determined. But the biochemical characterization of metabolic networks and capabilities present in an organism lags as compared to the rate at which genomes are being sequenced [8]. Current workhorse technologies have resulted in a rapid increase in the number of completely sequenced genomes. As of 2010, there are about 1500 completely sequenced genomes including those of eukaryotes, bacteria, and archaea present in the Kyoto Encyclopedia of
Genes and Genomes (KEGG) database [9]. Bioinformatics has stepped in to bridge the gap between genome sequencing and characterization of the metabolic potential of an organism. Several bioinformatics tools such as Pathway Tools [10], KEGG [11], Pathway Analyst [8, 12], BioMiner [13], and What Is There [14] aid in the reconstruction of metabolic networks present in the organism using the annotated genome.

In conjunction with the reconstructed metabolic network, four important variables, namely, mRNA transcripts, proteins, small metabolites, and metabolic flux contribute to the complete description of the metabolic potential present in an organism [7]. Metabolic engineering serves as a framework to integrate these four cellular variables [7, 15]. High-throughput omics technologies like transcriptomics, proteomics, and metabolomics have enabled the systemic measurements of mRNA transcripts, proteins, and small metabolites, respectively. However, an organism’s fluxome cannot be easily or directly measured because flux involves a dimension of time [16]. Due to the required expertise, time-consuming and expensive experimental setup, metabolic flux analysis has been restricted to only a few expert groups [17]. Consequently, the fluxome of the organism has to be inferred from other measurable quantities. Bioinformatics plays a vital role in estimating intracellular flux and also makes flux analysis accessible to non-expert groups. As a result, bioinformatics has contributed to the missing piece of the puzzle to complete metabolic analysis.

Metabolic flux can be analyzed at steady state or dynamic state. At steady state, the metabolite pools are considered to be unchanging [18], and thus the kinetic parameters and reaction mechanisms can be ignored in the model. The various analyses
of a metabolic network at steady state employ stoichiometric modeling as the basis [19]. Several computational tools for steady state flux analysis are available for educational and commercial purposes. DBSolve [20], FluxAnalyzer [21], YANA [22], Expa [23], SNA [24], Metatool [25, 26], MetaFluxNet [27, 28], and CycSim [29] are tools designed for steady state analysis of flux. Steady state analysis utilizes the topology of the network, and reveals network information such as extreme pathways, flux distribution, optimal flux distribution, and elementary modes present in the network under study; but the steady state assumption provides only a static snapshot of the metabolic potential of the organism.

In the post-genomic era, one main challenge is to understand the dynamic metabolic behavior of an organism and consequently be able to predict the behavior in unexplored genetic or environmental conditions. For this purpose, the kinetic and mechanistic information has to be incorporated into the metabolic model [30]. The software tools that, to date, have integrated the kinetic parameters and simulated the dynamic behavior of an organism are GEPASI [31], MIST [32], and SCAMP [33]. Owing to the network complexity, shortage of computational resources, and lack of technical tools for model development, not much work has been done in the area of metabolic analysis. In the past, dynamic models have been developed to study tricarboxylic acid cycle in Dictyoselium discoideum [34], threonine synthesis pathway in Escherichia coli [35], glycolysis in Trypanosoma brucei [36], and the human red blood cell metabolic network [37]. The major drawback is that these models are restricted to only the problem domain for which the model was constructed.
**Pathway analysis**

In order to apply rational and targeted modifications to the metabolic network of an organism, metabolic engineering requires the analysis of pathway topology [38]. Consequently, once the metabolic network has been successfully reconstructed and kinetic parameters have been estimated, few analyses help in understanding the metabolic potential present in the organism and aid in predicting transient metabolic behavior. These network analyses include but are not limited to 1) enumeration of the most probable paths from a source metabolite to desired product, 2) knock-out and amplification of enzymes, and 3) computation of the minimal enzyme set and the dispensable set.

**Most probable path**

A metabolic pathway is defined as a complex series of consecutive, controlled, and enzyme-catalyzed biochemical transformations. One of the goals of metabolic engineering is to improve the production of chemicals already present in an organism [2-5]. This is achieved by redirecting metabolite fluxes towards a desired product when there are two or more enzymes competing for the same metabolite, and accelerating or avoiding rate limiting reactions. To redirect the fluxes, the first step is to enumerate all feasible paths between a source metabolite and desired product metabolite. At steady state, the most probable paths and alternative routes through the network can be determined by calculating the elementary flux modes [38, 39]. Several studies at steady state have applied convex analysis to the stoichiometric matrix to calculate elementary flux modes [38] or basic reaction modes [40]. Flux modes are possible direct routes between two metabolites. The flux modes are termed elementary when the mode consists
of a minimal set of reactions and are nondecomposable. Since this study involves
dynamic state analysis and does not employ stoichiometric modeling, a graph theory
approach is used to calculate the most probable paths between a source metabolite and
desired product.

**Minimal and dispensable enzyme set**

Alteration of a pathway for the production of desired product must involve
identification of the enzymes involved in the biochemical transformation. The
identification of the most probable routes between two metabolites also reveals the
enzyme set that is essential for the functioning of the pathway in order to actively
produce the desired metabolite. The minimal enzyme set is composed of those enzymes
that are almost certainly essential for the production of the metabolite of interest, whereas
the set of enzymes that can be deleted simultaneously without affecting the production of
the desired metabolite is known as the dispensable set [40].

**Knock-out, attenuation or amplification of enzyme activity**

The goals of metabolic engineering can be accomplished by the following two
approaches [38]:

1. Enzymes can be blocked completely or removed by knocking out the
   corresponding genes,

2. Enzyme activities can be gradually changed.

The first approach is termed as knock-out and the second is amplification of enzyme
activity.
**Imputation of missing data**

Handling missing or incomplete experimental data is a challenge that scientists often face. Missing data poses a concern to the scientific community because inference from a real dataset containing missing values may not be accurate. On the other hand, setting up and running an experiment is also very time consuming and expensive. *Imputation* is an approach used to fill in missing data with estimated values in order to produce a complete dataset. Imputation of missing data is an important topic in machine learning and data mining [41]. The area of imputing missing data has received considerable focus in the recent past in statistics as well and several different strategies that have been developed to deal with this issue [42]. In this study, a simulation-optimization method is used for imputation of missing data.

**Network visualization**

Visual information conveys the message more aptly than tabulated data, especially in this case of analysis of a metabolic network. Manually drawing and visualizing a metabolic pathway is the best way to depict a complex pathway because humans may be able to produce better visualizations than automated layout algorithms. However, the problem with this approach is that drawing complex pathways is time-consuming, and modifications to the pathway, which is often the case, cannot be incorporated automatically. Metabolic network visualization is by itself an interesting area of research, and different graph layout algorithms have been devised to draw the pathway topologies [43, 44].
Thus, given the challenge of understanding the dynamic behavior of a metabolic network and a need for a software tool to enable the analysis of metabolism of an organism at dynamic state, this research focussed on developing a software tool to meet these needs. DynaFlux is a user-friendly software tool that is capable of performing dynamic analyses of metabolic networks. This tool integrates flux profiling and network analyses based on estimated network flux. The analyses include estimation of kinetic parameters, imputing missing experimental data, enumeration of the most probable paths from a source metabolite to desired product, knock-out and amplification of enzymes, computation of the minimal enzyme set, and the dispensable set. The graphical user interface also provides a visualization of the metabolic network and the analyses.

