



Original Research Article

A pipeline for testing drug mechanism of action and combination therapies: From microarray data to simulations via Linear-In-Flux-Expressions



Testing four-drug combinations for tuberculosis treatment

Christopher Denaro ^{a,*}, Nathaniel J. Merrill ^b, Sean T. McQuade ^a, Logan Reed ^c, Karim Azer ^d,
Benedetto Piccoli ^{a,c}

^a Center for Computational and Integrative Biology, Rutgers Camden, 201 S. Broadway, Camden, 08102, NJ, USA

^b Pacific Northwest National Laboratory, 902 Battelle Blvd, Richland, 99254, WA, USA

^c Department of Mathematical Sciences, Rutgers Camden, 311 N. Fifth Street, Camden, 08102, NJ, USA

^d Axcella, 840 Memorial Drive, Cambridge, 02139, MA, USA

ARTICLE INFO

MSC:
0000
1111

Keywords:
Linear in flux expressions (LIFE)
QSP
Computational pipeline
Metabolic networks

ABSTRACT

Computational methods are becoming commonly used in many areas of medical research. Recently, the modeling of biological mechanisms associated with disease pathophysiology have benefited from approaches such as Quantitative Systems Pharmacology (briefly QSP) and Physiologically Based Pharmacokinetics (briefly PBPK). These methodologies show the potential to enhance, if not substitute animal models. The main reasons for this success are the high accuracy and low cost.

Solid mathematical foundations of such methods, such as compartmental systems and flux balance analysis, provide a good base on which to build computational tools. However, there are many choices to be made in model design, that will have a large impact on how these methods perform as we scale up the network or perturb the system to uncover the mechanisms of action of new compounds or therapy combinations.

A computational pipeline is presented here that starts with available-omic data and utilizes advanced mathematical simulations to inform the modeling of a biochemical system. Specific attention is devoted to creating a modular workflow, including the mathematical rigorous tools to represent complex chemical reactions, and modeling drug action in terms of its impact on multiple pathways. An application to optimizing combination therapy for tuberculosis shows the potential of the approach.

1. Introduction

The field of mechanistic modeling lies at the intersection of mathematical and biological sciences [1–4]. The aim of mechanistic models is to capture and simulate underlying biological systems which are relevant to an applied problem. This field has been expanding as the availability of computational power and mechanistic data becomes more prevalent [5–9]. There has also been an increase in big data in the systems biology field, which makes establishing model priors more readily available. Note that mechanistic models contrast artificial intelligence models as they are typically built with reference to known biological systems — contrasting the “black box” behavior of artificial intelligence models. In this paper, we will introduce a pipeline that was created with the intention to investigate drug mechanism of action for single and combination therapies.

There exist many approaches to mechanistic modeling, and this area is rich and shows promises of expanding [10,11]. There has been a call

for this type of modeling as it offers a cheaper alternative to running translational and pre-clinical studies. Moreover, the accessibility of large genomic datasets makes populating necessary model priors more feasible.

Prior research in this field includes several methods of modeling — some more easily tied to underlying biology than others. Graph theory approaches, ordinary differential equation approaches, boolean network modeling, and agent-based modeling all serve to offer a foundation for mechanistic models [2]. In this paper, we will discuss a pipeline that uses a variation of the typical Flux-Balance-Analysis and ordinary differential equations approach.

As a motivation for the necessity of this pipeline, there is an ever-present interest in model-informed drug development (MIDD) [10,11]. Traditional methods of drug development are costly and time-consuming and do not translate well for some diseases. Some treatments for diseases also lack robust biomarkers for drug-target

* Corresponding author.

E-mail address: cad373@camden.rutgers.edu (C. Denaro).

engagement, making it difficult to gauge treatment performance during clinical trials [12]. Mechanistic modeling offers a possibility to alleviate both of these issues. For instance, it may be infeasible for certain diseases to have all or most of the potential treatments tested in a pre-clinical or translational model [13,14]. Specifically, the treatment for Tuberculosis may involve a combination of 3–4 antibiotics over the course of 6 months. In order to demonstrate how quickly the treatment space for combination therapies increases, we show an example of the number of possible 3 and 4-treatment combinations if there are a total of 20 treatments. If there are 20 potential antibiotics, then there are 20 choose 3 (1140) plus 20 choose 4 (4845), resulting in 5985 potential combinations to test. Mechanistic modeling offers an avenue to narrow down this wide field of potential combinations, aid in combination design, and prioritize combinations to test in pre-clinical models, hence de-risking and informing the translation into the clinic.

In the following sections, we will describe a conceptual, computational pipeline that describes how to capture pathologically-relevant metabolic pathways and utilizes available genetic data to predict treatment efficacy. As expected, the pipeline requires two key types of data: genetic data for gene expression levels and metabolic pathway data (e.g. RNAseq and metabolomics). This pipeline is built off of a variation of FBA techniques known as linear-in-flux-expressions (LIFE) which encompass more complex dynamics with respect to traditional FBA [15]. A description of the LIFE method can be found in Section 3.4. The LIFE methodology writes a system of ordinary differential equations as $\dot{x} = S(x) \cdot f$, as opposed to $\dot{x} = S(f) \cdot x$, leveraging the natural linearity of biological systems w.r.t. fluxes when encoding metabolic networks using hypergraphs. Non-linearities are captured via the implementation of Michaelis-Menten kinetics. In turn, the approach allows a better computational characterization of system equilibria and simulations [16,17].

The computational pipeline allows for the simulation of single and combination therapies over metabolic pathways and is composed of multiple stages. We begin the paper with a Section 2 that offers a rough overview of each of the steps, see Fig. 1. This serves as a soft introduction to the language of the paper. The following Section 3 provides a detailed explanation of these same steps as in Fig. 1. Then, in Section 4 we provide a concrete example by applying the pipeline to a specific problem: the treatment selection for Tuberculosis expanding the methods of [18].

2. Pipeline overview

In this section, we will give a cursory overview of the steps outlined in Fig. 1. Because the PBPK section follows the standard approaches in the field, we exclude a cursory overview and give a detailed explanation in component 3.

2.1. Genetic data

The computational pipeline begins with quantitative genetic data. An example of such quantitative genetic data is RNAseq or microarray data for the relative expression of genes. The genetic data must be quantitative as later in the pipeline, in the Data Curation and Drug Effect steps, it will be required to tie a specific value to genes or metabolites. Using the application of this pipeline to treatments for TB, we can create a demonstrative example.

2.2. Data curation

The next step in the pipeline is Data Curation or data processing. In most cases, this step is required because the source of the quantitative

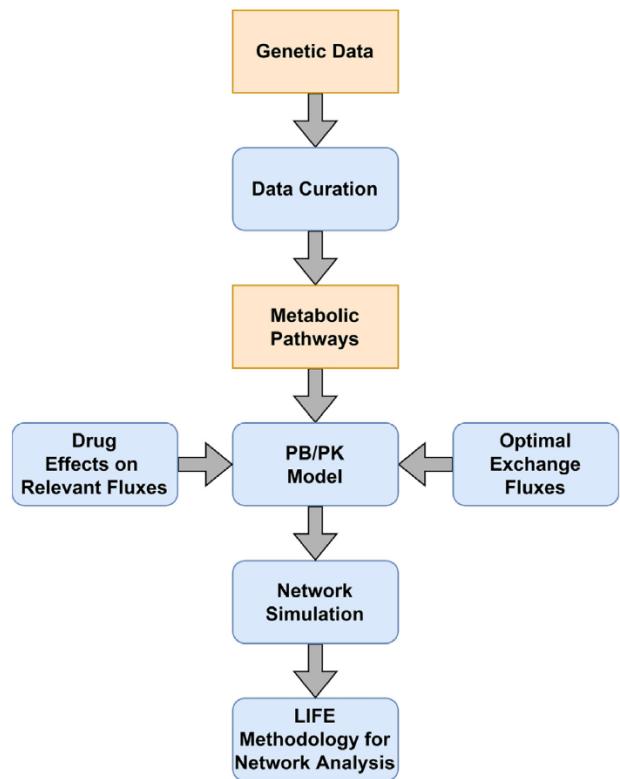


Fig. 1. This figure outlines the key components of the pipeline for evaluating combination therapy efficacy. Primary inputs into the pipeline are “Genetic Data” and “Metabolic Pathways”, highlighted in orange. Treatment PK data is a secondary input into the system.

genetic data can be noisy. This step offers an opportunity to normalize data from different sources or examine data for outliers. There are two key steps that should be carried out: (1) determining levels of significance and (2) filtering of expression data based on the agreeance of replicates. This filtering process can prove to be a useful tool in examining noisy data as it offers a simple, systematic way of determining whether or not values are noise or significant. Oftentimes, filtering is required to achieve consensus on the robustness of the differentially expressed list of genes.

Once significance has been determined, several filtering techniques can be carried out. In this section, we will outline several filtering techniques that may prove useful. The first filter we apply to the data is the “NaN filter”. If more than half of the replicates of an experiment are unknown for a gene, then this gene is not significant according to the filter. In order to determine significance, a 2-fold threshold of ± 1 was chosen. This threshold corresponds with a gene having more than two times (or less than one-half) of the expression value of the control. The following filters described reference this 2-fold threshold. The “Mean filter” deems microarray values to be significant if the mean of the replicates is significant according to the 2-fold threshold. The “Consensus filter” deems microarray values to be significant if more than half of the replicates are significant according to the 2-fold threshold. Significantly differentially expressed genes can be extracted from microarray data using standard approaches such as Significance Analysis of Microarrays (SAM) as well.

2.3. Metabolic pathways

The pipeline requires the input of a metabolic network. This can take many forms and is extendable to any generic network with nodes, edges, hyperedges, and uberedges. In practice, we also require a mapping between the genes from the data curation step and the nodes and

edges from the network. There are several online tools that accomplish this, but perhaps the most well-known is the Kyoto Encyclopedia of Genes and Genomes (KEGG) [19–21].

Using KEGG allows the linking between the genes and metabolic pathways. The metabolic pathways in KEGG are networks that can be represented by directed graphs, where the nodes represent metabolites and the edges represent metabolic reactions. This is used to identify which significant genes directly affect enzymes involved in reactions in the metabolic pathways.

Analyzing significantly affected genes and their corresponding pathways reveals which pathways are of interest.

