RECONSTRUCTION of SUBSTRATE COMPLEXES in BIOCHEMICAL NETWORKS from TIME-RESOLVED RELATIVE COMPOUND LEVELS

Jeanne M. O. Eloundou-Mbebi, Zoran Nikoloski

Systems Biology and Mathematical Modeling Group, Max Planck Institute of Molecular Plant Physiology, 14476 Potsdam-Golm, Germany

*** nikoloski@mpimp-golm.mpg.de**

Abstract

Reconstruction of networks of biochemical reactions each involving multiple substrates and products together with their stoichiometries, given time-resolved profiles of the compounds, is a first step in modeling and understanding the control of biochemical systems. The existing approaches largely focus on identification of (pairwise) relationships between compounds by using similarity-measures and regression-based methods, thus neglecting the stoichiometry and directed hypergraph structure of biochemical reactions. Here, we present the first computational approach for identifying the reaction substrates and their stoichiometries (*i.e.*, substrate complexes): The approach assumes mass action kinetics, largely applicable to non-enzymatic (bio)chemical systems, and employs replicated time-resolved relative levels for the considered compounds. It combines techniques from constrained regularized regression and mathematical programming to arrive at robustly identified substrate complexes. The proposed approach is validated on synthetic data from two biologically plausible networks. In addition, by using metabolomics profiles obtained from a glycine hydrothermal reaction, we show that the predicted substrate complexes are in line with chemical principles, thus, shedding light on prebiotic chemistry. The approach provides the basis for incorporation of elementary biochemical principles for accurate reconstruction of large-scale time-resolved biochemical networks.

Introduction

Biochemical networks facilitate the transformations necessary for performing various tasks and processes in biological systems. A biochemical network is a set of reactions denoting the processes that transform compounds, called *substrates*, to other compounds, called *products*. Knowledge of the biochemical reactions through which compounds are transformed is prerequisite to understanding the underlying mechanisms, behavior and control of biochemical systems [7, 9, 22]. Therefore, research efforts have focused on determining the set of biochemical reactions which take place in a given system given some read-outs from the involved components, known as the *biochemical network reconstruction problem*.

Biochemical networks are more general than signaling and transcriptional regulatory networks, usually modeled by graphs capturing bilateral interactions (*i.e.*, interactions between two proteins or genes) [8]. They are accurately represented by hypergraphs,

since biochemical reactions usually involve multiple substrates and products participating in many-to-many relationships [3, 10].

The rate at which a chemical reaction transforms the substrates into products is largely determined by the concentration and stoichiometry with which the compounds enter the reaction, the thermodynamic characteristics, and the kinetic law (including parameters dependent on temperature, pressure, and pH, to name a few). The problem of reconstructing biochemical reactions is that of determining (some of) the above-mentioned determinants given the levels of the involved compounds, measured by a variety of existing technologies [16]. Therefore, any method for reconstruction of biochemical networks which accounts only for bilateral relationships or neglects stoichiometry usually entails loss of information that may lead to inaccurate predictions and wrong interpretations.

Ever since the seminal work of [2], the existing solutions to the problem of reconstructing biochemical networks from (non-)stationary measurements of compound levels rely on applying various similarity measures to the obtained data profiles. The similarity measure can be used to determine statistically significant pairwise relationships between compounds. Therefore, these approaches neglect the many-to-many relationships between compounds in a reaction as well as the stoichiometry with which compounds enter the reaction.

Here we present a solution strategy for one aspect of the problem of reconstructing biochemical networks from time-resolved levels of the involved compounds accessible via metabolomics technologies. Under the assumption of mass action kinetics, our approach can be used to infer the set of substrates together with their stoichiometry, solely with the knowledge of the levels of the involved compounds at the given time points. Therefore, it improves the existing solutions based on similarity measures only. The proposed approach combines techniques from fitting and variable selection methods together with mathematical programming. To validate our approach, we apply it to synthetic data from biologically plausible toy networks. In addition, we use the approach to the time-resolved metabolomics data gathered from glycine degradation by hydrothermal reaction (HTR) [1], as a first step to understanding prebiotic chemistry.

1 Methods

1.1 Derivation of the proposed approach

We assume that we are given data on the levels of n compounds from l time-resolved experiments (replicates) with $T > 1$ time points. Let the n compounds be transformed through m reactions obeying mass-action law, applicable to modeling homogeneous well-mixed chemical systems. The rate of the i^{th} reaction, $1 \leq i \leq m$, at time point t is then given by

$$v_{i,t} = p_i \prod_{j=1}^{n} x_{j,t}^{\alpha_{ij}^t}, \tag{1}$$

where p_i denotes the (temperature and pressure-dependent) rate constant, $x_{j,t}$ stands for the concentration of the j^{th} compound at time point t, and α_{ij}^t is the stoichiometric coefficient of the j^{th} compound appearing in/as a substrate in the i^{th} reaction at time point t. Note that the reactants of the i^{th} reaction together with their stoichiometry are referred to as a *substrate complex*. Since we are interested in reconstructing the entire network, the effect of all m reactions simultaneously need to be considered; thus,

multiplying over all m reactions, we have:

$$\prod_{i=1}^{m} v_{i,t} = \prod_{i=1}^{m} \left(p_i \prod_{j=1}^{n} x_{j,t}^{\alpha_{ij}^t} \right). \tag{2}$$

Since our approach uses all available data replicates, the expression given by Eq. (2) also holds for every reaction rates $v_{i,t}^*$ for different concentration $x_{j,t}^*$ of the substrates (*e.g.*, replicate measurement), *i.e.*,

$$\prod_{i=1}^{m} v_{i,t}^* = \prod_{i=1}^{m} \left(p_i \prod_{j=1}^{n} x_{j,t}^{*\alpha_{ij}^t} \right). \tag{3}$$

Taking the logarithm of the ratio of Eqs. (2) and (3), one obtains

$$\log \frac{\prod_{i=1}^{m} v_{i,t}}{\prod_{i=1}^{m} v_{i,t}^*} = \sum_{j=1}^{n} \left[\sum_{i=1}^{m} \alpha_{ij}^t \right] \log \frac{x_{j,t}}{x_{j,t}^*}. \tag{4}$$

By setting $z_t = \log \dfrac{\prod_{i=1}^{m} v_{i,t}}{\prod_{i=1}^{m} v_{i,t}^*}$ and $Y_{j,t} = \log \dfrac{x_{j,t}}{x_{j,t}^*}$; then, Eq. (4) becomes

$$z_t = \sum_{j=1, j\neq k}^{n} \left[\sum_{i=1}^{m} \alpha_{ij}^t \right] Y_{j,t} + \sum_{i=1}^{m} \alpha_{ik}^t Y_{k,t}, \tag{5}$$

leading to

$$Y_{k,t} = \frac{z_t}{\sum_{i=1}^{m} \alpha_{ik}^t} - \sum_{j=1, j\neq k}^{n} \left[\frac{\sum_{i=1}^{m} \alpha_{ij}^t}{\sum_{i=1}^{m} \alpha_{ik}^t} \right] Y_{j,t}. \tag{6}$$

Note that through this transformation, the dependence on the parameters (rate constants) p is removed. The values for the variables $Y_{j,t}$, $(1 \leq j \leq n, 1 \leq t \leq T)$ in Eq. (6) can be obtained by taking the logarithms of the ratio of replicates of the measured levels for the corresponding compound. The relationship given by Eq. (6) can be seen as a linear regression model, where: (i) $Y_{k,t}$ is the response variable, (ii) $Y_{j,t}$ $(j \neq k)$ are the predictors, (iii) $\dfrac{\sum_{i=1}^{m} \alpha_{ij}^t}{\sum_{i=1}^{m} \alpha_{ik}^t}$s $(j \neq k)$ correspond the regression coefficients, and (iv) $\dfrac{z_t}{\sum_{i=1}^{m} \alpha_{ik}^t}$ is the intercept, at time point t.