**Implementation**

DynaFlux was developed using Wolfram Mathematica® 7.0 and the Integrated Development Environment Wolfram Workbench™ 2.0. The architecture of DynaFlux consists of five functional modules as shown in Figure 18. The modules are

1. Network input module
2. Automated differential equations generator module
3. Parameter estimation module
4. Network analysis module
5. Visualization/output module

The modules are accessible via a user-friendly graphical user interface (Figure 19).
**Network input module**

Metabolic pathway information and time course experimental data are two inputs for the tool that are essential to build the metabolic network for simulation, flux estimation, and network analysis. Major goals while developing the tool were that non-expert users should be able to use the tool with ease and that there should be minimum user interaction required for flux estimation and network analysis. In order to achieve these goals, the tool draws pathway information from an external software tool to help the user choose the pathways of interest, instead of forcing the user to define the pathway along with all the reactions manually that make up the pathway.

The metabolic pathway information of an organism is stored in the DynaFlux Database (DFDB) of the organism and consists of three flat files derived from Pathway Tools Pathway Genome Database (PGDB) of that particular organism [45]. The data needed for DFDB creation is exported from the desktop version of Pathway Tools software. The three DFDBs are

1. Pathway database
2. Reaction database
3. Enzymatic reaction database

The user is able to choose an organism from a list of organisms for which the DFDBs are available. Once the organism has been selected, the user is presented with the list of metabolic pathways present in the organism. The user can then select one or more pathways of interest (Chapter III). The DynaFlux software extracts the needed information from the three databases to reassemble the reactions along with the respective enzymes that make up the metabolic pathways for the purposes of estimating
flux through the pathways and other analyses of the network. DynaFlux is not organism-specific; rather the software allows the user to study the metabolism of any organism of interest, the only constraint is that the organism’s PGDB must be available in Pathway Tools. Even if the PGDB is unavailable, the user can create one by using the PathoLogic component of Pathway Tools if the annotated genome of the organism is available [10, 45, 46].

The second input to the tool is the time course data. When the pathways of interest have been selected by the user, DynaFlux identifies and lists the metabolites and enzymes present in the pathway(s) of interest in the order that they appear in the reaction set. The user is required to input the file containing the values corresponding to the concentrations of the metabolites and enzymes in the order provided in a tab-delimited format, with the columns being the different time points at which the measurements were taken, and the rows should corresponding to the reactants and enzymes. Here, ‘value’ can stand for concentrations, peak areas detected using a mass spectrometer, or any other units of data measured over time. This gives the user the flexibility of providing time course data measured using any technology.

**Automated differential equations generator module**

To accomplish the task of simulating a biochemical system on a computer, the system has to be transformed into a mathematical representation. Considering the facts that a metabolic system is non-linear in nature and the system is at dynamic state, the metabolic network is transformed into a set of coupled ordinary differential equations representing the pathway(s). The mathematical representation of the network is derived
by parsing the reactions individually in a linear sequence and generating the corresponding coupled differential equations [47]. There are four basic rules for generating the coupled differential equations for a reaction (Chapter IV). The automated differential equations generator (ADEC) uses those four rules to process the metabolic network and results in the complete set of coupled differential equations without any user intervention. A simulation of the biochemical system involves 1) transforming the system into a mathematical representation of the network, and 2) solving the mathematical representation. The first step is achieved by ADEC, and the second step is completed by the next module.

**Parameter estimation module**

An inverse problem is one in which the effects of the system are given and the causes that resulted in those effects are to be estimated. In this case, the time course data (effects) is given, and the kinetic parameters (causes), that yielded the time course data, are to be estimated and thus this problem falls into the category of Inverse Problems. The differential equations generated by parsing each reaction in the ADEC module is solved as a set of coupled differential equations using the NDSolve method in Mathematica (version 6.0 or above). NDSolve takes the following form:

\[
\text{NDSolve}[\text{eqns}, \text{vars}, \{t, \text{start}, \text{end}\}];
\]

where \(\text{eqns}\) is the set of coupled differential equations, \(\text{vars}\) is the set of all compounds involved in the pathway, \(t\) represents the time variable, \(\text{start}\) and \(\text{end}\) are the starting and ending time.
Network analysis module
Imputation of missing data

Imputation of missing data is done in two steps: estimation using slope method followed by estimation using simulation-optimization. As a first step, the “slope method” is used to find an initial estimate of the missing value of a metabolite at a given time point. The slope method was introduced in this study and uses the formula for calculating the slope of a straight line. Analysis of metabolite time course data indicated that a metabolite \( A \) produced or consumed by metabolite \( B \) usually follows a trend similar to metabolite \( B \) (Chapter IV), i.e., the rate of change of concentrations over time for the metabolites \( A \) and \( B \) were approximately the same; thus, the lines of the plots of concentrations of the metabolites were parallel. Mathematically, two parallel lines have the same slope. Applying this fact, the slope of the metabolite with measured concentration values was used to estimate the missing concentration of the other metabolite. The metabolite used for calculating the slope is called the neighbor to the missing metabolite. In situations wherein the metabolite is a branch point metabolite and as a result there are several neighbors available for the missing metabolite, a neighbor that is nearest to the missing metabolite is chosen based on the following criterion

\[
NN = \arg\min_{m_t \in \text{Neighbor}} \sum_{t=1}^{N} \text{Abs}[m_t(t) - c(t)]
\]  

where \( NN \) is the nearest neighbor, and \( c \) is the metabolite that has a missing value. The slope of the neighbor with measured concentrations at two time points is given by Equation 2
where $B_{t_1}$ and $B_{t_2}$ are the concentrations of metabolite $B$ at time points $t_1$ and $t_2$, respectively. Using the slope of metabolite $B$, the missing concentration of metabolite $A$ can be found using Equation 3

$$A_{t_1} = A_{t_2} - \text{slope} \cdot (t_2 - t_1)$$

With $A_{t_1}$ as the initial estimate, a simulation-optimization routine in the PEM is then used to determine the final estimate. The algorithm for imputing the final estimate is

1. Run the simulation-optimization routine. Determine the set of best-evaluated model parameters and $c$, the corresponding concentration of the missing metabolite.

2. If $\text{Absolute}[A_{t_1} - c] < 1.0$ and $\text{Absolute}[A_{t_1} - c_{\text{average}}] < 1.0$, then accept $c$ as one of the feasible final estimate. Here $c_{\text{average}}$ is the average of the feasible final estimates found thus far.