2.4. Drug effects on relevant fluxes

Once a metabolic network has been imported, we can then move on to implementing drug effects for the network. Drug effects are implemented as uberedges — edges that inhibit or promote other edges. Uberedges allow us to implement drugs/treatments as regulators of metabolic processes.

2.5. Optimal exchange fluxes

Prior to simulation, there is a critical step that includes the “completion” of a network. In many networks, there are two cases in which one cannot simply establish the system $\dot{x} = S(x) \cdot f$ while expecting biologically relevant results. (1) There may be nodes that do not have any incoming edges. If we were to initialize values for these nodes and then run the simulation according to the above expression, these nodes would simply deplete and then offer no dynamics. This behavior does not reflect the underlying biology. (2) In a similar case, there may be nodes with no outgoing edges. In this case, these nodes would simply accumulate in an unbounded way.

To address this situation, we optimize the exchange fluxes or “complete” the networks. The rough algorithm is as follows: We iterate through the nodes in the network and identify each node that has no incoming edges. For each node of this type, we create a “virtual” source going into the node. We then iterate through the network again, identifying all nodes without an outgoing edge. Again, for each of these nodes, we add a “virtual” sink going out of the node. If no nodes require a virtual sink, we traverse the network looking for the longest non-cyclic path, we then add a virtual sink at the end of this path — this ensures that we do not trivially accumulate mass. Finally, we check that if a node requires an upstream hyperedge to connect to a source, that all leading nodes in the hyperedge can also reach a source — this ensures that we do not trivially deplete a leading hyperedge node.

We assert that this process of adding “virtual” nodes, as sources and sinks, is representative of the underlying biology. For the sake of explanation, we will consider the case in which a metabolic network contains a node (metabolite) that does not have an outgoing edge or is not connected to a sink, while it is connected to a source. In this case, if we assume that the metabolic network is accurate to the underlying biology, then the metabolite level for that node would simply accumulate without bound — this behavior is likely not representative of healthy metabolic systems. It is much more likely that a biological process not captured by the metabolic network depletes this metabolite in some way; hence we add a “virtual” node acting as a sink for that metabolite. More precisely, such virtual nodes need to be added to the terminal components to which the node is connected. A very similar argument can be made for adding a “virtual” source node to metabolites that have no incoming edges.

2.6. Network simulation

This pipeline has two main capabilities: (1) Static ranking of treatment effects on a collection of pathways and (2) Simulations of network

dynamics. In the static ranking of effects on a collection of pathways, we can consider the overall impact that a treatment has based on the total gene expression change. In this way, each treatment can be assigned a ‘static score.’ With this format, treatments may be combined and compared to ensure the static scores reflect treatment-treatment interactions. The dynamic simulations allow for a singular treatment to be dynamically simulated over a pathway. Metabolites are initialized, then treatment effects are considered according to 2.4, then the system evolves according to $\dot{x} = S(x) \cdot f$. The dynamic simulations may also surface additional treatment effects in the network that were unknown, or specific metabolites that are significantly modified by the treatment.

2.7. LIFE methodology for network analysis

The inclusion of hyperedges and uberedges in the representation of metabolic networks allows for a greater level of detail to be captured through simulation. In Section 3.8 we highlight the difference between a network representation that only uses simple edges and the equivalent network using hyperedges.

LIFE expressions allow for a great level of analysis of metabolic networks [22]. As the name implies, LIFE networks are able to be scaled with a large number of edges — this allows for the capture of intricate networks with large amounts of interaction between nodes.

Traditionally, for Flux Balance Analysis (FBA), the null space of the resulting matrix offers a lot of information about the network. The null space of the system relates to the equilibria of the network. We have a similar result in the LIFE setting. We can compute the intersection of the positive orthant with the null space to observe only equilibria that correspond to positive network fluxes — this guarantees that biological reactions are respected.

3. Pipeline in detail

3.1. Genetic data

The pipeline requires a source of genetic data. Specifically, the genetic data must come from a source in which there is a “control” group and a “treated” group. This setting allows for relative expression data to be derived. In practice, the data need not be sourced from a specific experiment with a treatment group and control group. For instance, we may have genetic data sampled from a population that suffers from a disease — this could then be compared to (for the purpose of relative expression) against a population of healthy “controls”.

There are two major sources for data of this type: microarray data and RNAseq data. The way in which the genetic data is obtained is not necessarily important; however, a key requirement for the data is that it can be tied back to metabolic pathway information. For microarray data, this is typically done as wells are labeled with specific genes. With RNAseq data, it requires mapping reads back to the genome or transcriptome. We will touch on available methods for associating metabolic pathway information to genetic information in Section 3.3 — but the general picture is that online tools provide valuable resources for this mapping.

3.2. Data curation

Prior to implementation in the pipeline, the data is expected to have undergone some sort of quality control process — or data pre-processing. By nature, this type of data is susceptible to a lot of noise and low signal-to-noise ratio [23]. This can make it difficult to use the raw data directly and so we require a step to ensure that our data is well-behaved. In this section, we will outline several ways in which both microarray data and RNAseq data may need to be processed prior to implementation in the pipeline.

Microarray data can be noisy, and therefore we require some significance level cutoff to be set in order to label genes significantly affected

or not. How this level is determined can vary depending on the setting of both the experiment and the use case of this pipeline; however, we will outline the methods described previously. For experiments with multiple replicates of groups, it may be common for several replicates to offer differing results when examined with a simple “significance level cutoff”. In order to collate data from replicated experiments, we can apply several different types of filters to the data, providing a repeatable method of determining gene significance with replicated experiments. Note that the role of the filters is to determine the binary significance of a specific gene — this binary significance label facilitates easier integration of DE scores into the network; however, during the computation of the L_1 score and during the dynamic simulations, we will use the relative differential expression score for the significant genes. In practice, filters are implemented as binary masks over matrices that hold drug data and gene expression data. This allows for flexibility as multiple filters can be combined. Oftentimes with microarray data, valid data at a specific well may not be read. In cases like these, datasets are either left blank for the gene or are denoted “NaN” for “not a number” to mark that no data is available for that gene. With replicated experiments, some replicates may fail to read genes in which other experiments yield valid data. The use of a “NaN” filter is beneficial in this case. We can discard the significance of genes if fewer than half of the replicates have viable data. This helps to ensure that a noisy significant reading does not determine the significance of a gene — especially if other replicates fail to have a valid reading. We can also collate the replicates through the use of a “Mean” filter. As the name implies, the mean filter simply takes the average of the observed values of a gene over the replicates. In the implementation, replicates with no reading can either contribute 0 to the expression score or can be left out of the sum. In previous implementations of this pipeline, the second scenario was implemented. Namely, if we use the mean filter on a drug with multiple replicates, we only take the sum of the scores with valid readings, excluding replicates in which genes do not read. Additionally, a “Consensus” filter can be implemented. Using a consensus filter, a gene is labeled significant if over half of the replicates are above the “significance level cutoff”. This pipeline allows for both restrictive and loose filtering via the combination of the above filters.

RNAseq data that is published online can have different forms depending on what part of the RNAseq processing pipeline it originates from [24]. If RNAseq is already given in differentially expressed format then nothing more than setting a “significance level cutoff” for the expression level is needed. If an extra restriction is required, cutoff levels may be set for a false discovery rate (FDR) as well.

3.3. Metabolic pathways

Perhaps one of the most important components in the pipeline is the metabolic pathway. The simulation and analysis with the LIFE methodology is only as strong as the quality of the metabolic network. Metabolic pathways are implemented as graphs or networks. The metabolites and reactions of the pathway are implemented as nodes and edges, respectively. A reaction that has either multiple substrates or multiple products can be implemented via a hyperedge in the graph — this concept will be explained further in the following section. As one might expect, a fine-grain network will provide more precise information than a network with fewer nodes.

In order to use metabolic pathway information, it is a strict requirement that the genetic data be tied to the metabolic pathway information. In other words, we must be able to determine which metabolites the significantly affected genes affect. In practice, each gene may not be able to be mapped to a metabolic pathway in the case when gene data cannot be mapped to the metabolic pathway, we assume that the drug does not affect the pathway in a significant way and so we assume “normal” behavior of the network. In order to map the genes to the metabolites in the pathway, we can rely on

online tools and databases such as the Kyoto Encyclopedia of Genes and Genomes (KEGG) and Ingenuity Pathway Analysis (IPA). Specifically, KEGG offers a link to “DBGET” in which gene ids are searched for function in multiple databases.

After the completion of mapping gene ids to metabolic pathways, we are able to carry out preliminary analyses. We can order pathways based on the number of significantly affected genes that they contain. This gives only a very rough idea of which pathways are most significantly affected by the treatment. Pathways with the most occurrences of significantly affected genes can be examined in further detail — offering good candidates for network simulations. The process outlined above is very simple to carry out; there are other methods of determining which pathways are most significantly affected, specifically when we have RNAseq data. Gene Set Enrichment Analysis (GSEA) is a great tool for identifying significantly affected pathways and is standard in many RNAseq data pipelines. Many R packages contained within the library “Bioconductor” offer implementations of functions that enable the extraction of information from gene expression data, which are useful in determining which pathways to examine in further detail.

3.4. Drug effects on relevant fluxes

We model metabolic networks using an extended form of Flux Balance Analysis (briefly FBA) that we call Linear-In-Flux-Expressions (briefly LIFE). Generalizing the dynamics of FBA, we focus on systems of the following form:

$$\dot{x} = S(x) \cdot f, \quad (1)$$

where the stoichiometric matrix S is different from classical FBA models because it depends on x , and f is the column vector of fluxes. The values x_1, x_2, \dots represent the mass of metabolites or drug concentrations in various compartments in PBPK models, and so are constrained to \mathbb{R}_+ .

A number of mathematical approaches were shown to apply to such systems as (1). A detailed analysis is provided in [17] that combine techniques such as *continuous-time Markov chains* which describe linear systems without intakes nor excretions [25], and *compartmental systems* which describe linear systems with intakes and excretions [26,27]. Nonlinear systems are analyzed using the results of [28]. Metabolic networks contain exchange fluxes representing incoming mass from other parts of the network, or from the outside environment, and they are necessary for the existence of equilibria. For traditional metabolic networks described by a directed graph $G = (V, E)$, directed edges from a virtual node v_0 which acts as a source to some node in the graph are called intakes, and edges from a node in the graph to some virtual node v_{n+1} acting as a sink are called excretions.