Since the intercept term $\dfrac{z_t}{\sum_{i=1}^{m} \alpha_{ik}^t}$ is difficult to estimate due to the lack of data on reaction rates, we standardize the variables $Y_{j,t}$, yielding $\dfrac{z_t}{\sum_{i=1}^{m} \alpha_{ik}^t} = 0$ [14]. By denoting the standardized variables by $Y'_{j,t}$, the models to be analyzed are

$$-Y'_{k,t} = \sum_{j=1, j\neq k}^{n} \left[\frac{\sum_{i=1}^{m} \alpha_{ij}^t}{\sum_{i=1}^{m} \alpha_{ik}^t} \right] Y'_{j,t}, \tag{7}$$

for every time point t.

By setting

$$\beta_{j,t} = \frac{\sum_{i=1}^{m} \alpha_{ij}^t}{\sum_{i=1}^{m} \alpha_{ik}^t}, \tag{8}$$

for every compound j and time point t, Eq.(7) becomes,

$$-Y'_{k,t} = \sum_{j=1, j\neq k}^{n} \beta_{j,t} Y'_{j,t}, \tag{9}$$

for every time point t.

By solving (i.e., computing the $\beta_{j,t}$s) the linear regression problem given by Eq. (9), we obtain the possible connections that exist between the substrate complexes of the putative network at time point t. Once Eq. (9) is solved, we recover the stoichiometric coefficients α_{ij}^{t} of each substrate complex j at time point t from the $\beta_{j,t}$s, using Eq. (8).

Solving the linear regression problem given by Eq. (7) using the ordinary least squared (OLS) method, usually results in many non-zero coefficients. As substrate complexes involve only a limited number of compounds [3], a more appropriate solution should rely on solving Eq. (7) by shrinkage-based methods with advantage of sparsity, such as the least absolute shrinkage and selection operator (LASSO) (see Supplementary Information).

The models derived from metabolomics time-series relative levels based on the non-negatively constrained LASSO do not capture the dependence on time. The problem of inferring time-dependent model for a given compound can then be formulated as follows: Given a model at time point t for a compound k, we aim at obtaining the sparsest model for the compound at time point $t+1$ while fitting data and assuring smooth transition between models at the consecutive time points. This mimics the temporal activation of reactions, while minimizing the difference between the temporal networks. In other words, under the biochemically reasonable assumption that the estimation of the parameter β_t at time point t is given by the solution of the LASSO problem

$$\hat{\beta}_t = \underset{\beta_t}{\operatorname{argmin}} \{ \| Y'_{k,t} - \sum_{j=1, j\neq k}^{P} \beta_{j,t} Y'_{j,t} \|_2^2 + \lambda_t \|\beta_t\|_1 \}, \tag{10}$$

we aim at solving the following bi-level problem:

$$\begin{aligned} & \min_{\beta_{t+1}} \|\beta_{t+1} - \beta_t\|_2^2, \\ \text{s.t} \quad & \hat{\beta}_{t+1} = \underset{\beta_{t+1}}{\operatorname{argmin}} \{ \| Y'_{k,t+1} - \sum_{j=1, j\neq k}^{P} \beta_{j,t+1} Y'_{j,t+1} \|_2^2 + \\ & + \lambda_{t+1} \|\beta_{t+1}\|_1 \}, \end{aligned} \tag{11}$$

where $\beta_t = \left(\beta_{j,t}\right)_j$ and $\beta_{t+1} = \left(\beta_{j,t+1}\right)_j$ are the vectors of regression coefficients for the compounds participating in the network at time point t and $t+1$, respectively.

Solving the bi-level problem given by Eq. (19) is equivalent to solving the LASSO problem Eq. (12) below

$$\hat{\beta}_{t+1} = \underset{\beta_{\text{New}}}{\operatorname{argmin}} \{ \| Y - \sum_{j=1, j\neq k}^{2P-1} \beta_j Y'^{j} \|_2^2 + \lambda \|\beta\|_1 \}, \tag{12}$$

where $Y = y + \sum_{j=1, j\neq k}^{2P-1} \beta_{11}^{j} Y'^{j}$, $\lambda = \dfrac{\lambda_{t+1}}{1+\sqrt{\kappa}}$, and β_{New} is the vector made of the first $P-1$ components of β (see Supplementary Information for more details).

1.2 From models to stoichiometry of reactants

In the following, we present the last step of the approach whereby the stoichiometry of the reactants for each reaction are determined, thus, fully resolving the substrate complexes of the network. According to the relationship given by Eq. (8), one can compute the stoichiometric coefficients α_{ij}^t, by solving the (usually underdetermined) linear system

$$\sum_{i=1}^{m} \alpha_{ij}^t = \beta_{j,t}^{(r)} \sum_{i=1}^{m} \alpha_{ik}^t, \quad \text{at time point } t, \tag{13}$$

where j and r, $1 \leq j, r \leq n$, are the indices of the compounds and models, respectively. In addition, the substrate complexes reconstructed for each time point should also satisfy the following biochemically-meaningful constraints: (i) every substrate complex includes at least one compound, (ii) the number of molecules participating in each substrate complex is at most c_1, (iii) not every compound need participate in a substrate complex, (iv) the number of molecules of each compound over all substrate complexes is at most c_2, and (v) each stoichiometric coefficient is non-negative and of value at most c_3.

Finally, the obtained equalities from the statistical models do not necessarily guarantee a lack of conflicts which satisfy all of the listed constraints. In other words, the combination of these constraints with Eq. (13) is not expected to always result in a non-empty feasible space when solving for positive stoichiometric coefficients, due to the strong equality constraints. Therefore, to reconstruct the sparse substrate complexes, we allow for discrepancy, modeled by a real number $\epsilon_j^{(r)}$, to the regression coefficients from the statistical models, which we also aim to minimize. This yields the following linear program:

$$\begin{aligned} &\min \sum_{i=1}^{m}\sum_{j=1}^{P} \alpha_{ij}^t + \sum_{r=1}^{N}\sum_{j=1}^{P} |\epsilon_{j,t}^{(r)}|, \\ &\text{s.t} \quad \begin{cases} \sum_{i=1}^{m} \alpha_{ij}^t = \beta_{j,t}^{(r)} \sum_{i=1}^{m} \alpha_{ik}^t + \epsilon_{j,t}^{(r)} \\ 1 \leq \sum_{j=1}^{P} \alpha_{ij}^t \leq c_1 \\ 0 \leq \sum_{i=1}^{m} \alpha_{ij}^t \leq c_2 \\ 0 \leq \alpha_{ij}^t \leq c_3 \\ \epsilon_{j,t}^{(r)} \in \mathbb{R}, \end{cases} \end{aligned} \tag{14}$$

for all time points t, all reactions i, $1 \leq i \leq n$, all compounds j, $1 \leq j \leq m$, and all models r (N denotes the number of considered models) with c_1, c_2, and c_3 as positive constants.

