MODELING AND ANALYSIS OF AN OPIOID DETOXIFICATION SYSTEM FOR CELL-FREE METABOLIC ENGINEERING

A Thesis

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Master of Science

by

Sruti Dammalapati

August 2021

ABSTRACT

Opioids are a class of drugs highly valued for their potent analgesic properties; however, their misuse can lead to addiction, overdose incidents, and death. Although the opioid antagonist naloxone has been used in emergency medicine for over 50 years to reverse the effects of an overdose, access to this life-saving antidote is still limited due to its high cost and restricted availability. To address this issue, we designed a novel biosynthetic pathway as an alternative method for naloxone production. In addition, we formulated a mathematical model to simulate the expression of morphine dehydrogenase, the first enzyme of the proposed pathway. Due to its viability as a point-of-care (POC) solution, we used cell-free protein synthesis (CFPS) as our platform for protein production. However, to make CFPS a mainstream technology for POC manufacturing, the performance of these systems must be optimized. Toward this need, constraint-based approaches have become important for model-driven research. A key issue with such models is the existence of alternate optimal solutions which can result in high uncertainties in metabolic flux estimates. Therefore, in this study, we integrated kinetic parameters, enzyme levels and metabolite data as model constraints to generate accurate flux estimations. Since energy efficiency of CFPS is highly dependent on oxidative phosphorylation activity, we studied the effect of two different inhibitors of oxidative phosphorylation on cell-free metabolism. First, we tested the consistency of flux balance analysis (FBA) simulations with experimental measurements. Next, we used minimization of metabolic adjustment (MOMA) as an alternative to FBA, which removed the assumption of optimality of a specific biological objective. MOMA accurately

predicted the overall production of mRNA and protein along with changes in metabolic behavior in the presence of the inhibitors. This modeling approach can be extended to predict the effect of various pathway enzymes involved in the synthesis of naloxone. In addition, it can be used to identify possible negative effectors to improve CFPS yields. Taken together, we have developed an alternate strategy for the production of naloxone and successfully validated MOMA for model guided design and optimization of the proposed platform. Finally, MOMA can be used to engineer strains with improved CFPS performance, thus extending the scope of its application to cell-free metabolic engineering.

BIOGRAPHICAL SKETCH

Sruti Dammalapati received her Bachelors Degree in Chemical Engineering from the National Institute of Technology, Trichy, India in 2017. She joined the Varner Research Group, CBE in Fall 2019 and worked on building mathematical models to perform data-driven analyses of biological systems, specifically cell-free systems. Working under the guidance of Dr. Jeffrey Varner, Sruti graduated in August 2021 with a Master of Science Degree in Chemical Engineering.

I dedicate this to my parents.

ACKNOWLEDGEMENTS

Firstly, I would like to thank God for being my strength for the successful completion of this thesis. I extend my deepest gratitude to my parents, my brother Samvit, and my partner Aviroop, for always supporting me in every way they possibly could. I would specially like to thank my advisor, Dr. Jeffrey D. Varner for his guidance and constant encouragement, and Dr. Sijin Li for serving on my special committee. I would also like to thank Michael Vilkhovoy for making his experimental data and computational tools available for my research work. Finally, I would like to acknowledge some very special people who have helped me during my journey here. To Kasturi Mazumdar and Shreyanka Dhar- thank you for your unconditional support and friendship. To Abhishek Murti, Suthara Ramachandran, Ritika Jain, Naman Gupta, Sahil Desai and Abhishek Mangipudi- thank you for making my graduate school experience at Cornell University memorable. Finally, to Sandra Vadhin, Abhinav Adhikari, Zhiping Zhang, Aaron Wheeler, Zhihao Feng, Rachel LeCover, Anirudh Murali and Aasim Wani- I am so very grateful for knowing each one of you; thank you for making VarnerLab my best decision ever.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Cell-free Protein Synthesis

Cell-free protein synthesis (CFPS) is a powerful platform for engineering proteins and small molecules without the use of living cells. The major advantage of CFPS comes from the absence of a cell membrane. This characteristic of cell-free systems removes constraints associated with cell growth for protein production. Additionally, due to the open nature of the reaction system, direct access to the transcription and translation (TX-TL) environment is possible, providing tighter control over metabolism compared to *in vivo* processes [3, 4, 5]. Cell-free systems are also better suited to handle production of toxic products compared to *in vivo* systems, in which case toxicity can disrupt biosynthetic pathways and hinder cell growth [6]. For example, the expression of restriction endonucleases, cytolethal distending toxin, and human microtubule-binding protein was made possible due to CFPS [7]. Therefore, such advantages of CFPS make cell-free expression a promising alternative to conventional *in vivo* protein synthesis.

Cell-free biology, however, has been around for decades, serving as a useful tool for understanding complex mechanisms underlying gene expression. In the 1950s, cell-free systems were used to understand the incorporation of amino acids into proteins [8, 9, 10] and the role of ATP in protein production [11]. In 1961, Nirenberg and Matthaei discovered the genetic code which earned them a Nobel Prize for Physiology or Medicine [12]. Since then, cell-free biology underwent several advancements with various modifications made to cell-

extract preparation protocols and energy regeneration methods. In 1999, Kim and coworkers revealed that CFPS could be prolonged by regenerating ATP and eliminating deleterious activities that lower energy efficiency [13, 14]. Generating ATP with substrate level phosphorylation [15] and oxidative phosphorylation [16, 17] was shown to improve energy efficiency of E. coli CFPS. In 2007, Spirin and coworkers optimized cell-free systems by continuously removing synthesized products and replenishing reacted substrates [18]. The removal of unwanted reaction byproducts ensured continuous operation of the system. Although they succeeded at producing a single product, the system was energy deficient. Since cell-free systems require a tremendous amount of energy for protein synthesis, a cost-effective solution for sustained energy production remains one of its major challenges.

Towards this opportunity, Jewett and coworkers developed a Cytomin system, co-activating central metabolism, phosphorylation, and protein folding reactions [19]. Another platform, myTXTL [20], emerged more recently which uses a different metabolic process that couples ATP regeneration and inorganic phosphate recycling to extend the duration of protein synthesis. Since cell-free systems are derived from crude cell extracts, transcription and translation (TX-TL) processes rely on the cell's innate machinery. Therefore, to extend the durability of these processes, cell-free systems are often supplemented with buffer, amino acids, NTP, NAD, PEG, tRNA, and metabolic intermediates [21]. Therefore, with the advent of discoveries in metabolic engineering, CFPS has shown remarkable progress as a platform for protein synthesis over the last few decades.

Today, the scope of CFPS is not just limited to research but also extends to-

wards the production of industrially relevant biologics. For instance, CFPS is used for therapeutic protein and vaccine production [22, 23, 24, 25], industrial biocatalyst development [26], biosensing [27, 28] and many other applications [7]. In addition, the portability of cell-free systems has also been improved over the last decade, enabling its use for portable diagnostic testing and point-of-care treatment. For instance, cell-free extracts can be freeze-dried, allowing room temperature storage and distribution [29]. They can then be rehydrated with water at the time of need [30]. However, while the popularity of cell-free systems has grown dramatically, the platform still faces important biomanufacturing challenges. In particular, challenges towards scale-up, high extract cost, and limited post-translational modification capability. Towards this, new methods for the scale-up and optimization of CFPS are presently being researched. Notably, Voloshin and coworkers showed a new thin film technique to improve oxygen supply and produce higher protein yields [31]. Their approach increased the availability of hydrophobic surfaces which benefited protein expression and folding. Cell-free extract preparation methods have also undergone several advancements [32]. Typically, the choice of extract depends on the CFPS application. While E. coli cell extracts are commonly used due to their cost advantage and the availability of established extract preparation protocols, they possess limited post-translational modification capability. More expensive eukaryotic cell extracts are better at folding complex proteins. However, more recently, Guarino and DeLisa [33] have successfully expressed N-linked glycoproteins in an E. coli-based cell-free system. Taken together, such advancements in CFPS have paved the way for cell-free systems to become valuable tools for systems biology research and a promising platform for manufacturing of valuable proteins and chemicals.

In conclusion, for CFPS to become a mainstream technology for industrial bio-manufacturing, some limitations of these systems need to be addressed. Fortunately, the versatility of cell-free systems offers tremendous opportunity for computational modeling and analysis. Mathematical modeling has historically been used in the metabolic engineering community to maximize overall production yield, titer, and efficiency. One in particular, constraint-based modeling, has emerged as a promising tool to understand the performance limits and costs of various biological systems.

1.2 Mathematical modeling

Several mathematical models have been applied to understand complex metabolic processes occurring during protein synthesis [34, 35]. However, a majority of these were deterministic ODE models describing TX-TL processes based on Michaelis-Menten kinetics, as well as mRNA and protein degradation machinery [36, 37, 38]. At varying degrees of complexities, these models were all able to capture mRNA and protein dynamics in cell-free systems. Karzbrun and coworkers [39] derived a coarse-grained description of protein biosynthesis and degradation from which 10 rate constants and concentrations were estimated. They focused on the first hour of reaction since protein degradation dynamics did not reach steady state. They attributed the decay in synthesis to reagents depletion and waste accumulation. As a fixed amount of resources are present in cell-free systems, there is competition among biosynthetic processes which needs to be accounted for in mathematocal models [38, 36]. More recently, Vvoyvodic and coworkers [40] derived a coarse-grained model and included terms that accounted for resource competition for RNA polymerases

and ribosomes, production of toxic byproducts as well as energy consumption for mRNA production. However, the exact cause for decreasing protein yields still remained an open ended question.

While other detailed models emerged trying to predict the effect of resource limitation on CFPS, they could not be generalized for other reaction conditions [41]. The complexity of such models also made parameter estimation a challenging task. However, most importantly, these models where incomplete descriptions of the system because they did not integrate metabolism. Since the central carbon metabolism powers cell-free TX-TL processes, it must be included in mathematical models for more accurate predictions. Moreover, integrating metabolic pathways with descriptions of TX-TL processes can provide insights into resource constraints, energy efficiency and limitations of CFPS [42]. Towards this, constraint-based modeling has emerged as promising tool in systems biology to understand the working of complex metabolic networks [43, 42].

1.3 Constraint-based modeling

Constraint-based models are based on genome-scale reconstructions of metabolic networks which can be mathematically represented based on reaction stoichiometry. Thus, constraint-based models require little to no kinetic parameters, which is their major advantage. Flux balance analysis (FBA), metabolic flux analysis (MFA), as well as convex network decomposition approaches such as elementary modes and extreme pathways, model intracellular metabolism under the steady state assumption. FBA, a widely used constraint-based ap-

proach, typically formulated as a linear programming problem, has been historically successful at predicting yield, productivity, mutant behavior and growth phenotypes. FBA is an under-determined problem which requires the use of constraints on flux rates within the biochemical network [44]. These constraints define an allowable solution space and are determined from reaction thermodynamics, growth media constituents as well as from experimental measurements. For FBA, a linear objective function is typically chosen to arrive at the optimum solution using linear programming algorithms. For *in vivo* protein synthesis, several studies have compared the performance of various objective functions for FBA. In most cases, maximization of cellular biomass was concluded to be an appropriate objective function that could describe experimentally observed flux distributions [45, 46, 47]. However, Knorr and coworkers [48] determined the most probable objective function to be minimization of redox potential. Ow and coworkers [49] concluded that maximization of ATP dissipation is the best objective function. However, for CFPS, this objective function is set to maximize the rate of translation of protein [42]. According to Vilkhovoy and coworkers, the FBA under this objective function, coupled with sequence specific descriptions of TX-TL processes accurately predicted chloramphenicol acetyltransferase (CAT) protein and mRNA production.

More recently, constraint-based modeling has also been used for more challenging metabolic engineering applications. For example, FBA which is based on the assumption of optimality may not be suitable for predicting the metabolic state of organisms after a knock-out or perturbation. Towards this, approaches like regulatory on/off minimization (ROOM) [50] and minimization of metabolic adjustment (MOMA) [51] have emerged that seek to appropriately predict transient metabolic states after genetic perturbations. MOMA is de-

signed as a quadratic programming problem which assumes that the metabolic flux distribution immediately following a knock-out will resemble the wild type flux distribution with minimal adjustment. It can, thus, be used for predicting the behavior of perturbed metabolic networks, whose growth performance is in general sub-optimal. On the other hand, ROOM finds a flux distribution that satisfies the same constraints as FBA while minimizing the number of significant flux changes. ROOM is shown to provide more accurate flux predictions than FBA and MOMA for the final metabolic steady state [50]. Taken together, constraint-based models serve as useful tools for describing and predicting phenotypes in a metabolic network by considering physical, enzymatic, and kinetic constraints.

CHAPTER 2

DEVELOPMENT OF COMPUTATIONAL SOFTWARE TOOLS FOR CELL-FREE MODEL GENERATION

2.1 Introduction

[1] Cell-free biology is an emerging technology for research, and the point of care manufacturing of a wide array of macromolecular and small molecule products. A distinctive feature of cell-free systems is the absence of cellular growth and maintenance, thereby allowing the direct allocation of carbon and energy resources toward a product of interest. Moreover, cell-free systems are more amenable than living systems to observation and manipulation, hence allowing rapid tuning of reaction conditions. Recent advances in cell-free extract preparation and energy regeneration mechanisms have increased the versatility and range of applications of cell-free metabolic engineering (CFME). Thus, the cell-free platform has transformed from merely an investigative research tool to become a promising alternative to traditionally used living systems for biomanufacturing as well as biological research. In combination with the rise of synthetic biology, cell-free systems today have not only taken on a new role as a promising technology for just in time manufacturing of therapeutically important biologics and high-value small molecules, but have also been utilized for applications such as biosensing, prototyping genetic parts, and metabolic engineering.

[1] Adapted with permission from Adhikari, A.; Zhang, Z.; Murti, A.; Dammalapati, S.; Varner, J. CellFreeModelGenerationKit.jl: A Package for the generation and analysis of Sequence Specific Cell Free Models in the Julia Programming Language. In preparation.

2.1.1 Julia programming language

Julia is a high-level dynamic programming language specifically designed for scientific computing applications. It creates a new approach to numerical computing by combining the diverse fields of computer science and computational science. Julia is designed to be easy and fast. It compiles code to a machine readable language using a just-in-time (JIT) compiler. One of the unique features of Julia is its multiple dispatch through which methods can be dynamically dispatched based on the attributes of more than one of its arguments. The Julia REPL (read–evaluate–print loop) console is an an interactive text-based interface that takes an input, processes it and returns the results to the user.

The Julia ecosystem has upwards of 2,600 packages, with their functionality ranging from machine learning and DNA sequence analysis to mathematical modeling and simulations. Julia's built-in package manager, Pkg handles operations such as installing, updating and removing packages. Julia supports parallel and distributed computing while also providing direct calling of C and Fortran libraries. It also allows for importing functionality from other languages such as Python and R directly, by using packages such as Pycall (Python) and Rcall (R). Therefore, the code generators described in the following sections are written in the Julia programming language to take advantage of some of these unique features.

2.2 Code generators

2.2.1 VLConstraintBasedModelGenerationUtilities

[2]`VLConstraintBasedModelGenerationUtlities` is a Julia package holding constraint based model generation utility functions and types. This package includes methods to parse gene and protein sequences in standard FASTA format, and build reaction tables in four possible formats, viz., SBML, VFF, XML and JSON.

2.2.2 VLModelParametersDB

[3] `VLModelParametersDB` is a Julia package that wraps a SQLite database holding transcription (TX) and translation (TL) parameters (taken from BioNumbers and other sources), and enzyme kinetic information taken from the BRENDA enzyme database. This package allows for storage and manipulation of the relevant parameters required for building biophysical models of transcription and translation processes in three biological systems, namely, bacteria, human and cell-free. This package includes methods to transform the parameters from a CSV format to an intermediate dictionary and finally to a database, which can be accessed through SQL queries. The package is also supports JSON format so the user can generate their own database, which can be transformed into a data frame for manipulation.

[2]Varnerlab. Constraint-Based Model Generation Utilities (VLConstraintBasedModelGenerationUtilities). Available online at https://github.com/varnerlab/VLConstraintBasedModelGenerationUtilities.jl.

[3]Varnerlab. Model Parameters Database (VLModelParametersDB). Available online at https://github.com/varnerlab/VLModelParametersDB.jl.

2.2.3 JUGRNModelGenerator

[4] The `JUGRNModelGenerator` package is a code generation system that transforms simple JSON descriptions of the connectivity of gene regulatory networks into model code written in the Julia programming language. The input to the code generator is a JSON file which describes various components of the gene regulatory network; species present globally in the system, species involved in the genetic circuit, as well as transcription and translation models used to describe gene regulation. In other words, the model specification file in JSON format defines the biology of the model that gets generated. Table 2.1 describes the files generated by `JUGRNModelGenerator`.

2.2.4 CellFreeModelGenerationKit

Installation and testing

[5] `CellFreeModelGenerationKit.jl` is open source and available under a MIT software license. `CellFreeModelGenerationKit.jl` requires Julia version 1.6.x and above, and can be downloaded and installed as a package from the Julia package repository. `CellFreeModelGenerationKit.jl` requires the following dependencies: `DataFrames`, `CSV`, `Dates`, `Logging`, `WordTokenizers`, `DelimitedFiles`, `SQLite`. The dependencies and all other necessary packages and will be automatically installed when the code is run for the first time.

[4] Varnerlab. Gene Regulatory Network Model Generator in Julia (JUGRNModelGenerator). Available online at https://github.com/varnerlab/JUGRNModelGenerator.jl.

[5] Varnerlab. Cell-Free Model Generation Kit in Julia (CellFreeModelGenerationKit). Available online at https://github.com/varnerlab/CellFreeModelGenerationKit.jl.

Table 2.1: Description of `JUGRNModelGenerator` generated files

Filename	**Description**
`Balances.jl`	Material balance equations for genes, mRNA and proteins in the GRN
`Control.jl`	Encodes the control logic described in the GRN network file
`Parameters.jl`	Encodes the model parameters e.g., initial conditions or promoter function parameters in a Julia dictionary
`Degradation.dat`	Stoichiometric matrix for mRNA and protein degradation reactions
`Driver.jl`	Example script that can be used to solve the continuous material balance equations
`Include.jl`	Includes all the files into the current workspace
`Initialize.jl`	Adds all the required packages using the package manager
`Kinetics.jl`	Encodes the rate of transcription, translation and degradation for mRNA and protein species
`Network.dat`	Stoichiometric array for the transcription and translation reactions
`Solve.jl`	Solves the material balance equations using ODE solvers from the `DifferentialEquations.jl` package

To automate the development cycle of the `CellFreeModelGenerationKit.jl` package, we implemented a continuous integration (CI) pipeline using GitHub Actions. Therefore, every time changes are committed to version control, the CI workflow builds and tests the software to prevent any breaking changes to the code base. This also helps us identify and address potential issues during various phases of software development. While continuous integration tests the working of various modules in unison, we use Julia's standard library `Test` for unit testing or testing individual components of code. Through unit testing, we identify the correctness of code by verifying what the code does against the results we expect to see. Unit tests are declared in the `test/runtests.jl` file in the `CellFreeModelGenerationKit.jl` package. The package can then be tested from the Pkg REPL mode.

2.2.5 Code generation

Presently, `CellFreeModelGenerationKit.jl` transforms a structured text file (in the VFF format) into cell-free model code. The VFF format consists of delimited record types organized into five sections BIO-TYPE-PREFIXES, TXTL-SEQUENCE, METABOLISM, SPECIES-BOUNDS and GRN. Bio-type prefixes records are used to identify and declare the types of species. TXTL-SEQUENCE records are used to generate sequence specific transcription and translation reactions which are appended to the end of the metabolic reactions encoded in the METABOLISM section. METABOLISM records are used to encode metabolic reactions. GRN records are used to define the biology of the model being generated. In this section, the various types of species (promoters, genes and polymerases) involved in the regulatory circuit and their

regulatory action can be defined. Importantly, the reactions for tRNA charging, transcription, translation, and mRNA degradation, are generated from the GRN section of the VFF file and added to the metabolism reactions. Thus, the `CellFreeModelGenerationKit.jl` package reads a VFF file that describes the system in consideration with the corresponding TXTL sequence, metabolism and GRN sections and writes model code to an output directory specified by the user. Table 2.2 describes all the generated files. During this process, if a TOML file with default parameter values is not provided by the user, a `Defaults.toml` file is generated automatically which is populated with cell-free biophysical parameters taken from Adhikari and coworkers [52]. Additionally, the user can edit the generated `Defaults.toml` file with their own values. After generating code, if a directory already exists at the user specified location, it can be deleted or backed-up before new code is written based on user input. Finally, the output scripts that get generated can be used to run static flux balance analysis of a cell-free system.