In order to introduce our more general class of networks, we first define our dynamics when associated with a simple directed graph $G = (V, E)$ $V = \{v_1, \dots, v_n\}$ the set of nodes and $E \subset V \times V$ the set of directed edges, so that we may expand on this later. $S_{ve}(x)$ indicates the matrix entry corresponding to node v and edge e . The network equilibria correspond to the null space of $S(x)$ and thus depend both on the fluxes and metabolite levels.

Definition 1. Given a directed graph $G = (V, E)$, an edge $(v_0, v_1) \in E$ for the virtual node v_0 , the edge (v_0, v_1) is called an intake, and v_1 is an intake node. For the virtual node v_{n+1} an edge (v_j, v_{n+1}) is called an excretion, and v_j is an excretion node. The set of intake nodes is denoted I , and the set of excretion nodes is denoted J .

A general assumption is the following:

$$S_{ve}(x) = \begin{cases} -F_e(x_v) & e = (v, w), v \in V, \\ & w \in V \cup \{v_{n+1}\} \\ F_e(x_w) & e = (w, v), w \in V \\ 1 & e = (v_0, v) v \in I \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

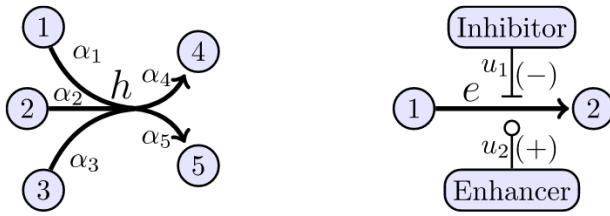


Fig. 2. The depicted features were added to LIFE dynamics in order to include important features of biochemical reactions. Left: A hyperedge h which connects three reactants to two products in a metabolic reaction. Each reactant and product has a weight corresponding to the stoichiometry of the reaction. Right: An enhancer(inhibitor) molecule promotes(inhibits) a chemical reaction. The edges u_1, u_2 are called “überedges” and connect a metabolite or drug to an edge e .

where $F_e : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is differentiable, strictly increasing, with $F_e(0) = 0$, for $\mathbb{R}_+ = \{x \in \mathbb{R} : x \geq 0\}$. (2) is a natural assumption on the system; the flow from a metabolite will depend only on that metabolite, but a metabolite with multiple edges may have different kinetics with each reaction. One can model nonlinear kinetics such as Michaelis-Menten [29] corresponding to Hill functions $F_v(x_v) = \frac{x_v^p}{K+x_v^p}$ with $p \in \mathbb{N}$. Each column corresponds to an edge (which corresponds to a reaction), and all columns of S have zero sum, except those representing intakes or excretions, which have positive and negative sum, respectively. Therefore, the dynamics (1) necessarily respect the conservation of mass. To be biologically meaningful, we further restrict to equilibria for which all components of f and x are positive, thus rendering the problem nonlinear even for linear dynamics. A previous result shows that with dynamics defined as (2), the existence of equilibria depends on the structure of the network.

We defined similar dynamics as (2) for a larger class of graphs that we call, *metabolic graphs* [18]. The more general features of metabolic graphs are designed to address two main limitations of modeling with simple graphs. 1. to capture, in a more natural way, the stoichiometry of reactions involving more than two compounds (left side of Fig. 2); 2. to model the effects of drugs enhancing or inhibiting reaction rates via their effect on gene expression of enzymes that control a reaction (right side of Fig. 2). Metabolic graphs are a combination of two ideas for additional structure: hypergraphs [30–32] which contain directed hyperedges with multiple initial and/or terminal nodes, and übergraphs [33] which include überedges connecting a node to other edges. Let us specify the key structure we will utilize going forward.

Definition 2. A hyperedge h is a set of nodes connected to each other, i.e. $h \in \mathcal{P}(V) \setminus \{\emptyset\}$, where $\mathcal{P}(V)$ is the powerset of V . Note that the set $h = \{v_i\}$ indicates a loop edge connecting v_i to itself.

Because chemical reactions of a metabolic network indicate a direction of flow, we require hypergraphs to include a direction.

Definition 3. A directed hyperedge is an ordered pair of two subsets of nodes, i.e. $h = (X, Y)$ with $X \in \mathcal{P}(V)$ or $X = \{v_0\}$, $Y \in \mathcal{P}(V \cup \{v_{n+1}\})$, where v_0 (v_{n+1}) is a virtual node called the source (sink). Elements of X (Y) are called initial nodes (terminal nodes) for the hyperedge h . The set of directed hyperedges is denoted \mathcal{H} .

When using a hyperedge to model reacting metabolites forming a product, we encode the stoichiometry relationship on the hyperedge via edge weights. It is convenient to have a notation for the cardinality of the sets of initial and terminal nodes for a hyperedge.

Definition 4. Given a directed hyperedge $h = (X, Y)$, with $X, Y \in \mathcal{P}(V)$, the indegree of h is defined as

$$d_{in}(h) = |X|, \quad (3)$$

and the outdegree of h is defined as

$$d_{out}(h) = |Y| \quad (4)$$

where $|\cdot|$ indicates the cardinality of a set.

Definition 5. A weighted directed hyperedge is a couple $\mathcal{H} \ni h = (X, Y)$ with $X \in \mathcal{P}(V)$ or $X = \{v_0\}$, $Y \in \mathcal{P}(V \cup \{v_{n+1}\})$, and corresponding weights $\Psi_h : h \mapsto (\Psi_h^{out}, \Psi_h^{in})$ where $\Psi_h^{out} : X \mapsto \mathbb{R}_+$ and $\Psi_h^{in} : Y \mapsto \mathbb{R}_+$.

We specify a mathematical representation of inhibitors and enhancers. More precisely, we want to consider metabolites influencing a given reaction. This can be captured by a new type of generalized edges, called an überedge linking nodes to directed hyperedges.

Definition 6. An e-i-überedge is a couple $u = (v, h)$ with $v \in V, h \in \mathcal{H}$. We denote the set of e/i-überedges by \mathcal{U} .

In metabolic graphs, an e/i-überedge will have a sign associated with it, to distinguish between their action as enhancer (+) or inhibitor (-). This subset of überedges from a node to a hyperedge are depth 2 überedges according to the general definition of übergraph (see Appendix). We now have defined the components of metabolic networks.

Definition 7. A metabolic graph is a weighted directed hypergraph endowed with signed depth-2 überedges connecting nodes to hyperedges. More precisely, a metabolic graph is an ordered quintuplet $G = (V, \mathcal{H}, \mathcal{U}, \Psi_H, \Psi_U)$ where $V = \{v_1, \dots, v_n\}$ is the set of nodes, \mathcal{H} is the set of directed hyperedges, $\Psi_H = \{\Psi_h : h \in \mathcal{H}\}$ is the set of functions assigning weights to hyperedges, \mathcal{U} is the set of e/i-überedges and $\Psi_U : \mathcal{U} \mapsto \{+, -\}$.

We assume that for every node v and hyperedge $h = (X, Y)$, with $v \in X$, the following holds. Let U_h be the set of nodes w such that there exists e/i-überedge $(w, h) \in \mathcal{U}$, then we have:

$$S_{vh}(x) = \begin{cases} -\alpha_v \mathbf{F}_h(x) \mathbf{K}_h(x) & v \in X \\ \alpha_v \mathbf{F}_h(x) \mathbf{K}_h(x) & v \in Y \\ 1 & X = \{v_0\}, v \in Y, \\ 0 & \text{otherwise,} \end{cases} \quad (5)$$

where $\alpha_w = \Psi_h^{in}(w)$ if $w \in X$ and $\alpha_w = \Psi_h^{out}(w)$ if $w \in Y$ are the stoichiometric coefficients, $\mathbf{F}_h : \mathbb{R}_{+}^{d_{in}(h)} \rightarrow \mathbb{R}_{+}$ is given by

$$\mathbf{F}_h(x) = \min_{w \in X} \left\{ F_{w,h}(x_w) \frac{1}{\alpha_w} \right\}, \quad (6)$$

$F_{w,h} : \mathbb{R}_{+} \rightarrow \mathbb{R}_{+}$ quantifies the potential flow of metabolite x_w due to reaction h , and

$$\mathbf{K}_h = \prod_{w \in U_h} K_{(w,h)}(x_w), \quad (7)$$

where $K_{(w,h)} : \mathbb{R}_{+} \rightarrow \mathbb{R}_{+}$ quantifies the action of metabolite x_w on h , with the convention that $K_h = 1$ if $U_h = \emptyset$.

The stoichiometric coefficients $\alpha_w \in \mathbb{R}$ for the reaction corresponding to hyperedge h are normalized such that $\sum_{w \in X} \alpha_w = 1$ and $\sum_{w \in Y} \alpha_w = 1$. The functions $F_{w,h}$, $w \in Y$ are continuously differentiable. The functions $K_{(w,h)}$, $w \in U_h$, are continuously differentiable, monotonic with $K_{(w,h)}(0) = 1$. More precisely if $\Psi_U((w, h)) = +$ then $K_{(w,h)}$ is increasing (enhancer case), otherwise $K_{(w,h)}$ is decreasing (inhibitor case).

Hyperedges and überedges can be used to more accurately model metabolic networks. Using hyperedges, we are able to represent chemical reactions with multiple substrates and/or products in a way that respects the relationship between substrates and products. The use of überedges enables us to capture the relationship between drug levels and effects on the metabolic network. More explicitly: a node is defined for drug level, then überedges are created, modifying chemical reactions that are regulated by genes that are significantly differentially expressed. Additionally, the strength of the überedge is determined by the DE score for the gene.

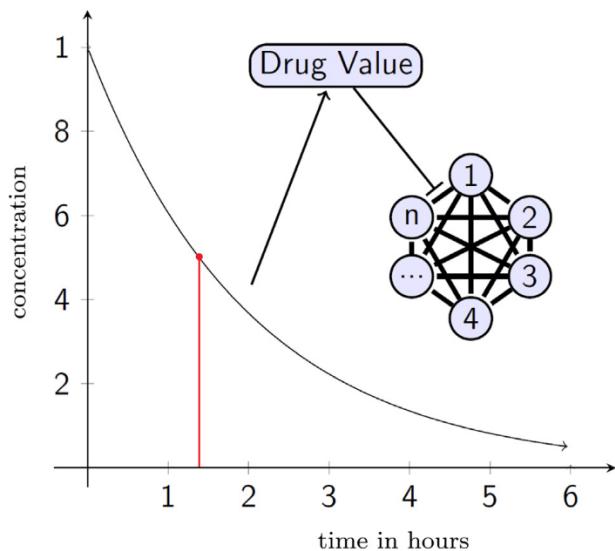


Fig. 3. An example of how we use pharmacokinetic (PK) data such as half-life and maximum drug concentration to simulate drug effects within the network. The half-life is indicated with the red line.