1.3 Selection of models and number of reactions

The number of models which are included in the program in Eq. (14) can be controlled by the statistical properties, such as the coefficient of determination, given by

$$R_{\beta_t}^2 = \frac{\sigma_{X\beta_t}^2}{\sigma_Y^2},$$

for any Y, X, and β_t being the response variable, the predictor variables and the LASSO coefficients at a given time point t, respectively, and σ^2 denotes the variance [12]. To this end, only models which have coefficient of determination above a threshold τ (*e.g.*, 0.8) can be considered in solving the proposed linear program.

Therefore, any conflicts which can be resolved by means of the introduced variables $\epsilon_{j,t}$ should only be due to models of high explanatory power. In addition, in the following comparative analysis, we explore the residual sum of squares (RSS), *i.e.*, the L_2-norm, of a model with coefficients β given by $\|Y - X\beta\|_2$ (whose value is sensitive to the magnitude of the variables).

Finally, the program in Eq. (14) requires specification of the number of substrate complexes (*i.e.*, the number of reactions) m. Usually, m is not known, and here we use the behavior of $\theta(m) = \frac{\sum_{r=1}^{N}\sum_{j=1}^{P}|\epsilon_{j,t}^{(r)}|}{m}$ for varying m to infer the likeliest number of reactions. The value for $\theta(m)$ specifies the relative relaxation per N statistical models selected based on their coefficient of determination. The "elbow criterion" for the curve $\theta(m)$ for varying m, similarly to the soft criterion for selection of principal components, indicates the value of m where unexpected discrepancy with respect to the statistical models may likely occur.

1.4 Proposed optimization-based algorithm

The algorithm for reconstruction of time-resolved substrate complexes can then be summarized in Algorithm 1, below.

Algorithm 1: Algorithm for reconstruction of time-resolved substrate complexes.

Data: time-resolved data matrix of centered log-transformed ratios Y';
threshold τ for R^2;
time points t and $t+1$;
number of reactions (*i.e.*, substrate complexes) m
Result: set of substrate complexes and stoichiometry for time point $t+1$
for every compound k at time point t **do**
  $\beta_t^k \leftarrow$ cross-validated positively-constrained LASSO;
  if model $R^2_{\beta_t^k} > \tau$ **then**
    $\beta_{t+1}^k \leftarrow$ cross-validated positively-constrained LASSO from Eq. (24);
  end
end
create the linear program in Eq. (14) with models from β_{t+1}^k;
solve for α_{ij} to obtain stoichiometric coefficients of the j^{th} compound in i^{th} complex;

1.5 Implementation

The computation of the LASSO models was performed by the `penalized` package in R [24]. The linear mathematical program was solved by using the `Rcplex` package in R [25]. However, usage of any convex optimization solver is possible at this point. Code snippets for the determination of the models and stoichiometric coefficients are provided in the Supplementary information.

2 Results and Discussion

2.1 Toy network I

To test the proposed approach for reconstruction of substrate complexes, *i.e.*, reactants and stoichiometry of reactions, we used the toy networks I and II given in Fig. 1 and Fig. 3, respectively. The network in Fig. 1 includes six compounds, denoted by A - F,

participating in 10 irreversible reactions. The reactions involve altogether 7 substrate complexes, of which 5 are with a single compound (*i.e.*, 2A, B, D, 2B, F) and the remaining 2 complexes (*i.e.*, B+C, A+E) include two compounds. The corresponding system of ordinary differential equations was integrated, assuming mass action kinetics for the reaction fluxes and with rate constants as shown in Fig. 1. The simulated concentration-time profiles from three different positive initial conditions, shown in Fig. 1 (inlay), were used as replicates.

Fig. 1. Toy network I and parameters. (A) The network includes 7 substrate complexes, 10 irreversible reactions and 6 compounds. The values for the rate constants p_i, $1 \leq i \leq 10$ are given next to the reactions. **(B)** The table includes the three initial conditions, used as replicates, for the concentration of the 6 compounds. **(C)** The levels of the compounds are simulated at time points $t_1 = 0.01$; $t_2 = 2$; $t_3 = 2.5$; $t_4 = 2.9$; $t_5 = 4$; $t_6 = 4.8$, $t_7 = 5$.

Following Algorithm 1, for every time point we extracted the six LASSO models, each with one of the six compounds as a response and the remaining five compounds as predictors. The performance of the model was quantified with respect to the residual sum of squares (RSS), corresponding to the L_2-norm, and the coefficient of determination, R^2. The optimum in Eq. (16) depends on the sparsity and the RSS. In the case of the paradigmatic network, over the first three time points, the LASSO models had coefficients of determination of at least 0.8 (Fig. 2A and S1 Table). The good predictive power of the models in the first three time points was confirmed by the correspondingly small RSS (S2 Table), which tends to increase with time (Fig. 2B inlay).

Fig. 2. Coefficient of determination (R^2) and residual sum of squares (RSS) for the toy network I. The distribution of the two statistics, (A) R^2 and (B) RSS, over the 6 models is shown in the histogram. The inlays illustrate the minimum (Min), average (Mean), and maximum (Max) of the statistics over the models for all time points

Since the stoichiometric coefficient for every compound in every substrate complex of the paradigmatic network in Fig. 1 is at most 2, we set $c_3 = 2$. Moreover, the sum of stoichiometric coefficients for the substrate complex of every reaction is at most 2, and, thus, $c_1 = 2$. Since the maximum sum of a compound occurrence in all reactant is 7 (compound B), $c_2 = 7$. Finally, the number of reactions in the network is given, *i.e.*, $m = 10$.

Solving the program in Eq. (14) with the time-dependent models, over all time points, we extracted nine complexes, including: 1.22F, D+E, 1.23C, 1.64B + 0.35C, A, C+F, A+B, A+F, B.The union of substrate complexes identified over all time points agrees with three complexes with respect to the participation of compounds (marked in blue in S5 Table) and one complex with respect to stoichiometry (marked in green in S5 Table). Altogether, among the five single compounds, three were identified by the approach and one complex out of the two complexes with two compounds was recovered. However, the approach identified four complexes that are not part of the network (marked in red in S5 Table). Altogether, three substrate complexes were not identified by the approach.

Upon perturbation of the time profiles with values following a normal distribution of zero mean and 0.05, 0.5 and 1 variance, we extracted similar substrate complexes, with a slight difference in the stoichiometric coefficients, for each time point. The small difference between the coefficients is in agreement with the small euclidean distance obtained between the stoichiometric matrices with and without noise at different time

points, provided by S1 Fig. Note that we extracted the same complexes from t_2 to t_6, thus we observed a similar behaviour in the euclidean distance between the stoichiometric matrices with and without noise from t_2 to t_6.