The static sequence specific flux balance analysis problem which can be run from `Static.jl` was formulated as a linear program:

$$\max_{\boldsymbol{w}} \left(w_X = \boldsymbol{\theta}^T \boldsymbol{w} \right) \tag{2.1}$$

subject to:

$$\mathbf{Sw} \quad = \quad \mathbf{0} \tag{2.2}$$

$$\mathcal{L}_i \leq \quad w_i \quad \leq \mathcal{U}_i \qquad i = 1, 2, \ldots, \mathcal{R} \tag{2.3}$$

where $\mathbf{S}$ denotes the stoichiometric matrix ($\mathcal{M} \times \mathcal{R}$) and σ_{ij} denotes the stoichiometric coefficient for species i in reaction j, $\mathbf{v}$ denotes the unknown flux vector ($\mathcal{R} \times 1$), $\boldsymbol{\theta}$ denotes the objective vector ($\mathcal{R} \times 1$), and $r_j(\mathbf{x}, \epsilon, \mathbf{k})$ denotes the rate of reaction j.

Table 2.2: Description of `CellFreeModelGenerationKit` generated files.

Filename	**Description**
`Include.jl`	Includes all the generated files into the current workspace
`Static.jl`	Solves the FBA problem
`Checks.jl`	Checks whether or not a file with the given name exists in the current directory
`Constraints.jl`	Encodes the bounds for the fluxes as well as the species for the FBA problem
`Control.jl`	Encodes the control logic described in the GRN network file
`Data.jl`	Encodes model parameters e.g., initial conditions or promoter function parameters in a dictionary
`Flux.jl`	Computes the optimal metabolic flux distribution given the constraints using `GLPK.jl`
`Kinetics.jl`	Encodes the rate of transcription, translation and degradation for mRNA and protein species
`Network.dat`	Stoichiometric array for the metabolism as well as transcription and translation reactions
`Solver.jl`	Contains the functions required for solving a static or dynamic FBA problem
`Types.jl`	Contains abstract and concrete data types used for model generation and calculation
`Utility.jl`	Encodes utility functions required for model calculation (e.g., computation of the Jacobian)

The optimization problem can be set up by updating the body of the functions present in the `Constraints.jl file`. The objective function can be updated by modifying the `objective_coefficient_array`. By default, the problem is set up as a minimization problem `(min_flag = true)` which can be updated as well, according to the objective. Finally, the constraints (both species and flux bounds) for the linear program can be modified according to the problem. If not updated, the code runs with default flux and species bounds.

2.2.6 Advanced functionalities

To extend the functionality of the code generator, advanced users can customize certain methods or write callback functions depending on their model code requirements. These methods can be used to modify the strategy of the parser or customize the content of the scripts written to the output directory.

2.3 Future work

At present, `CellFreeModelGenerationKit.jl` can be used to generate code that solves a static flux balance problem. Future developments to this project will allow code generation for dynamic flux balance analysis as well. This can be implemented by discretizing the problem, periodically updating the flux bounds and dynamically calling the `Static.jl`. A separate script, `Dynamic.jl`, which is easily customizable by the user, will be available to run the dynamic analysis. We also plan to broaden the scope of the code generator by extending its functionality to generate cell-free model code for use with

other programming languages, in addition to Julia. For example, by generating fully editable source code for languages like Octave and Python 3.x or a COBRA-compatible MAT-file for MATLAB.

Finally, for future versions of `CellFreeModelGenerationKit`, we plan to integrate `VLModelParametersDB` and `VLConstraintBasedModelGenerationUtlities` packages into the code generation processes, which was actually the motivation for developing those packages. In that case, the VFF format `json` file, which is an input to `CellFreeModelGenerationKit` in the current version, will be replaced with a JSON file. This has already been implemented in the `JUGRNModelGenerator` package.

2.4 Conclusions

`CellFreeModelGenerationKit` is a Julia package for generating model code for performing sequence specific constraint-based simulations of cell-free genetic circuits. This package and the associated dependencies are specifically designed to bridge the gap between computational and experimental research in biology, by generating model code based on simple descriptions of biological systems. This will offer two significant advantages. Primarily, computational researchers can focus their efforts on carrying out simulations and performing data analysis instead of writing code from scratch. Secondly, it allows for more collaborative opportunities between biologists and computational researchers by providing a framework for model-guided experimental design.

CHAPTER 3

INTEGRATED KINETIC CONSTRAINT-BASED MODELS OF E. COLI CELL-FREE PROTEIN SYNTHESIS

Abstract

[1] Cell-free protein expression has become a widely used research tool in systems and synthetic biology, and a promising technology for biomanufacturing of proteins. Cell-free protein synthesis relies on transcription and translation machinery to produce a protein of interest. However, to fuel this process requires biochemical enzymes and reactions that are involved in complex metabolic pathways. In this study, we integrated kinetic parameters, enzyme levels and metabolite data as model constraints to generate accurate flux estimations. Since energy efficiency of CFPS is highly dependent on oxidative phosphorylation activity, we studied the effect of two different inhibitors of oxidative phosphorylation on cell-free metabolism. First, we tested the consistency of flux balance analysis (FBA) simulations with experimental measurements. Next, we used minimization of metabolic adjustment (MOMA) as an alternative to FBA, which removed the assumption of optimality of a specific biological objective. MOMA accurately predicted the overall production of mRNA and protein along with changes in metabolic behavior in the presence of the inhibitors. This modeling approach can be used to identify possible negative effectors to improve CFPS yields or predict the effect of adding various enzymes on CFPS performance.

[1] Adapted with permission from Vilkhovoy, M.; Dammalapati, S.; Varner, J. An integrated kinetic constraint-based model of E. coli cell-free protein synthesis. In preparation.

3.1 Introduction

Energy generation is one of the major challenges in cell-free systems as the productivity of cell-free protein synthesis (CFPS) appreciably depends on ATP supply. Unlike *in vivo* systems, energy generation in cell-free is quite expensive, often requiring the addition of external sources of energy for prolonging protein synthesis [53, 54]. This is largely because transcription and translation (TX-TL) processes require continuous supply of ATP equivalents for protein synthesis and the failure to maintain this supply of energy halts productivity. Towards this, many studies have investigated energy regeneration methods in cell-free systems to reduce costs and maximize productivity of CFPS.

Historically, energy regeneration in cell-free systems was thought to depend on substrate-level phosphorylation, which is known to be an inefficient process [55, 56, 57]. Since then, several energy sources were studied as a means for prolonging CFPS. Kim and coworkers showed that supplementing cell-free extracts with phophoenol pyruvate (PEP) resulted in the accumulation of inorganic phosphate which eventually halted protein synthesis [14]. They were able to circumvent this problem when Pyruvate was used as an energy source, which extended protein synthesis by two hours [13]. However, this system was dependent on the exogenous supply of oxygen and pyruvate oxidase enzyme. The PANOx system (PEP, amino acids, NAD+, oxalic acid) addressed this limitation and generated ATP from pyruvate by the co-addition of NAD and CoA to the extract [15]. The addition of cofactors facilitated ATP regeneration by activating the E. coli pathway involving pyruvate dehydrogenase (PDH) and phophotransacetylase (PTA). The success with the PANOx system further inspired the utilization of glycolytic intermediates as an energy source. For

example, fructose 1,6-bisphosphate (FBP) and 3-phosphoglyceric acid (3PGA) showed promising results when used as energy sources for CFPS [15, 58].

More recently, glucose has been studied as an energy source for CFPS. However, due to the subsequent accumulation of organic acids in the commercial S30 E. coli extract, glucose was reported to be inefficient for CFPS [59]. Towards this, Kim and coworkers replaced S30 with the S12 E. coli extract which contained endogenous NAD and CoA co-factors. This led to substantial improvement in productivity of CFPS [60]. Another approach, referred to as the dual energy system, used two different ATP regeneration methods to prolong the production of chloramphenicol acetyltransferase (CAT) in the S30 cell-free E. coli extract [61]. This system produced 2-3 times more protein than methods using a single energy source. Using the S30 cell-free E. coli extract, Jewett and coworkers presented the Cytomim system, which co-activated central carbon metabolism, oxidative phosphorylation, and protein synthesis [19]. In effect, the Cytomim system attempted to mimic natural metabolism and simultaneously activated complex enzyme systems without live cells. They hypothesized that the inverted inner membrane vesicles (IMVs) that remain intact in the extract could be activated to support oxidative phosphorylation for protein synthesis. Towards this, they showed that CAT yield increased by 33% when the reaction was augmented with 10 mM phosphate. However, it was unknown whether the addition of phosphate enhanced oxidative phosphorylation or inhibited phosphatase reactions.

To better understand factors influencing energy efficiency in cell-free systems, Vilkhovoy and coworkers [42] simulated the production of CAT using sequence-specific constraint-based modeling. They found oxygen consumption

rate to be a critical factor controlling the energy efficiency of CFPS. The accumulation of acetate and lactate during CAT production suggested that CFPS was not operating with optimal oxidative phosphorylation activity. Moreover, their model showed that oxidative phosphorylation must be active to simultaneously satisfy the metabolic and protein production constraints in the system. These studies indicate that oxidative phosphorylation activity correlates with improved productivity of cell-free systems. Therefore, to further understand oxidative phosphorylation activity in CFPS to find opportunities for optimization, we examined the effect of biochemical inhibitors of oxidative phosphorylation on the production of green fluorescent protein (GFP) using the commercial myTXTL E. coli cell-free system [20].

In this study, we first used flux balance analysis (FBA) to predict GFP mRNA and protein production when the cell-free extract was treated with two different inhibitors of oxidative phosporylation- 2-4-dinitrophenol (DNP) and thenoyltrifluoroacetone (TTA). As with many constraint-based methods, the FBA objective function does not always lead to a unique optimal solution [42]. Therefore, we incorporated experimental metabolite measurements into FBA to constrain the solution towards the experimentally observed behaviors of the treatment groups. Then, we maximized the rate of protein translation to find the optimal flux distribution [42]. Under this objective, FBA results were consistent with the experimentally observed levels of mRNA and protein for the wild-type (or control) and perturbed groups (or treated groups). However, according to *in vivo* studies, FBA can make inaccurate flux predictions when cells are perturbed from their optimal wild type state by changing media conditions or gene knockouts. This hypothesis is based on the the fact that perturbed systems probably do not possess a mechanism for immediate regulation of fluxes toward the op-

timal objective [51]. Towards this, a variety of methods have emerged to predict fluxes that are close to the unperturbed wild type flux distribution [51, 50]. Of these, one such perturbation method called minimization of metabolic adjustment (MOMA), does not assume optimality of any metabolic function. Actually, MOMA is reported to more accurately predict flux distributions of perturbed systems *in vivo* compared to FBA [62, 51]. However, there is currently no known study that has employed MOMA to examine CFPS, specifically CFPS occurring under abnormal conditions.

Therefore, the crux of this study explores the application of MOMA to a cell-free system incubated with DNP and TTA. In particular, we formulated a quadratic program which minimizes the squared difference between fluxes in the perturbed state and the wild-type state. MOMA accurately captured the time-course behavior of the two treatment groups by predicting GFP mRNA and protein production. We compared the results from FBA and MOMA using two types of performance metrics, viz., energy efficiency and carbon yield. MOMA estimated lower energy efficiency for TX-TL processes. This was expected as FBA maximized the rate of translation, which directed energy resources towards TX-TL. Both approaches showed inefficiencies in energy distribution with significant wastage of resources which could otherwise be directed towards TX-TL processes. Carbon flux distributions showed variations between FBA and MOMA model solutions, with MOMA predicting a higher percentage of carbon flux directed towards the pentose phosphate pathway and organic acid secretion and a lower percentage of carbon flux through glycolysis for the treatment group. In effect, cell-free systems satisfied the MOMA objective as their flux distributions underwent minimal adjustment post-perturbation. Therefore, this work enables understanding of metabolic processes used for

powering TX-TL in cell-free systems from an alternate perspective compared to a standard FBA approach. Furthermore, future extensions of this model will be useful for redirecting energy resources for improving the performance of CFPS, especially CFPS using extracts prepared from genetically modified strains. Taken together, we present a modeling framework that describes and predicts CFPS metabolism with reasonable accuracy that can be potentially used to identify strategies for cell-free metabolic engineering applications.

3.2 Methods

FBA was formulated as a linear programming problem:

$$\max_{\boldsymbol{v}} \left(Z = \boldsymbol{\theta}^T \boldsymbol{v} \right) \tag{3.1}$$

subject to:

$$\begin{aligned} (\mathbf{S}\boldsymbol{v} - \dot{\mathbf{x}}) &\geq \mathbf{0} && (3.2) \\ \dot{x}_i &= \sum_{j=1}^{\mathcal{R}} \sigma_{ij} r_j(\mathbf{x}, \epsilon, \mathbf{k}) \qquad i = 1, 2, \ldots, \mathcal{M} && (3.3) \\ 0 \leq v_j &\leq r_j(\mathbf{x}, \epsilon, \mathbf{k}) \qquad j = 1, 2, \ldots, \mathcal{R} && (3.4) \end{aligned}$$

where $\mathbf{S}$ denotes the stoichiometric matrix ($\mathcal{M} \times \mathcal{R}$) and σ_{ij} denotes the stoichiometric coefficient for species i in reaction j, $\mathbf{v}$ denotes the unknown flux vector ($\mathcal{R} \times 1$), $\boldsymbol{\theta}$ denotes the objective vector ($\mathcal{R} \times 1$), and $r_j(\mathbf{x}, \epsilon, \mathbf{k})$ denotes the rate of reaction j. The reaction stoichiometry for TX-TL processes was based on the model of Allen and Palsson [63]. The objective of linear programming problem maximized the rate of maltodextrin consumption, transcription initiation, transcription, mRNA degradation, translation initiation and translation. For the DNP case, proton leak was also maximized in addition to other reactions. The

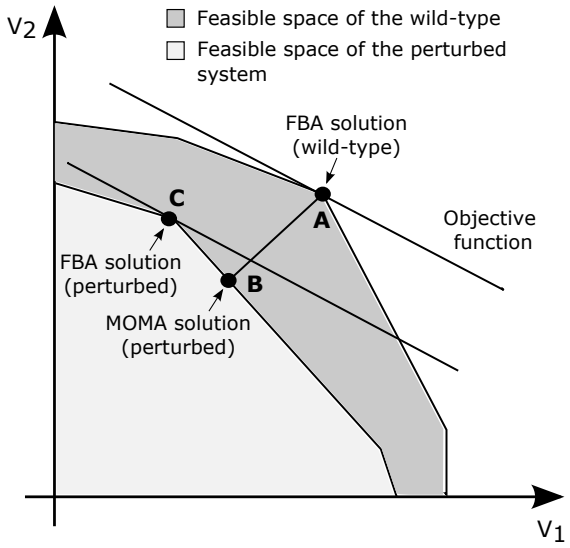

Figure 3.1: Schematic 2D representation of the feasible space for the wild-type (light grey polygon) and perturbed networks (dark grey polygon). The coordinates denote two arbitrary representative fluxes. Point **A** is the optimal FBA prediction for the wild type (or control case) and point **B** is the optimal FBA prediction for the perturbed system (with DNP/ TTA treatment). Point **C** is the alternative MOMA solution calculated through quadratic programming.

linear program for dynamic flux balance analysis was solved using the GNU Linear Programming Kit (GLPK) [64].

MOMA was formulated as a quadratic program (Figure 3.1). The MOMA formulation used the same constraints as FBA but relaxed the assumption of optimality of a specific biological objective.

$$\min \ \|\boldsymbol{w} - \boldsymbol{v}\|_2^2 \tag{3.5}$$

subject to:

$$(\mathbf{S}\boldsymbol{v} - \dot{\mathbf{x}}) \ \geq \ \mathbf{0} \tag{3.6}$$

$$\dot{x_i} \ = \ \sum_{j=1}^{\mathcal{R}} \sigma_{ij} r_j(\mathbf{x}, \epsilon, \mathbf{k}) \qquad i = 1, 2, \ldots, \mathcal{M} \tag{3.7}$$

$$0 \leq \ v_j \ \leq r_j(\mathbf{x}, \epsilon, \mathbf{k}) \qquad j = 1, 2, \ldots, \mathcal{R} \tag{3.8}$$

where $\boldsymbol{w}$ is the known wild type flux vector (control case) and $\boldsymbol{v}$ is the unknown flux vector of the perturbed system (either TTA or DNP case). The FBA solution to the control problem was used as the wild-type flux distribution input to MOMA. The quadratic programming problem was solved using the `Convex.jl` Julia package along with Gurobi optimizer [65].

The TX-TL and maltodextrin consumption reactions were captured by saturation kinetics; whereas all other metabolic reactions were calculated as the product of the turnover rate k_j and enzyme abundance ϵ_j. Species abundance in the cell-free system for species x was modeled as:

$$\mathbf{x}_{t+\Delta t} = \mathbf{x}_t + \mathbf{S}\boldsymbol{v}\Delta t \tag{3.9}$$

where Δt is the time step and t is the current time point. Constraints were placed on both species and flux rates. The initial maltodextrin species availability in the extract was set to 30 mM based on literature data. The maximum rate of change in species i was bounded by experimental metabolite data such that:

$$|\dot{x}_i| \leq \mathcal{B}_i \qquad i = 1, 2, \ldots, \mathcal{M} \tag{3.10}$$

This bound B_i was determined by fitting a regression spline to the experimental time series concentration data using the `SmoothingSplines` Julia package. This included absolute measurements for 63 metabolites encompassing central carbon metabolites, energy species and amino acids. From the regression spline, the rate of change of species concentration at time step t was determined using forward differences (from t to t + Δt). For nucleotide monophosphates and diphosphates, this upper bound was set to 0.02 mM/h to allow accumulation of mRNA.

Flux rates were bounded by the product of the turnover rate k_j and enzyme abundance ϵ_j. These values define the maximum allowable flux rates through

those reactions. The turnover rates for each reaction was taken from BRENDA [66] or Adadi and coworkers [67]. The enzyme abundance was identified for 104 reactions from Garenne and coworkers [68]. The enzyme abundance for all remaining enzymes were set to a median value of 50 nM. Based on this, the upper bound on the flux through reaction j was modeled as:

$$v_j \leq v_{max} \tag{3.11}$$

$$v_{max,j} = k_j \epsilon_j \tag{3.12}$$

In addition, constraints were placed on TX-TL reaction rates using effective biophysical models, following the work of Vilkhovoy and coworkers [42]. The upper bound on the transcription initiation rate was formulated as

$$r_{TX,init} = V_{TX,max} \frac{G}{\tau_{TX} K_{TX} + (\tau_{TX} + 1) G} \tag{3.13}$$

where G is the concentration of DNA plasmid in the cell-free reaction, K_{TX} is the transcription saturation coefficient, and τ_{TX} is the transcription time constant. $V_{TX,max}$, the maximum transcription rate was modeled as:

$$V_{TX,max} = R_{TX} \frac{\dot{n}_{TX}}{l_G} u(\kappa) \tag{3.14}$$

where R_{TX} is the RNA polymerase concentration, $\dot{n}_{TX}$ is the RNA polymerase elongation rate (nt/h) and l_G is the gene length (nt). $u(\kappa)$ is the transcription control function for the P70a promoter taken directly from Vilkhovoy and coworkers [42]. The transcription rate was formulated as:

$$r_{TX} = r_{TX,init} \prod_{s \in m_{TX}} \frac{x_s}{K_{s,TX} + x_s} \tag{3.15}$$

where m_{TX} is the set of reactants (ATP, CTP, GTP, and UTP) in the transcription reaction. $K_{s,TX}$ is the saturation constant for species s. The degradation of mRNA was modeled following first order kinetics:

$$r_d = k_d^{mRNA} . x_{mRNA} \tag{3.16}$$

where k_d denotes the mRNA degradation rate constant and x_{mRNA} denotes the mRNA concentration. Similarly, the translation initiation rate as well as the translation rate was formulated as:

$$r_{TL,init} = r_{TL} = V_{TL,max} \frac{x_{mRNA}}{\tau_{TL} K_{TL} + (\tau_{TL} + 1) x_{mRNA}} \tag{3.17}$$

where K_{TL} is the translation saturation coefficient, and τ_{TL} is the translation time constant. Here, the maximum translation rate, $V_{TL,max}$, was formulated as:

$$V_{TL,max} = K_p R_{TL} \frac{\dot{n}_{TL}}{l_p} \tag{3.18}$$

where K_p is the polysome amplification constant, R_{TL} is the ribosome concentration, $\dot{n}_{TL}$ is the ribosome elongation rate (aa/h) and l_p is the number of amino acids in the protein of interest.