3. Repeat steps 1 and 2 for a few times.

4. Report $c_{\text{average}}$ as the final estimate of the missing value of the metabolite.

5. Repeat steps 1 through 4 for the next missing metabolite, if any.

**Perturbation**

This network analysis involves determining enzyme activity by perturbation and predicting the corresponding changes in the rate of a reaction. Perturbation study involves amplifying the levels of the enzyme or knocking out the enzyme completely. This allows analyzing the activities of the enzyme, changes in the estimates of model parameters, and the concentrations of the metabolites involved in the reaction catalyzed by the enzyme at
different time points and the systemic changes. PEM’s simulation-optimization routine is used, and the *overall-best-evaluated* model parameters are estimated (Chapter V). Using the *overall-best-evaluated* model parameters and the amplified or knocked out enzyme concentration, a new set of data is generated for the network reactants and enzymes. PEM’s simulation-optimization routine is again employed and the *new overall-best-evaluated* model parameters are estimated for the perturbed biochemical system.

The kinetic parameters $K_R$, $K_{dy}$, $V_{maxf}$, $V_{maxr}$ and $v$ are calculated for the normal and the perturbed system. The definitions for $K_R$, $K_{dy}$, $V_{maxf}$ and $V_{maxr}$ can be found in (Chapter 4). Consider the uni-uni reaction in Equation 4

$$
\begin{align*}
    & k_{12} \\
    & E + A \rightleftharpoons EA \\
    & k_{21} \\
    & k_{23} \\
    & EA \rightleftharpoons P + E \\
    & k_{32}
\end{align*}
$$

The net rate of product formation for this reaction is obtained by subtracting the rate at which the product is consumed in the reaction from the rate at which the product is released in the reaction and is given by

$$
v = k_{23}[EA] - k_{32}([E] - [EA])[P]
$$

where $[EA]$, $[E]$, and $[P]$ are the concentrations of the complex, enzyme, and product [48].
**Most probable path**

The metabolic network is transformed into a graph with nodes representing the metabolites and the edges symbolizing the reactions of the network. All feasible routes between a source metabolite and a desired product are found using a *breadth-first search*. Breadth-first search is a graph-search algorithm that is a simple strategy in which the search begins at the root node, i.e., the source metabolite and expands this node, then all the neighboring nodes are expanded, next their neighbors, and so on until the desired product is found [49]. Once all the feasible paths are found by BFS, the paths are tested for validity. All the valid feasible paths are then ranked using the overall rate of product formation while taking the route to get to the desired product.

The fastest route from the source metabolite to the desired product is considered to be the best route. The enzymes that catalyze the reactions in this route are called the minimal enzyme set. The rest of the enzymes in the other feasible routes are termed as the dispensable enzyme set. At a branch point, an enzyme from the dispensable enzyme set can be knocked out resulting in routing the metabolite through the other faster enzyme, thus yielding more of the desired product.

**Visualization/output module**

This module displays the output of PEM and the other network analyses that can be done in DynaFlux. The kinetic parameters $K_R$, $K_{dy}$, $V_{maxf}$ and $V_{maxr}$ are output in a tabular form that can be saved into a text file or exported to a spreadsheet application for further analysis. The graph representation of the metabolic network is constructed using the GraphPlot function available in Mathematica (6.0 or above). The syntax for GraphPlot is
GraphPlot[{{vi1 -> vj1, vi2 -> vj2, ...}}];

where vertex \( v_i \) is connected to vertex \( v_j \). The rate of a reaction \( v \) is overlaid on the graph representation of the network for better comprehension of the rates.

**Results and Discussion**

One of the most important goals considered during the development of DynaFlux was to minimize user intervention. Several softwares for flux analysis at dynamic state like GEPASI [31], MIST [32], and SCAMP [33] require the user to define every reaction present in all the pathways in the metabolic network. Even softwares that deal with steady state analysis, such as FluxAnalyzer [21], Metatool [25, 26] and SNA [24], expect the users to define the reactions. On the contrary, DynaFlux pools in pathway information from Pathway Tools [10]. In this way, the user can pick the pathways of interest from the list provided and is not required to describe every reaction involved in the pathways present in the metabolic network. When the network’s complexity increases, the time taken to define the individual reactions also increases, but DynaFlux circumvents this drawback.

There are a few dynamic models that were used to study tricarboxylic acid cycle in *Dictyoselium discoideum* [34], threonine synthesis pathway in *Escherichia coli* [35], glycolysis in *Trypanosoma brucei* [36], and human red blood cell metabolic network [37]. These models were developed for the specific problem domain. But DynaFlux is not organism-specific, rather the software can be used for any organism of interest that has metabolic reconstructions, from the genome sequence, available in Pathway Tools software.
In this study, a time course dataset of the food-borne pathogen Salmonella typhimurium in normal growth conditions was used. The dataset was comprised of metabolomics and genomics data of three basal metabolic pathways, namely, aspartate biosynthesis, the tricarboxylic acid cycle, and the glyoxylate cycle. The three pathways shown in Figure 20 were constructed using the visualization module of DynaFlux. The metabolites and gene expression data were measured at 30 minutes, 60 minutes, and 120 minutes.

Flux analysis at dynamic state is the main feature of DynaFlux. The ADEG module of DynaFlux generates the mathematical model of the network represented by the set of coupled differential equations based on the pathways selected, without needing the involvement of the user. The PEM module uses a hybrid-mutation random-restart hill climbing algorithm to achieve the best fit between the experimental data and the model data (Chapter 4). Figure 21 shows the pathways overlaid with the estimated flux through the reactions.

Apart from flux estimation, DynaFlux can be used to impute missing data in experimental time course measurements and pathway analyses like enumeration of most probable paths between source and product, knock-out and amplification of enzymes, and computation of the minimal enzyme set and the dispensable set. In addition, DynaFlux provides network visualization for easy comprehension of the network under study.

Missing data is a common problem in real datasets. There are several methods that have been devised to handle this problem [42]. These methods are used to impute missing data, especially non response data in all types of medical research and social sciences. Recently, machine learning and data mining have been used to deal with
missing data [41]. All these approaches focus only on non response data from a sample population. But in this research, the missing data is from experimental measurements using high throughput technology and not from subjects of a sample population. So, the statistical approaches cannot be applied in this scenario. Thus, imputation of missing data is done using a novel approach that was introduced in this research.

The dataset used in this study was complete without any missing data. For the purposes of validating the imputation of missing data approach introduced in this study, the data points were forced to be missing. The approach used for imputation of missing data consists of two steps: 1) the slope method, and 2) the simulation method.

Table 12 lists the imputation of isocitrate at the third time point (120 minutes). The log₂ of the peak area of isocitrate at time point 3 was 12.4 units in the original dataset. The neighboring metabolites for isocitrate in the pathways under consideration are aconitate, glyoxylate, and ketoglutarate. Using the nearest neighbor approach introduced in this study, aconitate was found to be the nearest neighbor. Next, using aconitate as the neighboring metabolite, the slope method resulted in 12.4 units as an estimate for the missing value for isocitrate. Imputing the slope estimate at the missing spot in the dataset, PEM is invoked, and several estimates of the missing value are generated. The slope estimate remains the same for every simulation estimate. The simulation estimates are considered to be one set of data, and the slope estimates constitute the second set of data. Next, a paired t-test is applied to assess if the means of the two sets are statistically significantly different from each other. The p-value from the
t-test is 0.49, which is far greater than 0.05 and thus the two data sets are statistically the same. As a result, the final estimate of the missing value is given as 12.6 ± 0.1.