2.2 Toy network II: EnvZ-OmpR system

We turn the application of our approach to a prototypical two-component signaling system. The *Escherichia coli* EnvZ-OmpR system which consists of the sensor kinase EnvZ and the response-regulator OmpR, denoted by X and Y, respectively in the EnvZ-OmpR system [18,19]. Both the sensor and the response-regulator have phosphorylated forms, denoted X_p and Y_p. The network in Fig. 3 includes seven compounds denoted $X, XT, X_p, Y, X_pY, Y_p, XTY_p$, participating in 9 reactions. The reactions involve 6 substrate complexes, of which 4 are with single compounds (*i.e.*, X, XT, X_pY, XTY_p) and the remaining 2 complexes (*i.e.*, $X_p + Y$ and $XT + Y_p$) include two compounds. The corresponding system of ordinary differential equations was as well integrated, assuming mass action kinetics for the reaction fluxes and with rate constants as shown in Fig. 3. The simulated concentration-time profiles from three different positive initial conditions, shown in Fig. 3 (inlay), were used as replicates.

Fig. 3. The mass-action model underlying the EnvZ-OmpR model in which ATP is the cofactor in phospho-OmpR dephosphorylation [18, 19].(A) The network includes 6 substrate complexes and 7 compounds. The values for the rate constants p_i, $1 \leq i \leq 9$ are given next to the reactions (see S33 Table). **(B)** The table includes the three initial conditions, used as replicates, for the concentration of the 7 compounds. **(C)** The levels of the compounds are simulated at time points $t_1 = 0.01$; $t_2 = 2$; $t_3 = 2.5$; $t_4 = 2.9$; $t_5 = 4$; $t_6 = 4.8$, $t_7 = 5$.

Application of the proposed approach yields to similar results in terms of the performance of the model quantified with respect to the residual sum of squares and the coefficient of determination. In the EnvZ-OmpR model, over the first three time points, the LASSO models had coefficients of determination of at least 0.8 (Fig. 4A and S11 Table). The good predictive power of the models in the first three time points was confirmed by the correspondingly small RSS (S22 Table), which tends to increase with time (Fig. 4B inlay).

For the same reasons as in the toy network I, we chose the constant values as follows: $c_1 = 2, c_2 = 3, c_3 = 2$ and $m = 9$. For these constants, we extracted 16 different substrate complexes, including: $1.49XTY_p$, $0.63X_p + X_pY + 0.36Y_p$, $Y, 1.36XT + 0.63Y_p, 0.77X_p$, $1.52X + 0.47X_p$, $X_pY + XTY_p, Y + Y_p, X + X_p$, XT, $0.94X + 1.05X_p$, $0.05X + XT$, $0.91X + 1.08X_p$, $0.08X + XT$, $0.87X + 1.12X_p$, $0.12X + XT$ (see S55 Table). The union of substrate complexes identified over all time points agrees with three substrate complexes with respect to the participation of compounds (marked in blue in S55 Table) and with one complex with respect to stoichiometry (marked in green in S55 Table). Altogether, two out of four single compound complexes were identified and one out of two double compound complexes was identified by the approach. However, the approach identified eleven complexes that are not part of the network (marked in red in S55 Table). Note that the small difference in the coefficients of some complexes is due to the smoothness condition imposed in the extraction of the time dependent statistical models. Altogether, two substrate complexes were not identified by the approach.

More stable solutions over time are expected to result from enforcing the stoichiometric coefficient to be positive integers, not only positive reals and control for the number of reactants (rather than the sum of their stoichiometric coefficients). However, this solution strategy would result in a mixed-integer linear program which

Fig. 4. Coefficient of determination (R^2) and residual sum of squares (RSS) for the EnvZ-OmpR model. The distribution of the two statistics, (A) R^2 and (B) RSS, over the 7 models is shown in the histogram. The inlays illustrate the minimum (Min), average (Mean), and maximum (Max) of the statistics over the models for all time points

undoubtedly imposes computational difficulties. Therefore, the time-resolved metabolic profiles of glycine HTR are analyzed with the proposed Algorithm 1.

2.3 Glycine hydrothermal reaction (HTR)

Here we present the results of applying the proposed approach to recently obtained data sets on the relative levels of chemical products from the glycine HTR at the temperature regime of 180°C [1]. To this end, 1% solution of glycine was used in the experiments under 100 bar using a high pressure continuous flow reactor. Altogether, the levels of $P = 21$ compounds were measured by hydrogen-1 nuclear magnetic resonance and gas chromatography mass spectrometry based profiling in 3 replicates at 9 time points at $180°C$, $t_1 = 0.4$, $t_2 = 0.6$, $t_3 = 1.19$, $t_4 = 1.79$, $t_5 = 2.56$, $t_6 = 3.58$, $t_7 = 5.12$, and $t_8 = 7.16$ minutes. Missing values were due to instrument sensitivity, and were substituted by random positive numbers selected from the interval $[1, 100]$ (with 100 being the detection limit). The compounds identified during this period include four compound classes, including: carboxylic acids, amino acids, amides, and cyclic derivatives (see Tables in S9, S10, S11 and S12 Tables).

By using Algorithm 1, for every time point we extracted 21 LASSO models, each with one of the 21 compounds as a response and the remaining 20 compounds as predictors. The coefficients of the LASSO regression were 3−fold cross validated to produce robust estimates at time point t_i. The transformation given in Eq. (23) was then used to compute the coefficients at time point t_{i+1}, $1 \leq i \leq 8$. In this case, the LASSO regression was 10-fold cross validated since instead of 6, we had $(P-1)+6 = 26$ dependent variables. The best value of κ was approximated by sampling so that the λ_{t+1} obtained from solving Eq. (24) is the closest to the one obtained from Eq. (19), independently of the minimization condition.

Fig. 5. Coefficient of determination (R^2) and residual sum of squares (RSS) for the glycine HTR at 180 °C. The distribution of the two statistics, (A) R^2 and (B) RSS, over the 21 models is shown in the histogram. The inlays illustrate the minimum (Min), average (Mean), and maximum (Max) R^2 of the statistics over the models for all time points

For the case of glycine HTR 180°C, more than 50% of the models over all time points the corresponding coefficients of determination (R^2) were at least 0.80 (S7 Table and Fig. 5A). In addition, as shown in the inlay of Fig.5A, the average values of R^2 for the first seven time points were in the range of 0.2 to 0.9, with average over time of $\approx$ 0.8. Interestingly, for the eight time point, $t_8 = 5.12$ minutes, none of the models exhibited an R^2 greater than 0.8, which was due to the particular behavior of the data profiles at this time point. Therefore, the models from this time point were not used in the prediction of substrate complexes (referred to as "ignored" time point in S12 Table).

Moreover, the RSS for the derived statistical models was in the range [0, 2] for all time points except the second, $t_2 = 0.6$ minutes, matched by the small decrease in the R^2 in comparison to the average over time. Altogether, the behavior of RSS pointed out that the sparse models were indeed of high predictive power. Therefore, the derived

sparse non-negative regression models could be used for reliable prediction of substrate complexes based on the mathematical program in Eq. (14).