All parameters used above are listed in Table A.1. Since these parameters are taken from literature, there can be uncertainty in their values. Therefore, an ensemble set of flux solutions was generated in each case. For this, four important parameters, viz., RNA polymerase concentration, maximum transcription rate, ribosome concentration and maximum were randomly sampled within physiological ranges. RNA polymerase concentration levels were sampled between 0.060 and 0.075 μM, maximum transcription rates were sampled between 15 and 25 nt/s, ribosome concentration levels were sampled between 2.0 and 2.3 μM and maximum translation rates were sampled between 1 and 2 aa/s.

3.3 Results

Dynamic FBA simulations captured the time-dependent behaviour of the three cases examined here, viz., DNP, TTA and Control (Figure 3.2). In the case of

Control, the model failed to capture mRNA behavior at the 16 hour time point as there was a decline in the simulated levels of CTP, GTP and UTP in the model. However, the experimental system showed sustained mRNA levels. In the case of DNP and TTA, mRNA levels peaked at 2 hours and then declined over the course of the reaction. Since DNP and TTA are inhibitors of oxidative phosphorylation, the reduced GFP yield in their case supports our previous understanding that TX-TL is dependent on oxidative phosphorylation [42, 19]. TTA is an inhibitor of succinate dehydrogenase, which works by decoupling the TCA cycle from central carbon metabolism. On the other hand, DNP is a membrane gradient uncoupler which inhibits oxidative phosphorylation by preventing the proton gradient from forming across the membrane. Therefore, to capture the effect of DNP addition in our mathematical model, we introduced a reaction which leaked a charged proton to an uncharged proton in the metabolic network. Taken together, FBA sufficiently captured GFP mRNA and protein concentrations in the control and perturbed systems when the rate of translation was maximized.

However, FBA ignores the possibility that, under abnormal conditions, metabolic networks may not immediately regulate towards the optimal objective. Therefore, we introduced the method of MOMA to better understand the inhibitory effect of DNP and TTA on CFPS. As described earlier, the mathematical formulation of MOMA was based on the same stoichiometric constraints as FBA, but relaxed the assumption of maximizing the rate of translation of GFP. Additionally, we accounted for uncertainties arising from experimental measurements of flux constraints and literature estimates of model parameters. We also accounted for uncertainties in the control (or wild-type) FBA solution which is provided as an input to the MOMA formulation. MOMA simulations

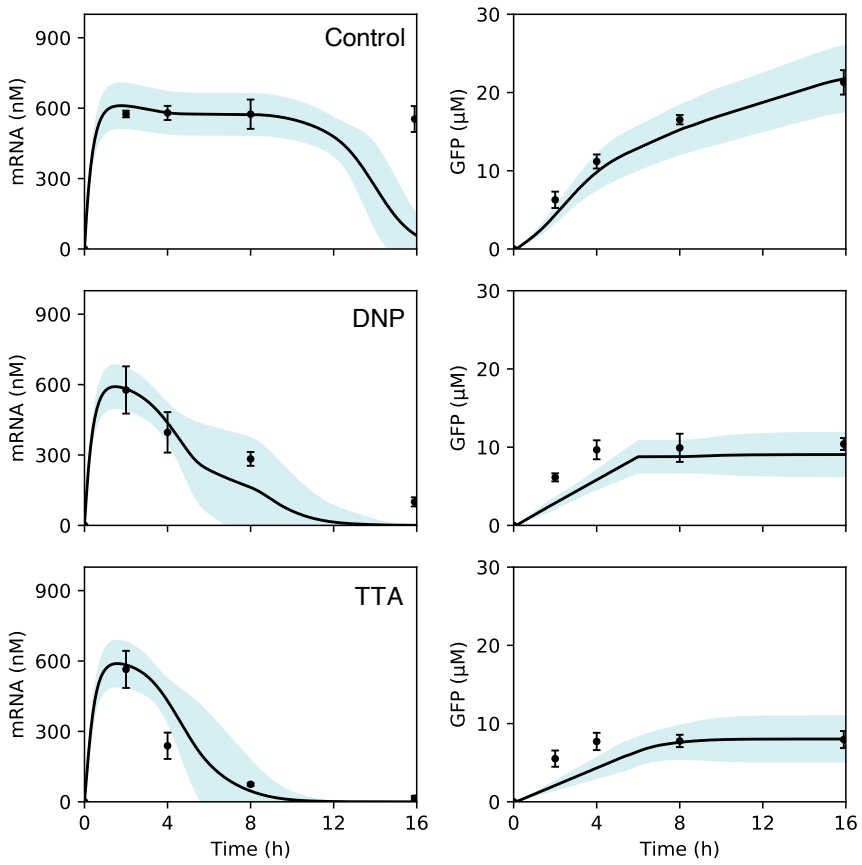

Figure 3.2: FBA mRNA and protein predictions for GFP in the three cases: (A) control; (B) TTA; (C) DNP. The lines represent the mean of the ensemble (set of 100 solutions or N=100) and the shaded regions represent the 95% confidence interval.

(Figure 3.3) captured the experimental mRNA and protein concentration data for both DNP and TTA, with only marginal differences between the FBA and MOMA predictions. Compared to the FBA solution for DNP, the MOMA solution showed a gradual increase in protein accumulation between the 6 hour and 16 hour time points. In the case of TTA, although MOMA marginally under-predicted GFP protein concentration, the ensemble solution set captured the experimental data points. Therefore, MOMA accurately predicted the optimal GFP production in the CFPS system when oxidative phosphorylation was

limited by the addition of DNP and TTA. Next, to understand the differences between the MOMA and FBA predictions, we calculated two different performance metrics: carbon yield and energy efficiency.

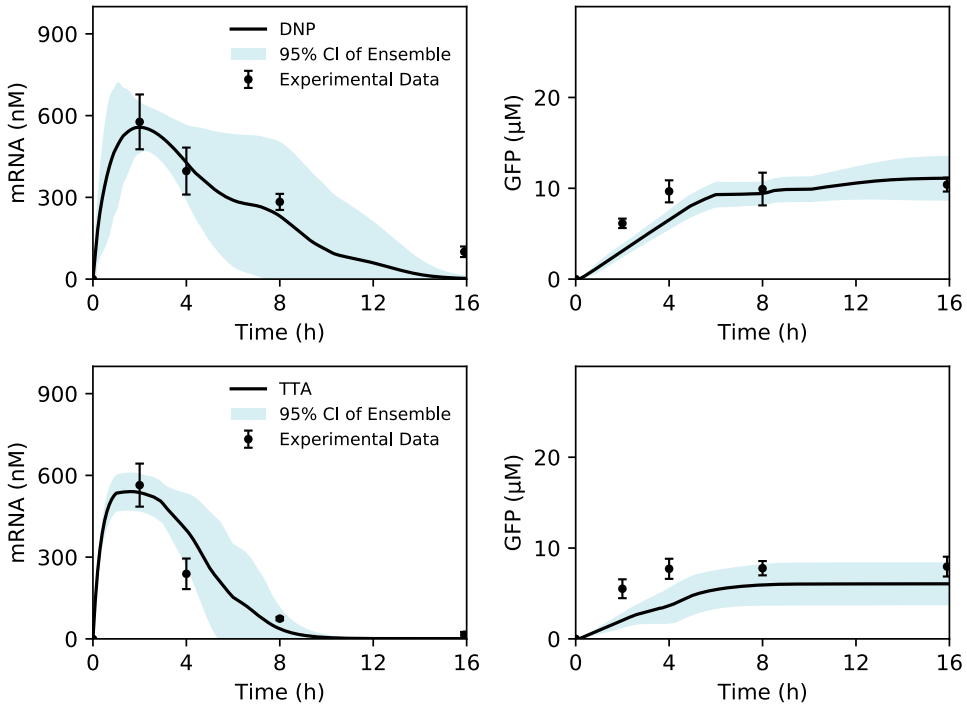

Figure 3.3: MOMA mRNA and protein predictions for GFP in the two cases: (top) DNP; (bottom) TTA. The lines represent the mean of the ensemble (set of 100 solutions or N=100) and the shaded regions represent the 95% confidence interval.

CFPS carbon yield was calculated for the duration of the reaction as a ratio of the concentration of carbon produced to the total concentration of carbon consumed. We observed differences in the ultimate fate of carbon in the perturbed systems as predicted with the FBA and MOMA objective functions (Figure 3.4). For the treatment group, MOMA predicted that the largest flux of carbon went towards the pentose phosphate pathway (29%, 17%, 33% for MOMA, FBA, Control), followed by purine, pyrmidine and chorismate metabolism (the other category; 23%, 33%, 19% for MOMA, FBA, Control), glycolysis (16%, 17%,

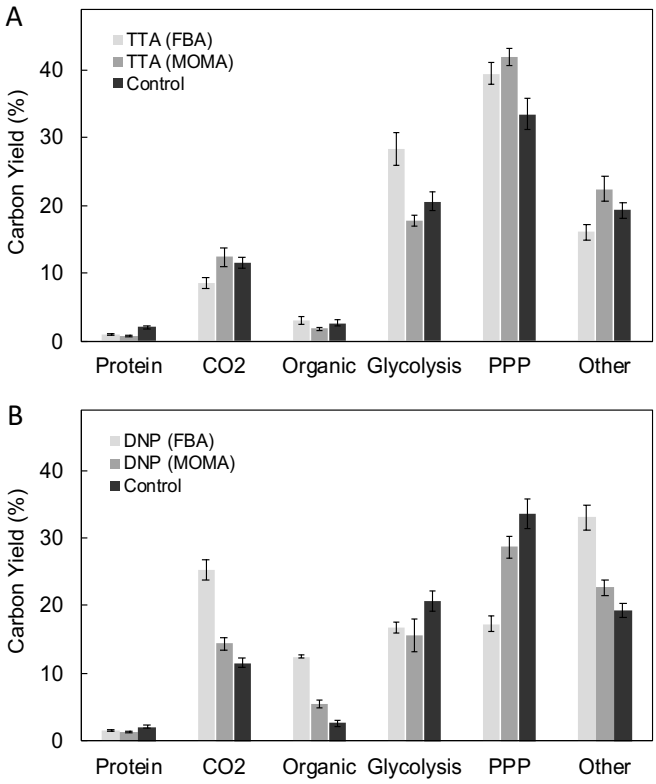

Figure 3.4: Mean carbon yield across an ensemble (N=100) estimated from FBA and MOMA simulations for (A) TTA; and (B) DNP. CFPS carbon yield was calculated for the duration of the reaction as a ratio of the concentration of carbon produced to the total concentration of carbon consumed. PPP denotes the Pentose Phosphate Pathway. Other includes purine, pyrimidine and chorismate metabolism

21% for MOMA, FBA, Control) and carbon dioxide (14%, 25%, 12% for MOMA, FBA, Control). In the TTA case, MOMA predicted a similar carbon distribution, except for a higher percentage of carbon yield in the pentose phophate pathway (42%, 39%, 33% for MOMA, FBA, Control). In addition, for the treatment group, the MOMA prediction of carbon yield for GFP protein synthesis (1.35% and 0.70% for DNP and TTA) was marginally lower than FBA (1.46% and 0.95% for DNP and TTA). Furthermore, MOMA predicted a reduction in the amount of carbon secreted in the form of organic metabolites (5% and 2% for DNP and

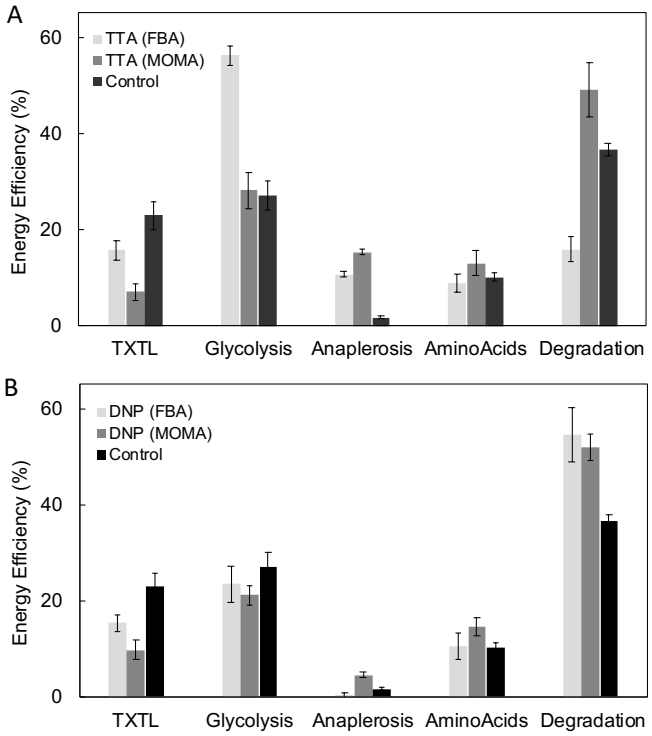

Figure 3.5: Mean energy efficiency across an ensemble (N=100) estimated from FBA and MOMA simulations for (A) TTA; and (B) DNP. Energy efficiency was calculated as a ratio for the entire duration of the reaction in terms of nucleotides triphosphates utilized for the corresponding category to ATP generation.

TTA) compared to FBA (12% and 3% for DNP and TTA). For some metabolites, the concentration profiles varied between the two models. For example, MOMA predicted higher accumulation of lactate compared to FBA for the TTA case, whereas the accumulation of acetate was lower for the DNP case. The differences observed in the concentration profiles of metabolites is likely driven by the MOMA objective to match the wild-type flux profile. In effect, MOMA attempted to consume metabolites at wild-type levels and partially secreted the difference when they could not be completely consumed. Taken together, MOMA predicted a carbon flux distribution closer to the wild type model compared to FBA for the treatment group.

CFPS energy efficiency was calculated as a ratio for the entire duration of the reaction in terms of nucleotides triphosphates utilized for the corresponding category to ATP generation. For both the treatment groups, MOMA predicted lower energy efficiency for TX-TL (10% and 7% for DNP and TTA) compared to FBA (15% for both DNP and TTA; Figure 3.5). This is consistent with the mathematical requirement that FBA maximizes the rate of translation for CFPS, therefore estimating higher energy utilization for GFP production. In addition, MOMA predicted higher energy utilization for anaplerosis and amino acid biosynthesis compared to FBA for both the DNP and TTA cases. Otherwise, there were no significant differences between the FBA and MOMA solutions for the DNP case. However, for the TTA case, MOMA predicted almost 55% energy wasted toward degradation, much higher than the FBA prediction of 16%. This means that FBA over-estimated the productivity of CFPS in the presence of TTA. MOMA also predicted a much lower energy utilization in glycolysis compared to FBA (23% and 56% for MOMA and FBA). Taken together, the simultaneous increase in energy utilization by amino acid biosynthesis and degradation pathways predicted by MOMA in the case of TTA shows a greater degree of inefficiency in the distribution of energy resources across the metabolic network.

3.4 Discussion

In this study, we showed that maximizing the rate of translation can be used as an objective function for genome-scale reconstructions of E. coli CFPS metabolism, consistent with previous studies [42]. However, the biological relevance of the selection of systemic optimality has always been questioned, especially for a system under perturbation. Based on published examples in the

literature, there exists no method for the immediate regulation of fluxes towards a specific biological objective following a perturbation. However, for CFPS systems, in silico models that describe metabolic perturbations have not been explored. Therefore, we used MOMA to analyze the performance of CFPS from a different perspective. We showed that, following a perturbation, cell-free systems follow the MOMA objective, with minimal readjustment in their flux distribution. In particular, we found that MOMA accurately predicted the overall production of mRNA and protein along with changes in metabolic behavior in the presence of the inhibitors. For our MOMA formulation, we calculated the wild type flux distribution by applying FBA to the control case and used this solution as an input. However, this input does not necessarily need to be an FBA solution. For instance, an experimentally determined wild-type flux distribution can be used instead. Finally, a comparison of the performance metrics for MOMA and FBA showed differences in the distribution of carbon flux and energy resources. The distribution of carbon yield and energy efficiency through the network for MOMA was closer to the wild-type distribution (or control group), since MOMA predicted fewer flux changes immediately after the addition of inhibitors. Analysis of energy efficiency under the MOMA objective showed an increase in the wastage of energy resources in the treatment groups through unnecessary reactions. This was also reflected by a decrease in the utilization of energy for transcription and translation. However, in spite of these differences in energy utilization, MOMA predictions were close to FBA estimations that assumed optimality of a specific metabolic function (which in our case was translation of GFP).

Our findings open several opportunities for the development of methods that aim to improve protein yields. One possibility is through the deletion of

enzymes that negatively affect CFPS. Cell-free extracts contain many enzymes that reduce the efficiency and productivity of CFPS, in addition to several essential enzymes that support TX-TL processes. For example, energy resources consumed for the degradation of nucleotides can otherwise be directed toward translation. Actually, recent CFPS studies have reported an improvement in protein yields by the deletion of enzymes that lead to the impairment of amino acid and nucleic acid degradation pathways. Therefore, such strategies that have been used for decades *in vivo* are only beginning to be used in CFPS [196, 2, 11]. Toward this, mathematical modeling strategies that integrate -omics data with constraint based models, as presented in this study, will reveal insights into the best strategies for optimizing CFPS through more efficient regulation of resources within the metabolic network. Moreover, using MOMA, the effects of inhibitors of particular proteins on CFPS can also be studied in a similar way. For example, these effects can be modeled by the addition of reactions that capture the effect of that specific inhibitor or by constraining the upper bounds of their fluxes to any defined fraction of the normal flux, depending on the extents of inhibition. Taken together, the implementation of MOMA in cell-free metabolic engineering provides an approach to analyse the effects of individual gene deletions on CFPS, identify possible negative effectors to improve CFPS yields, and analyse genetic interactions in synthetic circuits.

CHAPTER 4

COMPUTATIONAL FRAMEWORK FOR MULTISTEP METABOLIC PATHWAY DESIGN

Abstract

[1]*In silico* tools are indispensable for generating novel hypotheses and efficiently exploring alternatives in *de novo* metabolic pathway design. However, while many computational frameworks have been proposed for retrobiosynthesis, very few successful examples of algorithm-guided xenobiotic biochemical retrosynthesis have been reported in the literature. Interestingly, deep learning has significantly improved the quality of synthesis and retrosynthesis in organic chemistry applications. Inspired by this recent progress in computational organic chemistry, we explored the idea of combining deep learning of biochemical transformations with the traditional retrobiosynthetic workflow, in the hope of producing better *in silico* synthetic metabolic pathway designs. To develop our computational biosynthetic pathway design framework, we assembled metabolic reaction and enzymatic template data from public databases. A data augmentation procedure, adapted from literature, was carried out to enrich the assembled reaction dataset with artificial metabolic reactions generated by enzymatic reaction templates. Two neural network-based pathway ranking models were trained as binary classifiers, by outputting a scalar quantifying the likelihood of a 1-step/2-step pathway being plausible, to distinguish assembled reactions from artificial counterparts. This work discusses the neural network based 1-step pathway ranking model and how it was used to design a novel

[1]Adapted with permission from Zhang, Z.; Vadhin, S.; Dammalapati, S.; Varner, J. D. Computational framework for multistep metabolic pathway design. In preparation.

biosynthetic pathway for naloxone production.

4.1 Introduction

Manufacturing molecules through metabolic engineering is key to a sustainable society by converting renewable and widely-available starting materials into high-value products under milder conditions [69]. There are many successful metabolic engineering projects targeting the commercial production of bulk chemicals or pharmaceutical ingredients, such as the production of 1,4-butanediol (BDO) in E.coli [1] and the production of the antimalarial drug precursor artemisinic acid in engineered yeast [70]. However, with current technologies, the development of a new industrial bio-process typically costs several years and millions of dollars [71]. Towards this, *in silico* biosynthetic pathway design tools are indispensable for accelerating the design-build-test-learn cycle in metabolic engineering as they could help efficiently explore the whole design space and generate novel synthetic pathways [71, 69].