Table 13 lists four missing metabolites that were estimated using the slope-simulation approach. When the dataset contains two or more missing values, the slope method is first used to estimate all the missing values using the nearest neighbor approach. These estimates are then filled in at the missing places in the dataset. PEM is used repetitively to estimate each missing metabolite one at a time and replace the slope estimate. The final estimates are relatively close to the actual values listed in Table 13. The order in which the metabolites are estimated does not affect the final estimates.

The next goal was to find all feasible paths between a source metabolite and the desired product. Enumerating feasible or alternative routes between metabolites in a metabolic network has several applications in systems biology, for example in the design of knock-out experiments [50] and comparative analysis [51]. There are a number of well-established methods that exist to calculate the paths in a graph, but those cannot be applied to metabolic networks that are much more complicated than ordinary graphs. A few approaches apply shortest path algorithms to compute the metabolic routes using atomic mapping rules or structural information [52]. Some tools that use this approach are Pathway Hunter Tool [53] and Rahnuma [51]. At steady state, convex analysis has been applied to the stoichiometric matrix of the metabolic network to calculate the elementary flux modes. In this study, the network is at dynamic state represented as a graph. Consequently, a breadth-first search algorithm is used to enumerate all the feasible
paths between two metabolites. Furthermore, no other software tool calculates the shortest path or ranks the feasible paths using the fluxes present in the pathways.

Figure 22 shows the most probable paths between succinate and glutamate. There were three possible paths between the two metabolites and are listed in Table 14. The most probable paths found using BFS are highlighted in the metabolic network. The rank of the path and the total flux of the path are also recorded. The shortest path has only three transformations to the desired product and takes 13.16 flux units to produce glutamate. However, there are one or more individual reactions in all the three routes that have a negative flux (Figure 22). Negative values for flux indicate that the reaction was taking place in the reverse direction, i.e., the product was consumed to form the substrates of the reaction. Thus, a reaction with a negative flux in the pathway disrupts the flow of conversion, and the desired product is never achieved by taking that route.

Conversely in Figure 23a, there are three possible routes that exist between citrate and malate. None of the reactions in the three paths have a negative flux indicating that all the reactions are proceeding in the forward direction. The shortest path from citrate to malate is via aconitate and isocitrate (Table 15). This path has the minimum number of transformations and also the minimum value for the overall flux of the pathway, which is obtained by adding the individual fluxes. Consequently, the shortest path is the fastest path in going from citrate to malate. Figure 23b shows the fastest route in isolation from the pathways under study. The minimal enzyme set for the transformation of citrate to malate comprises aconitase B, isocitrate lyase, and malatesynthase G. All the other enzymes in the rest of the possible routes are considered to belong to the dispensable
enzyme set. The enzymes from the dispensable set can be knocked-out one-by-one to check if the metabolites in the fastest path can be routed efficiently towards the desired product.

Perturbation study involves knocking out an enzyme or amplifying the concentration of an enzyme. The enzyme aspartate transaminase was chosen for the perturbation study. Figure 24 shows the rates of the reactions overlaid on the metabolic network, which resulted from the perturbation study. The flux value listed below the enzyme’s name is the flux estimated using normal growth conditions. The next flux value is when aspartate transaminase was knocked out followed by the flux value wherein the enzyme concentration was amplified two-fold. The lower the flux value, the faster the reaction. From Figure 24, for aspartate transaminase, the flux was negative at normal growth conditions, indicating that the reaction was taking place in the reverse direction; thus, more oxaloacetate and glutamate were produced when compared to production of aspartate and ketoglutarate. The flux was nil as expected when the enzyme was knocked out. When the enzyme was amplified, the reaction was fast and proceeded in the forward direction. The perturbation study of aspartate transaminase not only affected the flux of this reaction but also changed the fluxes of other reaction. Thus, a change in one of the reactions has a systemic effect. Results from up to 5 perturbations can be overlaid on the same graphical representation for comparative studies.

The other important features of DynaFlux include integration of metabolomics and genomics data, mass balance of metabolites, the use of a benchmark metabolite to run the simulation, applying thermodynamic information to determine the direction of a
reaction to generate the set of coupled differential equations and visualization of
directionality in the network, and sending an email to the user in order to indicate the
completion of the simulation.

In summary, DynaFlux acts as a framework to integrate metabolic
reconstructions, genomic data, metabolic data, and thermodynamics, along with dynamic
state metabolic flux estimation and network analyses. This integration makes DynaFlux a
unique software for understanding metabolism at dynamic state and performing network
analyses that will aid in metabolic engineering.

Conclusions

DynaFlux is a user-friendly tool for conducting dynamic state metabolic flux
analysis. Due to the fact that flux analysis is done at dynamic state, the tool helps predict
the behavior of the metabolic system under unexplored genetic or environmental
conditions. The tool requires very less user interaction by mostly automating the
processes involved in flux estimations and network analyses. Thus, non-expert users can
use the tool without having to know the underlying mathematics employed. Flux analysis
was time-consuming and was mostly restricted to only a few specialized groups, but
DynaFlux makes flux analysis available to any user with a need to study fluxes of a
metabolic network. The tool imports pathway information from Pathway Tools, so the
initial setup is also effortless.

The tool integrates metabolomics and genomics data for flux estimation.
Furthermore, the tool can be used for imputation of missing experimental data and a few
network analyses such as perturbation study, finding all feasible paths between a source
metabolite and desired product, and visualization of the network. The results of the perturbation study will guide the researcher to formulate possible hypotheses about substrate limitation, abiotic stress, metabolic engineering options, and the optimal method to modify the metabolic system to produce the desired product.

Finding all feasible paths between two metabolites will help in directing the fluxes at branch points towards the metabolite of interest.

DynaFlux is an innovative tool aiding flux estimation and network analysis in the post-genomic era, thereby bridging the gap between genome sequencing and metabolic analysis. This in turn will hasten the process of metabolic engineering that is aimed at improving the metabolic potential of organisms for agricultural and industrial purposes.

References


9. **KEGG Organisms: Complete Genomes**

   [http://www.genome.jp/kegg/catalog/org_list.html]


17. Zamboni N, Fischer E, Sauer U: **FiatFlux--A software for metabolic flux analysis from 13C-glucose experiments.** *BMC Bioinformatics* 2005, **6**.


