

Systems biology

COVRECON: automated integration of genome- and metabolome-scale network reconstruction and data-driven inverse modeling of metabolic interaction networks

Jiahang Li ¹, Steffen Waldherr ¹, Wolfram Weckwerth ^{1,2,*}

¹Molecular Systems Biology Lab (MOSYS), Department of Functional and Evolutionary Ecology (FEE), University of Vienna, Djerassiplatz 1, 1030 Vienna, Austria

²Vienna Metabolomics Center (VIME), University of Vienna, Djerassiplatz 1, 1030 Vienna, Austria

*Corresponding author. Vienna Metabolomics Center (VIME), University of Vienna, 1030 Vienna, Austria. E-mail: wolfram.weckwerth@univie.ac.at (W.W.)

Associate Editor: Pier Luigi Martelli

Abstract

Motivation: One central goal of systems biology is to infer biochemical regulations from large-scale OMICS data. Many aspects of cellular physiology and organismal phenotypes can be understood as results of metabolic interaction network dynamics. Previously, we have proposed a convenient mathematical method, which addresses this problem using metabolomics data for the inverse calculation of biochemical Jacobian matrices revealing regulatory checkpoints of biochemical regulations. The proposed algorithms for this inference are limited by two issues: they rely on structural network information that needs to be assembled manually, and they are numerically unstable due to ill-conditioned regression problems for large-scale metabolic networks.

Results: To address these problems, we developed a novel regression loss-based inverse Jacobian algorithm, combining metabolomics COVariance and genome-scale metabolic RECONstruction, which allows for a fully automated, algorithmic implementation of the COVRECON workflow. It consists of two parts: (i) Sim-Network and (ii) inverse differential Jacobian evaluation. Sim-Network automatically generates an organism-specific enzyme and reaction dataset from Bigg and KEGG databases, which is then used to reconstruct the Jacobian's structure for a specific metabolomics dataset. Instead of directly solving a regression problem as in the previous workflow, the new inverse differential Jacobian is based on a substantially more robust approach and rates the biochemical interactions according to their relevance from large-scale metabolomics data. The approach is illustrated by *in silico* stochastic analysis with differently sized metabolic networks from the BioModels database and applied to a real-world example. The characteristics of the COVRECON implementation are that (i) it automatically reconstructs a data-driven superpathway model; (ii) more general network structures can be investigated, and (iii) the new inverse algorithm improves stability, decreases computation time, and extends to large-scale models.

Availability and implementation: The code is available in the website <https://bitbucket.org/mosys-univie/covrecon>.

1 Introduction

Recent studies in systems biology generate large datasets of molecular, genomic, and physiological variables, with the aim to understand complex diseases and regulatory interactions in biochemical networks from clinical studies (Linke *et al.* 2017, Elgendy *et al.* 2019). However, the functional interpretation of such datasets and the inference of how regulatory mechanisms in the underlying biochemical networks change in disease conditions relies on developing proper mathematical analysis and inference methods (Weckwerth 2003, Weckwerth 2011, Weckwerth 2019).

The primary method in genome-scale metabolomics data processing is statistics. In recent studies, both conventional and machine-learning methods have been implemented for metabolomics data analysis. The conventional statistical methods, such as *t*-test, clustering, and principal component analysis, are widely used but are not able to reveal the underlying biochemical regulatory interactions. Nevertheless, many recently developed machine learning techniques have

primarily enhanced the statistical power in metabolomics data analysis, such as deep learning (LeCun *et al.* 2015, Sidak *et al.* 2022), genetic algorithms (Mitchell 1998), and boosting machine-learning methods (Chen and Guestrin 2016). Yet, these methods provide limited insight into how the information in a biochemical network is transferred, what the critical regulatory steps are, and how a regulatory mechanism changes under different conditions. Here, a metabolic interaction network is defined from biochemical or regulatory interactions between identified metabolites. Interactions can be direct or involve several connected reactions (superpathways) (Nägele *et al.* 2014, Weckwerth 2019). As shown in Fig. 1a, many aspects of cellular physiology and organismal phenotype could be understood as a result of the metabolic interaction network dynamics (De Martino *et al.* 2018). Thus, analyzing the changes in a metabolic interaction network for different phenotypes can give insight into changes of the underlying regulatory mechanisms. The resulting differential metabolic interaction networks are usually influenced by

Received: August 24, 2022. Revised: June 12, 2023. Editorial Decision: June 15, 2023. Accepted: June 22, 2023

© The Author(s) 2023. Published by Oxford University Press.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.



requiring only data for the computation of the covariance matrix C and the estimation of a fluctuation matrix D .

In the last years, several studies have developed inverse differential Jacobian algorithms, which provide a convenient way to infer the dynamics of metabolic networks from metabolomics data (Steuer *et al.* 2003, Weckwerth 2003, Sun and Weckwerth 2012, Öksüz *et al.* 2013, Kügler and Yang 2014, Nägele *et al.* 2014, Sun *et al.* 2015, Khatibipour *et al.* 2018, Wilson *et al.* 2020, Weiszmann *et al.* 2023). This method is still at an early stage and requires further improvements and unification of methods (Weckwerth 2019). Here, we present a novel inverse differential Jacobian algorithm and integrate this algorithm with an automated implementation of a recently proposed workflow called COVRECON (Weckwerth 2019). Following this workflow very recently Wilson *et al.* identified a novel switch for macrophage polarization (Wilson *et al.* 2020). All existing studies using biochemical inference rely on structural network information that has been assembled manually, which can take weeks' effort and relies on a modeler's domain knowledge. In this work, we designed an automatic network reconstruction using pathway search algorithms in genome-scale metabolic networks and integrated this with a completely novel algorithm to inversely calculate the biochemical Jacobian and biochemical inference from metabolomics data. In previous implementations, the inference algorithms were numerically unstable due to ill-conditioned regression problems (Sun *et al.* 2015) for large-scale metabolic networks. Here, we address this problem by a novel regression loss-based inverse Jacobian algorithm. Putting these two parts together, we developed the complete automated COVRECON workflow and related Matlab toolbox. As shown in Fig. 1b, COVRECON consists of three sub-modules: (i) building of an organism-specific database, (ii) construction of a superpathway-based metabolic interaction model (Section 2.2 Sim-Network), and (iii) the inverse Jacobian computation (Section 2.3).

Accordingly, the COVRECON algorithm presents a convenient method to automatically construct a metabolic interaction model and integrating this with an inverse biochemical Jacobian reconstruction providing a more general network structure and perturbations of the same. As for the inverse Jacobian part, this work introduces a novel algorithm improving accuracy and stability, reduces computation time, and extends the method to large-scale models. In summary, this work presents the first Matlab toolbox integrating all required steps from building the interaction network up to solving the Jacobian problem.

2 Materials and methods

2.1 The differential Jacobian

Consider a biological system that consists of n compounds (metabolites, proteins) denoted by $\{X_i\}_{i=1\dots n}$. The system dynamics can be modeled with the set of ordinary differential equations:

$$\frac{d\mathbf{M}}{dt} = \mathbf{F}(\mathbf{M}) \rightarrow \begin{cases} \frac{dM_1}{dt} = f_1(M_1, M_2, \dots, M_n) \\ \frac{dM_2}{dt} = f_2(M_1, M_2, \dots, M_n) \\ \vdots \\ \frac{dM_n}{dt} = f_n(M_1, M_2, \dots, M_n), \end{cases} \quad (1)$$

where $\mathbf{M} = \{M_i\} = \{|X_i|\}$ are the concentrations of the n compounds, and $\mathbf{F} = f_i(M_i)$ are composed of the reaction

rates for these compounds (e.g. Michaelis–Menten kinetics, or mass action).