Given a desired compound of interest to produce, a retrobiosynthesis tool works backward to identify a series of enzymatic transformations that can construct the target from some available cellular metabolites or a biochemical feedstock [2, 72]. Computer-assisted retrobiosynthesis tools play an important role in helping scientists explore the whole design space for generating novel `de novo` biosynthetic pathways [2]. In analogy to chemical retrosynthesis, a general workflow for computational retrobiosynthesis consists of two parts - network generation and pathway pruning/ranking (Fig. 4.1) [1, 73, 2, 74]. The target is used as the seed for metabolic network generation in the pipeline,

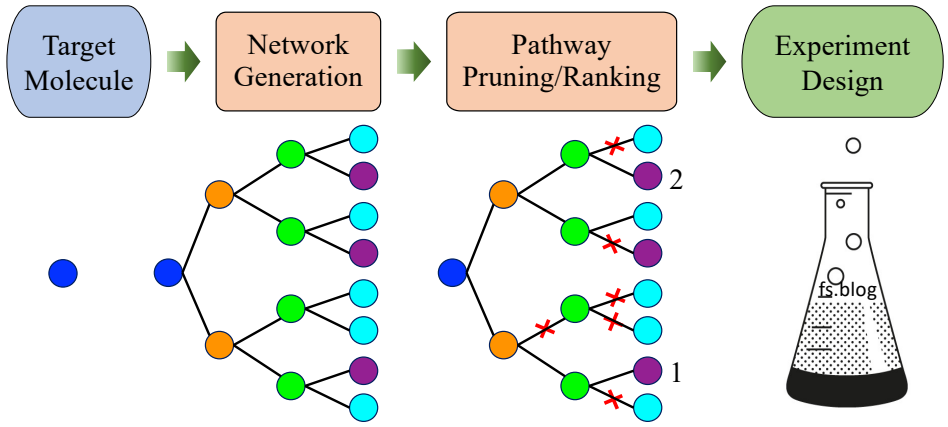

Figure 4.1: A general retrobiosynthesis workflow consists of two parts. Blue: target compound; orange/green/cyan: generated compound; purple: available compound. (adapted from [1, 2]).

which adopts some candidate selection algorithm to choose promising intermediates and applies generalized enzymatic reaction rules on those intermediate candidates for network expansion. This expansion continues iteratively until it reaches some stopping criteria such as a set of designated starting compounds, maximum depth of a network, or maximum allocated computational time. Then, all generated reactions and pathways would be evaluated by some qualitative and/or quantitative metrics. Reactions and pathways that can not meet the set thresholds of those metrics would be removed from the generated network. All potential remaining pathways are then extracted for qualitative and quantitative analyses. Such analyses may incorporate domain-specific knowledge, like structural similarity, enzyme availability, theoretical yield in the context of a metabolic model, and thermodynamic constraints, to analyze the plausibility of each pathway. In the pathway ranking step, a scoring function is designed to quantify the plausibility of all remaining pathways and give a list of ranked pathway candidates for guiding experimental design.

Some systems have already been developed for retrobiosynthesis. Developed by Genomatica, SimPheny was part of their Biopathway Predictor framework to engineer *E.coli* for the first direct biocatalytic synthesis of 1,4-Butanediol (BDO) from renewable carbohydrate feedstocks [1]. Metabolic models were used to analyze the performance of each proposed pathway, such as theoretical yield. Built upon public database Kyoto Encyclopedia of Genes and Genomes (KEGG) [75, 76, 77, 78, 79], the BNICE framework generated a publicly available biochemistry database ATLAS and claimed to be able to perform retrobiosynthesis but is not publicly or commercially available [80, 81, 82, 83, 84, 85]. PathMiner provides a heuristic search method for extracting metabolic routes from a network, although it does not involve network generation [86]. KEGG has a built-in pathway prediction server, PathPred, using SIMCOMP program and Reaction center-Different region-Matched region (RDM) pattern representation of reaction rules [87, 88]. RetroPath proposed a data-driven approach to automatically extract reaction rules in SMARTS from biochemical reactions and incorporated enzyme sequence consistency and compound similarity into pathway ranking [89]. They then validated this framework by using RetroPath to explore nine million possible enzyme combinations that could lead to the production of flavonoid pinocembrin and narrow it down to 12 candidates which were then used for experimental designs. Four out of those 12 top-ranked enzyme combinations proved to be able to produce the target compound with significant yields in the *E. coli* chassis. RetroPathRL adopted Monte Carlo Tree Search reinforcement learning method to explore biosynthetic space for longer pathway design [90]. Despite the availability of these tools, the production of 1,4-butanediol (BDO) in *E. coli* remains to be the only example of a retrosynthesis algorithm being used in a *de novo* pathway design for the commercial

production of a xenobiotic compound [72].

Due to its ability to learn knowledge representation from large amounts of data, deep learning is becoming popular in promoting chemical (retro)synthesis [91, 72]. Wei and coworkers combined convolutional neural network generated molecular fingerprints with a neural network for reaction type prediction to predict likely chemical reaction products given a set of reagents and reactants [92, 93]. Coley and coworkers introduced neural network based scoring function into reaction template based forward enumeration framework for predicting organic reaction outcomes and achieved 71.8% accuracy in rank 1 candidate [94]. The accuracy was improved to 85.6% later by a template-free graph-convolutional neural network model [95, 96]. Segler and coworkers designed a chemical retrosynthesis system (3N-MCTS) with Monte Carlo tree search and three different neural networks and achieved faster and better performance than all traditional computer-aided search methods [73].

Thereforem, inspired by the recent progress in computational organic chemistry, we explored the idea of combining deep learning of biochemical transformations with the traditional retrobiosynthetic workflow, in hope of producing better *in silico* designs for synthetic metabolic pathways.

4.2 Results

We adopted backward template-based enumeration for network expansion, and used deep learning based ranking models for network pruning and candidate ranking. To develop our computational biosynthetic pathway design framework, we assembled metabolic reaction and enzymatic template data from pub-

lic databases. A data augmentation procedure, adapted from literature, was carried out to enrich the assembled reaction dataset with artificial metabolic reactions generated by enzymatic reaction templates [97]. Two neural network-based pathway ranking models were trained as binary classifiers, by outputting a scalar quantifying the likelihood of a 1-step or 2-step pathway being plausible, to distinguish assembled reactions from artificial counterparts. Combining these two models with enzymatic templates, we built a multistep retrobiosynthesis pipeline and validated it by reproducing some natural and non-natural pathways computationally. However, this work is limited to the results and discussion of the neural network based 1-step pathway ranking model. The model was first validated with the glycolysis and BDO pathway, and was finally used to design a pathway for naloxone production.

4.2.1 Neural network based 1-step pathway ranking model (NN1PR)

A multilayer perceptron (MLP) model, consisting of 1 hidden layer with 256 neurons and a dropout layer, was trained as a binary classifier to distinguish assembled reactions from their generated counterparts (Fig. 4.2). The input to the network was the concatenation of ECFPs of target molecule and precursor candidate. The final scalar output from the trained model was then used as a quantitative likelihood measure of a reaction being plausible. For comparison, we developed a baseline model based on Tanimoto similarity [98]. The score for each target-precursor pair was given by the Tanimoto similarity between the target and proposed precursor. Both the baseline model and NN1PR model were

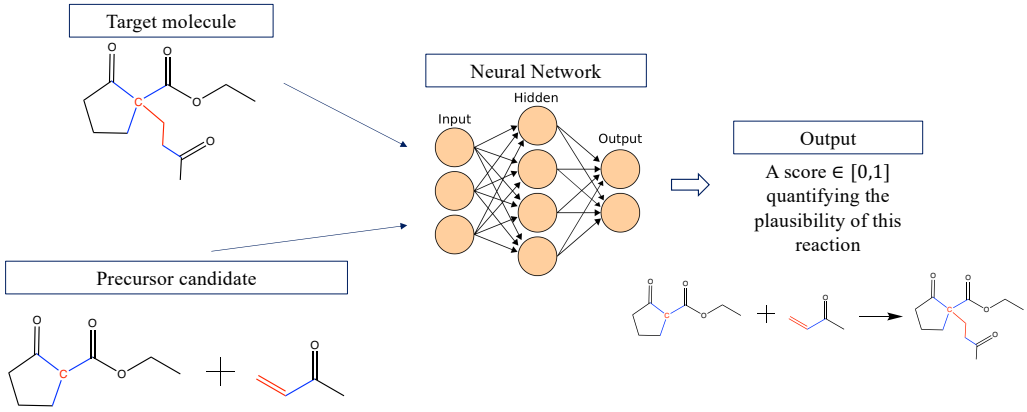

Figure 4.2: Schematic of the neural network based 1-step pathway ranking model (NN1PR).

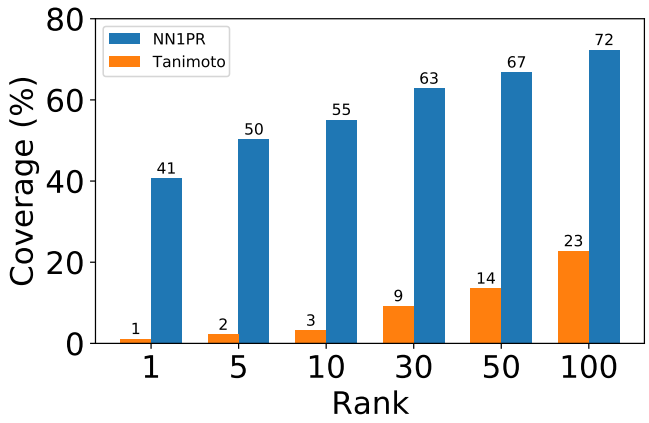

Figure 4.3: Performance comparison between NN1PR and its baseline on testing data.

used to rank positive reactions, which were one-step metabolic reactions from KEGG, among their corresponding negative counterparts generated through a data augmentation procedure. About 10% assembled KEGG reaction data were reserved for performance comparison. Assembled data were randomly shuffled before training/testing split to guarantee a fair coverage in both training and testing datasets. NN1PR outperformed its baseline model by a significant

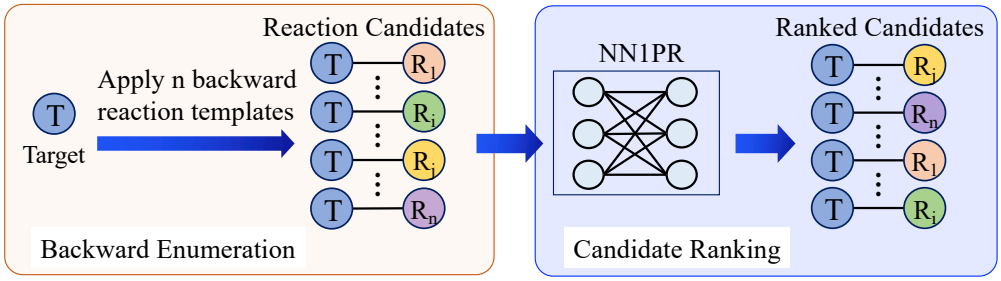

Figure 4.4: Schematic showing the one-step retrosynthesis framework that combines template based backward enumeration with neural network based 1-step pathway ranking (NN1PR) model.

margin (Fig.4.3). About 55% samples were ranked among top 10 by NN1PR, comparing to only 3% by the Tanimoto baseline. Top-100 candidates by NN1PR covered about 72% samples, which demonstrated a big room for improvement.

4.2.2 One-step pathway design framework with NN1PR

Combining template based backward enumeration with NN1PR, we built a pipeline for one-step retrosynthetic reaction design (Fig.4.4) [97]. In the backward enumeration, selected template set was applied on the given target for retrobiosynthesis to generate precursor candidates (R_1, R_i, R_j, R_n in Fig.4.4). Together with the target T, they formed target-precursor pairs ($T-R_1, T-R_i, T-R_j, T-R_n$ in Fig.4.4), that were then fed to NN1PR which would assign a scalar score in unit range to each pair. All candidates were then ranked based on their scores. For the purpose of multistep design, this one-step pathway design framework can be applied to design a pathway step-by-step or analyze a pathway proposed by human experts.

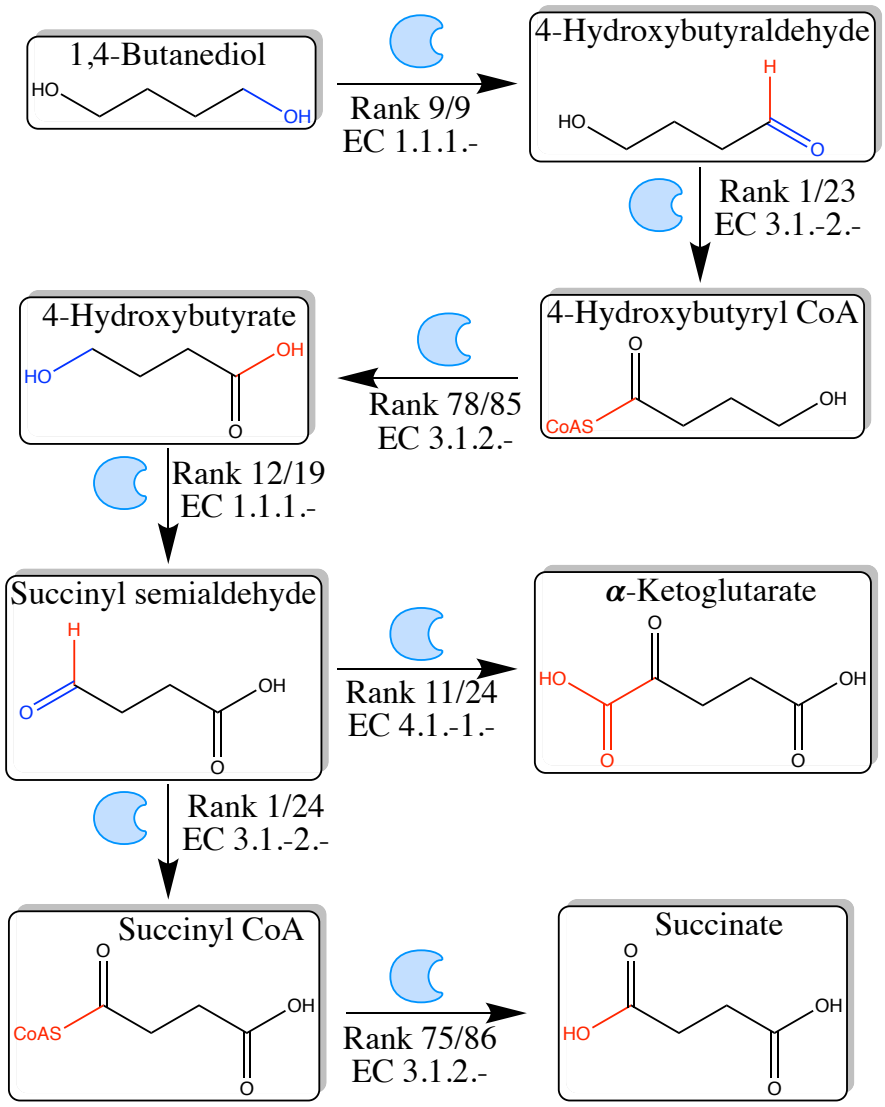

Figure 4.5: Ranks assigned by NN1PR to BDO pathway reported in [1] in a backward manner. Each step was annotated with a rank and corresponding EC number. Only the BNICE rule set (116 templates in total) was used in backward enumeration step to produce results here.

Computational validation with BDO pathway

We demonstrated designing a multistep pathway with this one-step pathway design framework by reproducing the BDO pathway reported by Yim et. et. [1]. Given the target, 1,4-Butanediol (BDO), for biosynthesis, we applied the

framework with BNICE rule set [99, 100] and found the desired precursor 4-Hydroxybutyraldehyde with EC $1.1.1.-$ at rank 9 out of 9 candidates (Fig.4.5). For the next step, the pipeline expanded on 4-Hydroxybutyraldehyde to generate precursors and found 4-Hydroxybutyryl CoA with EC $3.1. - 2.-$ and was ranked as 1 out of 23 alternatives. The network expansion went on for another 4 times to completely reproduce the reported BDO pathway. Note that BDO pathway is a non-natural pathway designed by human experts. Being able to reproduce BDO pathway demonstrated that this framework, although developed based on natural metabolic data, is capable of generating non-natural ones with pretty high ranks.

The performance of our framework remained quite good even if way more templates were included in candidate generation. The BNICE rule set only contained 116 rules which were quite small and limited in coverage. To see how our framework would perform as the size of the template set scales up, we added another 234384 templates and reran our pipeline on reproducing BOD pathway. The number of templates jumped up by about 2021 times, from 116 to 234500, and number of candidates increased by 71 times on average, but the absolute rank was at most 4 times worse (rank of 1,4-butanediol to 4-hydroxybutyraldehyde increased from 9 to 32). This comparison demonstrated that this framework performed reasonably well as the size of template set scaled up.

Computational validation with glycolysis pathway

We also demonstrated the application of this one-step pathway design framework on designing natural pathways by reproducing the glycolysis pathway.

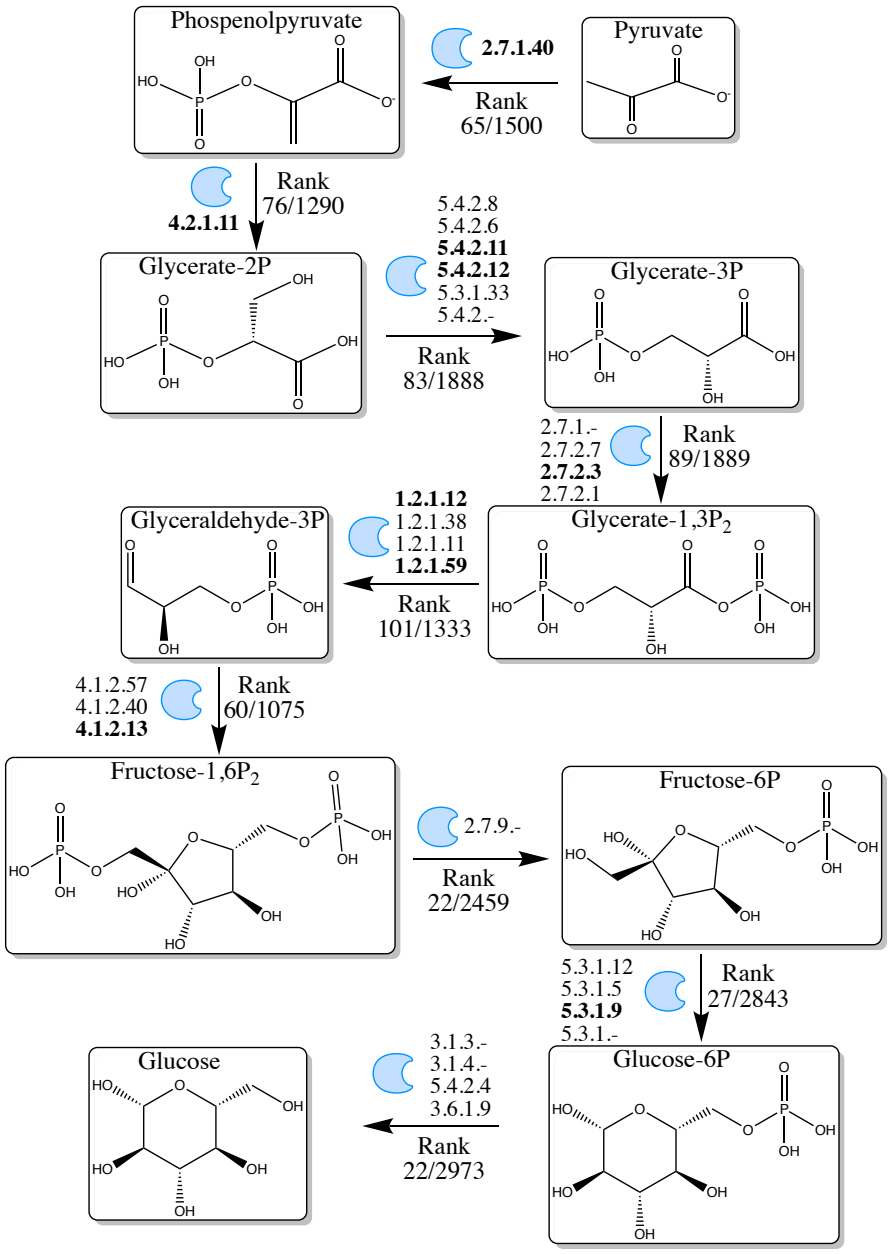

Figure 4.6: Ranks assigned by NN1PR for glycolysis pathway in a backward manner. 234501 templates were used in backward enumeration step to produce results here. Ranks here were the best ranks if there were alternatives leading to the same target. ECs were ECs associated with the corresponding rank, and ECs in bold were the ECs recorded in KEGG.