3.5. Optimal exchange fluxes

As mentioned previously, prior to simulation we must “complete” the metabolic network. Below, we will provide a general approach to the completion of a network. Recall that the simulation stage of the pipeline is a set of ordinary differential equations represented by $\dot{x} = S(x) \cdot f$, where x corresponds to the metabolites, f corresponds to the fluxes on the reactions, and S is a matrix capturing the stoichiometry of the network. If we were to attempt to simulate the flow of mass over the network, we would observe the complete depletion of several nodes — namely, those without any incoming edges, or sources. We would also observe a trivial accumulation of mass in several nodes — namely those without any outgoing edges, or sinks. To ensure that we do not have this type of behavior in our simulations, we “complete” the network. Network completion involves traversing over the network several times. The first time, we traverse the network searching for nodes with no incoming edges. If we encounter a node with no incoming edges, we add a “virtual” constant source; essentially, we ensure that the node is fed by some constant rate. Once every node has at least one incoming edge, we can guarantee that no node will trivially deplete to zero mass; however, nodes with no outgoing edges will now accumulate mass with no bound (as we have added mass to the system in the form of “virtual” nodes). To ensure that we do not have an unbounded gain of mass, we traverse through the graph, keeping a list of all nodes that do not have any outgoing edges. We then add a “virtual” sink to one of the nodes in the list — we check that no nodes are able to reach the new sink, then we repeat until each node in the list is able to reach a sink. In this way, we ensure that we will not have a trivial accumulation of mass in our system when we simulate.

As described in Section 2.5, this process of adding virtual sources and sinks can be thought of as accounting for the underlying biology that the metabolic network excludes. In some situations, care may need to be taken to constrain the values of the virtual nodes to ensure that this process respects the underlying biology.

3.6. PK model

In order to determine relevant values for treatment concentration for network simulation, it is necessary to consult literature. A pharmacokinetic (PK) model is required to determine the concentration of treatment in the compartments relevant to the metabolic pathways. If wanted, a physiological component can be added, describing behavior in different compartments of interest. Literature that addresses the average concentration of the treatment in various compartments is required to determine the value of the variable corresponding to the treatment. Note, that this should not simply be dosage information, as the treatment is metabolized after it enters the body and is subsequently available at the target site — the metabolic pathways. Fig. 3 demonstrates that drug half-life data will be considered when implementing drug effect on metabolite levels.

3.7. Network simulation

Now that we have constructed a matrix that captures the treatment effect on reactions and corresponds to a completed network, we are ready to run simulations. We have a system of ODEs in the form of $\dot{x} = S(x) \cdot f$. In order to simulate the network, we numerically solve this system. Note that if the treatment regimen calls for repeated doses administered, then the system must be simulated in time segments corresponding to those treatments. Doses of treatment being administered are implemented as an increase in the “metabolite”, x_n , that corresponds to the treatment. Care must be taken in the method used for solving this system as it can become very stiff — restricting step sizes in the numerical solvers quickly.

3.8. LIFE methodology for network analysis

In this section, we outline some of the advantages of using hyperedges to encode metabolic pathways.

3.8.1. Comparison of simple and hyperedge representations

We focus on a section of the central carbon metabolism, see Fig. 4, and illustrate the differences in terms of simulation results and equilibria for representation using hyperedges or simple edges.

The system state is given by $x \in \mathbb{R}^4$ with x_1 corresponding to Pyruvate, x_2 to Acetyl CoA, x_3 to Isocitrate, and x_4 to Oxaloacetate. The dynamics is written as $\dot{x} = S_i(x) \cdot f$, where:

$$S_1 = \begin{bmatrix} a & 0 & -x_1 & 0 & 0 & 0 \\ 0 & 0 & x_1 & -F_1(x) & F_2(x) & 0 & 0 \\ 0 & 0 & 0 & F_1(x) & -F_2(x) & -x_3 & 0 \\ 0 & b & 0 & -F_1(x) & F_2(x) & 0 & -x_4 \end{bmatrix}$$

$$S_2 = \begin{bmatrix} a & 0 & -x_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x_1 & -x_2 & x_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & -x_3 & -x_3 & x_4 & -x_3 \\ 0 & b & 0 & 0 & 0 & x_3 & -x_4 & 0 & -x_4 \end{bmatrix}$$

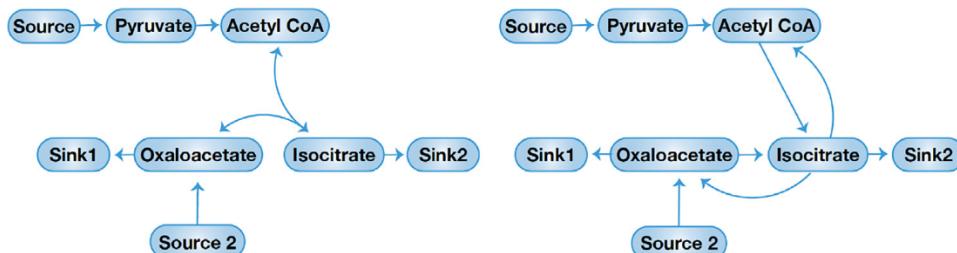
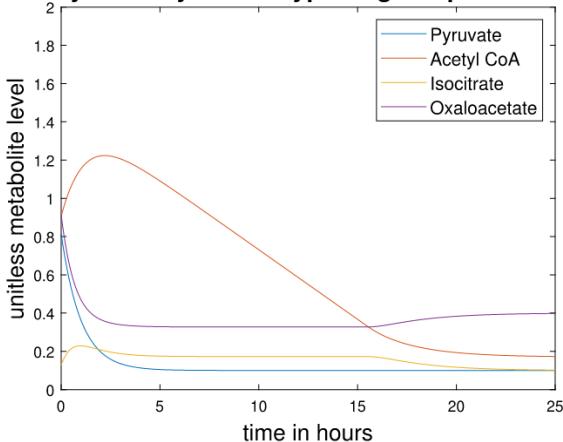


Fig. 4. Hyperedges (left) and simple edges (right) representation.

Acetyl CoA System - Hyperedge Representation



Acetyl CoA System - Simple Edge Representation

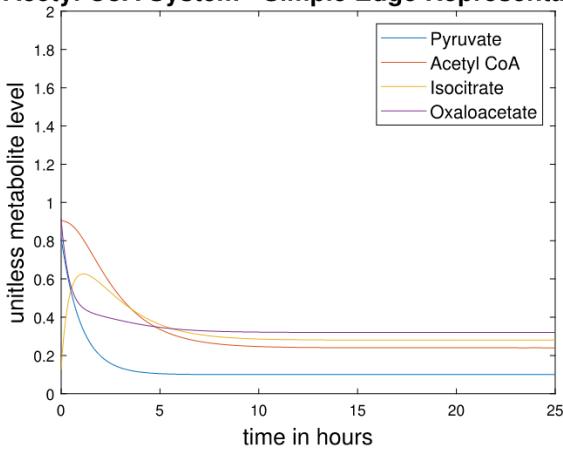


Fig. 5. Simulations for the hyperedge system (top) and the simple edge system (bottom).

with S_1 corresponding to hyperedges (Fig. 4 left), S_2 to simple edges (Fig. 4 right), $a, b > 0$ are intakes, excretions, and F_e models the reaction dynamics. We set $F_e = \min_{i \in e} F_{i,e}$, where the minimum is on x_i s incident to the edge e , $F_{i,e}$ being the corresponding flux. The hyperedge dynamics depend on all the involved metabolites in a nonlinear fashion via min.

We simulate the two systems for 25 h, see Fig. 5. For simple edges, we observe a “one-stage” evolution with exponential convergence to equilibrium. For the hyperedge system, two different phases appear. Namely, around time $t = 16$, Acetyl CoA crosses a threshold value and causes Isocitrate and Oxaloacetate to deflect to new equilibria.

3.8.2. Analysis of the space of equilibria

We illustrate how the different representations of the system produce different sets of equilibria. To do this, we assign weights to simple edges corresponding to hyperedges, and explore the whole space of weights. First introduce two variables on the edge weights: λ_1 and λ_2 . The corresponding equations and solutions for equilibria of the simple edge representation are:

$$\begin{cases} a - x_1 = 0 \\ x_1 - x_2 + x_3 = 0 \\ x_2 - x_3 - \lambda_1 x_3 + x_4 - x_3 = 0 \\ b + x_3 - \lambda_2 x_4 - x_4 = 0 \end{cases} \rightarrow$$

Table 1

Table detailing dosage information and the number of trials for various drug treatments. The rows in white correspond with 6-hour trials from [34]. The minimum inhibitory concentration (MIC) given is the minimum inhibition concentration reported in the paper [34] or [35]. Note that 3 columns corresponding to the number of replicates have been excluded to ensure readability. The number of trials is listed in the fourth column and refers to the number of times the experiment was carried out.

Drug	MIC	Dose 1	# of trials	Dose 2
Amikacin	1 ug/mL	5 ug/mL	2	10 ug/mL
Capreomycin	50 ug/ml	5 ug/mL	2	10 ug/mL
Clofazimine	1.25 ug/ml	10 ug/mL	1	13 ug/mL
EMB	0.6 ug/ml	10 ug/mL	2	
ETH	0.5 ug/ml	12 ug/mL	2	40 ug/mL
INH	0.02 ug/ml	0.2 ug/mL	1	0.4 ug/mL
Levofloxacin	1 ug/ml	10 ug/mL	2	
PZA	10 ug/ml	0.12 mg/mL	2	1.2 mg/mL
Pretomanid	0.4 ug/ml	0.2 ug/mL	2	0.4 ug/mL
Rifapentine	N.D.	0.1 ug/mL	2	0.5 ug/mL
Rifampicin	0.4 ug/ml	0.2 ug/mL	2	
Streptomycin	1 ug/ml	2 ug/mL	2	5 ug/mL
Ofoxacin	1 ug/ml	5 ug/mL	2	10 ug/mL

$$\begin{cases} x_1 = a \\ x_2 = \frac{a(1+\lambda_2)+b}{(1+\lambda_1)(1+\lambda_2)-1} + a \\ x_3 = \frac{a(1+\lambda_2)+b}{(1+\lambda_1)(1+\lambda_2)-1} \\ x_4 = \frac{b+\frac{a(1+\lambda_2)+b}{(1+\lambda_1)(1+\lambda_2)-1}}{1+\lambda_2} \end{cases}$$

As λ_1 and λ_2 vary over positive values, we obtain a surface of possible equilibria, which we then overlay to the unique equilibrium of the hyperedge system. The unique hyperedge equilibrium corresponds to a plane in the graph representation of equilibria, and the curve of intersection with the simple edge, surface gives the values of λ_1, λ_2 for which the two systems give the same equilibria. This is done for the three metabolites $x_{2,3,4}$ involved in the hyperedge dynamics, thus giving rise to three distinct curves, see Fig. 6 top. In order to better visualize the parameters λ_1, λ_2 for which the curves of intersection are defined, we project down onto the $\lambda_1 \lambda_2$ -plane, see Fig. 6 bottom. The three metabolites give rise to three distinct curves, with no common intersection. This implies that the simple edge system cannot simultaneously achieve the same equilibria as the hyperedge system. Otherwise, the three curves would intersect altogether in at least one point.