The steady-state Jacobian matrix \mathbf{J} of the model is defined as a $R^{n \times n}$ matrix in which J_{ij} is the first-order derivative of the rate f_i for the concentration of substances M_j at steady state, noted as $J_{ij} = \left. \frac{\partial f_i}{\partial M_j} \right|_{\text{steady}}$:

$$\mathbf{J} = \frac{\partial \mathbf{F}}{\partial \mathbf{M}_{\text{steady}}} = \begin{bmatrix} \frac{\partial f_1}{\partial M_1} & \frac{\partial f_1}{\partial M_2} & \dots & \frac{\partial f_1}{\partial M_n} \\ \frac{\partial f_2}{\partial M_1} & \frac{\partial f_2}{\partial M_2} & \dots & \frac{\partial f_2}{\partial M_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial M_1} & \frac{\partial f_n}{\partial M_2} & \dots & \frac{\partial f_n}{\partial M_n} \end{bmatrix}_{\text{steady}}. \quad (2)$$

Even if only evaluated at steady state, the Jacobian matrix of a system contains useful information about its dynamics, such as regulatory interactions between the different compounds. In a previous study, Steuer *et al.* (2003) established a relation between the covariance matrix \mathbf{C} of the metabolic data in the network and the steady-state Jacobian matrix of the system \mathbf{J} as the Lyapunov equation given by

$$\mathbf{J} \cdot \mathbf{C} + \mathbf{C} \cdot \mathbf{J}^T = -2\mathbf{D}. \quad (3)$$

Thereby, $\mathbf{C} \in R^{n \times n}$ is the covariance matrix of the compounds' concentrations M_j near its steady-state value M_j^{steady} , and the fluctuation matrix \mathbf{D} is the covariance of noise sources acting on the system.

Here, we focus on the differences in Jacobian matrices for two biological conditions, e.g. a health and disease condition, abbreviated as “h” and “d.” Using steady-state metabolomics data of a biological network, the objective is to evaluate the differences in the Jacobian matrices and thus the changes in biochemical interactions between the two conditions.

The differences between the two conditions are quantified by the differential Jacobian $D\mathbf{J}$, the elements of which are defined from the Jacobians in first, e.g. health, condition “h” \mathbf{J}_h and in the second, e.g. disease, condition “d” \mathbf{J}_d as

$$D\mathbf{J}_{ij} = \begin{cases} \frac{(\mathbf{J}_d)_{ij} - (\mathbf{J}_h)_{ij}}{(\mathbf{J}_h)_{ij}} & \text{if } (\mathbf{J}_h)_{ij} \neq 0 \\ 1, & \text{if } (\mathbf{J}_h)_{ij} = 0. \end{cases} \quad (4)$$

In the Lyapunov Equation (3), the Jacobian matrix \mathbf{J} has $n \times n$ unknown variables, while the covariance matrix \mathbf{C} is a symmetric matrix, and thus has only $\frac{n(n+1)}{2}$ independent variables to be determined from measurement data. This fact indicates that in the direct inverse approach of calculating \mathbf{J} from \mathbf{C} , we would generally have only $\frac{n(n+1)}{2}$ equations but $n \times n$ unknown variables, thus the inverse Jacobian approach would generally be under-determined and the reconstructed Jacobian matrix non-unique.

However, for realistic biological networks the Jacobian matrix structure is commonly a sparse matrix (Sun and Weckwerth 2012, Nägele *et al.* 2014). Thus, the first step of COVRECON, Sim-Network, is made to automatically construct a simplified conceptualized network model for the measured metabolites, which is used to constrain the Jacobian matrix structure. In most cases, the resulting structure leaves fewer non-zero entries than independent variables in the

covariance matrix, thus making the Lyapunov equation overdetermined and requiring the use of regression methods.

The COVRECON method consists of two major steps:

- 1) Determine the Jacobian structure. The key objective is to reduce the network structure from a genome-scale network to the specific biochemical species that are included in the considered dataset.
- 2) Analyze the differential Jacobian matrix. We establish a new method that focuses on identifying the major components in the differential Jacobian, instead of pursuing a full quantitative reconstruction.

The following sections will describe these two steps in more detail.

2.2 Determination of the Jacobian structure with the software Sim-Network

The Jacobian matrix structure is determined by the software tool Sim-Network, which we implemented in Matlab. It constructs a conceptualized metabolic model for the metabolites in the considered dataset using a pathway search approach. Methods for pathway design and prediction have a long history since the emerging genomics, proteomics, and metabolomics databases [e.g. BIGG models (King et al. 2016), KEGG (Ogata et al. 1999), Metacyc (Caspi et al. 2020), and ModelSEED (Seaver et al. 2021)].

To construct the conceptualized metabolic model, Sim-Network makes use of reaction data from BIGG genome-scale models (King et al. 2016), the KEGG database (Ogata et al. 1999). For relevant reactions thermodynamics data (reaction irreversibility, direction and estimated reaction delta Gibbs free energy), we utilize the ModelSEED dataset (Seaver et al. 2021).

Figure 1b illustrates a scheme of the Sim-Network tool; the aim of Sim-Network is reducing the genome-scale model (with more than 1000 metabolites) to a superpathway-based metabolic interaction network for the measured metabolites set (with <100 metabolites). In general, Sim-Network contains three main steps: network information gathering, path search, and pruning. We first generate a reaction database from a BIGG model or all the reactions related to a specific organism in KEGG. Then Sim-Network will gather relevant metabolites, and assemble a directional weighted network representation. Next, through shortest path search algorithm, Sim-Network will compute shortest paths (in both directions) for all pairs of metabolites in a specific metabolomics dataset. Finally, regarding the predefined cost threshold (Moriya et al. 2010, Wang et al. 2017, Karp et al. 2021), Sim-Network will prune the network (assuming long-distance interactions are negligible), and construct a metabolic interaction network for the considered dataset. The detailed workflow of Sim-Network consists of the following steps:

Step 1: Initially, one needs to choose an organism-based genome-scale metabolic model as a database. If transcriptomic data is available, the model is modified and trimmed by discarding reactions whose enzyme is not activated, using the GIMME method (Becker and Palsson 2008). When no transcriptomic data are available, the KEGG database offers a broader database including more enzyme specific reactions. By choosing the organism in COVRECON toolbox according to the experimental dataset (e.g. hsa for *Homo sapiens*, mmu for *Mus musculus*), Sim-Network will search and include all

the enzymes and reactions of that organism into a database model.

Step 2: The model stoichiometric matrix is transformed into its network representation. Meanwhile, a weighting strategy is applied to the reactions. Firstly, the side-metabolites (e.g. H_2O , H^+ ...) are excluded during the pathway search (Wang et al. 2017, Karp et al. 2021). This is because those side-metabolites concentrations are influenced by many other metabolites, which makes the influence from each specific metabolite negligible. The predefined side-metabolites are listed in the [Supplementary Material](#). For each reaction, the reaction direction and irreversibility information is obtained from the BIGG model or from the ModelSEED dataset. The forward direction is given Weight 1. If the reaction is reversible, the reverse reaction is given a user-defined weight (default 2). If the Gibbs free energy difference of the reaction ΔG is available in ModelSEED database, we adjust the path cost for the reverse reaction according to the strategy in MRE web tool (Kuwahara et al. 2016). Generally speaking, for the reverse reaction whose Gibbs free energy difference is larger than 100 kcal/mol, we will add the log value of the delta Gibbs free energy to the user-defined reverse reaction weight. Based on the weighting strategy, Sim-network will connect every pair of reactant-product in each forward or reverse reaction with directional connections, and give the related reaction weight to the directional connections. Since the reaction rate is influenced by all reactants, two metabolites are also connected in Sim-Network when they are both reactants in a reaction, or both products in a reversible reaction.