Given the target, Pyruvate, for biosynthesis, we applied the framework with BNICE and Retro rule set [101] and found the desired precursor phosphoenolpyruvate with EC 2.7.1.40, as reported in KEGG, at rank 65 out of 1500 candidates (Fig.4.6). The network expansion went on repeatedly for another 8 times to completely reproduce the well-known glycolysis pathway. For 7 steps in this reproduction, the recorded ECs in KEGG were included in the best ranked candidates found by the pipeline. While for the other two steps (fructose 1,6-bisphosphate to fructose 6-phosphate and glucose 6-phosphate to glucose), the recorded ECs were included in other alternative candidates.

Computational evaluation of naloxone producing pathway

Presently, there is no known complete biosynthetic pathway for the production of the opioid antagonist naloxone. Toward this end, we used the one-step pathway design framework to design a novel biosynthetic pathway that produces naloxone with morphine as the precursor (4.7). Some steps had multiple matched candidates, while others only had one. Only the best ranks and corresponding EC numbers are shown here. The ranks evaluated by the pipeline could provide a quantitative understanding of the difficulty in achieving each step, and the corresponding EC numbers help in the identification of pathway enzymes.

4.3 Discussion

The pipeline proposed here provides a way of leveraging big-data in metabolism to facilitate metabolic pathway design by combining template-

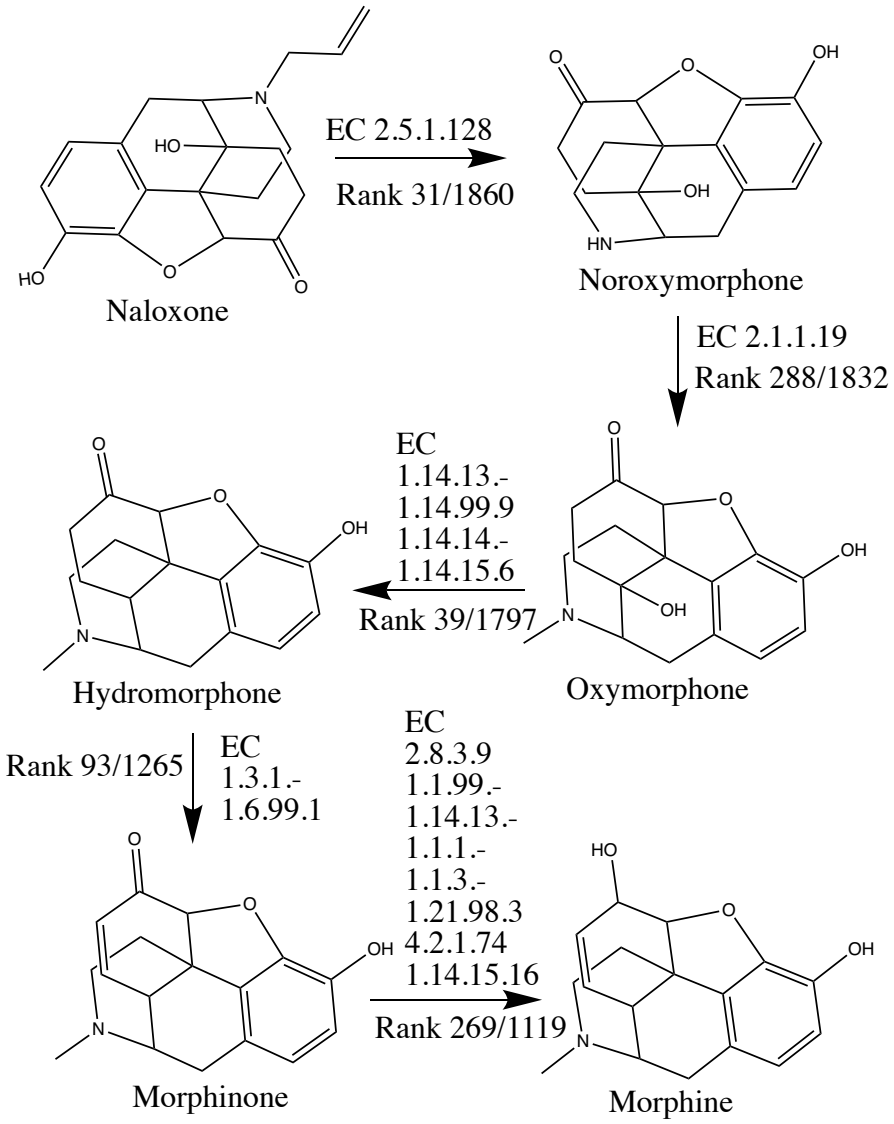

Figure 4.7: Ranks assigned by NN1PR for the naloxone production pathway in a backward manner. Ranks here were the best ranks if there were alternatives leading to the same target. ECs were ECs associated with the corresponding rank, and ECs in bold were the ECs recorded in KEGG.

based backward enumeration with deep learning based ranking models. There are some limitations with the framework and many further developments could be pursued to improve the efficiency, capability, and bio-feasibility of this pipeline.

The capability depends on the quality of templates. The performance of this pipeline depends heavily on the quality of its template set, but there is no standard way of quantifying the quality of such a dataset. Also, pre-selecting a template set for designing an unknown pathway is challenging. It would be better if the template set was not assembled from literature, like in our current pipeline. Instead, inserting a module that could extract templates from metabolic reactions could truly automate the pipeline further. Although this practice has been adopted for chemical synthesis [94, 97, 73], template extraction is much harder in biochemistry because many reactions in public databases are usually not atom balanced. Falon et al. reported extracting enzymatic templates but their methodology was too constrained, resulting in very conservative templates [100].

Runtime as more templates involved. Runtime is yet another concern with this framework. The capability of a pipeline could be significantly improved if there is a flexible template extraction module, but the runtime would be a serious issue as more templates become available. One potentially effective way of reducing running time is to develop a template selection model to screen all templates and select only the most suitable ones to apply on a target. Template selection models should guarantee faster screening for each template than running it on the target. Otherwise, the runtime benefit may not be great. Segler and Waller developed such a model using neural networks, although details

(like the runtime) of this model are unknown [73]. However, generally speaking, neural network models are quite fast with predictions, which should make them a good choice for a template selection model [102].

Better extension to multiple-step design. The demonstrated application on multiple-step design only utilized a one-step ranking model which did not take more available information into consideration. For better performance, neural network based n-step pathway ranking (NNnPR) models could be developed to rank n-step pathways. Knowledge from metabolism should be incorporated to rule out some popular and/or common precursor candidates at early stages, so that the pipeline could focus more on exploring the unknown world. The number of precursor candidates generated for each given target could be reduced so as to improve the efficiency of the pipeline. Many candidates differ from each other only by a trivial co-substrate that does not impact further backward exploration biologically. Henceforth, the pipeline should be improved by merging those candidates together to focus backward enumeration on main substrates. A handy way to achieve this design is to create several knowledge-based lists of equivalent substrates and use these lists to rule out equivalent candidates in network generation. Monte-Carlo tree search could also be explored for generating longer pathways if available metabolites are also given [90].

Other deep learning architectures promising better performance. A promising way of getting rid of both capability and runtime issues is to adopt template-free methods [95, 96]. Instead of using templates, this type of framework tries to predict what bonds and atoms in a target are going to change in a reaction, and then enumerate all possible outcomes. Jin and Coley developed a graph-convolutional neural network model that had comparable performance

with template-based models [96]. Transfer learning has been proved to be an effective way of knowledge learning when large datasets are not available [103]. Considering the published literature on chemical synthesis, it is worthwhile exploring transfer learning for boosting biochemical synthesis. One possibility is to adapt learnt fingerprint models using convolutional networks to generate better fingerprints for biochemical compounds [104, 105, 92].

Incorporating more knowledge and addressing more biological concerns. In this work, only substructure information encoding as fingerprints was incorporated for ranking. However, there are more quantitative structure-attribute relations that should be considered. More importantly, metabolic reactions happen inside a chassis that contains complicated biological environment, including pH and other substrates. All of these factors do more or less impact the progress of each reaction, but have not been incorporated into the pipeline. To make this kind of a model more helpful in metabolic pathway design for human experts, more information on enzyme design and genome sequence should be considered [69]. Models for evaluating the synthesizability and toxicity of a molecule can be incorporated in network generation to prune out some undesirable molecules [106, 107]. Thermodynamic constraints, like Gibbs free energy, should be considered for evaluating a reaction candidate [1]. Pathway yield and compatibility in a chassis can be analyzed for pathway selection if a chassis is specified [89].

4.4 Conclusions

Combining backward template-based enumeration with neural network based ranking models, we developed a new framework for computer-aided metabolic pathway retrosynthesis. The deep learning ranking model, trained on KEGG metabolic reactions, outperformed the Tanimoto similarity-based method by a significant margin. The framework was demonstrated to be suitable for multistep pathway design by reproducing one natural and one non-natural pathway. In addition, the framework was used to design a pathway for Naloxone production. However, the pipeline was not without its flaws. Several limitations, especially dependence on the quality of templates and runtime concern, were discussed. While the experimental validation of our prediction with the glycolysis pathway is still work in progress, many directions are under consideration for improving the accuracy, capability, and efficiency of our multistep computational pipeline. In general, this work demonstrated the applicability and advantage of introducing deep learning into metabolic pathway design.

4.5 Materials and Methods

In the work, we developed a new framework for *de novo* metabolic pathway design by leveraging tools developed by cheminformaticians and the power endowed by deep learning. Given a target compound of interest for production through metabolic engineering, one has to first design a metabolic pathway that connects the target to some available metabolite(s) in chassis before any further experiments could be carried out. The basic idea here was to use backward reaction template-based enumeration to generate possible precursor

candidates and use neural network based ranking models to pick out high-rank candidates for next step expansion. Deep-learning based ranking models were trained on KEGG metabolic reactions to learn the underlying connections between metabolic reactants and products. Then the probability score generated by the trained neural network-based pathway ranking model (NN1PR) given any reactant-product pair was used for pruning and ranking all candidates. Given a target compound, for the first-step reaction, some backward reaction templates were applied to generate potential candidates that were represented as three light orange circles in the figure. Then NN1PR assigned each candidate a score in unit interval, which was used to rank those candidates. Depending on computational time requirement or user's settings, only a portion of high-ranked candidates would be chosen for designing the second-step reaction. In the illustrating schematic, only the first 2 candidates were chosen for next step expansion which yielded 6 possible candidates as represented by green circles. And again, NN1PR was used to prune some candidates (2 in the example) and rank the rest (ranks 1 to 4 in the figure). The process went on and on until finishing preset number of steps. Generated pathways would then be analyzed to see if any could lead to known cellular metabolites. If that is the case, at least one synthetic pathway can be reconstructed through backtracking.

Dataset assembly.

The backward reaction templates and metabolic reactions were assembled from literature [99, 101, 100] and public databases [108, 109, 80].

Metabolic reaction dataset.

KEGG (Kyoto Encyclopedia of Genes and Genomes) is a computerized resource for understanding high-level functions and utilities of the biological system [108, 75, 76, 110, 77, 111, 112, 113, 114, 115, 78, 116, 117, 118, 119, 79] It is a collection of databases dealing with many aspects of information in biology, including biological pathways and chemical substances [108, 120]. We assembled 11475 reactions from KEGG Reaction Database and 323 pathways from KEGG Module Database. Each reaction has its unique KEGG reaction ID, corresponding enzyme as EC number, and compound names as KEGG compound ID. The size of this dataset was not significant, but our purpose here was to build a working pipeline as a proof-of-concept study. We expect that incorporation of additional training data will improve the performance of the pipeline developed here. A better way to resolve this issue would be to assemble additional data directly from various databases like ChEBI [121], XTMS [122], Rhea [123, 124], and MINEs.

Backward reaction templates dataset.

Building a pipeline for reaction template extraction was not a trivial task. Therefore, we assembled the backward reaction templates dataset from literature. The Faulon group published multiple sets of reaction rules based on metabolic reaction database MetaNetX v3.0 [100, 125, 126, 127, 128]. We used their reaction rules dataset which handles hydrogen explicit, with a template diameter from 2 to 16 [101]. The dataset had around 350k templates, including both forward and backward ones. Among them, there were 234k backward templates, referred to as 'RetroRule' rule set. We also used a dataset reported by Henry [99] consisting

of 116 backward rules referred to as the 'BNICE' rule set in this paper.

Data preprocessing.

Querying KEGG for SMILES of each compound in metabolic reaction dataset.

As mentioned in section *Dataset assembly*, compounds in the metabolic reaction dataset were in KEGG compound IDs which follow KEGG nomenclature but do not contain any cheminformatic information directly. To get SMILES of each compound, we queried KEGG for its compound list which had 18749 entries in total. We set up a pipeline in Python to automatically query KEGG for the SMILES string of each KEGG compound listed.

Process reactions into mono-product reactions.

The backward reaction templates dataset we assembled were mono-product templates. In order to use this template dataset, we had to process our metabolic reactions dataset into mono-product reactions. For each multiple-product reaction, we split it into multiple children reactions sharing the same reactants but each just inherited one product from their parent reaction. With this processing, the original 11475 reactions were expanded into 21848 single-product reactions with 7105 unique products.

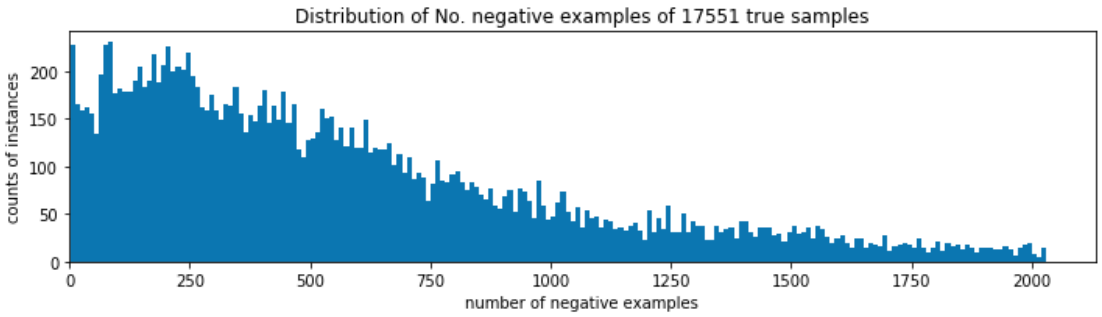

Figure 4.8: Distribution of number of negative examples of 17551 compounds. The number varied in a wide range from 0 to over 2000. Totally, there were 113336477 negative reactions, about 760 for each positive one on average.

Data augmentation.

In our metabolic reaction dataset, there were only reactions that already exist in nature (also called positive reactions in this paper). While to build a machine learning model, one also needs negative metabolic reactions which are not favored or do not exist in nature so as to train a model to distinguish positive ones from negative counterparts. To create negative metabolic reactions, we adapted a data augmentation procedure reported by Coley et. al. [94]. We applied all reaction templates - both forward and backward - on all unique SMILES strings, totally 18058 cases, to generate hypothetical precursors for each compound. This augmentation process - applying hundreds of thousands of reaction templates on thousands of targets one-by-one - was very time consuming, taking several hours in our implementation. For each mono-product reaction, the set of hypothetical precursors of its product was used to generated negative reactions by simply linking each hypothetical precursor with the product by '>>' to form a valid reaction SMILES. The original precursors were removed from hypothetical precursors beforehand. To guarantee that all hy-

pothetical precursors were hypothetical but not found in nature, a strict check was carried out to remove all positive cases from generated counterparts. There were 200 compounds that couldn't be modified by any templates, and most of them were small molecules. After this augmentation step, some compounds had more than 2000 augmented precursor sets, but it was about 760 cases for each compound on average (Fig.4.8). The statistics here signaled the importance of a well-performed ranking model which would be tasked to rank the positive ones as high as possible among hundreds of candidates. Overall, we got $113M$ negative reactions.

Reaction fingerprints.

Most of existing machine learning models require input data to be in tensor format [129, 102, 130], so our datasets in SMILES and SMARTS had to be vectorized for using machine learning model. As briefly mentioned in Section Manipulation molecules in cheminformatics, molecular fingerprints [131], such as Extended-connectivity fingerprints (ECFPs) [132] that were derived from the Morgan Algorithm [133] and learned fingerprints [104, 134], have been widely used to convert molecules in SMILES into vectors. Although some researchers had developed machine learning models that used SMILES/SMARTS sequences as input directly, such as in Liu's work [135]. Reaction fingerprints can be derived from the concatenation of reactant and product fingerprints [92], the difference between reactant and product fingerprints [136], or many other ways [137, 138]. In this work, we used an implementation of ECPFs in RDKit to generate molecular fingerprints, and represented each reaction as the concatenation of its reactant and product fingerprints. Each group of reactants or

products were represented as a 512-bit vector and henceforth each reaction was a 1024-bit vector. Each bit represented a feature and took value 1 if corresponding feature existed, otherwise 0.

Baseline model.

We developed a baseline model on top of Tanimoto similarity metric as mentioned in section *Pathway ranking*. Two backward rule sets, RetroPath and BNICE, were applied on product(s) of each positive reaction to generate negative derivatives, and then Tanimoto scores were calculated with each reaction represented as a 1024-bit vector, as discussed in *Reaction fingerprints*. The final ranks of positive reactions were solely based on Tanimoto scores.

Deep learning model.

Neural network based 1-step pathway ranking model (NN1PR).

The dataset had $113M$ negative examples but only 21848 positive counterparts, which was very unbalanced. For balanced training, we randomly shuffled negative examples and used the top 30% of these for training. The deep learning model was a feedforward neural network consisting of one hidden dense layer with 'relu' as the activation function [102] (Fig.4.9). The hidden layer had 256 neurons, followed by a dropout layer to regulate overfitting. The output layer was a dense layer with 1 neuron using 'sigmoid' as the activation function, so as to represent the probability of a reaction being a positive one. The input layer was a trivial one that didn't have any parameters. The optimizer was Adam,

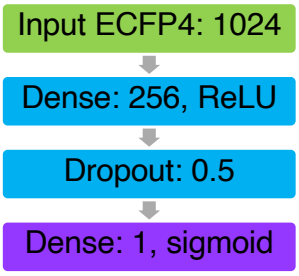

Figure 4.9: Architecture of the neural network based one-step ranking model (NN1PR). The model had 1 hidden layer and a dropout layer. The output layer was a dense layer with 'sigmoid' as the activation function. The input layer was a trivial one that didn't have any parameters.

which is a stochastic gradient descent method that is based on adaptive estimation of first-order and second-order moments, with a learning rate of 0.001 [139]. The loss function was the built-in 'binary_crossentropy' in TensorFlow. During the training, we also monitored classification accuracy. The model was trained for 50 epochs with a batch size of 128. The performance based on either loss or classification accuracy improved significantly in the first 10 epochs and then improved very slowly afterwards. Based on observation, 30 epochs was a good place to stop training. We observed that the training curves would vary a bit at the beginning in different runs, but all reached similar final performance. The difference could be explained by the random initialization of model parameters.

NN1PR was trained as a binary classifier, so we tested its capability of classifying reactions on reserved positive reactions and other unused negative reactions. The probability distributions showed that NN1PR was able to assign low scores to the vast majority of negative reactions while most positive ones got very high scores. Both the baseline model and the machine learning model

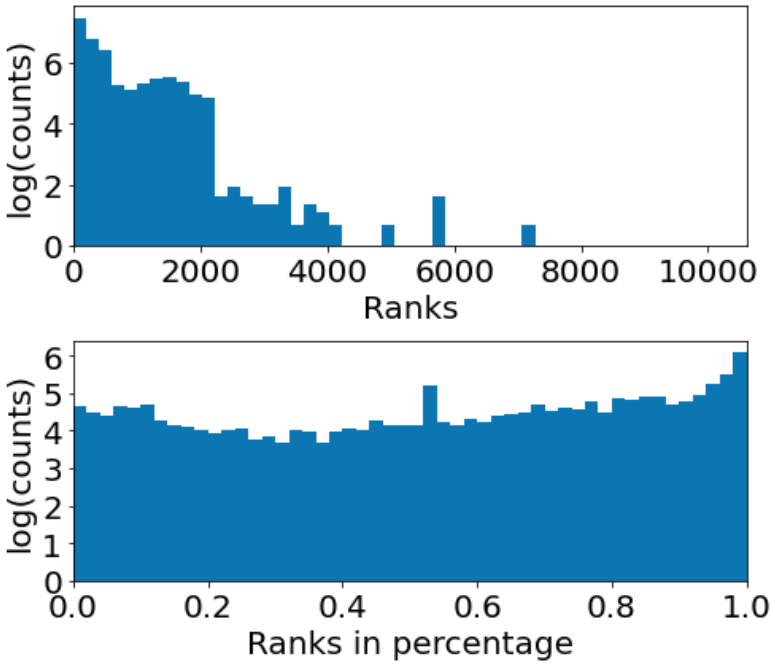

Figure 4.10: Performance of baseline model on ranking testing samples. Top: the distribution of ranks of 4796 testing samples by the baseline. Bottom: the distribution of ranks in percentage of all testing samples by the baseline.

were then used to ranking positive reactions together with negative reactions. Each positive reaction was mixed with its negative counterparts and two models were used to rank all reactions. For machine learning model, most cases were rank within 700 (Fig.4.11), and almost all positive reactions had ranks within 2000. While for the baseline model, some reactions were ranked higher than 10000 (Fig.4.10). Machine learning model outperformed baseline model by a significant margin (Fig.4.3). More than 50% samples were ranked in top-10 by the deep learning model and more than 70% were within rank 100. While for the baseline, only about 23% were within top-100.