29. Le Fevre F, Smidtas S, Combe C, Durot M, d'Alche-Buc F, Schachter V: 
   
   CycSim—An online tool for exploring and experimenting with genome-scale 

   reaction systems: Algebraic properties, validated calculation procedure and 
   example from nucleotide metabolism. Journal of Mathematical Biology 2002, 

31. Mendes P: Biochemistry by numbers: Simulation of biochemical pathways 


33. Sauro HM: SCAMP: A general-purpose simulator and metabolic control 

34. Wright B, Butler M, Albe K: Systems analysis of the tricarboxylic acid cycle in 
   Dictyostelium discoideum. I. The basis for model construction. Journal 

35. Chassagnole C, Fell DA, RaÁ’s B, Kudla B, Mazat JP: Control of the 
   threonine-synthesis pathway in Escherichia coli: A theoretical and 

36. Bakker BM, Michels PAM, Opperdoes FR, Westerhoff HV: Glycolysis in 
   bloodstream form Trypanosoma brucei can be understood in terms of the


43. Bourqui R, Cottret L, Lacroix V, Auber D, Mary P, Sagot M-F, Jourdan F: 


Table 12 - Imputation of missing value of isocitrate at the third time point
STD – standard deviation, COV – coefficient of variation, SEM – standard error of the mean.

<table>
<thead>
<tr>
<th>Run</th>
<th>Slope Estimate</th>
<th>Simulation Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12.4</td>
<td>12.8</td>
</tr>
<tr>
<td>2</td>
<td>12.4</td>
<td>12.7</td>
</tr>
<tr>
<td>3</td>
<td>12.4</td>
<td>13.3</td>
</tr>
<tr>
<td>4</td>
<td>12.4</td>
<td>12.5</td>
</tr>
<tr>
<td>5</td>
<td>12.4</td>
<td>11.7</td>
</tr>
<tr>
<td>6</td>
<td>12.4</td>
<td>12.9</td>
</tr>
<tr>
<td>7</td>
<td>12.4</td>
<td>11.5</td>
</tr>
<tr>
<td>8</td>
<td>12.4</td>
<td>12.9</td>
</tr>
</tbody>
</table>

Average 12.4 12.6
STD 0.0 0.6
COV 0.0 5.0
Count 8 8
SEM 0.0 0.1
T-test 0.49
Table 13 - Imputation of missing data

<table>
<thead>
<tr>
<th>No. of missing values</th>
<th>Metabolite</th>
<th>Timepoint</th>
<th>Actual Value</th>
<th>Slope Estimate</th>
<th>Simulation Estimate</th>
<th>Final Estimate</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Isocitrate</td>
<td>2</td>
<td>11.9</td>
<td>11.9</td>
<td>11.9</td>
<td>11.9±0.1</td>
<td>0.92</td>
</tr>
<tr>
<td>1</td>
<td>Isocitrate</td>
<td>3</td>
<td>12.4</td>
<td>12.4</td>
<td>12.6</td>
<td>12.6±0.1</td>
<td>0.50</td>
</tr>
<tr>
<td>2</td>
<td>Oxaloacetate</td>
<td>3</td>
<td>19.5</td>
<td>19.9</td>
<td>19.8</td>
<td>19.8±0.1</td>
<td>0.68</td>
</tr>
<tr>
<td>2</td>
<td>Glyoxylate</td>
<td>2</td>
<td>11.8</td>
<td>12.1</td>
<td>11.6</td>
<td>11.6±0.1</td>
<td>0.08</td>
</tr>
</tbody>
</table>
Table 14 - Feasible paths between succinate and glutamate

<table>
<thead>
<tr>
<th>Rank</th>
<th>Path</th>
<th>Flux</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SUC -&gt; THREODSISOCITRATE -&gt; KETOGLUTARATE -&gt; GLT</td>
<td>13.16</td>
</tr>
<tr>
<td>2</td>
<td>SUC -&gt; FUM -&gt; MAL -&gt; GLYOX -&gt; THREODSISOCITRATE -&gt; KETOGLUTARATE -&gt; GLT</td>
<td>506.51</td>
</tr>
<tr>
<td>3</td>
<td>SUC -&gt; FUM -&gt; MAL -&gt; OXALACETICACID -&gt; CIT -&gt; CISACONITATE -&gt; THREODSISOCITRATE -&gt; KETOGLUTARATE -&gt; GLT</td>
<td>2965.99</td>
</tr>
</tbody>
</table>
Table 15 - Feasible paths between citrate and malate

<table>
<thead>
<tr>
<th>Rank</th>
<th>Path</th>
<th>Flux</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CIT -&gt; CISAONITATE -&gt; THREODSISOCITRATE -&gt; GLYOX -&gt; MAL</td>
<td>584.4</td>
</tr>
<tr>
<td>2</td>
<td>CIT -&gt; CISAONITATE -&gt; THREODSISOCITRATE -&gt; SUC -&gt; FUM -&gt; MAL</td>
<td>1033.7</td>
</tr>
<tr>
<td>3</td>
<td>CIT -&gt; CISAONITATE -&gt; THREODSISOCITRATE -&gt; KETOGLUTARATE -&gt; SUCCOA -&gt; SUC -&gt; FUM -&gt; MAL</td>
<td>1280.5</td>
</tr>
</tbody>
</table>
Figure 18 - DynaFlux architecture
Figure 19 - User interface of DynaFlux
Figure 20 - Three pathways: Aspartate biosynthesis, TCA cycle and glyoxylate cycle
Figure 21 - Metabolic flux estimates overlaid on the network
Figure 22 - Feasible routes between succinate and glutamate
Figure 23 - Feasible routes between citrate and malate
a) Metabolic network with the three routes highlighted; b) the fastest route isolated from the network.
Figure 24 - Metabolic network overlaid with fluxes from normal, knocked-out and amplified aspartate transaminase
CHAPTER VI
CONCLUSION

Introduction
Metabolism is one of the most complex cellular processes by which an organism obtains energy from intaken food in order to perform activities so as to maintain a living state [1]. The process of metabolism consists of two phases or states: 1) dynamic state and 2) steady state. Metabolism can be analyzed at either the steady state or the dynamic state of metabolism. Steady state analysis provides only a static snapshot of the metabolic capabilities because the analysis includes only the stoichiometry of the network and few constraints [2], and excludes kinetic properties and regulatory information in order to reduce the complexity thus limiting the usefulness. On the contrary, dynamic analysis sheds light on the transient behavior of metabolism by including kinetic and regulatory information.

Analysis of metabolism aids in achieving the goals of metabolic engineering. However, there is a lag between the two phases of metabolic engineering, namely synthesis and analysis, because tools for molecular biology are available to bring about the genetic changes, but analytical techniques to identify the optimal genetic modifications are still naïve [3]. In addition, the cost of experiments, needed skills, and laborious setup of experiments has restricted flux measurements to only a few laboratories [4]. Bioinformatics helps to bridge this gap through modeling and simulation. At present, there are several mathematical methods and computational tools that describe network flux at steady state (Chapter II). Owing to the limitations of steady
state analysis, there has been an increasing demand to examine metabolism during the dynamic state [3, 5-7] because of the predictive capabilities at unexplored conditions available through such models. This research focused on dynamic state analysis of metabolism.