Step 3: For the selected metabolite subset $\{X_s\}$ of the database metabolites set $\{X\}$, we determine and save all the shortest path costs and routes for every directional pair of metabolites in $\{X_s\}$. Sim-Network discards all routes of which the cost is higher than a preset threshold (default is 3). Then, based on the search result, we construct the metabolic interaction model and generate the related Jacobian matrix structure based on a simplified model strategy. For example, if the shortest route from metabolite X_a to X_c contains another metabolite X_b in the selected metabolites set $\{X_s\}$, we will discard this connection, assuming that the influence from X_a to X_c can be reflected by the co-influences of (X_a-X_b) and (X_b-X_c) . The resulted network is saved in SBML format (Hucka et al. 2003) with the Sim-Biology Matlab toolbox.

With the conceptualized network, one can determine the Jacobian structure. The diagonal components of the related Jacobian matrix are non-zero components, and each connection in the conceptualized network represents a non-zero off-diagonal component. In [Supplementary Material S3](#), we provide a toolbox manual and case studies.

2.3 Inverse differential Jacobian evaluation

This section introduces a new algorithm to determine the major components of the differential Jacobian matrix, using the Jacobian structure information as determined by Sim-Network and a covariance estimation from metabolomics data.

In the inverse Jacobian approach, the Lyapunov Equation (3) is solved for the Jacobian matrix J with given covariance and fluctuation matrices C and D . This is a linear equation, which as outlined in previous studies (Sun and Weckwerth 2012, Kügler and Yang 2014) can be rewritten in the form

$$Aq = b, \quad (7)$$

where the vector q consists of all the non-zero components in the Jacobian matrix as unknown variables. The matrix A is calculated from the values in the covariance matrix C ; its dimension is $(\frac{n(n+1)}{2}, L)$, where L is the dimension of the vector q . Since the Jacobian structure is sparse, we usually have $L < \frac{n(n+1)}{2}$, and Equation (7) is overdetermined. The vector b is constructed from the fluctuation matrix D ; its dimension is $\frac{n(n+1)}{2}$.

Similarly, to evaluate the differential Jacobian in two conditions denoted by “h” and “d,” the corresponding Lyapunov equations can be rewritten as

$$\begin{aligned} A_h q_h &= b_h, \\ A_d q_d &= b_d. \end{aligned} \quad (8)$$

Previous inverse Jacobian algorithms (Steuer *et al.* 2003, Sun and Weckwerth 2012, Kügler and Yang 2014, Nägele *et al.* 2014, Sun *et al.* 2015, Wilson *et al.* 2020) have assumed that independent stochastic noise affects each metabolite individually, giving rise to a diagonal fluctuation matrix D in the Lyapunov Equation (3). Thus, these studies used an average result of randomly sampled diagonal perturbation matrices D (Sun and Weckwerth 2012, Nägele *et al.* 2014), or applied a L-p optimization of the two conditional fluctuation matrixes D_h and D_d (Kügler and Yang 2014) to calculate the differential Jacobian matrix DJ . These methods work through directly solving the linear Equation (8). However, when the condition number of the matrix A is large, the linear equations’ solution will be unstable against small perturbations in the data. This makes previous methods (Sun and Weckwerth 2012, Kügler and Yang 2014, Nägele *et al.* 2014, Sun *et al.* 2015) numerically sensitive and not adequate to deal with large-scale models.

As an alternative to a direct solution of the linear Equation (8) and subsequent calculation of the differential Jacobian according to (4), we introduce the use of the regression loss as a measure for the relevance of individual components in the differential Jacobian. Solving an overdetermined linear equation of the general form as in Freedman (2009) by linear regression, we obtain the solution

$$q^* = (A^T A)^{-1} A^T b,$$

and we define the (linear) regression loss r as

$$r = \|b - Aq^*\|.$$

As shown in the [Supplementary Material](#), under numerical variations in b the variation of the regression solution q is proportional to the condition number of A , while the variation in the regression loss r does not scale with the condition number.

This property can be used to make the determination of large elements in the differential Jacobian more robust against fluctuations in the data. To this end, we construct a “regression loss matrix” R that aims to capture the relative importance of individual elements in the differential Jacobian. The regression loss matrix has the same dimension and sparsity structure as the Jacobian J , determined by Sim-Network. Each non-zero element of the regression loss matrix is computed as in Equation (9), where A_c is calculated by combining

A_h and A_d in Equation (8) with the additional constraint that only that single element J_{ij} may differ between the Jacobians J_h and J_d , while all other elements are equal.

$$\begin{aligned} q_s^* &= (A_c^T A_c)^{-1} A_c^T b_s \\ R_{ij}^* &= \min_{b_s} \|b_s - Aq_s^*\|. \end{aligned} \quad (9)$$

To solve the regression problem, one needs to use specific values for the fluctuation matrices D_1 and D_2 . However, in practice these are not known. Therefore, similar as in previous studies (Sun and Weckwerth 2012, Nägele *et al.* 2014), we sample over possible values b_s of the fluctuation matrices (diagonal elements distributed between 0 and 1), and take the minimum regression loss as the overall result for the regression loss matrix as in Equation (9). A more comprehensive description of the new algorithm is presented in [Supplementary Material S1](#).

For comparison, we replicated the L-p optimization used in the previous work (Sun and Weckwerth 2012, Nägele *et al.* 2014). The general idea of the L-p optimization is that the L-p cost optimization will induce most components in the optimized differential Jacobian matrix [refer to (DJ-1) in Equation (4)] close to zero. Furthermore, previous studies do L-p optimization based on diagonal D matrix; we extended it into diagonal-dominant matrix optimization and applied an improved L-p algorithm, which integrates several global optimization approaches. The details of the original and improved L-p optimization approach are presented in [Supplementary Material S1](#).

2.4 The inverse differential metabolic interaction network

The differential metabolic interaction network illustrates the change of the metabolic network between two phenotypes and is visualized by circular interaction plots. Here, each node i represents a metabolite in the dataset; the thickness of the line $j \rightarrow i$ is proportional to DJ_{ji} in Equation (4) or entry R_{ji}^* of the regression loss matrix (9); the node size i is proportional to DJ_{ii} in Equation (4) or the entry R_{ii}^* of the regression loss matrix (9). The generation of these circular plots is included in the COVRECON toolbox with different settings for the resulting differential metabolic interaction network as shown in the [Supplementary Material S3](#) (toolbox manual). The detailed enzyme and gene information of each superpathway can be checked interactively in the circular plot.

2.5 Model case studies

To evaluate the new regression loss Jacobian algorithm, we utilize an abstract test model and several published models obtained from the EBI BioModels database (Malik-Sheriff *et al.* 2020). Seven differently sized models are utilized in this evaluation. For each model, we introduced different reaction parameters as described in [Supplementary Material S1](#) (refer to [Supplementary Section S3](#)) to obtain the two conditions (“h” and “d”).

Here, we note that for each model, all the constant compounds and conserved compounds are excluded based on the method (Waldherr 2014). The basic information of all the test models is summarized in Table 1. Similar to previous studies (Sun and Weckwerth 2012, Kügler and Yang 2014), we generate artificial covariance data for evaluation using two different approaches. The first approach generates the *in silico* covariance matrix C_1 and C_2 from the Lyapunov equation

Table 1. Overviews of the evaluation models.