4. Application to tuberculosis

In this section, we will review the application of the pipeline to the design of Tuberculosis drug combinations.

4.1. TB: Genetic data

In the application of the pipeline to the treatment of Tuberculosis, we gathered genetic data in the form of microarray data. Genetic data were collected from the Gene Expression Omnibus (GEO) — which hosted experiments that tested the gene expression levels of *Mycobacterium Tuberculosis* when exposed to individual antibiotics. Table 1 holds information collected on 16 drugs, focusing on expression data collected at six hours post-exposure.

4.2. TB: Data curation

The genetic data included several replicates of some experiments. We determined a way to collate these experiments into one simple measure for a given antibiotic. We did this using several filters for significance. First, a drug’s impact on a gene was considered significant if the \log_2 ratio of the gene was differently expressed by greater than 1 in absolute value — note, this corresponds to either a doubling or

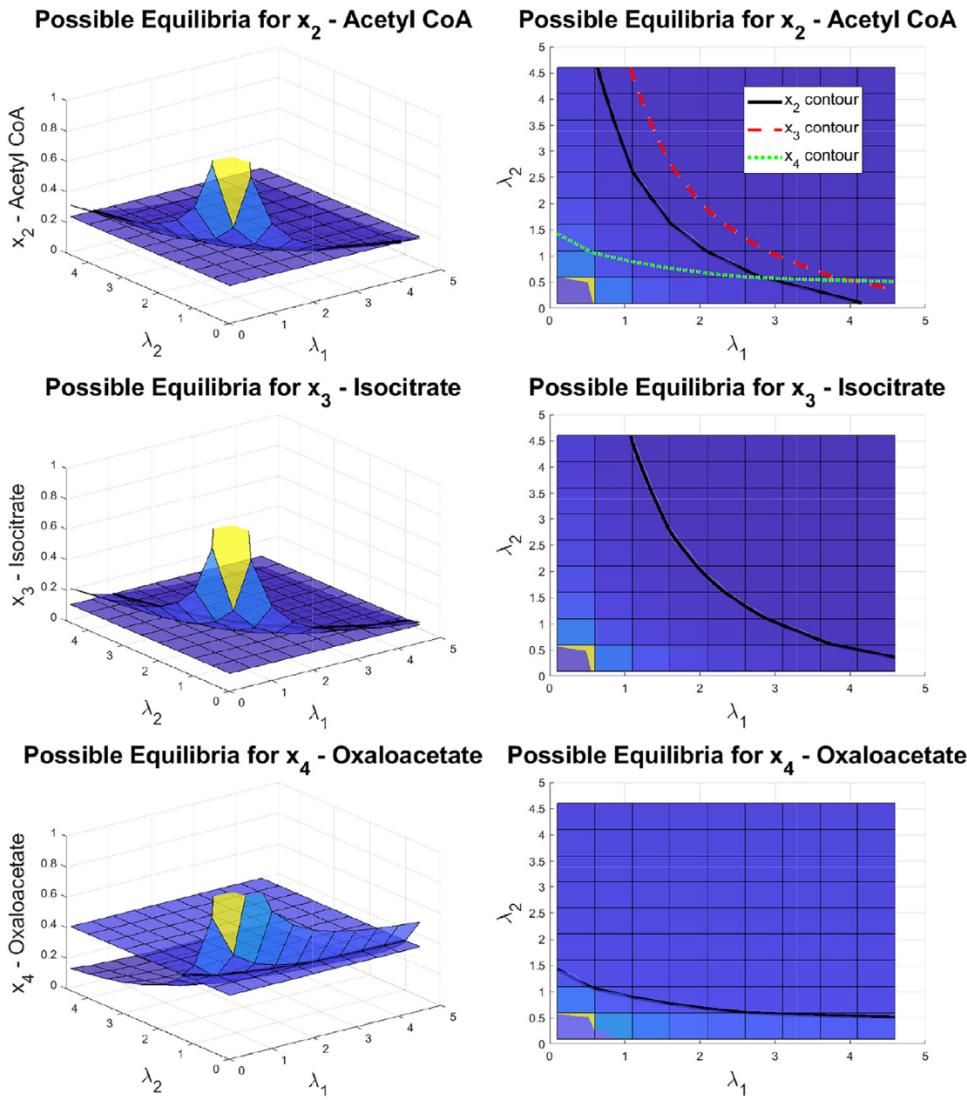


Fig. 6. Surfaces of equilibria of metabolites $x_{2,3,4}$ (respectively top left, top center, and top right) for hyperedges and simple edges. Projections on $\lambda_1\lambda_2$ -plane (bottom) with intersection curves in black. The intersection curves for x_3 and x_4 are also shown in the first panel for comparison with red dashed and green dotted lines respectively.

halving of a gene's expression against the control. Average Significance per gene is shown in Fig. 7, note that the “significance level” can be adjusted from 1 according to the level of power needed in the analysis.

We then used several filters, see Table 2, to collate the replicates for a given drug. For simplicity and consistency during implementation, we chose to use the mean filter for all drugs.

4.3. TB: Metabolic pathways

Metabolic pathways were imported from the Kyoto Encyclopedia of Genes and Genomes (KEGG). KEGG pathways encode metabolic pathways in the following way: nodes in the network correspond to metabolites, while edges in the network correspond to chemical reactions. KEGG pathways are also labeled in a way that makes it possible to tie the gene data from the microarrays back to the metabolic pathways that those genes affect. There are many genes that are not directly associated with any pathway and are likely not involved with enzyme creation or protein translation, but instead, are involved with other processes such as gene regulation. Genes may also be associated with a particular pathway, but the association has yet to be determined or is unknown to KEGG. Fig. 8 shows how many genes were able to be mapped to pathways from KEGG.

Once genes have been tied to metabolic pathway data, we are able to determine drug effects on specific pathways. We determine the number of significant genes that occur on each pathway, then sum the significant genes for each drug that appear on that pathway. In this manner, we can get a rough estimate of a drug's impact on a pathway. Fig. 9 shows a ranking of 4-drug combinations over the top ten most affected pathways — note, in the figure, we use the SAM filter for determining significance when we have multiple replicates of experiments. To rank the combinations, we use the L_1 metric, with is defined as: $L_1(\mathbf{x}) = \sum_i |x_i|$. As a remark, the drug combinations with the highest L_1 scores are not necessarily the combinations that deflect the network the most in simulation. L_1 scores do not capture complex interactions between metabolites; for instance, negative or positive feedback loops are not captured. Ranking treatments by their L_1 scores is not meant to capture their entire biological behavior; instead, the ranking is used to determine a list of possible treatment candidates for simulation.

To examine the results of this process, we compare the list of the top ten most affected pathways to known drug mechanisms of action. Included in the most affected list of pathways are: mtu02020, mtu02010, mtu03010, and mtu00190, corresponding to the Two-Component System, ABC Transporters, the Ribosome, and Oxidative Phosphorylation, respectively. These metabolic networks align roughly with known drug

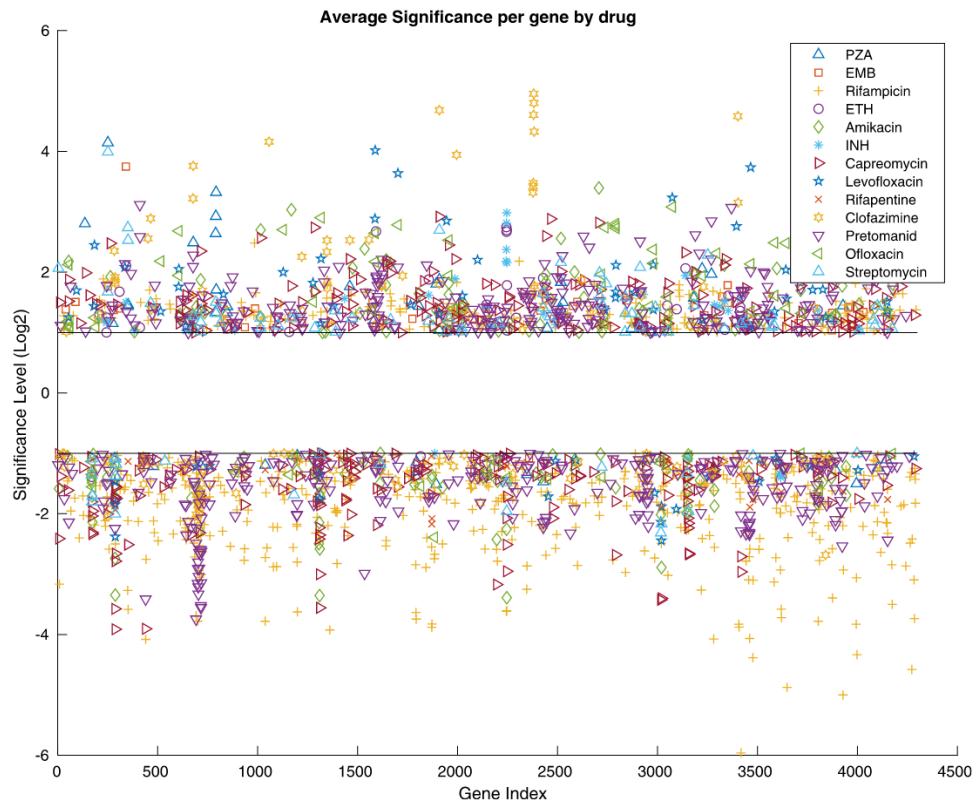


Fig. 7. Plot of average (average over replicates) value of significant gene expressions for the 13 drugs from [34]. The colors and shapes indicate different drugs.