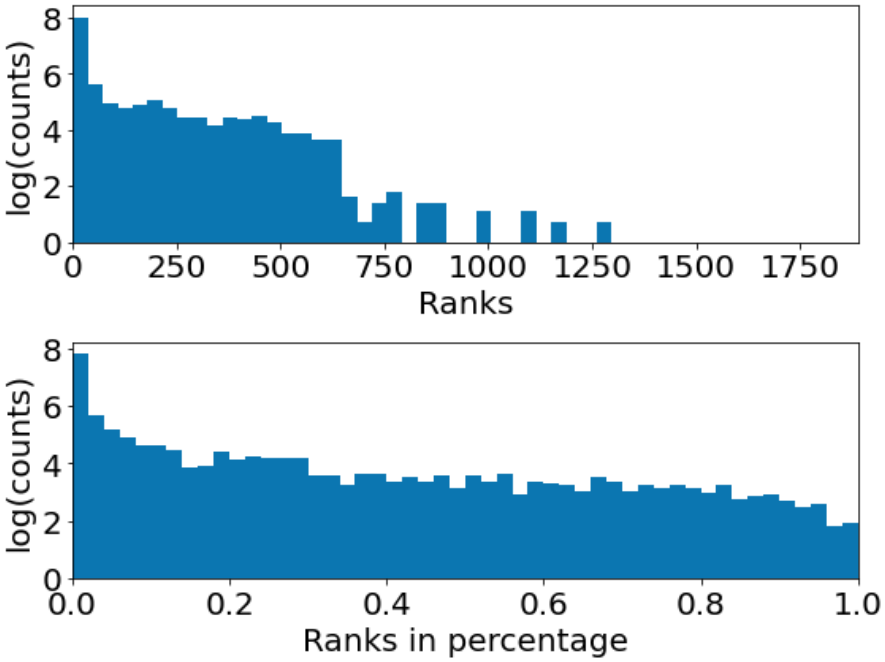

Figure 4.11: Performance of NN1PR on ranking testing samples. Top: the distribution of ranks of 4796 testing samples by NN1PR. Bottom: the distribution of ranks in percentage of all testing samples by NN1PR.

Computational tools.

All scripts were written in Python. RDKit was used for manipulating molecules and reactions and accomplishing various cheminformatics calculations [140]. Keras [141, 129] using the Tensorflow 2 [142, 143] backend was used for building the machine learning models.

CHAPTER 5

MODELING THE EXPRESSION OF MORPHINE DEHYDROGENASE IN AN E. COLI CELL-FREE SYSTEM

Abstract

Opioids are a class of drugs highly valued for their potent analgesic properties; however, their misuse can lead to addiction, overdose incidents, and death. Although the opioid antagonist naloxone has been used in emergency medicine for over 50 years to reverse the effects of an overdose, access to this life-saving antidote is still limited due to its high cost and restricted availability. To address this issue, we propose a novel biosynthetic pathway for the production of naloxone, using morphine as a precursor. To experimentally validate the first step of the proposed pathway, we produced morphine dehydrogenase - an enzyme that catalyzes the oxidation of morphine to morphinone - by cell-free protein synthesis (CFPS), taking advantage of the speed of CFPS compared to cell-based culture. In this work, we formulated a mathematical model to simulate the sigma factor 70 induced expression of morphine dehydrogenase. We used experimental protein concentration data to estimate unknown model parameters by minimizing the difference between simulated and experimentally measured protein concentrations. Then, we used global sensitivity analysis for a detailed insight into the influence of individual model parameters on the expression dynamics of the system. Taken together, we have developed an effective model for the expression of an enzyme that could serve as the first enzyme in a novel biosynthetic pathway for the production of naloxone. Ultimately, this model can be adapted for the expression of other pathway enzymes as well.

5.1 Introduction

Opioids, including the illegal drug heroin, are a class of highly potent drugs known for their pain-relieving properties. Opioids work by blocking pain signals between the brain and body, which produces a feeling of relaxation. However, due to its highly addictive nature, opioid usage often leads to abuse, overdose and eventually, death. According to the Centers for Disease Control and Prevention (CDC), nearly 500,000 Americans died from an opioid-overdose between 1999–2019, including prescription opioids and illicit drugs [144]. Since its approval in 1971, the opiate antagonist naloxone, commonly sold under the brand name Narcan®, is used in emergency situations to reverse the effects caused by overdoses of heroin, morphine or other opioids. However, despite an increase in naloxone prescription in recent years, CDC reports that not enough naloxone is being dispensed in many areas of the country that need it the most [145]. The recent price increase of the naloxone auto-injector, Evzio, marked by its market exclusivity, further restricts access to this life-saving medication [146]; a two-dose Evzio package priced at $690 in 2014 is $4,500 today, which is a price increase of more than 500% [147]. Therefore, to address such limitations, accessible point-of-care manufacturing of naloxone is needed considering the severity of the opioid crisis. Toward this need, we propose the development of a biosynthetic route for naloxone production.

Naloxone is usually semisynthesized from naturally derived opiates, such as morphine, thebaine, or oripavine, but there is currently no known fully biosynthetic pathway for naloxone production [148]. Towards this opportunity, we designed a biosynthetic pathway *in silico* for the production of naloxone (Figure 4.7, and predicted various enzymes involved in the biotransformation of

morphine to naloxone; however, the experimental validation of this pathway is still underway. We chose morphine as our starting compound because it is the major degradation product of heroin in the human body [149]. In this work, we focused on morphine dehydrogenase (*mdh*), an enzyme which catalyzes the conversion of morphine to morphinone, that could serve as the first step towards a novel opioid detoxification pathway.

In order to support experimental efforts and model guided optimization, we developed a mathematical model based on the work of Adhikari and coworkers [52], to simulate expression of *mdh* in a cell-free system. Cell-free systems have undergone several advancements since the mid 1990s, and ever since its introduction, CFPS has been used to formulate the central dogma, probe transcription and translation, and construct systems level metabolic models [32]. Here, we used a CFPS based platform to take advantage of its speed compared to cell cultures. We used sigma-factor based gene regulation for the expression of *mdh* (Figure 5.1). Model parameters not available in literature were estimated by minimizing the difference between simulated and experimentally measured protein concentrations. The model predicted mRNA concentration during the course of the reaction. In order to provide a detailed insight into the influence of individual model parameters on the expression dynamics of the system, we performed Morris sensitivity analysis. While we only considered the expression of *mdh* in this study, the model presented here can be adapted for the expression of other pathway enzymes as well.

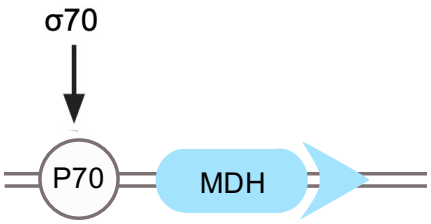

Figure 5.1: Schematic of the cell-free gene expression circuit used in this study for sigma factor 70 (σ70) induced expression of *mdh*

.

5.2 Methods

A mechanistic model for TX-TL was constructed based on the previous work of Adhikari et al. [52], where mRNA and protein balances for gene j ($\mathcal{G}_j$) were formulated as:

$$\frac{dm_j}{dt} = r_{X,j}u_j - \theta_{m_j}m_j \tag{5.1}$$

$$\frac{dp_j}{dt} = r_{L,j}w_j - \theta_{p_j}p_j \tag{5.2}$$

Here, m_j and p_j represent mRNA and protein concentrations respectively, and $r_{X,j}u_j$ and $r_{L,j}w_j$ denote the rate of transcription (X) and translation (L) of gene j. $\theta_{m_j}m_j$ and $\theta_{p_j}p_j$ denote the degradation rate of mRNA and protein. The rate of transcription $r_{X,j}u_j$ was modeled as the product of a kinetic limit, $r_{X,j}$, and a control term $u_j \in [0,1]$. The kinetic limit of transcription was derived from elementary reactions leading to the formation of m_j, similar to McClure [150]:

$$r_{X,j} = V_{X,j}^{max}\left(\frac{\mathcal{G}_j}{\tau_{X,j}K_{X,j} + (\tau_{X,j}+1)\,\mathcal{G}_j}\right) \tag{5.3}$$

where $\tau_{X,j}$ denotes the time constant for transcription and $K_{X,j}$ denotes the transcription saturation constant. The maximum transcription rate, $V_{X,j}^{max}$, was given

by:

$$V_{X,j}^{max} = \left[R_X \left(\frac{\dot{v}_X}{l_{G,j}} \right) \right] \tag{5.4}$$

where R_X denotes the RNA polymerase concentration, $\dot{v}_X$ denotes the transcription elongation rate and $l_{G,j}$ denotes the gene length. The control function $u\left(\ldots\right)$ describes transcriptional regulation and is the fraction of all possible configurations that lead to expression. $u\left(\ldots\right)$ was formulated as:

$$u\left(\ldots\right)_j = \left(\sum_{i \in \{X\}} W_i f_i\left(\ldots\right) \right) \left(\sum_{j \in \mathcal{C}_j} W_j f_j\left(\ldots\right) \right)^{-1} \tag{5.5}$$

where W_i (dimensionless) denotes the weight of configuration i, while $f_i\left(\cdots\right)$ (dimensionless) is a binding function which describes the fraction of bound activator/inhibitor for configuration i. Similar to Ackers and coworkers [151], W_i is modeled as the Gibbs energy of configuration i: $W_i = \exp\left(-\Delta G_i / RT\right)$ where ΔG_i denotes the molar Gibbs free energy for configuration i, R denotes the ideal gas constant, and T denotes the system temperature.

By analogy, the kinetic limit of translation was formulated as:

$$r_{L,j} = V_{L,j}^{max} \left(\frac{x_{mRNA}}{\tau_{L,j} K_{L,j} + \left(\tau_{L,j} + 1\right) x_{mRNA}} \right) \tag{5.6}$$

where x_{mRNA} denotes the mRNA concentration, $\tau_{L,j}$ denotes the time constant for translation and $K_{L,j}$ denotes the translation saturation constant. $V_{L,j}^{max}$, which denotes the maximum translation rate, was formulated as:

$$V_{L,j}^{max} = \left[K_P R_L \left(\frac{\dot{v}_L}{l_{P,j}} \right) \right] \tag{5.7}$$

where R_L denotes the total ribosome pool, $\dot{v}_X$ denotes the translation elongation rate, K_p denotes the polysome amplification constant and $l_{P,j}$ denotes the length of protein.

The model equations, written in the Julia programming language, were generated automatically using the JUGRNModelGenerator package from the Varnerlab GitHub repository[1]. The model equations were solved numerically using the `DifferentialEquations.jl` Julia package. Known model parameters for cell-free gene regulation were taken directly from the literature (Table A.2). Unknown model parameters that influence *mdh* gene expression were determined from protein concentration measurements using the `POETS.jl` Julia package. `POETS.jl` is an implementation of the Pareto Optimal Ensemble Technique in the Julia programming language (JuPOETs). The objective function calculated the squared difference between the model simulations and experimental data for MDH protein concentration at time index i. Thus, the objective function was formulated as:

$$\mathcal{Z}(\mathbf{k}) = \sum_{i=1}^{\mathcal{N}} \left(\hat{\mathcal{M}}_i - \hat{y}_i(\mathbf{k})\right)^2 \tag{5.8}$$

which was subjected to constraints derived from model equations, initial conditions and parameter bounds. JuPOETs was run for 10 generations and in each generation, all parameter sets with a Pareto rank less than or equal to two were collected. A total of 11 unknown parameters were estimated including time constants, degradation modifiers, translation half-life and translation saturation constant.

To understand the effect of parameters on model performance, Morris sensitivity analysis or Morris's one at a time (OAT) method was used. The `DiffEqSensitivity.jl` Julia package was used for Morris Sensitivity Analysis. Morris sensitivity analysis is considered a global method because the final

[1]Varnerlab. Gene Regulatory Network Model Generator in Julia (JUGRNModelGenerator). Available online at https://github.com/varnerlab/JUGRNModelGenerator.jl.

measure is obtained by averaging elementary effects. Elementary effects are local sensitivity measures which are calculated by measuring perturbation in the output of the model on changing one input parameter. Therefore, elementary effects (or local measures) are computed at different points, such that, a wide range of the input parameter space is explored for analysis.

$$EE_i = \frac{f(x_1, x_2, ..x_i + \delta, .., x_n) - y}{\delta} \tag{5.9}$$

Finally, to account for uncertainty in the parameters taken directly from literature (Table A.2), parameter values were sampled within physiological limits to generate the ensemble solution set. RNA polymerase concentration levels were sampled between 0.060 and 0.075 μM, maximum transcription rates were sampled between 15 and 25 nt/s, ribosome concentration levels were sampled between 2 and 2.3 μM and maximum translation rates were sampled between 1 and 2 aa/s.

5.3 Results

The model simulations adequately captured the dynamics of σ70-induced *mdh* protein expression (Figure 5.2). An ensemble set (N=100) of 11 unknown parameters was estimated using experimental *mdh* protein training data. The means and standard deviations of the estimate parameters are summarized in Table A.3. *mdh* protein concentration increased almost linearly for the first 6 h of the reaction, after which it began saturating, reaching a concentration value of $\sim$25 μM at 10 h. The mean half-life of *mdh* protein was estimated to be roughly 3.5 days, close to previously reported values; Adhikari et al. [52] reported a half-life value of 11 days for the expression of dual emission green fluorescent protein

(deGFP) using an E. coli based myTXTL cell-free system, whereas Horvath et al. [152] reported a value of 6 days for the expression of chloramphenical acetyl transferase (CAT) using a modified version of the PANOxSP protocol. The mean half-life of translational capacity $\tau_{L,1/2}$, which quantified the decrease in the rate of protein production over time, was estimated to be 6 h. The decrease in translation capacity can be attributed to the depletion of metabolic resources in the system essential for supporting translation, among other factors. On the other hand, transcription of mRNA was predicted by the model. The mRNA concentration was observed to increase sharply to a value of 1200 nM within the first 1 h of the reaction, after which it remained steady. The mean mRNA half-life was estimated to be 4.4 min. Therefore, in this case, the cell-free reaction was observed to have continuous transcriptional activity.

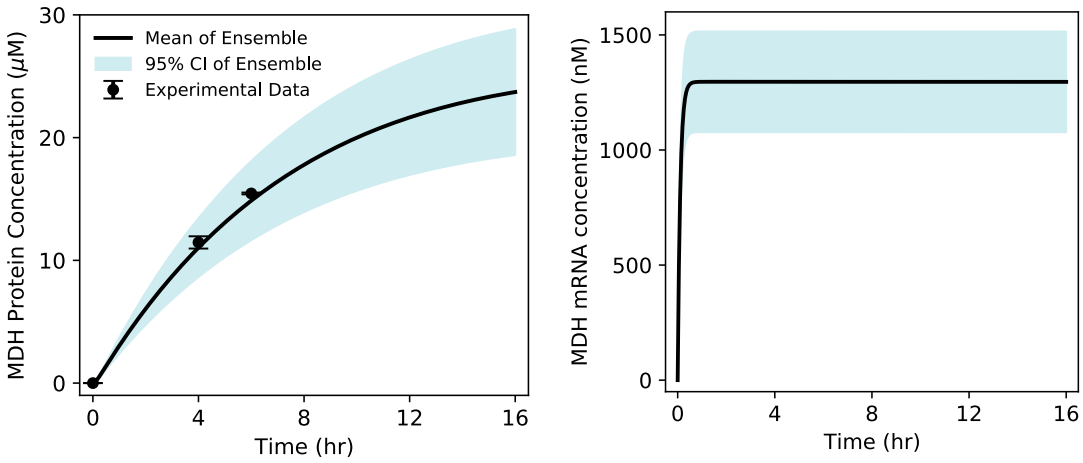

Figure 5.2: Model simulations for σ70 induced *mdh* expression. Left: Simulated and measured *mdh* protein concentration versus time. B: Simulated and measured *mdh* mRNA concentration versus time. An ensemble set of solutions (N=100) was generated.

The importance of model parameters was quantified using Morris sensitivity analysis, a global sensitivity analysis method (Figure 5.3). The Morris method

computes the influence of each parameter on the system dynamics. This influence is calculated as an elementary effect on a specified model performance function. In this case, the performance function was defined as the integrated area under the curve (AUC) for each mRNA and protein species in their respective timeplots. The mean of the elementary effect represents the direct affect of the parameter on the specified species; a higher mean implicates a higher influence of the parameter. Meanwhile, variance represents the effect of the parameter; a higher variance implies that the effects are non-linear or the result of interactions with other parameters.

The Morris sensitivity measures (mean and variance) were binned into categories based upon their relative magnitudes, from no influence (white) to high influence (black). The translation saturation coefficient K_L, translational capacity half-life $\tau_{L,1/2}$, translation time constant τ_L, and protein degradation constant $\theta_{p,mdh}$, influenced only the protein concentration. Among these parameters, the translation saturation coefficient and translation time constant had the smallest and largest effects respectively, on protein level. The $\sigma 70$ degradation constant $\theta_{p,\sigma 70}$, mRNA degradation constant $\theta_{m,mdh}$ and mRNA time constant modifier τ_X influenced both transcription and translation, although their influence on transcription was only marginal. Therefore, the parameters which directly affected the transcription of mRNA also indirectly affected the downstream translation process. Taken together, Morris sensitivity analysis highlighted the coupled nature of the transcription and translation processes, also showing the global importance of experimentally tunable parameters like the mRNA time constant and degradation modifiers.

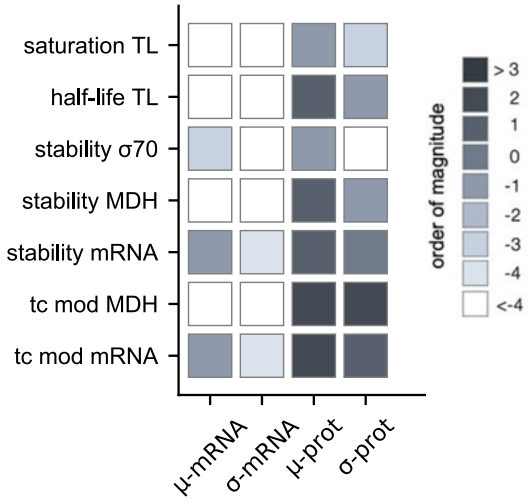

Figure 5.3: Global sensitivity analysis of the estimated parameters using the Morris method. Morris sensitivity coefficients were calculated for the unknown model parameters, where the range for each parameter was established from the ensemble.

5.4 Discussion

A mechanistic model was developed to simulate the expression of *mdh* in a cell-free system. The model captured the dynamics of the system using only *mdh* protein concentration data, which showed an accumulation of protein in the reaction system. All of our parameter estimates were in the same order of magnitude as those obtained by Adhikari et al.; their study used the same cell-free reaction system for σ70 induced deGFP expression. Surprisingly for us, certain parameters like the binding dissociation constant K and cooperativity parameter n which appear in the transcriptional control function were not influential on the system. It is possible that incorporating mRNA training data into the objective function would provide more accurate parameter estimates, especially for transcription. In that case, we also expect K and n to have a more significant effect on the system.

As a preliminary study, we have only accounted for the expression of *mdh* to catalyze the conversion of morphine to morphinone. This method can be adapted for other pathway enzymes involved in the production of naloxone. Future experiments directed towards the expression and quantification of other enzymes will provide us with additional training data for modeling the complete pathway. In that case, training objectives can be formulated to estimate other model parameters using multi-objective optimization or JuPOETs [153]. In that case, the parameter estimates determined from this study can be used to constrain the parameter search space for the larger network.