Model type	Independent components number	Reaction number	Non-zero Jacobian components number	Matrix condition number scale
Test model	6	7	20	1.00E + 02
The upper glycolysis pathway model	5	8	16	5.00E + 01
EGFR/ERK signaling pathway model	13	13	31	1.00E + 04
Carbohydrate energy metabolism model	12	18	59	1.00E + 06
AMPK-mTOR pathway model	31	48	159	1.00E + 05
Hepatic glucose metabolism model	27	35	108	1.00E + 10
Blood cell metabolism model	33	29	252	1.00E + 10

through giving random fluctuation matrix D_1 and D_2 with different randomness ε_D (details in [Supplementary Material](#)). The second approach generates *in silico* steady-state data from stochastic differential equations (SDE) simulation ([Higham 2008](#)).

3 Results

3.1 The correlation matrix does not recover the Jacobian

Correlation network topology analysis is one of the frequently used multivariate statistical methods to study metabolic interactions ([Weckwerth *et al.* 2004](#), [Weckwerth 2010](#), [Weckwerth 2011](#), [Poldrack *et al.* 2015](#), [Kitagawa *et al.* 2019](#), [Tofte *et al.* 2019](#)). However, correlation network analysis alone is not suitable to study the differential metabolic regulation between two conditions. Using the Models 2, 4, 6, and 7, we generated the differential correlation matrices under the two conditions. Then, we compared the differential Jacobian matrices and the differential correlation matrices ([Supplementary Fig. S1](#)). The results show that the differential correlation matrix cannot recover the differential Jacobian matrix. Here, we note that, a correlation matrix or network can still be utilized to show potential interactions when the Jacobian structure information is not clear ([Poldrack *et al.* 2015](#), [Kitagawa *et al.* 2019](#), [Tofte *et al.* 2019](#)).

3.2 The regression loss Jacobian is more reliable than the L-p-based Jacobian

In the following, we describe the results of applying either the regression loss Jacobian algorithm or the improved L-p optimization algorithm for the differential Jacobian reconstruction on the different models as described in Section 2.4. Since in previous studies ([Steuer *et al.* 2003](#), [Sun and Weckwerth 2012](#), [Kügler and Yang 2014](#), [Nägele *et al.* 2014](#), [Sun *et al.* 2015](#), [Wilson *et al.* 2020](#), [Weismann *et al.* 2023](#)), the fluctuation matrix D is assumed to be a diagonal matrix, we restrict this comparison to a diagonal structure of D .

In this evaluation, artificial data are generated using Gaussian perturbations acting on individual compounds only for all the evaluation models in Section 2.5. The covariance matrices for Models 1 and 2 are computed through SDE simulation with 100 and 300 samples, respectively, while in other models the covariance matrices are determined from the Lyapunov equation with $\varepsilon_D = 0.4$ (Section 2.5). The Jacobian reconstruction made use of structure information derived directly from the underlying model. Both the L-p optimization method and the new regression loss Jacobian algorithm were applied to that data. As shown in [Supplementary Table S1](#), the new inverse method achieves a more accurate result with less computation time than the L-p optimization. The table

shows the accuracy of three inverse differential Jacobian approaches: Kügler *et al.* ([Kügler and Yang 2014](#)) (details in [Supplementary Material S1](#)), an improved L-p optimization approach, and our new inverse Jacobian algorithm. The indicated cost is the L-p optimization loss; the target cost is the L-p loss calculated with the real b_b and b_d (details in [Supplementary Material S1](#)). From the table, we can see that the improved L-p optimization approach can obtain a better result than Kügler *et al.* ([Kügler and Yang 2014](#)), but requires more computation time. Our new inverse Jacobian algorithm will need fewer samples to reach a similar accuracy (100 compared to 1000) while also cutting down the computation time.

The inverse differential metabolic interaction network for both the L-p algorithm and our new regression loss Jacobian algorithm with all the models are shown in [Fig. 2](#). The results show that because of the instability of the regression solution with ill-conditioned matrices, the L-p optimization will be inadequate when either the condition number $K(J)$ of the Jacobian is large, or the model dimension is large. The regression loss Jacobian algorithm can achieve a better accuracy and numerical stability by utilizing the regression loss matrix R^* to recover the relevant components of the differential Jacobian DJ . The actual matrices (in place of the circular interaction plots) are shown in [Supplementary Fig. S2](#). Furthermore, [Supplementary Fig. S3](#) shows a scatter plot between the real differential Jacobian components values, scaled to the interval (0, 1) and the calculated $r(i, j)$ in our regression loss Jacobian algorithm for all models. This confirms that with enough samples, the regression loss Jacobian algorithm can find most differential Jacobian components for the two conditional Jacobian matrixes, except for some false negative components in the larger models 6 and 7.

To test the numerical stability of the new algorithm, we calculated the statistical accuracy by using different noise levels in the D matrix, $\varepsilon_D = 0.2, 0.3, 0.4$, and 0.5 . For each model and ε_D , we repeated the evaluation 100 times and calculated the replicability of the Top 1, 3, and 5 differential Jacobian components in R^* [refer to Equation (S4*)]. The results are listed in [Table 2](#), where we observe that the new algorithm can identify the Top 1, 3, and 5 components with high accuracy. Of note, even in the larger models 6 and 7, which have several false negatives, these false negatives still have a high consistency, indicating that the noise level in the D matrix does not significantly impact the results.

Finally, we tested the regression loss Jacobian method with covariance data generated from stochastic simulations for the Models 1, 3, 4, and 5 with a stochastic second-third order Runge–Kutta implicit method. For each model, we added the stochastic Gaussian noise perturbations to every component at each time step ([Higham 2008](#)), which corresponds to the