**Research Objectives**

To accomplish the goals of this research, the first step was to identify the metabolites present in the metabolic pathways that were actively participating in the metabolism of an organism at the given condition. Metabolome Searcher is a web-based application for metabolite identification and pathway mapping. The tool addresses the challenge of metabolic analysis using mass spectrometry data by not only identifying the metabolites present in the organism but also maps the metabolites to the associated metabolic pathways. This was achieved by searching publicly available genome-constructed metabolic databases (Chapter III). The searches can be further refined by restricting the queries to be specific to the experiment’s MS analysis conditions. In addition, the tool aids in visualizing the active pathways by redirecting the user to ProCyc [8] and KEGG [9].

Large compound databases such as Pubchem [10] and Chemspider [11] allow searches of single mass and other query types, but they do not allow queries from large lists of masses or connect putative compounds to metabolic pathways. As the query list expands, as it does in metabolome data sets, data analysis using single queries becomes unrealistic and tedious. Additionally, querying compound databases that contain millions of non-biological molecules can impede a researcher’s ability to overlay a metabolic
context onto metabolomic data. Metabolome Searcher is a novel tool that eliminates these limitations. The versatility of accepted query types, the provision of pathways mapped to queries, and providing direct links to visualize the pathways are unique to the Metabolome Searcher.

Once the researcher identifies the pathways of interest by using Metabolome Searcher, the next step is to analyze metabolic flux. A few tools like GEPASI, MIST and SCAMP have been developed to analyze dynamic state metabolic flux [6, 12, 13]. The limitations of these tools are that the structure, kinetics, and kinetic parameters are required as input from the users. The mechanistic details and kinetic parameters of a metabolic pathway are mostly unknown [14-16]. DynaFlux overcomes these limitations and enables the analysis of metabolic flux by estimating the kinetic parameters that resulted in the time course metabolomics and genomics data. The metabolic pathways are mathematically represented as a set of coupled differential equations and solved. Hybrid-mutation random-restart hill-climbing, a novel variant of hill-climbing, is introduced and used for flux estimation (Chapter 4). Recent advancements in computational capability, such as high-performance computing [17] and the availability of several technical computing softwares, have made genome-scale dynamic state metabolic modeling feasible. The estimation of kinetic parameters was validated by applying proof of concept.

Having solved the inverse problem of estimating kinetic parameters, the metabolic network is ready for perturbation studies. Dynamic state metabolic flux analysis not only portrays the transient behavior of the network but also serves as a guide
to predict alterations in flux during unexplored conditions and the resulting metabolic phenotype. Perturbations studies included knock-out and attenuation of enzymes. This study shows that a modification in one of the network enzymes resulted in a local effect that rippled through the entire network to create a systemic effect. Imputation of missing data was an important problem that DynaFlux handles and was validated using a statistical approach. Also, all feasible paths between two metabolites are enumerated, which in turn helps in directing the fluxes at branch points towards the metabolite of interest.

DynaFlux is a one-of-a-kind tool that combines estimation of flux and analysis of the metabolic network. DynaFlux is a user-friendly software tool that was developed for the scientific community interested in metabolic flux analysis, and was built using Wolfram Mathematica® 7.0 and the Integrated Development Environment Wolfram Workbench™ 2.0. The features available in the tool include: 1) deriving metabolic reconstructions for the simulation from Pathway Tools; 2) automated building of the mathematical model of the metabolic network; 3) parameter estimation using hybrid-mutation random-restart hill climbing algorithm; 4) perturbation studies of enzyme activities; 5) enumeration of feasible routes between two metabolites 6) minimal enzyme set and 7) visualization of the network.

**Recommendations**

**Automated import of data from pathway tools**

A script to communicate with Pathway Tools [18] via the PerlCyc module [19] to create the DynaFlux Database of an organism would be a useful addition to DynaFlux. Currently, the information from Pathway Tools Pathway/Genome Databases was
manually exported and then automatically pooled within DynaFlux to create the organism’s DynaFlux Database.

**Modification of metabolic pathways**

Allowing the user to manually edit the imported pathways, such as insertion or deletion of metabolic reactions and metabolites, would provide more flexibility in flux analysis.

**Optimization algorithms**

The present study analyzed four different variants of hill-climbing search. Hybrid-mutation random-restart hill-climbing search proved to be the best among the four variants. Other optimization algorithms like simulated annealing and genetic algorithms could be employed and compared with the current approach. The best algorithm could then be adopted for estimating the kinetic parameters.

**Representation format of the computational model**

The model can now be backed-up, saved or retrieved as a flat file. The Systems Biology Markup Language (SBML) is an XML-based format for representing biochemical networks [20]. Implementing SBML would enable the model to be portable between different simulation and analysis tools for performing different kinds of metabolic analysis.
Integration of transcriptional regulatory network

Integration of the transcriptional regulatory network into the metabolic network modeling would increase the accuracy of the model, and thereby imitate the biology more precisely.

Different kinetic mechanisms

Incorporation of different kinetic mechanisms would provide more flexibility to the user who knows the underlying mechanism. If the mechanism were unknown, the current mechanism, namely sequential ordered, would be the default.

Reverse mapping of metabolic phenotype to genotype

The ultimate end for this research would be to reverse map the desired metabolic phenotype to the genotype of the organism to achieve the goals of metabolic engineering.

References


APPENDIX A
SUPPLEMENTARY DATA FOR CHAPTER IV

Proof for Ranjitha’s Shortcut Method for kinetic parameter $K_R$, $V_{maxf}$ and $V_{maxr}$

Based on the definitions of $K_m$ for reactants of uni-uni and bi-bi reaction [1], a generalized formula was determined using Ranjitha’s Shortcut Method, and was used to quantify enzyme affinity and rate of product formation. The definition for the kinetic parameter $K_R$ is

$$K_R = \frac{k_{DFP} \cdot k_{DLC}}{k_{AR} \cdot k_{DP}}$$  \hspace{1cm} (1)

where $k_{DFP}$ = forward rate constant representing dissociation of final product from complex

$k_{DLC}$ = Sum of the forward and reverse rate constants representing dissociation of largest enzyme-substrate complex. Special case: For the first substrate in a reaction with more than one substrate, only the forward rate constant representing dissociation of product from largest complex

$k_{AR}$ = Forward rate constant representing association of this reactant with enzyme to form complex

$k_{DP}$ = Sum of the forward rate constants representing dissociation of product(s) from respective complex(es)

$K_R$ was proved using the concept presented in Chapter 9 Section A of [2], which describes the King-Altman method of deriving steady-state velocity equations. The formula in Equation 1 can be applied to deduce the kinetic constant $K_R$ of a reaction
consisting of any number of substrates and products. In addition, Equation 1 is valid for reactions at dynamic state and steady state. When at steady state, $K_R$ turns out to be the Michelis constant $K_m$ of the reaction.

**Bi-Bi reaction**

\[
\begin{align*}
E + A & \rightleftharpoons EA \\
& \quad (k_1) \\
EA + B & \rightleftharpoons EAB \\
& \quad (k_2) \\
EAB & \rightleftharpoons P + EQ \\
& \quad (k_3) \\
EQ & \rightleftharpoons Q + El \\
& \quad (k_4)
\end{align*}
\]

(2)

The method in [2] was applied to a bi-bi reaction (Equation 2), and the velocity equations were obtained using the five steps as follows:

1. Figure 25 shows the geometric representation of a bi-bi reaction
2. Figure 26 shows all possible patterns containing one less line than basic figure shown in Figure 25.