Finally, in this study, we have neglected the role of metabolism. CFPS relies on central carbon metabolism for energy generation and therefore, TX-TL models are more descriptive when integrated with metabolism. Although more recent studies have explored this idea in cell-free systems [152, 42], these models are not as comprehensive as the model presented here. Therefore, a future extension of this study could consider coupling this description of TX-TL with the availability of metabolic resources in the CFPS reaction, following a sequence specific based approach. Towards this, metabolite values for different time points could be determined and used in metabolic modeling to optimize reaction conditions [154].

Therefore, this work takes us one step closer towards a more descriptive, systems level metabolic model. With subsequent developments, though the inclusion of other pathway enzymes and descriptions of metabolism, the resultant model will aid in model guided development of a CFPS platform for the production of naloxone at a therapeutically relevant level. Ultimately, the expressed enzymes will work to sequentially catalyze the conversion of heroin, or

its degradation products, to produce the opioid antidote naloxone. Taken together, future adaptations of this model can then be used to optimize reaction conditions to achieve better CFPS yields and make this anti-opioid technology a feasible point-of-care solution.

CHAPTER 6

CONCLUSIONS AND FUTURE DIRECTIONS

Metabolism is the set of chemical reactions occurring in cells which allow them to grow, reproduce and respond to their environment. To perform these diverse functions, cells arrange chemical reactions in highly interconnected and complex metabolic networks. As a result, metabolic modeling emerged as a tool to better understand various cellular processes and how they can be manipulated to increase productivity and efficiency. For instance, today, through the identification of pathways controlling a specific phenotype, cells can be artificially reprogrammed to produce valuable therapeutics and vaccines, among other products. Currently, various cell-based culture systems are used for large-scale production; however, transportation and storage at low temperatures makes their distribution expensive and problematic, especially for applications in remote locations. To this end, cell-free protein synthesis (CFPS) has paved the way for the development of point-of-care technologies that enable immediate and accessible protein production at or near the patient's bedside.

Since the mid-1990s, the advantages of CFPS have enabled numerous applications; CFPS was used to formulate the central dogma, probe transcription and translation, and construct systems level metabolic models. In particular, the lack of a living cell allows us to direct resources towards the product of interest. In 2018, Vilkhovoy et al. published a constraint-based model integrating transcription and translation with metabolism that identified translation rate and oxidative phosphorylation as the key factors in productivity and energy efficiency, respectively. Therefore, to gather more insights into energy efficiency of CFPS, we expanded this model by integrating experimental measurements of 63

metabolites along with kinetic parameters, enzyme levels, and enzyme activity assays. First, flux balance analysis (FBA), a widely used constraint-based model, was used to simulate the metabolic state of the system with two different oxidative phosphorylation inhibitors. Then, we tested the objective of minimization of metabolic adjustment (MOMA) for CFPS, which is based on the premise that there is no method for immediate regulation of fluxes towards a specific biological objective, following a perturbation. We found MOMA to accurately predict the overall production of mRNA and protein along with changes in metabolic behavior in the presence of the inhibitors.

Next, we proposed CFPS as a platform for producing naloxone, an opioid overdose antitode, taking advantage of its potential to rapidly express bioactive recombinant DNA (rDNA) proteins. Considering the ongoing opioid crisis, paired with the insufficient availability and high cost of naloxone, we developed a novel biosynthetic pathway for the production of naloxone and predicted possible enzymes that could catalyze various steps of the proposed pathway. Future work will experimentally validate this pathway, followed by characterization and quantification of the enzymes involved. We also developed a mathematical model, accounting for transcription and translation, to simulate the cell-free expression of morphine dehydrogenase, the first enzyme of the designed pathway. This model can be adapted for the expression of other pathway enzymes as well. A future extension of this study could consider coupling this description of transcription and translation with the availability of metabolic resources in the CFPS reaction. Towards this, metabolite values for different time points could be determined and used in metabolic modeling to optimize reaction conditions.

Based on our findings, MOMA opens several directions for future work. One possible way would be to adapt this MOMA formulation to the naloxone pathway we designed. In particular, MOMA can be used to predict the perturbation effect of adding various pathway enzymes identified for the synthesis of naloxone. In addition, MOMA can help in the development of strategies that improve protein yields by the deletion of enzymes that negatively affect CFPS. For example, energy resources consumed for the degradation of nucleotides can otherwise be directed toward translation. Thus, MOMA can reveal insights into the best strategies for optimizing CFPS through more efficient regulation of energy resources within the metabolic network.

Taken together, we have developed a strategy for the point-of-care production of naloxone using cell-free metabolic engineering, pending further experimental validation and enzyme characterization. We have also validated the approach of MOMA for model guided development and optimization of this proposed platform. Finally, MOMA can be used to engineer strains with improved CFPS performance, thus extending the scope of its application to cell-free metabolic engineering.

APPENDIX A

APPENDIX

Table A.1: Parameters for sequence specific flux balance analysis and minimization of metabolic adjustment

Parameter	**Value**
RNA polymerase concentration (R_{TL})	60-75 nM [20]
Ribosome concentration (R_{TX})	2-2.3 μM [20]
Transcription elongation rate $(\dot{n}_{TX})$	15-25 nt/s [20]
Translation elongation rate $(\dot{n}_{TL})$	1-2 aa/s/ribosome [20]
Transcription time constant (τ_{TX})	0.021 - 0.05 (calculated)
Translation time constant (τ_{TL})	0.063 - 0.126 (calculated)
Transcription saturation coefficient (K_{TX})	0.3 μM [150]
Translation saturation coefficient (K_{TL})	600.0 μM (estimated)
Polysome number (K_P)	10 (estimated)
mRNA degradation rate constant (k_d^{mRNA})	2.38 h^{-1} [20]
Maltodextrin saturation constant (K_m)	8.3 mM (BRENDA)
Transcription saturation constant (K_s^{TX})	0.03 mM estimated
Weight RNA polymerase binding alone P70a (K_1)	0.014 (estimated)
Weight bound RNAP-σ_{70} P70a (K_2)	10 (estimated)
σ_{70} concentration (σ_{70})	35 nM [20]
σ_{70} dissociation constant (K_D)	130 nM [155]
σ_{70} hill coefficient (n)	1 [155]
Gene concentration (G)	5 nM (experiment)

Table A.2: Literature parameters used for TX-TL model equations.

Parameter	**Value**	**Reference**
RNA Polymerase concentration, R_X	0.06-0.07 μM	[20]
Ribosome concentration, R_L	$<$2.3 μM	[20]
σ70 concentration, σ70	$<$35 nM	[20]
Transcription elongation rate, $\dot{v}_X$	12-30 nt/s	[20]
Translation elongation rate, $\dot{v}_L$	1-2 aa/s	[20]
Transcription saturation coefficient, K_X	0.036 μM	[150]
Polysome amplification constant, K_P	1.0	[52]
Transcription initiation time, $k_{init,X}$	22 s	[150]
Translation initiation time, $k_{init,L}$	1.5 s	[52]

Table A.3: Estimated parameters for *mdh* gene regulation

Description	**Parameter**	**Value ($\mu \pm \sigma$)**
Translation saturation coefficient	K_L	312.79 $\pm$ 3.16 μM
Half-life translation	$\tau_{L,1/2}$	5.87 $\pm$ 0.06 h^{-1}
Time constants		
mdh transcription	τ_X	0.612 $\pm$ 0.006
mdh translation	τ_L	0.0502 $\pm$ 0.0005
mRNA and protein half life		
mRNA *mdh*	$ln(2)/\theta_{m,mdh}$	4.37 $\pm$ 0.04 min
Protein *mdh*	$ln(2)/\theta_{p,mdh}$	3.527 $\pm$ 0.035 $days$
Protein σ_{70}	$ln(2)/\theta_{p,\sigma_{70}}$	2.182 $\pm$ 0.021 $days$
Binding energies		
RNAP + *mdh* gene	$\Delta G_{mdh,RX}$	45.967 $\pm$ 0.464 $kJmol^{-1}$
RNAP + σ_{70} + *mdh* gene	$\Delta G_{mdh,\sigma 70}$	-20.012 $\pm$ 0.201 $kJmol^{-1}$
Binding parameters		
Hill coefficient	n	1.118 $\pm$ 0.011
Dissociation constant	K	7.350 $\pm$ 0.074

BIBLIOGRAPHY

[1] Harry Yim, Robert Haselbeck, Wei Niu, Catherine Pujol-Baxley, Anthony Burgard, Jeff Boldt, Julia Khandurina, John D Trawick, Robin E Osterhout, Rosary Stephen, et al. Metabolic engineering of escherichia coli for direct production of 1, 4-butanediol. *Nature chemical biology*, 7(7):445, 2011.

[2] Noushin Hadadi and Vassily Hatzimanikatis. Design of computational retrobiosynthesis tools for the design of de novo synthetic pathways. *Current opinion in chemical biology*, 28:99–104, 2015.

[3] Erik D. Carlson, Rui Gan, C. Eric Hodgman, and Michael C. Jewett. Cell-free protein synthesis: Applications come of age. *Biotechnology Advances*, 30(5):1185–1194, 2012.

[4] Simon J. Moore, James T. MacDonald, and Paul S. Freemont. Cell-free synthetic biology for in vitro prototype engineering. *Biochemical Society Transactions*, 45(3):785–791, 06 2017.

[5] Ke Yue, Yiyong Zhu, and Lei Kai. Cell-free protein synthesis: Chassis toward the minimal cell. *Cells*, 8(4), 2019.

[6] Emanuel G. Worst, Matthias P. Exner, Alessandro de Simone, Marc Schenkelberger, Vincent Noireaux, Nediljko Budisa, and Albrecht Ott. Residue-specific incorporation of noncanonical amino acids into model proteins using an escherichia coli cell-free transcription-translation system. *Journal of Visualized Experiments*, 2016(114), 2016.

[7] Khushal Khambhati, Gargi Bhattacharjee, Nisarg Gohil, Darren Braddick, Vishwesh Kulkarni, and Vijai Singh. Exploring the potential of cell-free protein synthesis for extending the abilities of biological systems. *Frontiers in Bioengineering and Biotechnology*, 7:248, 2019.

[8] Walter C Schneider and George H Hogeboom. Cytochemical studies of mammalian tissues; the isolation of cell components by differential centrifugation: a review. *Cancer Research*, 11(1):1–22, 1951.

[9] T Winnick et al. Studies on the mechanism of protein, synthesis in embryonic and tumor tissues. 1. evidence relating to the incorporation of labeled amino acids into protein structure in homogenates. *Arch. Biochem.*, 27, 1950.

[10] T Winnick et al. Studies on the mechanism of protein synthesis in embryonic and tumor tissues. 2. inactivation of fetal rat liver homogenates by dialysis, and reactivation by the adenylic acid system. *Arch. Biochem.*, 28:338–347, 1950.

[11] Mahlon B Hoagland, Elizabeth B Keller, and Paul C Zamecnik. Enzymatic carboxyl activation of amino acids. *Journal of Biological Chemistry*, 218(1):345–358, 1956.

[12] J Heinrich Matthaei and Marshall W Nirenberg. Characteristics and stabilization of dnaase-sensitive protein synthesis in e. coli extracts. *Proceedings of the National Academy of Sciences of the United States of America*, 47(10):1580, 1961.

[13] Dong-Myung Kim and James R Swartz. Prolonging cell-free protein synthesis with a novel atp regeneration system. *Biotechnology and Bioengineering*, 66(3):180–188, 1999.

[14] Dong-Myung Kim and James R Swartz. Prolonging cell-free protein synthesis by selective reagent additions. *Biotechnology progress*, 16(3):385–390, 2000.

[15] Dong-Myung Kim and James R Swartz. Regeneration of adenosine triphosphate from glycolytic intermediates for cell-free protein synthesis. *Biotechnology and bioengineering*, 74(4):309–316, 2001.

[16] Michael C Jewett and James R Swartz. Mimicking the escherichia coli cytoplasmic environment activates long-lived and efficient cell-free protein synthesis. *Biotechnology and bioengineering*, 86(1):19–26, 2004.

[17] Michael C Jewett and James R Swartz. Substrate replenishment extends protein synthesis with an in vitro translation system designed to mimic the cytoplasm. *Biotechnology and bioengineering*, 87(4):465–471, 2004.

[18] Alexander S Spirin and James R Swartz. *Cell-free protein synthesis: methods and protocols*. John Wiley & Sons, 2007.

[19] Michael C Jewett, Kara A Calhoun, Alexei Voloshin, Jessica J Wuu, and James R Swartz. An integrated cell-free metabolic platform for protein production and synthetic biology. *Molecular systems biology*, 4(1):220, 2008.

[20] Jonathan Garamella, Ryan Marshall, Mark Rustad, and Vincent Noireaux.

The all e. coli tx-tl toolbox 2.0: a platform for cell-free synthetic biology. *ACS synthetic biology*, 5(4):344–355, 2016.

[21] Zachary Z Sun, Clarmyra A Hayes, Jonghyeon Shin, Filippo Caschera, Richard M Murray, and Vincent Noireaux. Protocols for implementing an escherichia coli based tx-tl cell-free expression system for synthetic biology. *Journal of visualized experiments: JoVE*, e50762(79), 2013.

[22] Dong-Myung Kim and James R Swartz. Efficient production of a bioactive, multiple disulfide-bonded protein using modified extracts of escherichia coli. *Biotechnology and bioengineering*, 85(2):122–129, 2004.

[23] Gang Yin and James R Swartz. Enhancing multiple disulfide bonded protein folding in a cell-free system. *Biotechnology and bioengineering*, 86(2):188–195, 2004.

[24] John P Welsh, Yuan Lu, Xiao-Song He, Harry B Greenberg, and James R Swartz. Cell-free production of trimeric influenza hemagglutinin head domain proteins as vaccine antigens. *Biotechnology and bioengineering*, 109(12):2962–2969, 2012.

[25] Yuan Lu, John P Welsh, and James R Swartz. Production and stabilization of the trimeric influenza hemagglutinin stem domain for potentially broadly protective influenza vaccines. *Proceedings of the National Academy of Sciences*, 111(1):125–130, 2014.

[26] Peter L. Bergquist, Sana Siddiqui, and Anwar Sunna. Cell-free biocatalysis for the production of platform chemicals. *Frontiers in Energy Research*, 8:193, 2020.

[27] David K Karig. Cell-free synthetic biology for environmental sensing and remediation. *Current opinion in biotechnology*, 45:69–75, 2017.

[28] Shimyn Slomovic, Keith Pardee, and James J Collins. Synthetic biology devices for in vitro and in vivo diagnostics. *Proceedings of the National Academy of Sciences*, 112(47):14429–14435, 2015.

[29] Mark Thomas Smith, Scott D Berkheimer, Christopher J Werner, and Bradley C Bundy. Lyophilized escherichia coli-based cell-free systems for robust, high-density, long-term storage. *Biotechniques*, 56(4):186–193, 2014.

[30] Keith Pardee, Alexander A Green, Tom Ferrante, D Ewen Cameron, Ajay

DaleyKeyser, Peng Yin, and James J Collins. based synthetic gene networks. *Cell*, 159(4):940–954, 2014.

[31] Alexei M Voloshin and James R Swartz. Efficient and scalable method for scaling up cell free protein synthesis in batch mode. *Biotechnology and bioengineering*, 91(4):516–521, 2005.

[32] Michael Vilkhovoy, Abhinav Adhikari, Sandra Vadhin, and Jeffrey D Varner. The evolution of cell free biomanufacturing. *Processes*, 8(6):675, 2020.

[33] Cassandra Guarino and Matthew P DeLisa. A prokaryote-based cell-free translation system that efficiently synthesizes glycoproteins. *Glycobiology*, 22(5):596–601, 2012.

[34] Andreas Karoly Gombert and Jens Nielsen. Mathematical modelling of metabolism. *Current opinion in biotechnology*, 11(2):180–186, 2000.

[35] Bernhard O Palsson and Edwin N Lightfoot. Mathematical modelling of dynamics and control in metabolic networks. i. on michaelis-menten kinetics. *Journal of theoretical biology*, 111(2):273–302, 1984.

[36] O Borkowski, C Bricio, M Murgiano, B Rothschild-Mancinelli, GB Stan, and T Ellis. Cell-free prediction of protein expression costs for growing cells, 2018.

[37] Tobias Stogbauer, Lukas Windhager, Ralf Zimmer, and Joachim O Radler. Experiment and mathematical modeling of gene expression dynamics in a cell-free system. *Integrative Biology*, 4(5):494–501, 2012.

[38] Simon J Moore, James T MacDonald, Sarah Wienecke, Alka Ishwarbhai, Argyro Tsipa, Rochelle Aw, Nicolas Kylilis, David J Bell, David W McClymont, Kirsten Jensen, et al. Rapid acquisition and model-based analysis of cell-free transcription–translation reactions from nonmodel bacteria. *Proceedings of the National Academy of Sciences*, 115(19):E4340–E4349, 2018.

[39] Eyal Karzbrun, Jonghyeon Shin, Roy H Bar-Ziv, and Vincent Noireaux. Coarse-grained dynamics of protein synthesis in a cell-free system. *Physical review letters*, 106(4):048104, 2011.

[40] Peter L Voyvodic, Amir Pandi, Mathilde Koch, Ismael Conejero, Emmanuel Valjent, Philippe Courtet, Eric Renard, Jean-Loup Faulon, and

Jerome Bonnet. Plug-and-play metabolic transducers expand the chemical detection space of cell-free biosensors. *Nature communications*, 10(1):1–8, 2019.

[41] Mathilde Koch, Jean-Loup Faulon, and Olivier Borkowski. Models for cell-free synthetic biology: make prototyping easier, better, and faster. *Frontiers in bioengineering and biotechnology*, 6:182, 2018.

[42] Michael Vilkhovoy, Nicholas Horvath, Che-Hsiao Shih, Joseph A Wayman, Kara Calhoun, James Swartz, and Jeffrey D Varner. Sequence specific modeling of e. coli cell-free protein synthesis. *ACS synthetic biology*, 7(8):1844–1857, 2018.

[43] David Dai, Nicholas Horvath, and Jeffrey Varner. Dynamic sequence specific constraint-based modeling of cell-free protein synthesis. *Processes*, 6(8):132, 2018.

[44] Jeffrey D Orth, Ines Thiele, and Bernhard Ø Palsson. What is flux balance analysis? *Nature biotechnology*, 28(3):245–248, 2010.

[45] Anthony P Burgard and Costas D Maranas. Optimization-based framework for inferring and testing hypothesized metabolic objective functions. *Biotechnology and bioengineering*, 82(6):670–677, 2003.

[46] Erwin P Gianchandani, Matthew A Oberhardt, Anthony P Burgard, Costas D Maranas, and Jason A Papin. Predicting biological system objectives de novo from internal state measurements. *BMC bioinformatics*, 9(1):1–13, 2008.

[47] Carlos Eduardo García Sánchez, César Augusto Vargas García, and Rodrigo Gonzalo Torres Sáez. Predictive potential of flux balance analysis of saccharomyces cerevisiae using as optimization function combinations of cell compartmental objectives. *PLoS One*, 7(8):e43006, 2012.

[48] Andrea L Knorr, Rishi Jain, and Ranjan Srivastava. Bayesian-based selection of metabolic objective functions. *Bioinformatics*, 23(3):351–357, 2007.

[49] Dave Siak-Wei Ow, Dong-Yup Lee, Miranda Gek-Sim Yap, and Steve Kah-Weng Oh. Identification of cellular objective for elucidating the physiological state of plasmid-bearing escherichia coli using genome-scale in silico analysis. *Biotechnology progress*, 25(1):61–67, 2009.

[50] Tomer Shlomi, Omer Berkman, and Eytan Ruppin. Regulatory on/off minimization of metabolic flux changes after genetic perturbations. *Proceedings of the national academy of sciences*, 102(21):7695–7700, 2005.

[51] Daniel Segre, Dennis Vitkup, and George M Church. Analysis of optimality in natural and perturbed metabolic networks. *Proceedings of the National Academy of Sciences*, 99(23):15112–15117, 2002.

[52] Abhinav Adhikari, Michael Vilkhovoy, Sandra Vadhin, Ha Eun Lim, and Jeffrey D Varner. Effective biophysical modeling of cell free transcription and translation processes. *Frontiers in bioengineering and biotechnology*, 8, 2020.