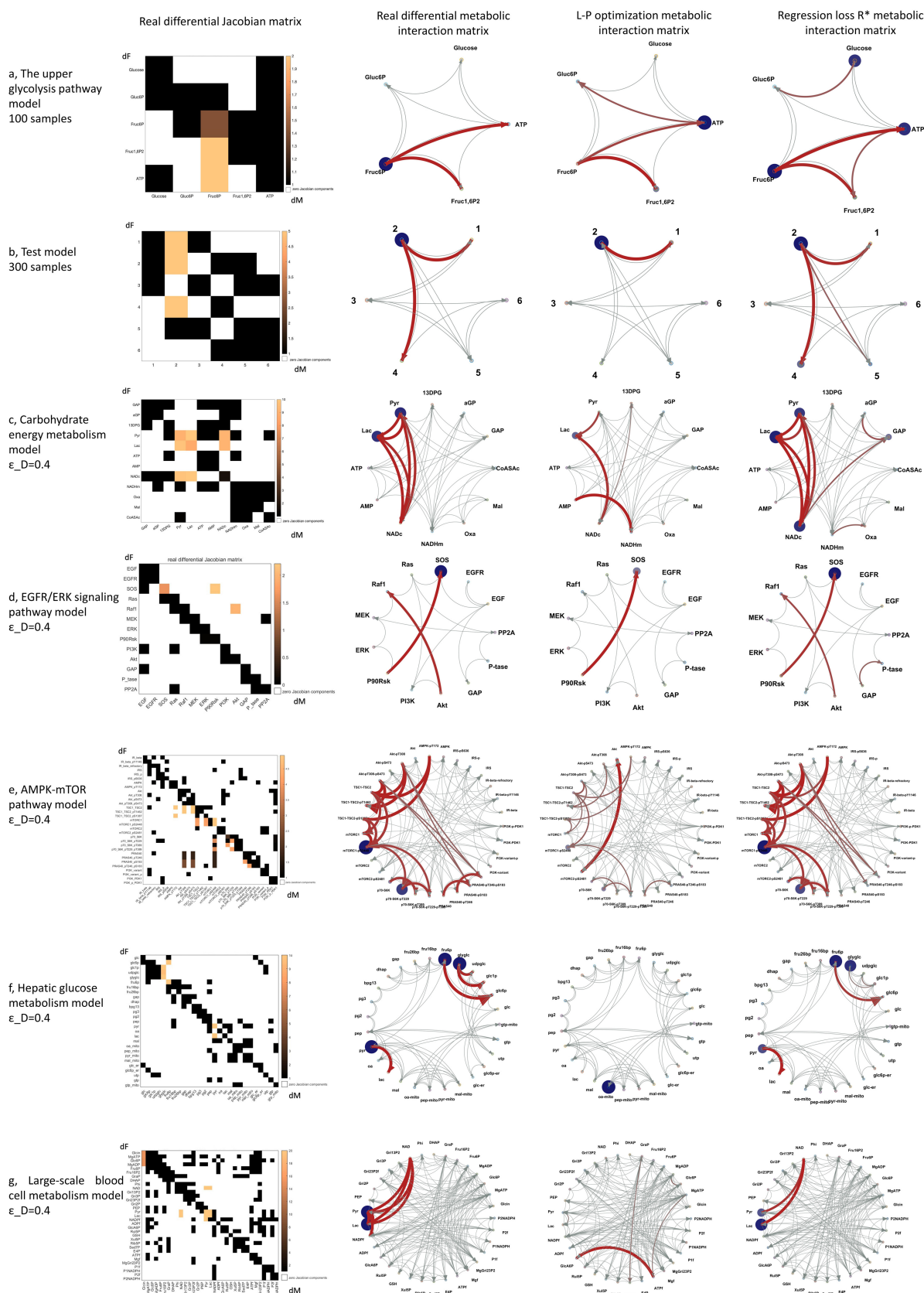


Figure 2. The inverse differential metabolic interaction network with both L-P optimization and the new regression loss Jacobian algorithm for all evaluation models in Section 2.5. The evaluation models are: (a) textbook model of the upper glycolysis pathway; (b) test model; (c) carbohydrate energy metabolism model; (d) EGFR/ERK signaling pathway; (e) AMPK-mTOR pathway model; (f) hepatic glucose metabolism model; and (g) large-scale blood cell metabolism model. In each sub-graph, the left two subplots give the real differential Jacobian matrix and differential interaction network; the right two subplots show the calculated differential interaction network through the L-p optimization approach and the regression loss Jacobian algorithm, respectively (refer to Section 2.3).

Table 2. The regression loss Jacobian 100 repeats accuracy results with diagonal theoretical D for the evaluation models 4–7 as in Section 2.5.

Evaluation model	Accuracy in 100 repeats (Top 1, Top 3, Top 5)			
	$\varepsilon_D = 0.2$	$\varepsilon_D = 0.3$	$\varepsilon_D = 0.4$	$\varepsilon_D = 0.5$
Carbohydrate energy metabolism model	1, 0.99, 0.96	0.88, 0.88, 0.84	0.86, 0.83, 0.76	0.72, 0.70, 0.63
AMPK-mTOR pathway model	1, 1, 1	1, 1, 1	1, 1, 1	1, 1, 0.99
Hepatic glucose metabolism model	1, 1, 0.98	1, 1, 0.96	0.99, 0.88, 0.84	0.97, 0.81, 0.84
Blood cell metabolism model	1, 1, 0.95	1, 1, 0.93	1, 0.96, 0.88	0.97, 0.95, 0.84

nominal D matrix being a diagonal matrix. For each model, we repeated the computation with covariances computed from 100 and 1000 samples to evaluate the effect of sample size on the results. [Supplementary Fig. S4](#) demonstrates the results of the new inverse Jacobian algorithm. As shown in [Supplementary Fig. S4a](#) and [Fig. 4b](#), for small models (about 10 variables) the algorithm correctly finds the large differential Jacobian components through large values in the regression loss R^* using only 100 samples. For larger models, the algorithm is only able to find some of the relevant differential Jacobian components with 100 or 1000 samples, as shown in [Supplementary Fig. S4c and d](#). Even though the detection is improved with 1000 compared to 100 samples, there are still some false negatives even with 1000 samples. We can conclude that this new inverse Jacobian algorithm gives reliable results using about 100 samples for metabolic models with about 10 variables. For larger models, we expect the required sample number to grow with the square of the model size, presenting a practical challenge for larger models. However, even with an insufficient number of samples, we can still find several relevant differential Jacobian components through large values of the regression loss R^* .

3.3 COVRECON workflow case study

As in the first step of COVRECON, Sim-Network will reconstruct a reduced model; we need to test the influence of this network reduction approach (see “5 Effect of the network reduction on the Jacobian reconstruction” in [Supplementary Material S1](#)). The result verifies the feasibility of this first step (Sim-network) in the COVRECON approach. Even when using the reduced network structure for the Jacobian reconstruction, the algorithm is able to detect most of the relevant interactions in the original network.

In the next step, the complete COVRECON workflow is tested starting from a data covariance matrix, without additional structure information. The test is done with Models 6 and 7 (hepatic glucose metabolism model and blood cell metabolism model). With these models, we generate the covariance matrices C_b and C_d for both conditions “d” and “h” with the Lyapunov equation using $\varepsilon_D = 0.5$. In the reconstruction, we first use Sim-Network to determine the Jacobian structure information, relying biochemical pathway information in the KEGG database for *H.sapiens*. As algorithm parameters, we use a cost threshold of 2; discard side-metabolites (listed in [Supplementary Material](#)); apply reverse reaction weight of 2; and use thermodynamics to modify the reverse reaction weight. The reconstructed metabolic interaction networks and Jacobian structure matrices of the two models are shown in [Fig. 3](#).

Generally, we find that the network structure determined by Sim-network contains more interactions than the related literature model. A few examples are labeled in [Fig. 3](#) and discussed in the following. Labels a1 and b1 correspond to one-step

reaction interactions: a1, D-Glucose 6-phosphate to D-Glucose (KEGG R00303) and b1, 2,3-Bisphospho-D-glycerate to 2-phospho-D-glycerate (KEGG R09532). In addition, in cases a2 and b2, the metabolites are connected because they occur together in a reaction: a2, Pyruvate and UTP in (KEGG R00659) and b2, Glycerone phosphate and D-Erythrose 4-phosphate in (KEGG R01829). These reactions are not included in the literature models, but are registered in KEGG and are thus found by applying Sim-Network. On the other hand, in case b3, the connection from D-Glyceraldehyde 3-phosphate (GraP) to D-Fructose 1,6-bisphosphate (Fru16P2) is found by Sim-Network as GraP->D-Fructose 6-phosphate (Fru6P)-> Fru16P2 (step1: KEGG R08575, step 2: KEGG R00756); but it is subsequently discarded as an indirect interaction since the intermediate metabolite Fru6P is also in our network (according to the network simplification strategy explained in the method part). As for case b4, the Jacobian interaction between Fru16P2 and Pyruvate & Phosphoenolpyruvate is not discovered by Sim-network. This is because Fru16P2 will activate the catalyzing enzyme Pyruvate kinase through an allosteric regulation, but Sim-network does currently not take interactions resulting from enzyme regulations into account. The circular plot in matlab format is in [Supplementary Materials S4 and S5](#), where the detailed information of all superpathways can be checked. In addition, the text version of the superpathways information is presented in [Supplementary Material S2](#).