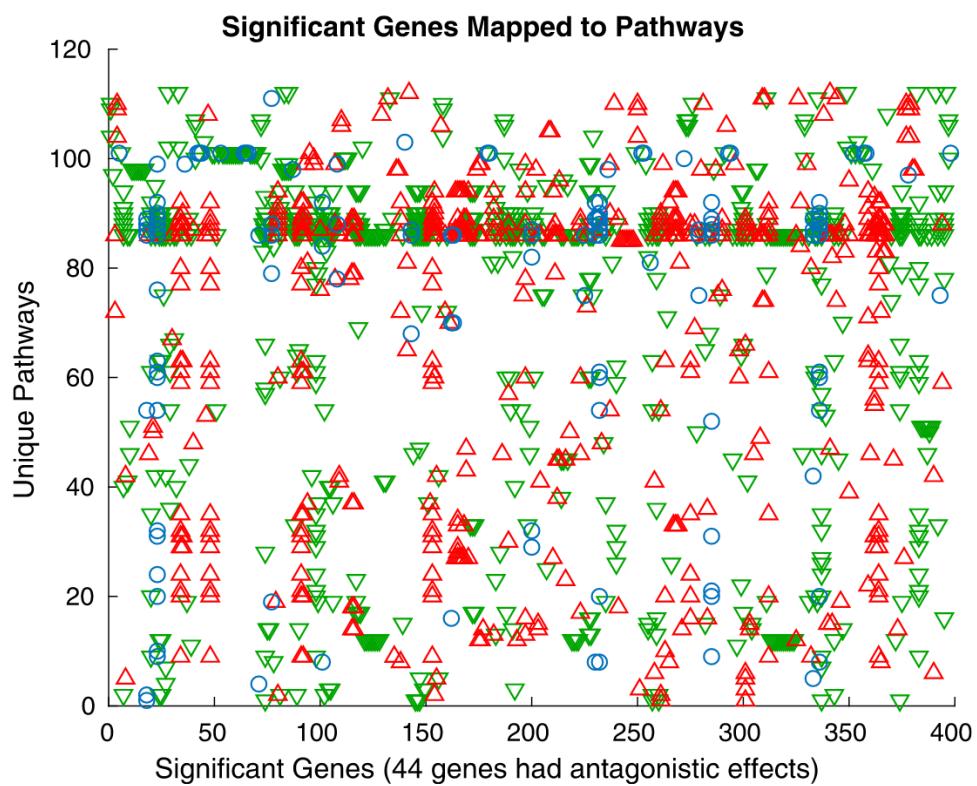


Fig. 8. This image shows the mapping of the 398 genes to the metabolic pathways. Downregulated genes are shown in green downward-facing triangles, upregulated genes are shown in red upward-facing triangles, and genes that were upregulated by some drugs and downregulated by others are shown as blue circles.

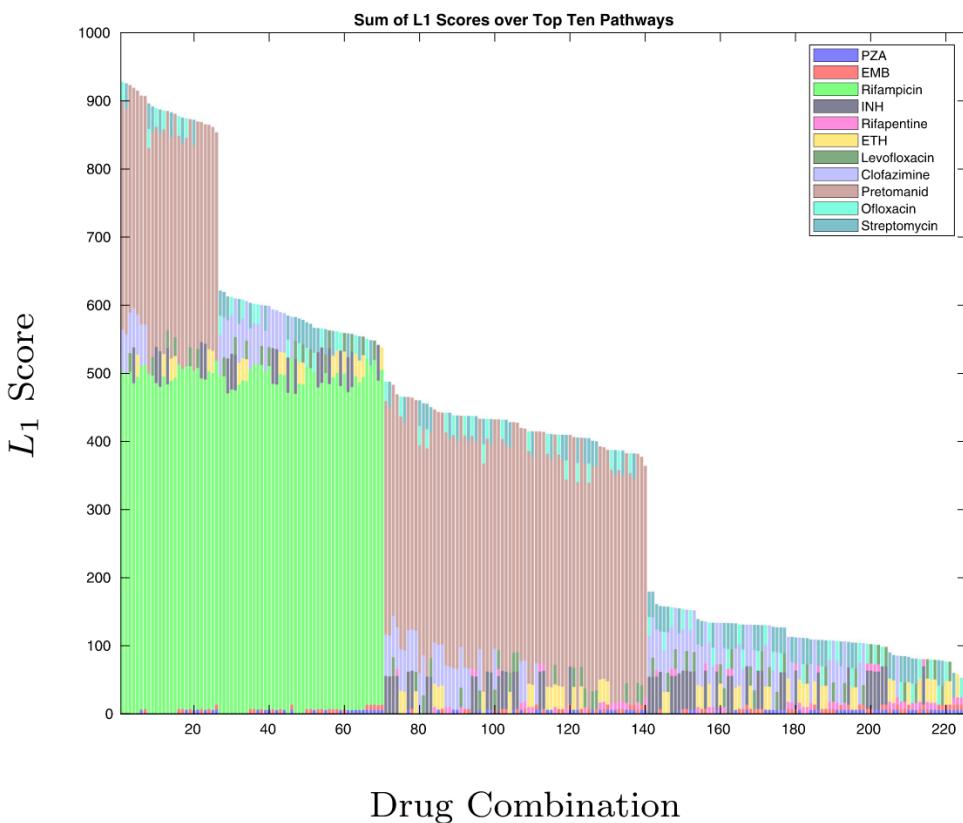


Fig. 9. 225 four-drug combinations were scored regarding the genes significantly affected by drugs across the top ten significant metabolic pathways. This bar graph shows the L_1 score of each drug combination, and the contributions to the score from each of the four drugs. Injectables such as Amikacin and Capreomycin were excluded. The ordering of treatments looks promising, as the standard treatment – Rifampin (Rifampicin), Isoniazid (INH), Pyrazinamide (PZA), and Ethambutol (EMB) (RIPE) – ranks high according to the L_1 score at position 26 [37].

Table 2

A summary of the number of significant genes identified by various methods is shown for a given drug. The highlighted cells correspond to the number of genes that were used in the computation of L_1 scores and in simulation. Values are the number of significant genes.

Drug	Consensus	Mean	SAM	Consensus & Mean	SAM & Mean
PZA	43	160	647	43	47
EMB	48	63	6	46	0
Rifampicin	659	959	176	650	75
ETH	48	94	606	48	35
Amikacin	241	481	916	240	168
INH	65	105	625	64	36
Capreomycin	224	603	2236	224	361
Levofloxacin	68	109	2	68	0
Rifapentine	24	157	372	23	26
Clofazimine	251	388	858	251	133
Pretomanid	95	712	2485	95	499
Ofloxacin	72	129	1313	72	69
Streptomycin	23	187	1134	23	80

mechanisms of action, as Pyrazinamide, Streptomycin, and linezolid potentially target the ribosome, while Clofazimine may affect oxidative phosphorylation [36]. This verification suggests that using L_1 scores over networks captures a reasonable level of detail relating to the mechanism of action of the therapy.

4.4. TB: Optimizing exchange fluxes

While KEGG offers a great library of metabolic pathways, the pathways offered are often not directly compatible with simulation. We use the method described in Section 3.5 to complete each metabolic network that we simulate over. This ensures that no metabolite will trivially deplete or trivially accumulate mass. After network completion, the behavior of the simulation aligns more with the underlying biology, as it is unexpected to have zero metabolites or strictly increasing metabolites.

4.5. TB: Drug effects on relevant fluxes

In order to determine a drug's effect on a pathway pulled from KEGG, we combine the experimental microarray data with the KEGG nodes. Genes from the microarray are mapped to the nodes in the metabolic networks. Note that this process is lossy, as there are many wells from the microarray data that have unknown functions and so they cannot be mapped to the metabolic networks. Despite this complication, we are still able to map 398 significant genes to their pathways.

Drug effect on network edges is determined using the following process: if a gene is differentially expressed, we consider the edges in the network which relate to that gene and add a drug-effect uberedge altering the edges on the metabolite. For instance, if a metabolite is significantly inhibited by a drug, then we introduce an uberedge that modifies the edges leading to the inhibited metabolite. The strength of the uberedges is determined by the strength of expression of the metabolites.

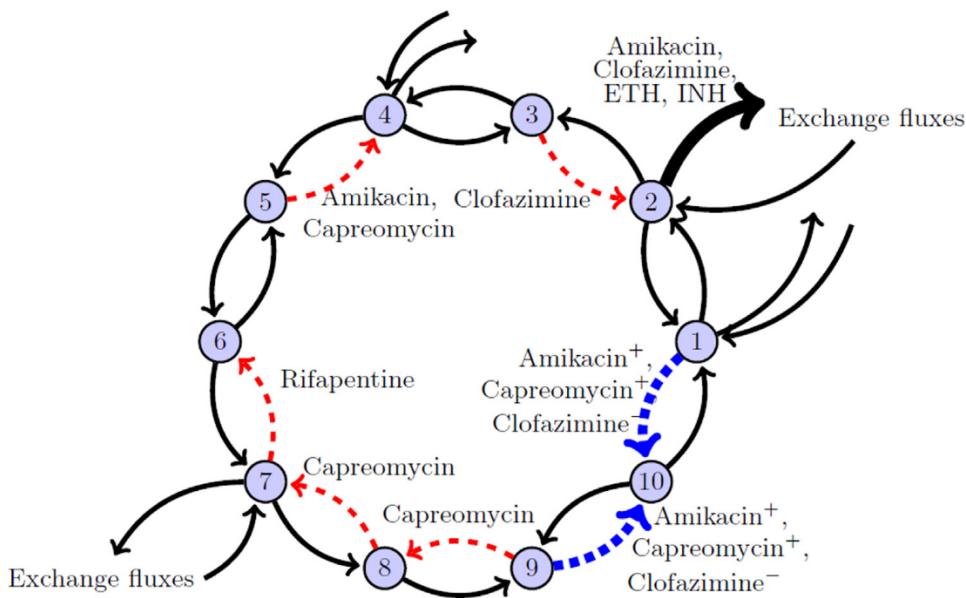


Fig. 10. An example pathway from KEGG, mtu01130, “Biosynthesis of Antibiotics”, which shows drugs inhibiting (red), enhancing (thick black), and mixed (blue) edges. Non-effected reactions are shown with thin black lines. Nodes correspond to chemical compounds found within the Biosynthesis of Antibiotics pathway.