The most frequently occurring compounds in the models with R^2 greater than 0.8, from the first two (early) time points, $t_1 = 0.4$ and $t_2 = 0.6$ minutes, included the carboxylic acid derivatives, *i.e.*, glyoxylic acid (with 12 occurrences), oxamic acid (11), *N*-glycyl-glycine (8), followed by the natural amino acids, *i.e.*, alanine (8), and the cyclic derivatives, *i.e.*, 2-5-dihydroxypyrazine (6). Glycine appeared as predictor in only 2 of the statistical models (Table in S9 Table). The large frequency of occurence for *N*-glycyl-glycine (8) could be explained by the fact that it is one of the main first products of glycine degradation, and, thus, serves as a proxy for glycine.

With the evolution of the glycine HTR, the *N*-carboxy-methylamine increased the occurrence from 12 models in the early time points to 15 and 24 models in the intermediary time points $t_3 - t_4$ and $t_5 - t_6$, respectively, while alanine also increased the occurence to 15 and 23 models. Interestingly, for the intermediary time points, glycine occurred as a predictor in altogether 6 models, and we observed a shift of the predictors towards natural amino acids and cyclic derivatives from the carboxylic acid derivatives, predominant in the early time points. This was representative for the last three (late) time points $t_7 - t_9$ (Table in S9 Table). Therefore, the interpretation of the statistical models indicated that the predictors capture the biochemically reasonable progression of the glycine degradation process, starting from glycine and carboxylic acids, in the early time points, to cyclic derivatives and natural amino acids, in the later stages of the reaction.

Selection of the number of reactions, m, in the case of the glycine HTR was based on the behavior of $\theta(m)$, as indicated in Section 1.3, for values of m in the range $[1, 21]$ (with $P = 21$ denotes the number of compounds). The upper bound for $\epsilon_j^{(r)}$ was set to 1 over all time points, which ensured feasibility of the linear program Eq. (14) for the used range of m. For each time point, m was selected following the "elbow" criterion.

An illustration of the choice of m for three time points, t_3, t_7 and t_9, is shown in Fig. 6: At time point $t_3 = 0.9$ minutes, the elbow in the curve of $\theta(m)$ appeared at $m = 5$, which was used to reconstruct five substrate complexes following Algorithm 1. However, for $t_7 = 3.58$ minutes, the number of reactions was identified to be 7 or 8, while for $t_9 = 7.16$ minutes, it was 3 or 5. In the latter two cases, we used the larger number of reactions in the reconstruction of the substrate complexes.

With the determined number of reactions $m = 5$ at t_3, we identified the following five substrate complexes: 0.95 C15 +0.42 C17, 0.97 C4 + 0.02 C15, 0.93 C17 + 0.06 C20, 0.03 C3 + 0.05 C5 + 0.91 C20, and C3, with C3, C4, C5, C15, C17 and C20 denoting alanine, *N*-carboxy-methylamine, glycine, 2,5-diketopiperazine, hydantoin, and 2,3,5-trihydroxy-3,6-dihydropyrazine, respectively (Table in S11 Table). At t_7, eight substrate complexes were determined: 0.03 C3 + 0.96 C16, C11, 0.02 C4 + 0.97 C5, C6, C2, 0.96 C3 + 0.03 C4, C10, C4, where C2, C6, C10, C11, and C16 denote glyoxylic acid, sarcosine, 2,5-dihydroxypyrazine, 3-methylpiperazine-2,5-dione, and *N*-carboxy-glycine, which are largely cyclic derivatives reacting together with one of the first derivatives of glycine degradation, *N*-carboxy-glycine. Finally, at t_9, we identified 5 substrate complexes, 0.096 C2 + 0.003 C3, 0.008 C3 + 0.04 C4 + 0.94 C13, 0.05 C3 + 0.94 C18, 0.94 C3 + 0.005 C5, C4 with C13 and C18 denoting glycine-*N*-methylamide and iminodiacetic acid.

The predicted substrate complexes at the early time points were next examined for their feasibility following basic chemical principles. We identified *N*-glycyl-glycine as one of the substrate complexes at time $t_2 = 0.6$ minutes, which could be explained by the chemical reaction leading reversibly to glycine, shown in in Fig. 7. The substrate complexes can be completed for the product side based on the mass balance principle and feasibility of the likely chemical reaction, which can be employed for the

Fig. 6. Time-dependent selection of the number of reactions m for the glycine HTR. Shown are the values for $\theta(m)$ at three different time points, t_3, t_7, and t_9, of the glycine HTR as the number of reactions, m, increases in the range [1,21]. The values are obtained by solving Eq. (14) at each of the considered time points. The circles on the lines mark the number of reactions which are used to reconstruct the substrate complexes following the "elbow" criterion.

interpretation of the remaining predictions.

Fig. 7. Substrate complex at time point $t_2 = 0.6$ minutes. The predicted substrate complexes containing N-glycyl-glycine is transformed through a reversible reaction into glycine.

3 Conclusion

We devised a novel computational approach for predicting the reaction substrates and their stoichiometries by assuming mass action kinetics for the reaction rates. The proposed optimization-based approach combines statistical techniques from regularized regression with mathematical programming and provides for inclusion of biochemical principles and constraints for accurate reconstruction of time-resolved biochemical networks. The approach can readily be extended to other kinetic laws of multiplicative form, and future attempts will be directed to completion of the reactions with the corresponding product complexes and inference of consensus networks over a given time domain. The analysis of the performance provides the insight that only some of the complexes can be accurately reconstructed in terms of composition and stoichiometry, while for others, the majority of the participating compounds can be identified. This is in line with the possibility that multiple network structures may give rise to the same dynamics studied in the context of unidentifiability [21].

References

1. Antonietti M, Meret M, Kopetzki N, Degenkolbe T, Kleessen S, Nikoloski Z., Tellstroem V, Barsch A, Kopka J, Willmitzer L. From System Biology to System Chemistry: Metabolomic procedures enable insights into complex chemical reaction networks in water. Royal Society of Chemistry Advances. 2014 Feb 06; 4:16777-16781. doi: 10.1039/C3RA42384K .

2. Arkin A, Shen P, Ross J. A Test Case of Correlation Metric Construction of a Reaction Pathway from Measurements. Science. 1997 Aug 29. Vol. 277 no. 5330 pp. 1275-1279. doi: 10.1126/science.277.5330.1275.

3. Basler G, Grimbs S, Ebenhöh O, Selbig J, Nikoloski Z. Evolutionary Significance of Metabolic Networks Properties. Journal of the Royal Society Interface. 12012 Jun 7;9(71):1168-76. doi: 10.1098/rsif.2011.0652.

4. Tunahan Cakir, Margriet M. W. B. Hendriks, Johan A. Westerhuis, Age K. Smilde. Metabolic network discovery through reverse engineering of metabolome data. Metabolomics. 2009 Feb 21; 5(3): 318–329. doi: 10.1007/s11306-009-0156-4.

5. Castellini A, Zucchelli M, Busato M, M, Vincenzo. From time series to biological network regulations: an evolutionary approach. Mol. BioSyst. 2013 Feb 2;9(2):225-33. doi: 10.1039/c2mb25191d.