[53] Ho-Cheol Kim and Dong-Myung Kim. Methods for energizing cell-free protein synthesis. *Journal of bioscience and bioengineering*, 108(1):1–4, 2009.

[54] James R Swartz. Advances in escherichia coli production of therapeutic proteins. *Current opinion in biotechnology*, 12(2):195–201, 2001.

[55] Geoffrey Zubay. In vitro synthesis of protein in microbial systems. *Annual review of genetics*, 7(1):267–287, 1973.

[56] Carl W Anderson, J William Straus, and Bernard S Dudock. Preparation of a cell-free protein-synthesizing system from wheat germ. *Recombinant DNA Methodology*, pages 677–685, 1989.

[57] Lyubov A Ryabova, Leonid M Vinokurov, Ekaterina A Shekhovtsova, Yuly B Alakhov, and Alexander S Spirin. Acetyl phosphate as an energy source for bacterial cell-free translation systems. *Analytical biochemistry*, 226(1):184–186, 1995.

[58] Tae-Wan Kim, Jung-Won Keum, In-Seok Oh, Cha-Yong Choi, Ho-Cheol Kim, and Dong-Myung Kim. An economical and highly productive cell-free protein synthesis system utilizing fructose-1, 6-bisphosphate as an energy source. *Journal of biotechnology*, 130(4):389–393, 2007.

[59] Kara A Calhoun and James R Swartz. Energizing cell-free protein synthesis with glucose metabolism. *Biotechnology and bioengineering*, 90(5):606–613, 2005.

[60] Tae-Wan Kim, Ho-Cheol Kim, In-Seok Oh, and Dong-Myung Kim. A highly efficient and economical cell-free protein synthesis system using

the s12 extract of escherichia coli. *Biotechnology and Bioprocess Engineering*, 13(4):464–469, 2008.

[61] Tae-Wan Kim, In-Seok Oh, Jung-Won Keum, Yong-Chan Kwon, Ju-Young Byun, Kyung-Ho Lee, Cha-Yong Choi, and Dong-Myung Kim. Prolonged cell-free protein synthesis using dual energy sources: Combined use of creatine phosphate and glucose for the efficient supply of atp and retarded accumulation of phosphate. *Biotechnology and bioengineering*, 97(6):1510–1515, 2007.

[62] Rafael U Ibarra, Jeremy S Edwards, and Bernhard O Palsson. Escherichia coli k-12 undergoes adaptive evolution to achieve in silico predicted optimal growth. *Nature*, 420(6912):186–189, 2002.

[63] Timothy E Allen and Bernhard Ø Palsson. Sequence-based analysis of metabolic demands for protein synthesis in prokaryotes. *Journal of theoretical biology*, 220(1):1–18, 2003.

[64] Andrew Makhorin. Gnu linear programming kit, 2018.

[65] LLC Gurobi Optimization. Gurobi optimizer reference manual, 2021.

[66] Lisa Jeske, Sandra Placzek, Ida Schomburg, Antje Chang, and Dietmar Schomburg. Brenda in 2019: a european elixir core data resource. *Nucleic acids research*, 47(D1):D542–D549, 2019.

[67] Roi Adadi, Benjamin Volkmer, Ron Milo, Matthias Heinemann, and Tomer Shlomi. Prediction of microbial growth rate versus biomass yield by a metabolic network with kinetic parameters. *PLoS Comput Biol*, 8(7):e1002575, 2012.

[68] David Garenne, Chase L Beisel, and Vincent Noireaux. Characterization of the all-e. coli transcription-translation system mytxtl by mass spectrometry. *Rapid Communications in Mass Spectrometry*, 33(11):1036–1048, 2019.

[69] Jay D Keasling. Manufacturing molecules through metabolic engineering. *Science*, 330(6009):1355–1358, 2010.

[70] Dae-Kyun Ro, Eric M Paradise, Mario Ouellet, Karl J Fisher, Karyn L Newman, John M Ndungu, Kimberly A Ho, Rachel A Eachus, Timothy S Ham, James Kirby, et al. Production of the antimalarial drug precursor artemisinic acid in engineered yeast. *Nature*, 440(7086):940–943, 2006.

[71] Jens Nielsen and Jay D Keasling. Engineering cellular metabolism. *Cell*, 164(6):1185–1197, 2016.

[72] Geng-Min Lin, Robert Warden-Rothman, and Christopher A Voigt. Retrosynthetic design of metabolic pathways to chemicals not found in nature. *Current Opinion in Systems Biology*, 2019.

[73] Marwin HS Segler, Mike Preuss, and Mark P Waller. Planning chemical syntheses with deep neural networks and symbolic ai. *Nature*, 555(7698):604, 2018.

[74] Lin Wang, Satyakam Dash, Chiam Yu Ng, and Costas D Maranas. A review of computational tools for design and reconstruction of metabolic pathways. *Synthetic and systems biotechnology*, 2(4):243–252, 2017.

[75] Minoru Kanehisa. Toward pathway engineering: a new database of genetic and molecular pathways. *Sci. Technol. Jap.*, 59:34–38, 1996.

[76] Minoru Kanehisa. A database for post-genome analysis. *Trends Genet.*, 13:375–376, 1997.

[77] Minoru Kanehisa and Susumu Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30, 2000.

[78] Minoru Kanehisa, Susumu Goto, Yoko Sato, Miho Furumichi, and Mao Tanabe. Kegg for integration and interpretation of large-scale molecular data sets. *Nucleic acids research*, 40(D1):D109–D114, 2012.

[79] Minoru Kanehisa, Yoko Sato, Masayuki Kawashima, Miho Furumichi, and Mao Tanabe. Kegg as a reference resource for gene and protein annotation. *Nucleic acids research*, 44(D1):D457–D462, 2016.

[80] Noushin Hadadi, Jasmin Hafner, Adrian Shajkofci, Aikaterini Zisaki, and Vassily Hatzimanikatis. Atlas of biochemistry: a repository of all possible biochemical reactions for synthetic biology and metabolic engineering studies. *ACS synthetic biology*, 5(10):1155–1166, 2016.

[81] Jasmin Hafner, Homa MohammadiPeyhani, Anastasia Sveshnikova, Alan Scheidegger, and Vassily Hatzimanikatis. Updated atlas of biochemistry with new metabolites and improved enzyme prediction power. *ACS Synthetic Biology*, 2020.

[82] Vassily Hatzimanikatis, Chunhui Li, Justin A Ionita, Christopher S Henry, Matthew D Jankowski, and Linda J Broadbelt. Exploring the diversity of complex metabolic networks. *Bioinformatics*, 21(8):1603–1609, 2005.

[83] Stacey D Finley, Linda J Broadbelt, and Vassily Hatzimanikatis. Computational framework for predictive biodegradation. *Biotechnology and bioengineering*, 104(6):1086–1097, 2009.

[84] Keng Cher Soh and Vassily Hatzimanikatis. Dreams of metabolism. *Trends in biotechnology*, 28(10):501–508, 2010.

[85] Stacey D Finley, Linda J Broadbelt, and Vassily Hatzimanikatis. In silico feasibility of novel biodegradation pathways for 1, 2, 4-trichlorobenzene. *BMC systems biology*, 4(1):7, 2010.

[86] Daniel C McShan, S Rao, and Imran Shah. Pathminer: predicting metabolic pathways by heuristic search. *Bioinformatics*, 19(13):1692–1698, 2003.

[87] Yuki Moriya, Daichi Shigemizu, Masahiro Hattori, Toshiaki Tokimatsu, Masaaki Kotera, Susumu Goto, and Minoru Kanehisa. Pathpred: an enzyme-catalyzed metabolic pathway prediction server. *Nucleic acids research*, 38(suppl_2):W138–W143, 2010.

[88] Masahiro Hattori, Yasushi Okuno, Susumu Goto, and Minoru Kanehisa. Development of a chemical structure comparison method for integrated analysis of chemical and genomic information in the metabolic pathways. *Journal of the American Chemical Society*, 125(39):11853–11865, 2003.

[89] Baudoin Delépine, Thomas Duigou, Pablo Carbonell, and Jean-Loup Faulon. Retropath2. 0: A retrosynthesis workflow for metabolic engineers. *Metabolic engineering*, 45:158–170, 2018.

[90] Mathilde Koch, Thomas Duigou, and Jean-Loup Faulon. Reinforcement learning for bioretrosynthesis. *ACS Synthetic Biology*, 9(1):157–168, 2019.

[91] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[92] Jennifer N Wei, David Duvenaud, and Alán Aspuru-Guzik. Neural networks for the prediction of organic chemistry reactions. *ACS central science*, 2(10):725–732, 2016.

[93] Sophia Ananiadou, Sampo Pyysalo, Jun'ichi Tsujii, and Douglas B Kell. Event extraction for systems biology by text mining the literature. *Trends in biotechnology*, 28(7):381–390, 2010.

[94] Connor W Coley, Regina Barzilay, Tommi S Jaakkola, William H Green, and Klavs F Jensen. Prediction of organic reaction outcomes using machine learning. *ACS central science*, 3(5):434–443, 2017.

[95] Connor W Coley, Wengong Jin, Luke Rogers, Timothy F Jamison, Tommi S Jaakkola, William H Green, Regina Barzilay, and Klavs F Jensen. A graph-convolutional neural network model for the prediction of chemical reactivity. *Chemical science*, 10(2):370–377, 2019.

[96] Wengong Jin, Connor Coley, Regina Barzilay, and Tommi Jaakkola. Predicting organic reaction outcomes with weisfeiler-lehman network. In *Advances in Neural Information Processing Systems*, pages 2607–2616, 2017.

[97] Connor W Coley, Luke Rogers, William H Green, and Klavs F Jensen. Computer-assisted retrosynthesis based on molecular similarity. *ACS central science*, 3(12):1237–1245, 2017.

[98] Taffee T Tanimoto. Elementary mathematical theory of classification and prediction. 1958.

[99] Christopher S Henry, Linda J Broadbelt, and Vassily Hatzimanikatis. Discovery and analysis of novel metabolic pathways for the biosynthesis of industrial chemicals: 3-hydroxypropanoate. *Biotechnology and bioengineering*, 106(3):462–473, 2010.

[100] Thomas Duigou, Melchior Du Lac, Pablo Carbonell, and Jean-Loup Faulon. Retrorules: a database of reaction rules for engineering biology. *Nucleic acids research*, 47(D1):D1229–D1235, 2019.

[101] RetroRules. Retrorules: Delivering reaction rules to engineer biology. https://retrorules.org/dl, 2020. [Online; accessed 24-September-2020].

[102] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[103] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[104] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. In *Advances in neural information processing systems*, pages 2224–2232, 2015.

[105] Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

[106] Wenhao Gao and Connor W Coley. The synthesizability of molecules proposed by generative models. *Journal of chemical information and modeling*, 60(12):5714–5723, 2020.

[107] Jidon Jang, Geun Ho Gu, Juhwan Noh, Juhwan Kim, and Yousung Jung. Structure-based synthesizability prediction of crystals using partially supervised learning. *Journal of the American Chemical Society*, 142(44):18836–18843, 2020.

[108] Kanehisa Labs. Kegg: Kyoto encyclopedia of genes and genomes. https://www.genome.jp/kegg/, 2020. [Online; accessed 24-September-2020].

[109] Laboratory of Computational Systems Biotechnology. Atlas of biochemistry. https://lcsb-databases.epfl.ch/atlas/Downloads, 2020. [Online; accessed 24-September-2020].

[110] Hiroyuki Ogata, Susumu Goto, Kazushige Sato, Wataru Fujibuchi, Hidemasa Bono, and Minoru Kanehisa. Kegg: Kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 27(1):29–34, 1999.

[111] Minoru Kanehisa, Susumu Goto, Shuichi Kawashima, and Akihiro Nakaya. The kegg databases at genomenet. *Nucleic acids research*, 30(1):42–46, 2002.

[112] Minoru Kanehisa, Susumu Goto, Shuichi Kawashima, Yasushi Okuno, and Masahiro Hattori. The kegg resource for deciphering the genome. *Nucleic acids research*, 32(suppl_1):D277–D280, 2004.

[113] Minoru Kanehisa, Susumu Goto, Masahiro Hattori, Kiyoko F Aoki-Kinoshita, Masumi Itoh, Shuichi Kawashima, Toshiaki Katayama, Michihiro Araki, and Mika Hirakawa. From genomics to chemical genomics:

new developments in kegg. *Nucleic acids research*, 34(suppl_1):D354–D357, 2006.

[114] Minoru Kanehisa, Michihiro Araki, Susumu Goto, Masahiro Hattori, Mika Hirakawa, Masumi Itoh, Toshiaki Katayama, Shuichi Kawashima, Shujiro Okuda, Toshiaki Tokimatsu, et al. Kegg for linking genomes to life and the environment. *Nucleic acids research*, 36(suppl_1):D480–D484, 2007.

[115] Minoru Kanehisa, Susumu Goto, Miho Furumichi, Mao Tanabe, and Mika Hirakawa. Kegg for representation and analysis of molecular networks involving diseases and drugs. *Nucleic acids research*, 38(suppl_1):D355–D360, 2010.

[116] Minoru Kanehisa. Toward understanding the origin and evolution of cellular organisms. *Protein Science*, 28(11):1947–1951, 2019.

[117] Minoru Kanehisa, Yoko Sato, Miho Furumichi, Kanae Morishima, and Mao Tanabe. New approach for understanding genome variations in kegg. *Nucleic acids research*, 47(D1):D590–D595, 2019.

[118] Minoru Kanehisa, Miho Furumichi, Mao Tanabe, Yoko Sato, and Kanae Morishima. Kegg: new perspectives on genomes, pathways, diseases and drugs. *Nucleic acids research*, 45(D1):D353–D361, 2017.

[119] Minoru Kanehisa, Susumu Goto, Yoko Sato, Masayuki Kawashima, Miho Furumichi, and Mao Tanabe. Data, information, knowledge and principle: back to metabolism in kegg. *Nucleic acids research*, 42(D1):D199–D205, 2014.

[120] Wikipedia contributors. Kegg — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=KEGG&oldid=957775243, 2020. [Online; accessed 24-September-2020].

[121] EMBL-EBI. Chebi. https://www.ebi.ac.uk/chebi/, 2020. [Online; accessed 24-September-2020].

[122] XTMS. Extended metabolic spaces. https://xtms.micalis.inrae.fr/downloads/, 2020. [Online; accessed 24-September-2020].

[123] Rhea. Rhea. https://www.rhea-db.org/download, 2020. [Online; accessed 24-September-2020].

[124] Anne Morgat, Thierry Lombardot, Kristian B Axelsen, Lucila Aimo, Anne Niknejad, Nevila Hyka-Nouspikel, Elisabeth Coudert, Monica Pozzato, Marco Pagni, Sébastien Moretti, et al. Updates in rhea—an expert curated resource of biochemical reactions. *Nucleic acids research*, page gkw990, 2016.

[125] MetaNetX. Metanetx: Automated model construction and genome annotation for large-scale metabolic networks. https://www.metanetx.org/, 2020. [Online; accessed 24-September-2020].

[126] Sébastien Moretti, Olivier Martin, T Van Du Tran, Alan Bridge, Anne Morgat, and Marco Pagni. Metanetx/mnxref–reconciliation of metabolites and biochemical reactions to bring together genome-scale metabolic networks. *Nucleic acids research*, 44(D1):D523–D526, 2016.

[127] Mathias Ganter, Thomas Bernard, Sébastien Moretti, Joerg Stelling, and Marco Pagni. Metanetx. org: a website and repository for accessing, analysing and manipulating metabolic networks. *Bioinformatics*, 29(6):815–816, 2013.

[128] Thomas Bernard, Alan Bridge, Anne Morgat, Sébastien Moretti, Ioannis Xenarios, and Marco Pagni. Reconciliation of metabolites and biochemical reactions for metabolic networks. *Briefings in bioinformatics*, 15(1):123–135, 2014.

[129] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019.

[130] Chollet Francois. Deep learning with python, 2017.

[131] Roberto Todeschini and Viviana Consonni. *Handbook of molecular descriptors*, volume 11. John Wiley & Sons, 2008.

[132] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

[133] Harry L Morgan. The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. *Journal of Chemical Documentation*, 5(2):107–113, 1965.

[134] Steven Kearnes, Kevin McCloskey, Marc Berndl, Vijay Pande, and Patrick

Riley. Molecular graph convolutions: moving beyond fingerprints. *Journal of computer-aided molecular design*, 30(8):595–608, 2016.

[135] Bowen Liu, Bharath Ramsundar, Prasad Kawthekar, Jade Shi, Joseph Gomes, Quang Luu Nguyen, Stephen Ho, Jack Sloane, Paul Wender, and Vijay Pande. Retrosynthetic reaction prediction using neural sequence-to-sequence models. *ACS central science*, 3(10):1103–1113, 2017.

[136] Hans Kraut, Josef Eiblmaier, Guenter Grethe, Peter Löw, Heinz Matuszczyk, and Heinz Saller. Algorithm for reaction classification. *Journal of chemical information and modeling*, 53(11):2884–2895, 2013.

[137] Matthew A Kayala, Chloé-Agathe Azencott, Jonathan H Chen, and Pierre Baldi. Learning to predict chemical reactions. *Journal of chemical information and modeling*, 51(9):2209–2222, 2011.

[138] Matthew A Kayala and Pierre Baldi. Reactionpredictor: prediction of complex chemical reactions at the mechanistic level using machine learning. *Journal of chemical information and modeling*, 52(10):2526–2540, 2012.

[139] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[140] RDKit. Rdkit: Open-source cheminformatics software, 2020. [Online; accessed 16-September-2020].

[141] Aurélien Géron. Keras. https://keras.io/, 2020. [Online; accessed 17-September-2020].

[142] Google Brain. Tensorflow. https://www.tensorflow.org/, 2020. [Online; accessed 17-September-2020].

[143] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283, 2016.

[144] Holly Hedegaard, Arialdi M Miniño, Margaret Warner, et al. Drug overdose deaths in the united states, 1999-2019. *CDC*, 2020.

[145] Centers for Disease Control, Prevention, et al. Still not enough naloxone where it's most needed. *Vital Signs*, 2019.

[146] Alex Wang and Aaron S. Kesselheim. Government patent use to address the rising cost of naloxone: 28 u.s.c. § 1498 and evzio. *Journal of Law, Medicine & Ethics*, 46(2):472–484, 2018.

[147] Ravi Gupta, Nilay D Shah, and Joseph S Ross. The rising price of naloxone—risks to efforts to stem overdose deaths. *New England Journal of Medicine*, 375(23):2213–2215, 2016.

[148] Mary Ann A Endoma-Arias, D Phillip Cox, and Tomas Hudlicky. General method of synthesis for naloxone, naltrexone, nalbuphone, and nalbuphine by the reaction of grignard reagents with an oxazolidine derived from oxymorphone. *Advanced Synthesis & Catalysis*, 355(9):1869–1873, 2013.

[149] Peter-John Holt, Neil C Bruce, and Christopher R Lowe. Bioluminescent assay for heroin and its metabolites. *Analytical chemistry*, 68(11):1877–1882, 1996.

[150] William R McClure. Rate-limiting steps in rna chain initiation. *Proceedings of the National Academy of Sciences*, 77(10):5634–5638, 1980.

[151] Gary K Ackers, Alexander D Johnson, and Madeline A Shea. Quantitative model for gene regulation by lambda phage repressor. *Proceedings of the National Academy of Sciences*, 79(4):1129–1133, 1982.

[152] Nicholas Horvath, Michael Vilkhovoy, Joseph A Wayman, Kara Calhoun, James Swartz, and Jeffrey D Varner. Toward a genome scale sequence specific dynamic model of cell-free protein synthesis in escherichia coli. *Metabolic engineering communications*, 10:e00113, 2020.

[153] David M Bassen, Michael Vilkhovoy, Mason Minot, Jonathan T Butcher, and Jeffrey D Varner. Jupoets: a constrained multiobjective optimization approach to estimate biochemical model ensembles in the julia programming language. *BMC systems biology*, 11(1):1–11, 2017.

[154] Michael Vilkhovoy, David Dai, Sandra Vadhin, Abhinav Adhikari, and Jeffrey D Varner. Absolute quantification of cell-free protein synthesis metabolism by reversed-phase liquid chromatography-mass spectrometry. *JoVE (Journal of Visualized Experiments)*, (152):e60329, 2019.

[155] Marco Mauri and Stefan Klumpp. A model for sigma factor competition in bacterial cells. *PLoS computational biology*, 10(10):e1003845, 2014.