In conclusion, compared with manually built models, Sim-Network has several advantages. First, the resulting model is all-inclusive, for it will explore all the reactions in one specific organism and locate all possible routes, without relying on a modeler’s domain knowledge. On the other hand, the automatically constructed model will be more complex than a manually built model, and it might include rare reactions and routes. Domain knowledge could thus still be helpful to remove individual interactions, starting from the detailed information given by the Sim-network reconstruction.

Moreover, the enzyme regulatory interactions are not included in the current tool. In fact, these interactions are still rarely known except for several widely studied enzyme activators or inhibitors, such as fructose bisphosphate for pyruvate kinase. Future versions of COVRECON could make use of an enzyme regulation database to also take these interactions into account.

With the reconstructed network information, we then apply the regression loss Jacobian algorithm to the covariance matrices. As shown in [Fig. 4](#) and [Supplementary Fig. S5](#), the COVRECON approach shows similar results to the results with exact Jacobian structure information obtained from the model. As a further evaluation, we also consider different Sim-network settings to reconstruct the network (cost threshold 1, reverse reaction weight 1, no thermodynamic strategy, see [Supplementary Material S1](#) “6 Sim-Network Matlab Interface and default settings”). This will result in more

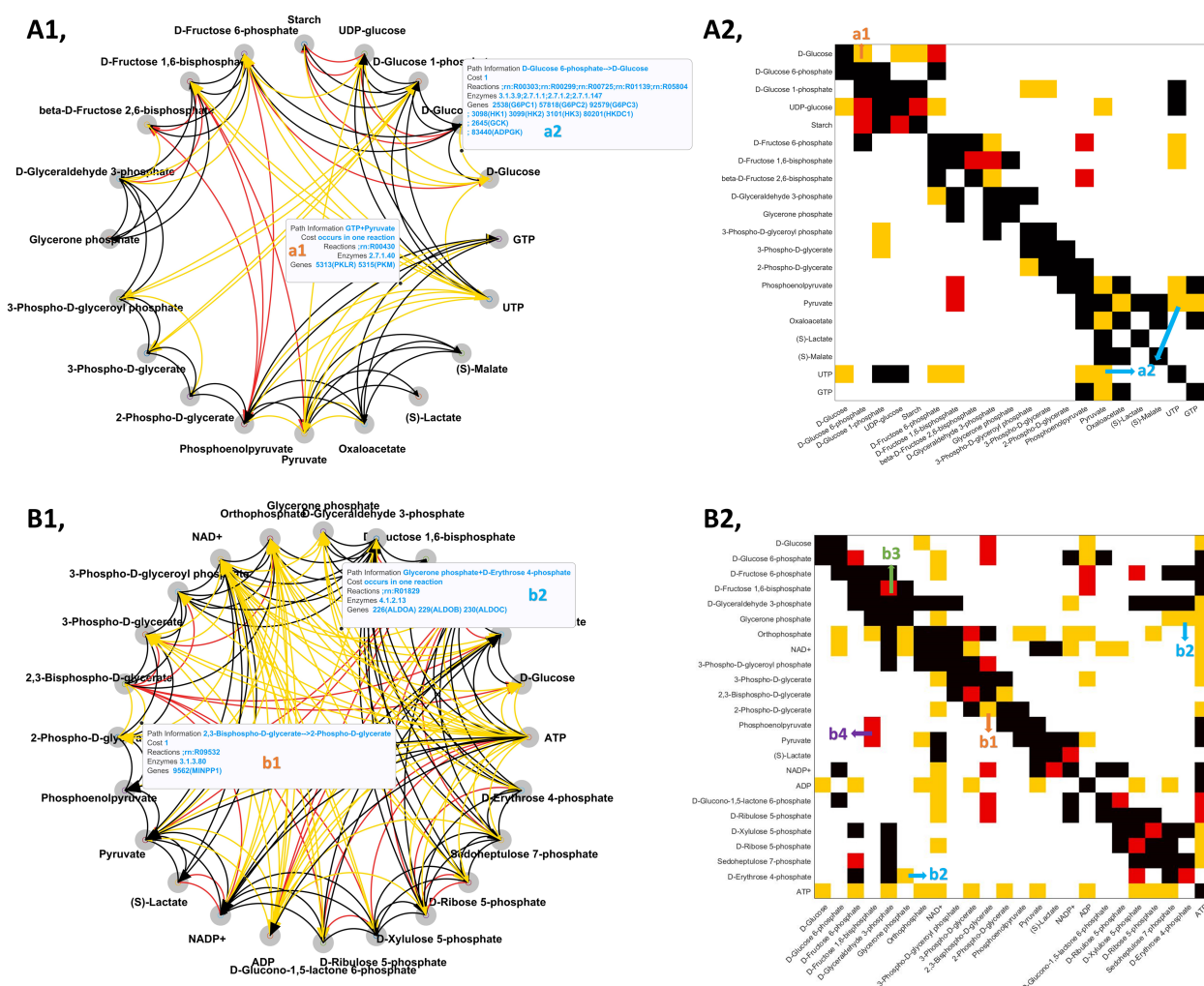


Figure 3. Biochemical pathway network reconstruction and Jacobian structure matrix of model 6 and 7 with Sim-network tool. (A) and (B) give two example cases for the case study of Sim-network. (A1) and (B1) present the reconstructed networks. (A2) and (B2) show the related Jacobian structure information compared to the literature Jacobian structures. An orange Jacobian line (left subplots) or component (right subplots) represents a new added non-zero Jacobian interaction, a red Jacobian line (left subplots) or component (right subplots) represents an unconnected non-zero Jacobian interaction. The marked Jacobian components a1, a2, and b1–b4 are the typical cases for illustration.

connections because we treat the forward and reverse reaction direction in the same way. As shown in [Supplementary Fig. S6](#), with this setting, the reconstructed model will miss fewer components present in the literature model but have even more components that are not present in the literature model. The resulting regression loss matrix R^* remains similar in each case. Overall, this analysis verifies that the COVRECON workflow can recover relevant interactions in a differential Jacobian from only metabolite covariance data.

3.4 Application to an experimental dataset from literature

Finally, we applied the COVRECON to a real experimental dataset from a breast cancer study ([Di Filippo et al. 2022](#)). We analyzed the differential Jacobian matrix between two breast cancer cell lines: non-tumorigenic breast epithelial cell line (MCF102A) and pleural effusion metastasis of a breast adenocarcinoma (MCF7). For the reconstruction, we use the KEGG dataset with organism *H.sapiens* (KEGG code: hsa), the Sim-network settings are left at their default values, and the transcriptomic dataset is used to discard inactivate reactions with GIMME method (refer to Section 2.2). The

discovered superpathways with relevant reactions, enzymes, and genes information are listed in [Supplementary File S2](#). A COVRECON toolbox manual of the case study is presented in [Supplementary File S3](#).