6. Hempel S, Koseska A, Nikoloski Z, Kurths J. Unraveling Gene Regulatory Networks from Time-Resolved Gene Expression Data - Measures Comparison Study. BMC Bioinformatics. 2011 July 19 ;12:292. doi: 10.1186/1471-2105-12-292.

7. M. Hendrickx D, Margriet M. W. B. Hendriks, Paul H. C. Eilers, Age K. Smilde, Huub C. J. Hoefsloot. Reverse engineering of metabolic networks, a critical assessment. Molecular BioSystems. 2011 Oct 11; 7: 511-520. doi: 10.1039/C0MB00083C.

8. Jeong H, Tombor B, Albert R, Z. N. Oltvai, A. L. Barabási. The large-scale organization of Metabolic Networks. Nature. 2000 July 18; 407:651-654. doi: 10.1038/35036627.

9. Hiroaki K. Computational Systems Biology. Nature. 2002 Nov 14; 420:206-210. doi: 10.1038/nature01254.

10. Klamt S, Utz-Uwe Haus, Theis F. Hypergraphs and Cellular Networks. PLoS Computational Biology. 2009 May;5(5):e1000385. doi:10.1371/journal.pcbi.1000385.

11. Lykou A, Ntzoufras L. On Bayesian Lasso Variable Selection and the Specification of the Shrinkage Parameter. Statistics and Computing. 2013 May; 23(3):361-390. doi:10.1007/s11222-012-9316-x.

12. Lykou A, Ntzoufras L. On Bayesian Lasso Variable Selection and the Specification of the Shrinkage Parameter. Statistics and Computing. 2013 May; 23(3):361-390. doi:10.1007/s11222-012-9316-x.

13. Pan W, Yuan Y, Guy-Bart Stan. Reconstruction of Arbitrary Biochemical Reaction Networks: A Compressive Sensing Approach. IEEE conference on Decision and Control. 2012 May 15; 2334-2339. doi:10.1109/CDC.2012.6426216.

14. Howard J. Seltman. Experimental Design and Analysis. IEEE conference on Decision and Control. Carnegie Mellon University 2012.

15. Shlomi T, Moran N Cabili, Markus J Herrgård, ØPalsson B,Ruppin E. IEEE conference on Decision and Control. Nature Biotechnology. 2008 August 17; 26:1003-1010. doi:10.1038/nbt.1487.

16. Shulaev, V. Metabolomics technology and bioinformatics. Briefings in Bioinformatics. 2006 May 18; 7(2):128-139. doi: 10.1093/bib/bbl012.

17. Tibshirani R. Regression Shrinkage and Selection via the LASSO. Journal of the Royal Statistical Society: Series B. 1996; 58(1):267-288.

18. Pratt L, Silhavy TJ. Two-Component Signal Transduction, J. A. Hoch, T. J. Silhavy, Eds. (American Society for Microbiology, Washington, DC, 1995), 105–127.

19. Stock AM, Robinson VL, Goudreau VL. Two-component signal transduction. Annu. Rev. Biochem. 69, 183 (2000). doi:10.1146/annurev.biochem.69.1.183 pmid:10966457

20. Tibshirani R. The Lasso Problem and Uniqueness. Electronic Journal of Statistics. 2013; 7(0):1456-1490. doi:10.1214/13-ejs815.

21. Craciun G, Casian P. Identifiability of chemical reaction networks.Journal of Mathematical Chemistry. 2008; 44:244-259. DOI 10.1007/s10910-007-9307-x.

22. Verma M, Zakhartsev M, Reuss M, Hans V. Westerhoff. Domino Systems Biology and the A of ATP. Biochimica et Biophysica Acta (BBA) - Bioenergetics. 2013 January; 1827(1):19-29. doi:10.1016/j.bbabio.2012.09.014.

23. Proceedings of the 51th IEEE Conference on Decision and Control, CDC 2012, December 10-13, 2012, Maui, HI, USA.IEEE. 2012. url:http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6416474.

24. Goeman J, Meijer R, Chaturvedi M. L1 and L2 Penalized Regression Models. cran.r-project.org. 2012.

25. Hector Corrada Bravo, Stefan Theuss, Kurt Hornik. R interface to CPLEX. cran.r-project.org. 2013.

4 Supplementary Information

Supplementary Text

This section is devoted to providing some preliminaries of the least absolute shrinkage and selection operator (LASSO) and a computational approach for time-dependent networks.

LASSO

Let us consider the linear regression problem

$$R = \sum_{j=1}^{P} \beta_j S_j \tag{15}$$

where R and $(S_j)_{j=1}^{P}$ are the response variables and P predictor variables, respectively. Under the assumption that the predictors $(S_j)_{j=1}^{P}$ are standardized, *i.e.*, $\frac{\sum_j S_{ji}}{P} = 0$ and $\frac{\sum_j S_{ji}^2}{P} = 1$, the LASSO estimates of the coefficients β_j, are given by:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}\{\|R - \sum_{j=1}^{P} \beta_j S_j\|_2^2 \quad \text{subject to} \quad \|\beta\|_1 \leq \eta, \tag{16}$$

where $\|.\|_1$ and $\|.\|_2$ stand for the L_1 and L_2 norms, respectively [17]. The non-negative parameter η, referred as the tuning parameter, controls the amount of shrinkage imposed on the coefficients. If the shrinkage level is large enough, the coefficients of the predictors with weak effect on the response are forced to be zero. By shrinking some coefficients to zero, LASSO improves the prediction accuracy and simplifies the interpretation of the model, due to the reduced subset of predictors. There exists some methods designed to estimate the parameter η, for instance, cross validation and generalized cross-validation [17]. Note that it can be establish that one can always find a positive real λ such Eq. (17) below is equivalent to Eq. (16)

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}\{\|R - \sum_{j=1}^{P} \beta_j S_j\|_2^2 + \lambda\|\beta\|_1\}. \tag{17}$$

Time dependent LASSO

The models derived from metabolomics time-series relative levels based on the non-negatively constrained LASSO do not capture the dependence on time. The problem of inferring time-dependent model for a given compound can then be formulated as follows: Given a model at time point t for a compound k, we aim at obtaining the sparsest model for the compound at time point $t+1$ while fitting data and assuring smooth transition between models at the consecutive time points. This mimics the temporal activation of reactions, while minimizing the difference between the temporal networks. In other words, under the biochemically reasonable assumption that the estimation of the parameter β_t at time point t is given by the solution of the LASSO problem

$$\hat{\beta}_t = \underset{\beta_t}{\operatorname{argmin}}\{\|Y'_{k,t} - \sum_{j=1, j\neq k}^{P} \beta_{j,t} Y'_{j,t}\|_2^2 + \lambda_t\|\beta_t\|_1\}, \tag{18}$$

we aim at solving the following bi-level problem:

$$\min_{\beta_{t+1}} \|\beta_{t+1} - \beta_t\|_2^2,$$

$$\text{s.t} \qquad \hat{\beta}_{t+1} = \underset{\beta_{t+1}}{\operatorname{argmin}} \{\|Y'_{k,t+1} - \sum_{j=1, j\neq k}^{P} \beta_{j,t+1} Y'_{j,t+1}\|_2^2 + \lambda_{t+1}\|\beta_{t+1}\|_1\}, \qquad (19)$$

where $\beta_t = \left(\beta_{j,t}\right)_j$ and $\beta_{t+1} = \left(\beta_{j,t+1}\right)_j$ are the vectors of regression coefficients for the compounds participating in the network at time point t and $t+1$, respectively.