Using Equation IX-1 of [2],

$n = 4$ and $m = 4$ for a bi-bi reaction

Number of patterns with (n-1) lines $= \frac{m!}{(n-1)!(m-n+1)!}$
\[ \begin{align*}
\therefore \text{Number of patterns with 3 lines} &= \frac{4!}{(4-1)!(4-4+1)!} \\
&= 4 \quad \text{(as shown in Figure 26)}
\end{align*} \]

3. Applying step 3,

\[ \begin{align*}
\frac{[E]}{[E_1]} &= \frac{k_{-1} k_3 k_4 + k_4 k_{-2} k_{-1} + k_2[B] k_3 k_4 + k_{-3}[P] k_{-2} k_{-1}}{\text{denominator}} \\
\frac{[EA]}{[E_1]} &= \frac{(k_3 k_4 k_1[A] + k_{-2} k_4 k_1[A] + k_{-4}[Q] k_{-3}[P] k_{-2} + k_1[A] k_{-3}[P] k_{-2})}{\text{denominator}} \\
\frac{[EAB]}{[E_1]} &= \frac{(k_{-1} k_{-4}[Q] k_{-3}[P] + k_4 k_1[A] k_2[B] + k_2[B] k_{-4}[Q] k_{-3}[P] + k_1[A] k_2[B] k_{-3}[P])}{\text{denominator}} \\
\frac{[EQ]}{[E_1]} &= \frac{(k_{-1} k_{-4}[Q] + k_{-2} k_{-1} k_{-4}[Q] + k_{-4}[Q] k_2[B] k_3 + k_1[A] k_2[B] k_3)}{\text{denominator}}
\end{align*} \]

\[ \text{denominator} = k_{-1} k_3 k_4 + k_4 k_{-2} k_{-1} + k_2 k_3 k_4[B] + k_{-3} k_{-2} k_{-1}[P] + (k_3 k_4 k_1 + k_{-2} k_4 k_1)[A] + (k_{-4} k_{-3} k_{-2} k_{-1}[P][Q] + k_1 k_{-3} k_{-2}[A][P] + (k_4 k_1 k_2 + k_4 k_1 k_2)[A][B] + k_2 k_{-4} k_{-3}[B][P][Q] + k_1 k_2 k_{-3}[A][B][P] + (k_{-1} k_{-4} + k_{-2} k_{-1} k_{-4})[Q] + k_{-4} k_{-2} k_3[B][Q] \]

4. Thus,

\[ \begin{align*}
\text{constant} &= k_{-1} k_3 k_4 + k_4 k_{-2} k_{-1} \\
\text{coeff}_{AB} &= k_1 k_2 (k_3 + k_4) \\
\text{coeff}_{PQ} &= k_{-3} k_{-4} (k_{-1} + k_{-2}) \\
\text{coeff}_A &= k_1 k_4 (k_3 + k_{-2}) \\
\text{coeff}_B &= k_2 k_3 k_4 \\
\text{coeff}_P &= k_{-1} k_{-2} k_{-3} \\
\text{coeff}_Q &= k_{-1} k_{-4} (k_3 + k_{-2})
\end{align*} \]
5. Referring to Section C in Chapter 9 of [2], the Michaelis constants are

\[
K_mA = \frac{k_3 k_4}{k_1 (k_3 + k_4)}
\]

\[
K_mB = \frac{k_4 (k_3 + k_{-2})}{k_2 (k_3 + k_4)}
\]

\[
K_mP = \frac{k_{-1} (k_3 + k_{-2})}{k_{-3} (k_{-1} + k_{-2})}
\]

\[
K_mQ = \frac{k_{-1} k_{-2}}{k_{-4} (k_{-1} + k_{-2})}
\]

The formulae from step 5 can be obtained using Ranjitha’s Shortcut Method for $K_R$ at dynamic state by skipping the five derivation steps and can also be verified with [1] for a ternary-complex mechanism.

Also,

\[
v = k_4[EQ] - k_{-4}[Q][E]
\]

\[
\frac{v}{[E_i]} = \frac{k_1 k_2 k_3 k_4[A][B] - k_{-1} k_{-2} k_{-3} k_{-4}[P][Q]}{\text{denominator}}
\]

\[
um1 = k_1 k_2 k_3 k_4[E_i]
\]

\[
um2 = k_{-1} k_{-2} k_{-3} k_{-4}[E_i]
\]

\[
V_{maxf} = \frac{\num1}{\text{coeff}_{AB}} = \frac{k_3 k_4[E_i]}{(k_3 + k_4)}
\]

\[
V_{maxr} = \frac{\num2}{\text{coeff}_{PQ}} = \frac{k_{-1} k_{-2}[E_i]}{(k_{-1} + k_{-2})}
\]
Likewise, the formulae for $V_{\text{max}f}$ and $V_{\text{max}r}$ can be obtained using Ranjitha’s Shortcut Method for $V_{\text{max}f}$ and $V_{\text{max}r}$ at dynamic state without having to go through a tedious derivation method and can be also verified with [1] for a ternary-complex mechanism.

Ranjitha’s Shortcut Method for kinetic parameters for a uni-uni, uni-bi and ter-ter reaction were also proved in a similar manner.
References


Figure 25 - Geometric representation of a bi-bi reaction
Figure 26 - Patterns containing one less line than basic figure
APPENDIX B

LETTERS OF PERMISSION

---

**Permission Letter from Co-author**

**To:** A Ranjitha Dhanasekaran  
**From:** Jon Pearson  
**Date:** Thu, Dec 1, 2011 at 1:35 PM

I, Jon Pearson, hereby give my permission to Arockia Ranjitha Dhanasekaran to use the manuscript entitled “Metabolome searcher: a high throughput tool for metabolite identification and metabolic pathway mapping” co-authored by Jon L. Pearson, Balasubramanian Ganesan and Bart C. Weiner, in her dissertation.

Hope that works for you. Good luck.

Jon

[Unread text hidden]

---

**Permission Letter from Co-author**

**To:** Ranjitha Dhanasekharan  
**From:** Balasubramanian Ganesan  
**Date:** Thu, Dec 1, 2011 at 1:04 PM

I, Balasubramanian Ganesan, hereby give my permission to Arockia Ranjitha Dhanasekaran to use the manuscript entitled “Metabolome searcher: a high throughput tool for metabolite identification and metabolic pathway mapping” co-authored by Jon L. Pearson, Balasubramanian Ganesan and Bart C. Weiner, in her dissertation.
AROCKIA RANJITHA DHANASEKARAN
CURRICULUM VITAE

EDUCATION

Doctor of Philosophy, Computer Science
Utah State University, Logan, Utah
Dissertation: “A Dynamic State Metabolic Journey: From Mass Spectrometry to Network Analysis via Estimation Of Kinetic Parameters”
Advisor: Dr. Bart C. Weimer

Master of Science, Computer Science
Utah State University, Logan, Utah
Thesis: “Cooption and Irreducibility in Regulatory Networks for Cellular Pattern Development”
Advisor: Dr. Nicholas S. Flann

Bachelor of Engineering, Computer Science & Engineering
University of Madras, Chennai, India
Project: “Browserless Mail Service”
Advisor: Dr. Anthony Irudhayaraj

RESEARCH INTERESTS

Bioinformatics & Computational Biology

RESEARCH EXPERIENCE

Center for Integrated BioSystems
Utah State University, Logan, Utah

DynaFlux: Simulation and Analysis of Dynamic State Metabolic Flux
A software tool for estimation of flux through a metabolic network at non-steady state and analyses of network properties using Mathematica 7.0 & Workbench 2.0.