Figure 5 illustrates the inverse differential metabolic interaction network, where highlighted components imply major differences in metabolic interactions between the two datasets. Here, different from Section 2.4, the node size represents the Variable Importance [calculated as $-\log(P)$, where P is the P -value of a t -test from the metabolomics datasets comparison]. The corresponding matrix is presented in [Supplementary Fig. S7](#). Moreover, we attached the detailed enzyme gene information of each metabolic interaction line in [Supplementary Material S8](#), where one can interactively check detailed information about each metabolic interaction. In addition, the information is also presented in [Supplementary Material S2](#). As shown in [Fig. 5](#), we found several biochemical perturbations between the two different cell lines MCF102A and MCF7. These predicted biochemical perturbations are well in agreement with significantly different gene expression levels shown as bar plots (see also [Supplementary Fig. S8](#)). Accordingly, COVRECON is able to identify important perturbation points in the experimental

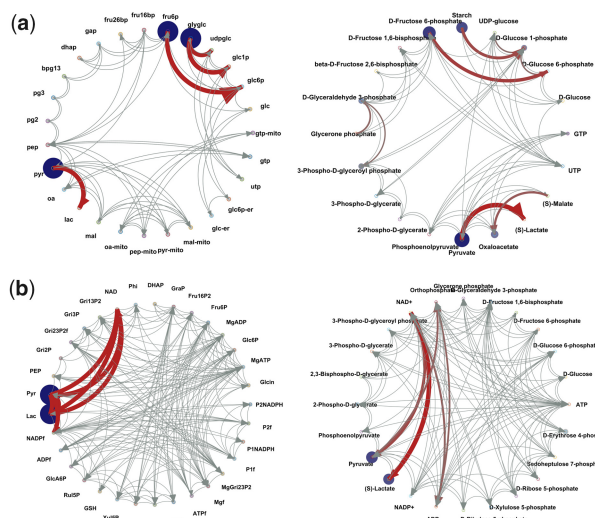


Figure 4. Complete workflow of the COVRECON strategy. SIM-Network is used for network reduction and metabolic interaction matrix generation. Subsequently, a covariance data matrix is used to calculate the differential Jacobian in form of the Regression loss matrix R^* . Inverse differential metabolic interaction network determined from COVRECON (right) and the exact differential metabolic interaction network (left). The circular plot in matlab figure format is in [Supplementary Materials S6 and S7](#).

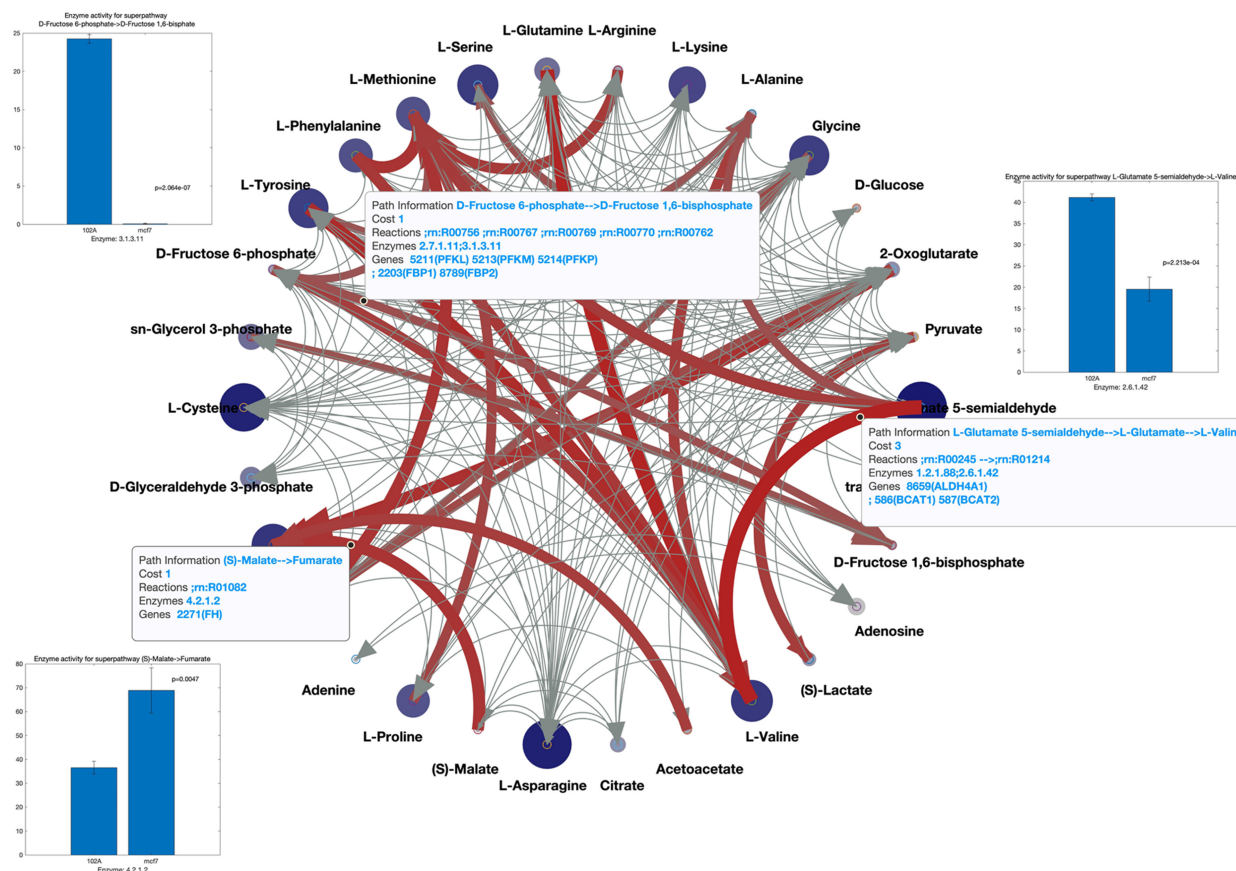


Figure 5. The differential metabolic interaction network plot, where the node size represents the variable importance [calculated as $-\log(P)$, P is the P -value of t -test from metabolomics datasets comparison], and the line widths describe the metabolic interaction difference. We chose and list the superpathways details of three highlighted metabolic interactions and performed a t -test for relevant transcriptomic data. The t -test results show significant ($P < 0.05$). The matlab format circular plot with all superpathway information is in [Supplementary Material S8](#).

system. In [Supplementary Fig. S8](#), we list all the t -test results of the transcriptomic profile for interactions with highlighted value above 0.5.

4 Conclusion

In this article, we have developed a new approach for the inverse differential Jacobian algorithm: COVRECON. Unlike the widely used constraint-based analysis for the reaction-flows optimization, this approach endeavors to discover the changes in metabolic interactions between two conditions of the system. This new approach offers an alternative mathematical approach to process and interpret large-scale metabolomics data. The open-source matlab-tool is available in the website <https://bitbucket.org/mosys-univie/covrecon/>. [Supplementary File S3](#) gives a toolbox manual.

The main subject of this new approach is large-scale metabolomics data. Meanwhile, we can also integrate different OMICS datasets. In Sim-Network, the transcriptomic data can be used to exclude reactions with low activity; and the important enzyme regulations can be added into pathway search for a better network reconstruction.

From our knowledge, COVRECON is the first method to integrate different OMICS data and automatically construct a metabolic interaction model providing a more general

network structure and perturbations of the same. As for the inverse Jacobian part, this work introduces a novel algorithm improving accuracy and stability, reduces computation time, and extends the method to large-scale models.

Acknowledgements

We thank Dr Xiaoliang Sun for valuable discussions.

Author contributions

W.W. and J.L. conceived the study. J.L. developed the method. J.L. and S.W. implemented and performed the analysis and interpretation of the results. S.W. and J.L. wrote the first version of the manuscript. S.W., W.W. and J.L. reviewed and approved the final version of the manuscript.