To solve the problem in Eq. (19), we first transform it into a uni-level problem. It is easy to establish, like in the case of LASSO, that one can always find a positive real κ such that Eq. (19) is equivalent to

$$\hat{\beta}_{t+1} = \underset{\beta_{t+1}}{\operatorname{argmin}} \{\|Y'_{k,t+1} - \sum_{j=1, j\neq k}^{P} \beta_{j,t+1} Y'_{j,t+1}\|_2^2 + \lambda_{t+1}\|\beta_{t+1}\|_1 + \kappa\|\beta_{t+1} - \beta_t\|_2^2\}, \quad (20)$$

If we set,

$$y = \begin{bmatrix} Y'_{t+1,k} \\ \mathbf{0}_{(P-1)\times 1} \end{bmatrix}, \eta = \begin{bmatrix} \beta_{t+1} \\ \sqrt{\kappa}(\beta_{t+1} - \beta_t) \end{bmatrix},$$
$$Y' = \begin{bmatrix} Y'_{t+1,k} & \mathbf{0}_{N\times(P-1)} \\ \mathbf{0}_{(P-1)\times(P-1)} & I_{(P-1)\times(P-1)} \end{bmatrix},$$

then Eq. (20) becomes

$$\hat{\beta}_{t+1} = \underset{\beta_{t+1}}{\operatorname{argmin}} \{\|y - \sum_{j=1, j\neq k}^{2P-1} \eta_j Y'^j\|_2^2 + \lambda_{t+1}\|\beta_{t+1}\|_1\}, \qquad (21)$$

where and Y'^j is the j^{th} column of Y'.

Since,

$$\begin{bmatrix} \beta_{t+1} \\ \sqrt{\kappa}(\beta_{t+1} - \beta_t) \end{bmatrix} \Longrightarrow \|\beta_{t+1}\|_1 = \frac{1}{1+\sqrt{\kappa}}\|\eta + \beta_{11}\|_1, \qquad (22)$$

where, $\beta_{11} = \begin{bmatrix} \mathbf{0}_{P\times P} \\ \sqrt{\kappa}\beta_t \end{bmatrix}$, substituting Eq.(22) in Eq. (21), leads to,

$$\hat{\beta}_{t+1} = \underset{\eta_1}{\operatorname{argmin}} \{\|y - \sum_{j=1, j\neq k}^{2P-1} \eta_j Y'^j\|_2^2 + \frac{\lambda_{t+1}}{1+\sqrt{\kappa}}\|\eta + \beta_{11}\|_1\}. \qquad (23)$$

Now, by setting $\beta = \eta + \beta_{11}$, we have

$$\hat{\beta}_{t+1} = \underset{\beta_{\text{New}}}{\operatorname{argmin}} \{\|Y - \sum_{j=1, j\neq k}^{2P-1} \beta_j Y'^j\|_2^2 + \lambda\|\beta\|_1\}, \qquad (24)$$

where $Y = y + \sum_{j=1, j\neq k}^{2P-1} \beta_{11}^j Y'^j$, $\lambda = \frac{\lambda_{t+1}}{1+\sqrt{\kappa}}$, and β_{New} is the vector made of the first $P-1$ components of β. Thus, solving the bi-level problem given by Eq. (19) is equivalent to solving the LASSO problem Eq. (24).

Therefore, knowing the estimated coefficients $\hat{\beta}_t$ of the model at time point t, one could compute the estimated coefficients $\hat{\beta}_{t+1}$ of the model at time point $t+1$ through the usage of Eq. (24). In other words, having the knowledge of the predictors in a network at time point t, one could obtain the knowledge of the predictors in the network at time point $t+1$, while assuring the smooth transition between both time points.

Supplementary Tables

S1 Table. Coefficient of determination (R^2) of the paradigmatic example over all model and time points. Model i ($1 \leq i \leq 6$) corresponds to the model where A to F is the response, respectively.

	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
Response compounds	A	B	C	D	E	F
$t_1 = 0.01$	0.9577919	0.9474565698	9.522925e-01	0.9596405	0.947168770	0.89399921
$t_2 = 2$	0.9577919	0.9474565698	9.522925e-01	0.9596405	0.947168770	0.89399921
$t_3 = 2.5$	0.0000000	0.0123936961	7.897752e-02	0.0000000	0.037647458	0.08693510
$t_4 = 2.9$	0.0000000	0.0141898563	4.889770e-02	0.0000000	0.018083754	0.05689915
$t_5 = 4$	0.0000000	0.0055721160	4.901351e-02	0.0000000	0.017161122	0.03215535
$t_6 = 4.98$	0.0000000	0.0005565881	1.086786e-02	0.0000000	0.001534894	0.01086786
$t_7 = 5$	0.0000000	0.0000000000	4.084991e-06	0.0000000	0.000000000	0.00000000

S2 Table. Residual sum of squares (RSS) of the paradigmatic example over all model and time points. Model i ($1 \leq i \leq 6$) corresponds to the model where A to F is the response, respectively.

	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
Response compounds	A	B	C	D	E	F
$t_1 = 0.01$	0.0194625	0.01042265	0.01206605	0.01587093	0.01800102	0.0720041
$t_2 = 2$	0.0194625	0.01042265	0.01206605	0.01587093	0.01800102	0.0720041
$t_3 = 2.5$	0.2279560	4.74296523	10.16708076	2.84767017	3.25010254	4.0010932
$t_4 = 2.9$	0.3168745	4.86503968	6.90768045	2.31531966	2.82261911	3.9671770
$t_5 = 4$	0.3653750	4.94860957	4.89496179	2.01615844	2.60165329	3.9638048
$t_6 = 4.98$	0.4315421	5.05450529	1.80376951	1.56981397	2.36767476	3.9295958
$t_7 = 5$	0.4507236	5.09090222	0.71171033	1.43441130	2.30214685	4.0174639

S11 Table. Coefficient of determination (R^2) of the EnvZ-OmpR model over all model and time points. Model i ($1 \leq i \leq 7$) corresponds to the model where X, XT, X_p, Y, X_pY, Y_p and XTY_p is the response, respectively.