Metabolome Searcher
Developed a web-based application to identify putative compound names & associated pathways from mass spectrometry data using Perl & Perl CGI.

Western Dairy Center
Utah State University, Logan, Utah

Modeling of Brine Diffusion in Cheese
Calculated the diffusion coefficients based on Fick’s laws of diffusion using Mathematica 7.0 & Workbench 2.0.
RESEARCH EXPERIENCE (CONTINUED)

Department of Computer Science
Utah State University, Logan, Utah

01/2005-07/2007

Computational Modeling of Tumor-Induced Angiogenesis

Derived potential anti-angiogenesis therapies by disrupting the model and exploring the resultant changes using Cellular Potts Model in CompuCell.

Cooption and Irreducibility in Regulatory Networks for Cellular Pattern Development

Analyzed Genetic Regulatory Networks via knockouts, subgraph isomorphism and subcycles using C#.

Biological Development of Mosaic Cell Patterns: Exploring the Space of Genetic Regulatory Networks

Examined the space available for evolutionary exploration of Genetic Regulatory Networks using C#.

TEACHING EXPERIENCE

Department of Computer Science
Utah State University, Logan, Utah

08/2011-Present

Instructor, Computational Science: JAVA/Internet (CS3410)

Delivering lectures, assigning homeworks, preparing exams, and administering all grades (43 students).

HONORS AND AWARDS

School of Graduate Studies Dissertation Fellowship
Utah State University, Logan, Utah

08/2010-04/2011

Merit Admission & Scholarship, Bachelor of Engineering
University of Madras, Chennai, India

08/1998-05/2002

PUBLICATIONS


Presentations


Paper Presentation, “Cooption and Irreducibility in Regulatory Networks for Cellular Pattern Development,” Intermountain Graduate Research Symposium, Utah State University, 2007. (Second Prize: College of Science)

Paper Presentation, “Biological Development of Mosaic Cell Patterns: Exploring the Space of Genetic Regulatory Networks,” Intermountain Graduate Research Symposium, Utah State University, 2006. (Third Prize: College of Science)

Professional Training

Intensive English Language Institute
Utah State University, Logan, Utah

Bioinformatics Workshop 08/14/2007-08/16/2007
Center for Integrated BioSystems
Utah State University, Logan, Utah

Leadership Training 08/20/2007-08/21/2007
Associated Students of Utah State University
Utah State University, Logan, Utah

Student Orientation, Advising & Registration (SOAR) 01/2004-04/2004
A-team facilitator
Utah State University, Logan, Utah

National Institute of Information Technology (NIIT) 06/2003-12/2003
Enterprise Application Developer
Chennai, India

IBM Advance Career Education 09/1998-09/2001
Application Developer Curriculum
Chennai, India

Computer Skills

Languages
Java, C, C++, C#, Perl, LISP, R, SAS, PHP, HTML, XML, Perl CGI, ASP.Net, VB.Net, SQL, Unix Shell, BASIC, Pascal, COBOL, FORTRAN

Softwares
<table>
<thead>
<tr>
<th>COMPUTER SKILLS (CONTINUED)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platforms</td>
</tr>
<tr>
<td>Windows NT/95/98/2000/XP/2003 Server/Vista/7, AIX, LINUX, Mac OS, Solaris, MS-DOS</td>
</tr>
<tr>
<td>Packages</td>
</tr>
<tr>
<td>MS-Office 98/XP/2003/2007, OpenOffice.org</td>
</tr>
<tr>
<td>Databases</td>
</tr>
<tr>
<td>Oracle 9i, MS SQL</td>
</tr>
<tr>
<td>Web Servers</td>
</tr>
<tr>
<td>Apache, Microsoft IIS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OTHER EXPERIENCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Website Developer</td>
</tr>
<tr>
<td>Department of Animal, Dairy and Veterinary Science</td>
</tr>
<tr>
<td>Utah State University, Logan, Utah</td>
</tr>
<tr>
<td>Website Developer &amp; Systems Administrator</td>
</tr>
<tr>
<td>Office of International Students and Scholars</td>
</tr>
<tr>
<td>Utah State University, Logan, Utah</td>
</tr>
<tr>
<td>Note Taker for Psychology and Comparative Cultures</td>
</tr>
<tr>
<td>Disability Resource Center</td>
</tr>
<tr>
<td>Utah State University, Logan, Utah</td>
</tr>
<tr>
<td>Teaching Assistant</td>
</tr>
<tr>
<td>Advanced Artificial Intelligence (CS6600)</td>
</tr>
<tr>
<td>Department of Computer Science</td>
</tr>
<tr>
<td>Utah State University, Logan, Utah</td>
</tr>
<tr>
<td>A-team facilitator</td>
</tr>
<tr>
<td>SOAR</td>
</tr>
<tr>
<td>Utah State University, Logan, Utah</td>
</tr>
<tr>
<td>Student Assistant</td>
</tr>
<tr>
<td>Office of International Students and Scholars</td>
</tr>
<tr>
<td>Utah State University, Logan, Utah</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LEADERSHIP &amp; SERVICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graduate Student Senate, Utah State University</td>
</tr>
<tr>
<td>• Department Representative, Computer Science 2007-Present</td>
</tr>
<tr>
<td>• International Senator 2007-2009</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>International Student Council</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Active General Council Member 2007-2008</td>
</tr>
<tr>
<td>• Volunteered for International Banquet 2005 &amp; 2008</td>
</tr>
<tr>
<td>• Fund Raising Activities</td>
</tr>
<tr>
<td>Bangladesh flood relief 2007</td>
</tr>
<tr>
<td>Tsunami Relief 01/2005</td>
</tr>
<tr>
<td>• Mr. &amp; Ms. International 2nd Runner-up 2004-2005</td>
</tr>
</tbody>
</table>
LEADERSHIP & SERVICE (CONTINUED)

Indian Student Association
- Volunteered for organizing Diwali 2009
- **Best Organizer** Award Diwali 2005 (750 participants)
- Website Developer 2005-2006
- General Secretary 2004-2005

Association for Computing Machinery Committee on Women (ACMW, Dept. of Computer Science)
- Website Developer 2006-2007
- Hosted a **website design workshop** 2007

LANGUAGES

**English** (fluent), **Tamil** (native), **Hindi, Telugu** and **Bengali** (intermediate)

Willing to Relocate – Available December 15 2011