4.6. TB: PK model

For simplicity and clarity, we limit our implementation to only include a pharmacokinetic aspect. If required, separate model parameters can be established if multiple compartments are of interest. In order to populate the simulation with valid model priors, a literature search was done to find the expected blood concentration levels of antibiotics. Refer to Table 1 columns two and five for the corresponding minimum inhibitory concentration and administered drug dose information, respectively. The dynamic simulations described in Section 4.7 provide the ability to simulate recurring treatments with combination therapies. The PK profiles determined in this section allow for an accurate implementation of recurring drug administration. PK data from literature is used to determine the concentration of drug in the compartment of interest (eg: lungs, liver, kidney, etc.); these drug concentration values are then replenished every treatment period.

We can also integrate existing PK resources in our pipeline. KEGG also offers pathways relevant to implementing drug effects on metabolic pathways. Fig. 10 provides an example of a relevant pathway for therapy metabolism — the “Biosynthesis of Antibiotic” pathway.

4.7. TB: Network simulation

Recall, as described in Section 3, the pipeline provides the ability to simulate the flow of mass through a metabolic network while considering the effects of a specific therapy — note that the pipeline also allows for the consideration of combination therapies. Fig. 11 provides a sample output from the pipeline for a combination of antibiotics — Isoniazid, Rifapentine, Levofloxacin, and Clofazimine. Note that the figure is able to capture which reactions (edges) were inhibited or promoted and which metabolites (nodes) have increased or decreased. As we are simulating over a specified time horizon, we can create a time-lapse of the figures to create an animation that demonstrates the effects of a treatment. The pipeline also supports graphing metabolite levels over time as in 12.

4.8. TB: LIFE methodology for network analysis

The pipeline culminates with dynamic simulations that are able to simulate the flow of mass through a specified metabolic pathway under the effects of a specific treatment — or a combination of treatments.

We can evaluate a treatment based on the deflection of metabolites from their equilibrium state. As a result of the use of hyperedges and uberedges, we observe interesting dynamics during simulation. Fig. 12 shows an example of a graph output of metabolite levels over the course of 336 h of treatment. Focusing on the blue metabolite, we see a transitory period between 10 h and 140 h where the metabolite approaches a new equilibrium under treatment. This behavior can provide insights into how well a therapy or therapy combination perturbs homeostatic metabolic function. The graphs can also enable a rough assessment of a therapeutic combination by observing the number of significantly deflected metabolites. While metabolite deflection may provide insight into the efficacy of a treatment, there may be some limitations to the approach. For instance, there is quite a degree of separation between treatment with an antibiotic and observed gene differential expression. We assume that any gene differential expression is due solely to exposure of the drug; however, it may be the case that gene differential expression is due to a gene regulatory network and not solely due to drug exposure. Future work on this pipeline will look to integrate gene regulatory network information in order to identify instances like this.

5. Conclusion

In this paper, we have conceptualized a computational pipeline that integrates experimental gene expression data with publicly available metabolic networks to predict combination therapy efficacy.

The pipeline may accept any type of gene expression data — in the example application, we utilize microarray data of gene expression levels of *Mycobacterium Tuberculosis* post-treatment with a single antibiotic. The pipeline may also accept any type of metabolic network — we show the application to KEGG pathways; however, the pipeline can potentially accept custom metabolic networks if specialization is required. We reiterate that publicly available metabolic networks often require an additional step in “completing” the networks to enable simulation of the flow of mass through the network. The usefulness of the pipeline is shown via the application to combination therapies for Tuberculosis — providing a preliminary ranking of combination therapies. In general, the estimation step of computing the L_1 scores roughly aligns with the known mechanism of action of antibiotics, providing a useful first-pass summary of which networks to simulate. The pipeline then provides a more detailed estimation of the drug effect

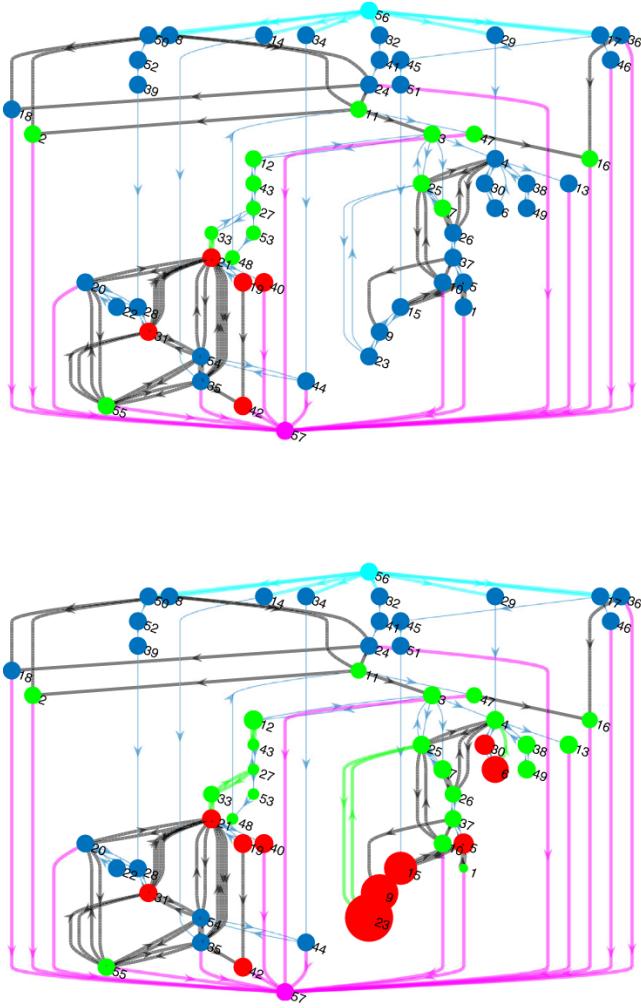


Fig. 11. Carbon metabolism networks highlighting the metabolites that were affected by the drug combinations, Top: INH, Rifapentine, Levofloxacin, Clofazimine and Botom: Rifampicin, ETH, Levofloxacin, Pretomanid. The red nodes represent metabolites that increased in mass, while the green nodes decreased. Cyan lines represent intakes to the system, while magenta represents excretions. Black edges represent hyperedges. Green edges represent enhanced reactions. For the figure on the bottom, the metabolite that increased the most was Fumarate, while the metabolite that decreased the most was Carbon dioxide.

on each network by simulating the change of metabolite levels over time. For future work, we would like to investigate the importance of network topology on the treatment effect; this can be facilitated by the inclusion of visual tools into the pipeline. We aim to provide the capability of animating all network simulations. Future work will also include the introduction of gene regulatory network (GRN) information — this will aid in identifying if gene expression change is due directly to drug effect or downstream regulation. Additionally, we will aim to validate the results of the dynamic simulation by determining if there is a correlation between metabolite deflection and pre-clinical bactericidal effects.

While this paper outlines a conceptual pipeline, we make example code available for its specific application to combination therapies of antibiotics in *Mycobacterium Tuberculosis*.

The MATLAB source code for the example application of this pipeline to multi-drug combinations in *Mycobacterium tuberculosis* is made available at https://github.com/chrisDev-2B/LIFE_computational_pipeline.

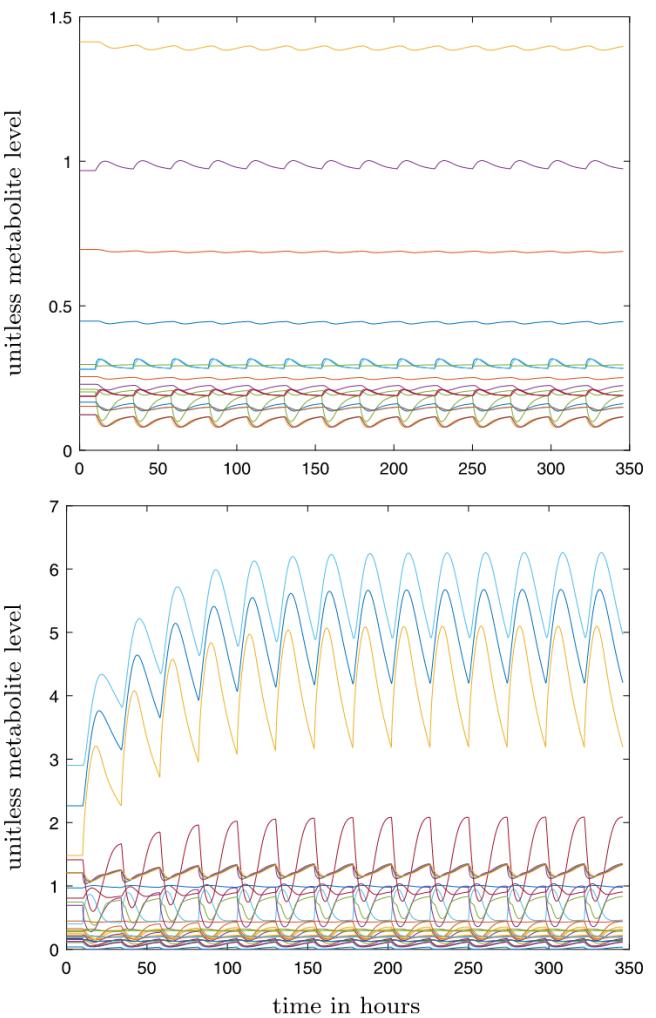


Fig. 12. 10 h of equilibrium followed by 336 h of daily drug treatment. Trajectories of metabolites in central carbon metabolism. The top shows the combination of INH, Rifapentine, Levofloxacin, and Clofazimine, and the bottom shows the combination of Rifampicin, ETH, Levofloxacin, and Pretomanid. Due to the system having many metabolites, only the metabolites that had significant changes are shown for each simulation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

We thank Chanchala Kaddi of Sanofi for her expertise and insightful discussions.

References

- [1] K. Azer, C.D. Kaddi, J.S. Barrett, J.P.F. Bai, S.T. McQuade, N.J. Merrill, et al., History and future perspectives on the discipline of quantitative systems pharmacology modeling and its applications, *Front. Physiol.* 12 (2021) 637999, <http://dx.doi.org/10.3389/fphys.2021.637999>.