Supplementary data

Supplementary data are available at *Bioinformatics* online.

Conflict of interest

None declared.

Funding

This work was supported by Ph.D. scholarships provided by China Scholarship Council (CSC) [grant number: 201806010428 to J.L.]. Part of this work was supported by the department fundings of Molecular Systems Biology Lab (MOSYS).

Data availability

The data underlying this article are available in the article and in its online supplementary material.

References

- Becker SA, Palsson BO. Context-specific metabolic networks are consistent with experiments. *PLoS Comput Biol* 2008;4:e1000082.
- Caspi R, Billington R, Keseler IM *et al.* The MetaCyc database of metabolic pathways and enzymes—a 2019 update. *Nucleic Acids Res* 2020;48:D445–53.
- Chen T, Guestrin C. XGBoost: a scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 785–94. 2016.
- De Martino D, Mc Andersson A, Bergmiller T *et al.* Statistical mechanics for metabolic networks during steady state growth. *Nat Commun* 2018;9:2988.
- Di Filippo M, Pescini D, Galuzzi BG *et al.* INTEGRATE: model-based multi-omics data integration to characterize multi-level metabolic regulation. *PLoS Comput Biol* 2022;18:e1009337.
- Elgendy M, Cirò M, Hosseini A *et al.* Combination of hypoglycemia and metformin impairs tumor metabolic plasticity and growth by modulating the PP2A-GSK3 β -MCL-1 axis. *Cancer Cell* 2019;35:798–815.e795.
- Freedman DA. *Statistical Models: Theory and Practice*. Cambridge University Press, 2009.
- Higham DJ. Modeling and simulating chemical reactions. *SIAM Rev* 2008;50:347–68.
- Hucka M, Finney A, Sauro HM *et al.*; SBML Forum. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* 2003;19:524–31.
- Karp PD, Midford PE, Billington R *et al.* Pathway tools version 23.0 update: software for pathway/genome informatics and systems biology. *Brief Bioinform* 2021;22:109–26.
- Khatibipour MJ, Kurtoğlu F, Çakır T. JacLy: a Jacobian-based method for the inference of metabolic interactions from the covariance of steady-state metabolome data. *PeerJ* 2018;6:e6034.
- King ZA, Lu J, Dräger A *et al.* BiGG models: a platform for integrating, standardizing and sharing genome-scale models. *Nucleic Acids Res* 2016;44:D515–22.
- Kitagawa H, Ohbuchi K, Munekage M *et al.* Phenotyping analysis of the Japanese Kampo medicine maoto in healthy human subjects using wide-targeted plasma metabolomics. *J Pharm Biomed Anal* 2019;164:119–27.
- Kügler P, Yang W. Identification of alterations in the Jacobian of biochemical reaction networks from steady state covariance data at two conditions. *J Math Biol* 2014;68:1757–83.
- Kuwahara H, Alazmi M, Cui X *et al.* MRE: a web tool to suggest foreign enzymes for the biosynthesis pathway design with competing endogenous reactions in mind. *Nucleic Acids Res* 2016;44:W217–25.
- Lamichhane S *et al.* An overview of metabolomics data analysis: current tools and future perspectives. *Compr Anal Chem* 2018;82:387–413.
- LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature* 2015;521:436–44.
- Linke M, Pham HTT, Katholnig K *et al.* Chronic signaling via the metabolic checkpoint kinase mTORC1 induces macrophage granuloma formation and marks sarcoidosis progression. *Nat Immunol* 2017;18:293–302.
- Malik-Sheriff RS, Glont M, Nguyen TVN *et al.* BioModels—15 years of sharing computational models in life science. *Nucleic Acids Res* 2020;48:D407–15.
- Mitchell M. *An Introduction to Genetic Algorithms*. MIT Press, 1998.
- Moriya Y, Shigemizu D, Hattori M *et al.* PathPred: an enzyme-catalyzed metabolic pathway prediction server. *Nucleic Acids Res* 2010;38:W138–43.
- Nägele T, Mair A, Sun X *et al.* Solving the differential biochemical Jacobian from metabolomics covariance data. *PLoS One* 2014;9:e92299.
- Ogata H, Goto S, Sato K *et al.* KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res* 1999;27:29–34.
- Öksüz M, Sadıkoğlu H, Çakır T. Sparsity as cellular objective to infer directed metabolic networks from steady-state metabolome data: a theoretical analysis. *PLoS One* 2013;8:e84505.
- Poldrack RA, Laumann TO, Koyejo O *et al.* Long-term neural and physiological phenotyping of a single human. *Nat Commun* 2015;6:1–15.
- Seaver SMD, Liu F, Zhang Q *et al.* The ModelSEED biochemistry database for the integration of metabolic annotations and the reconstruction, comparison and analysis of metabolic models for plants, fungi and microbes. *Nucleic Acids Res* 2021;49:D575–88.
- Sidak D, Schwarzerová J, Weckwerth W *et al.* Interpretable machine learning methods for predictions in systems biology from omics data. *Front Mol Biosci* 2022;9:926623.
- Steuer R, Kurths J, Fiehn O *et al.* Observing and interpreting correlations in metabolomic networks. *Bioinformatics* 2003;19:1019–26.
- Sun X, Länger B, Weckwerth W. Challenges of inversely estimating Jacobian from metabolomics data. *Front Bioeng Biotechnol* 2015;3:188.
- Sun X, Weckwerth W. COVAIN: a toolbox for uni- and multivariate statistics, time-series and correlation network analysis and inverse estimation of the differential Jacobian from metabolomics covariance data. *Metabolomics* 2012;8:81–93.
- Tofte N, Suvitaival T, Trost K *et al.* Metabolomic assessment reveals alteration in polyols and branched chain amino acids associated with present and future renal impairment in a discovery cohort of 637 persons with type 1 diabetes. *Front Endocrinol (Lausanne)* 2019;10:818.
- Waldherr S. A guideline to model reduction by stoichiometric decomposition for biochemical network analysis. In: *Proceedings of the 21st International Symposium on Mathematical Theory of Networks and Systems*. 490–5. 2014.
- Wang L, Dash S, Ng CY *et al.* A review of computational tools for design and reconstruction of metabolic pathways. *Synth Syst Biotechnol* 2017;2:243–52.
- Weckwerth W. Metabolomics in systems biology. *Annu Rev Plant Biol* 2003;54:669–89.

- Weckwerth W. Metabolomics: an integral technique in systems biology. *Bioanalysis* 2010;**2**:829–36.
- Weckwerth W. Unpredictability of metabolism—the key role of metabolomics science in combination with next-generation genome sequencing. *Anal Bioanal Chem* 2011;**400**:1967–78.
- Weckwerth W. Toward a unification of system-theoretical principles in biology and ecology—the stochastic Lyapunov matrix equation and its inverse application. *Front Appl Math Stat* 2019;**5**:29.
- Weckwerth W, Loureiro ME, Wenzel K *et al.* Differential metabolic networks unravel the effects of silent plant phenotypes. *Proc Natl Acad Sci USA* 2004;**101**:7809–14.
- Weiszmann J, Walther D, Clauw P. Metabolome plasticity in 241 *Arabidopsis thaliana* accessions reveals evolutionary cold adaptation processes. *Plant Physiol* 2023;kiad298.
- Wilson JL, Nägele T, Linke M *et al.* Inverse data-driven modeling and multiomics analysis reveals Phgdh as a metabolic checkpoint of macrophage polarization and proliferation. *Cell Rep* 2020;**30**:1542–52.e1547.