- [2] P. Bloomingdale, T. Karelina, M. Cirit, S.F. Muldoon, J. Baker, W.J. McCarty, et al., Quantitative systems pharmacology in neuroscience: Novel methodologies and technologies, *CPT Pharmacomet. Syst. Pharmacol.* 10 (5) (2021) 412–419, <http://dx.doi.org/10.1002/psp4.12607>.
- [3] C. Lacroix, T. Soeiro, M. Le Marois, R. Guilhaumou, C. Casse-Perrot, E. Jouve, et al., Innovative approaches in CNS clinical drug development: Quantitative systems pharmacology, *Therapie* 76 (2) (2021) 111–119, <http://dx.doi.org/10.1016/j.therap.2020.12.007>.
- [4] P. Sorger, S. Allerheiligen, D. Abernethy, R. Altman, K. Brouwer, A. Califano, et al., Quantitative and systems pharmacology in the post-genomic era: new approaches to discovering drugs and understanding therapeutic mechanisms, in: An NIH White Paper By the QSP Workshop Group, NIH Bethesda, 2011, pp. 1–48.
- [5] E. Bradshaw, M. Spilker, R. Zang, L. Bansal, H. He, R. Jones, et al., Applications of quantitative systems pharmacology in model-informed drug discovery: perspective on impact and opportunities, *CPT Pharmacomet. Syst. Pharmacol.* 8 (2019) 777–791.
- [6] V. Balbas-Martinez, L. Ruiz-Cerdá, I. Irurzun-Arana, I. González-García, A. Vermeulen, J.D. Gómez-Mantilla, et al., A systems pharmacology model for inflammatory bowel disease, *PLoS One* 13 (3) (2018) 1–19, <http://dx.doi.org/10.1371/journal.pone.0192949>.
- [7] C.D. Kaddi, B. Niesner, R. Baek, P. Jasper, J. Pappas, J. Tolsma, et al., Quantitative systems pharmacology modeling of acid sphingomyelinase deficiency and the enzyme replacement therapy olipudase alfa is an innovative tool for linking pathophysiology and pharmacology, *CPT Pharmacomet. Syst. Pharmacol.* 7 (2018) 442–452.
- [8] R. Coletti, L. Leonardelli, S. Parolo, L. Marchetti, A QSP model of prostate cancer immunotherapy to identify effective combination therapies, *Sci. Rep.* 10 (1) (2020) 9063, <http://dx.doi.org/10.1038/s41598-020-65590-0>.
- [9] J.E. Ming, R. Abrams, D.W. Barlett, T. Nguyen, K. Kudrycki, A. Kadambi, et al., A quantitative systems pharmacology platform to investigate the impact of alirocumab and cholesterol-lowering therapies on lipid profiles and plaque characteristics, *Gene Regul. Syst. Biol.* (2017).
- [10] M.C. Peterson, M.M. Riggs, FDA advisory meeting clinical pharmacology review utilizes a quantitative systems pharmacology (QSP) model: A watershed moment? *CPT Pharmacomet. Syst. Pharmacol.* 4 (3) (2015) 189–192, <http://dx.doi.org/10.1002/psp4.20>.
- [11] V.R. Knight-Schrijver, V. Chelliah, L. Cucurull-Sánchez, N. Le Novere, The promises of quantitative systems pharmacology modelling for drug development, *Comput. Struct. Biotechnol. J.* 14 (2016) 363–370, <http://dx.doi.org/10.1016/j.csbj.2016.09.002>.
- [12] M. Kuchimanchi, M. Monine, K. Kandadi Muralidharan, C. Woodward, N. Penner, Phase II dose selection for alpha synuclein-targeting antibody cinnapenemab (BIIIB054) based on target protein binding levels in the brain, *CPT Pharmacomet. Syst. Pharmacol.* 9 (9) (2020) 515–522, <http://dx.doi.org/10.1002/psp4.12538>.
- [13] K. Gadkar, D. Kirouac, N. Parrott, S. Ramanujan, Quantitative systems pharmacology: a promising approach for translational pharmacology, *Drug Discov. Today* 21–22 (2016) 57–65, <http://dx.doi.org/10.1016/j.ddtec.2016.11.001>.
- [14] C. Thiel, I. Smit, V. Baier, H. Cordes, B. Fabry, L.M. Blank, et al., Using quantitative systems pharmacology to evaluate the drug efficacy of COX-2 and 5-LOX inhibitors in therapeutic situations, *npj Syst. Biol. Appl.* 4 (1) (2018) 28, <http://dx.doi.org/10.1038/s41540-018-0062-3>.
- [15] S.T. McQuade, R.E. Abrams, J.S. Barrett, B. Piccoli, K. Azer, Linear-in-flux-expressions methodology: Toward a robust mathematical framework for quantitative systems pharmacology simulators, *Gene Regul. Syst. Biol.* 11 (2017) 1177625017711414.
- [16] S.T. McQuade, Z. An, N.J. Merrill, R.E. Abrams, K. Azer, B. Piccoli, Equilibria for large metabolic systems and the LIFE approach, in: 2018 Annual American Control Conference, ACC, IEEE, 2018, pp. 2005–2010.
- [17] N.J. Merrill, Z. An, S.T. McQuade, F. Garin, K. Azer, R. Abrams, et al., Stability of metabolic networks via linear-in-flux-expressions, *Netw. Heterog. Media* 14 (1) (2019) 101–130.
- [18] S.T. McQuade, N.J. Merrill, B. Piccoli, Metabolic graphs, LIFE method and the modeling of drug action on Mycobacterium tuberculosis, in: J. Kotas (Ed.), *Advances in Nonlinear Biological Systems: Modeling and Optimal Control*, Vol. 11, in: AIMS Applied Mathematics Book Series, 2020, pp. 121–143.
- [19] M. Kanehisa, Y. Sato, M. Furumichi, K. Morishima, M. Tanabe, New approach for understanding genome variations in KEGG, *Nucleic Acids Res.* 47 (D1) (2018) D590–D595, <http://dx.doi.org/10.1093/nar/gky962>.
- [20] M. Kanehisa, M. Furumichi, M. Tanabe, Y. Sato, K. Morishima, KEGG: new perspectives on genomes, pathways, diseases and drugs, *Nucleic Acids Res.* 45 (D1) (2016) D353–D361, <http://dx.doi.org/10.1093/nar/gkw1092>.
- [21] M. Kanehisa, S. Goto, KEGG: Kyoto encyclopedia of genes and genomes, *Nucleic Acids Res.* 28 (1) (2000) 27–30, <http://dx.doi.org/10.1093/nar/28.1.27>.
- [22] S. McQuade, N. Merrill, B. Piccoli, Metabolic graphs, the LIFE method, and modeling drug action on mycobacterium tuberculosis, in: J. Kotas (Ed.), 2020.
- [23] D. Venet, V. Detours, H. Bersini, A measure of the signal-to-noise ratio of microarray samples and studies using gene correlations, *PLoS One* 7 (12) (2012) <http://dx.doi.org/10.1371/journal.pone.0051013>.
- [24] L.A. Corchete, E.A. Rojas, D. Alonso-Lopez, J. De Las Rivas, N.C. Gutierrez, F.J. Burguillo, Systematic comparison and assessment of RNA-seq procedures for gene expression quantitative analysis, *Sci. Rep.* 10 (1) (2020) <http://dx.doi.org/10.1038/s41598-020-76881-x>.
- [25] E. Cinlar, *Introduction to Stochastic Processes*, Courier Corporation, 2013.
- [26] F. Bullo, Lectures on network systems, 2016, Online at <http://motion.me.ucsb.edu/book-lns>, with contributions by J. Cortes, F. Dorfler and S. Martinez.
- [27] J.A. Jacquez, C.P. Simon, Qualitative theory of compartmental systems, *SIAM Rev.* 35 (1) (1993) 43–79.
- [28] H. Maeda, S. Kodama, Y. Ohta, Asymptotic behavior of nonlinear compartmental systems: nonoscillation and stability, *IEEE Trans. Circuits Syst.* 25 (6) (1978) 372–378.
- [29] S. Kou, B.J. Cherayil, W. Min, B.P. English, X.S. Xie, Single-Molecule Michaelis-Menten Equations, ACS Publications, 2005.
- [30] A. Bretto, *Hypergraph Theory: An Introduction*, in: Mathematical Engineering, Springer, Cham, 2013.
- [31] V.I. Voloshin, *Introduction to Graph and Hypergraph Theory*, Nova Science Publ., 2009.
- [32] S. Feng, E. Heath, B. Jefferson, C. Joslyn, H. Kvinge, H.D. Mitchell, et al., Hypergraph models of biological networks to identify genes critical to pathogenic viral response, *BMC Bioinformatics* 22 (1) (2021) <http://dx.doi.org/10.1186/s12859-021-04197-2>.
- [33] C. Joslyn, K. Nowak, Ubergraphs: A definition of a recursive hypergraph structure, 2017, arXiv preprint [arXiv:1704.05547](https://arxiv.org/abs/1704.05547).
- [34] H.I. Boshoff, T.G. Myers, B.R. Copp, M.R. McNeil, M.A. Wilson, C.E. Barry, The transcriptional responses of mycobacterium tuberculosis to inhibitors of metabolism novel insights into drug mechanisms of action, *J. Biol. Chem.* 279 (38) (2004) 40174–40184.
- [35] S. Ma, K.J. Minch, T.R. Rustad, S. Hobbs, S.-L. Zhou, D.R. Sherman, et al., Integrated modeling of gene regulatory and metabolic networks in mycobacterium tuberculosis, *PLoS Comput. Biol.* 11 (11) (2015) e1004543.
- [36] J.C. Palomino, A. Martin, Drug resistance mechanisms in mycobacterium tuberculosis, *Antibiotics-Basel* 3 (3) (2014) 317–340, <http://dx.doi.org/10.3390/antibiotics3030317>.
- [37] P. Nahid, S.E. Dorman, N. Alipanah, P.M. Barry, J.L. Brozek, A. Cattamanchi, et al., Executive summary: Official American thoracic society/centers for disease control and prevention/infectious diseases society of america clinical practice guidelines: Treatment of drug-susceptible tuberculosis, *Clin. Infect. Dis.* 63 (7) (2016) 853–867, <http://dx.doi.org/10.1093/cid/ciw566>.