	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7
Response compounds	X	XT	X_p	Y	X_pY	Y_p	XTY_p
$t_1 = 0.01$	0.956188520	0.9312082	0.9142930	0.8105377	0.95563382	0.9326498	0.9061196
$t_2 = 2$	0.956188520	0.9312082	0.9142930	0.8105377	0.95563382	0.9326498	0.9061196
$t_3 = 2.5$	0.085291957	0.0000000	0.7704946	0.8396404	0.01039598	0.0000000	0.0000000
$t_4 = 2.9$	0.004936784	0.0000000	0.7712560	0.7712560	0.00000000	0.0000000	0.0000000
$t_5 = 4$	0.000000000	0.0000000	0.8036430	0.8036430	0.00000000	0.0000000	0.0000000
$t_6 = 4.98$	0.000000000	0.0000000	0.7939760	0.7939760	0.00000000	0.0000000	0.0000000
$t_7 = 5$	0.000000000	0.0000000	0.7864949	0.7864949	0.00000000	0.0000000	0.0000000

S22 Table. Residual sum of squares (RSS) of the EnvZ-OmpR model over all model and time points. Model i ($1 \leq i \leq 7$) corresponds to the model where X, XT, X_p, Y, X_pY, Y_p and XTY_p is the response, respectively.

	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7
Response compounds	X	XT	X_p	Y	X_pY	Y_p	XTY_p
$t_1 = 0.01$	0.007116517	0.019768102	0.02855982	0.04859614	0.0160649	0.03110153	0.03110153
$t_2 = 2$	0.007116517	0.019768102	0.02855982	0.04859614	0.0160649	0.03110153	0.03110153
$t_3 = 2.5$	2.532225457	0.002379416	0.28467772	0.31065403	2.0924878	10.02062636	7.20009166
$t_4 = 2.9$	3.061562463	1.313885418	0.27503177	0.35204696	1.5334377	4.94214358	7.17393383
$t_5 = 4$	3.477026041	3.085744497	0.26919394	0.35688553	1.3473650	2.54399087	6.49494070
$t_6 = 4.98$	4.963143917	6.380422763	0.30322544	0.43065224	1.3550858	0.32757314	4.20037765
$t_7 = 5$	6.138949957	7.191653390	0.31536498	0.46198895	1.4654422	0.06011917	3.02068931

S3 Table. Values of the rate constants attributed to the reactions in the paradigmatic example.

Rate constants	Values
p_1	1
p_2	1
p_3	1
p_4	1
p_5	1
p_6	1
p_7	1
p_8	1
p_9	1
p_{10}	1

S33 Table. Values of the rate constants attributed to the reactions in the EnvZ-OmpR model.

Rate constants	Values
p_1	1
p_2	1
p_3	1
p_4	1
p_5	1
p_6	1
p_7	1
p_8	1
p_9	1

S4 Table. Initial conditions for the species involved in the paradigmatic example.

Initial conditions	A	B	C	D	E	F
IC 1	5	5	3	1	1	2
IC 2	1	3	4	5	1	1
IC 3	4	3	3	3	5	4

S44 Table. Initial conditions for the species involved in the EnvZ-OmpR model.

Initial conditions	X	XT	X_p	Y	X_pY	Y_p	XTY_p
IC 1	4	4	1	4	2	3	5
IC 2	3	3	5	4	5	4	4
IC 3	4	4	3	2	5	2	3

S5 Table. Identified substrate complexes in the paradigmatic example network at the different time points t_1 to t_7 for the rate constants p_1 to p_{10}.

Reactions	$t_1 = 0.01$	$t_2 = 2$	$t_3 = 2.5$	$t_4 = 2.9$	$t_5 = 4$	$t_6 = 4.98$	$t_7 = 5$
r_1	1.22F	0	0	0	0	0	0
r_2	0	0	0	0	0	0	0
r_3	D+E	D+E	D+E	D+E	D+E	D+E	D+E
r_4	1.23 C	0	0	0	0	0	0
r_5	0	0	0	0	0	0	0
r_6	0	0	0	0	0	0	C
r_7	1.64B + 0.35C	C + F	C + F	C + F	C + F	C + F	A + F
r_8	0	0	0	0	0	0	0
r_9	0	0	0	0	0	0	0
r_{10}	A	A + B	A + B	A + B	A + B	A + B	B

S55 Table. Identified substrate complexes in the EnvZ-OmpR network at the different time points t_1 to t_7 for the rate constants p_1 to p_9.

Reactions	$t_1 = 0.01$	$t_2 = 2$	$t_3 = 2.5$	$t_4 = 2.9$	$t_5 = 4$	$t_6 = 4.98$	$t_7 = 5$
r_1	$1.49XTY_p$	$X_pY + XTY_p$	$X_pY + XTY_p$	$X_pY + XTY_p$	$X_pY + XTY_p$	$X_pY + XTY_p$	$X_pY + XTY_p$
r_2	0	0	0	0	0	0	0
r_3	$0.63X_p + X_pY + 0.36Y_p$	$Y + Y_p$	$Y + Y_p$	$Y + Y_p$	$Y + Y_p$	$Y + Y_p$	$Y + Y_p$
r_4	Y	0	0	0	0	0	0
r_5	0	0	0	0	0	0	0
r_6	$1.36XT + 0.63Y_p$	$X + X_p$	$0.94X + 1.05X_p$	$0.94X + 1.05X_p$	$0.91X + 1.08X_p$	$0.87X + 1.12X_p$	$0.87X + 1.12X_p$
r_7	$0.77X_p$	0	0	0	0	0	0
r_8	0	0	0	0	0	0	0
r_9	$1.52X + 0.47X_p$	XT	$0.05X + XT$	$0.05X + XT$	$0.08X + XT$	$0.12X + XT$	$0.12X + XT$

S6 Table. 21 compounds measured at Glycine HTR at 180 °C

Compound abbreviations	Compound names
C_1	Glyoxylic acid
C_2	Glycolic acid
C_3	Alanine
C_4	N-Carboxy-methylamine (2TMS) OR [C2H5NO2]
C_5	Glycine
C_6	Sarcosine
C_7	Carbonic acid
C_8	Oxamic acid
C_9	Serine
C_{10}	2,5-Dihydroxypyrazine
C_{11}	3-Methylpiperazine-2,5-dione
C_{12}	N-Carboxy-alanine
C_{13}	Glycine-N-methylamide
C_{14}	Glycineamide
C_{15}	2,5-Diketopiperazine
C_{16}	N-Carboxy-glycine
C_{17}	Hydantoin
C_{18}	Iminodiacetic acid
C_{19}	2,3,5-Trihydroxypyrazine
C_{20}	2,3,5-Trihydroxy-3,6-dihydropyrazine
C_{21}	N-Glycyl-glycine

S7 Table. Coefficient of determination (R^2) from the Glycine HTR data at 180 °C over all model and time points. Model i ($1 \leq i \leq 21$) corresponds to the model where C_i is the response, respectively.

TPs (min)	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}	C_{17}	C_{18}	C_{19}	C_{20}	C_{21}
$t1 = 0.4$	0.89	0.55	0.54	0.87	0.92	0.88	0.93	0.97	0.88	0.89	0.71	0.76	0.90	0.87	0.88	0.92	0.74	0.81	0.85	0.47	0.76
$t2 = 0.6$	0.89	0.55	0.54	0.87	0.92	0.88	0.93	0.97	0.88	0.89	0.71	0.76	0.90	0.87	0.88	0.92	0.74	0.81	0.85	0.47	0.96
$t3 = 0.9$	0.26	0.39	0.83	0.84	0.50	0.78	0.60	0.05	0.77	0.79	0.61	0.47	0	0.04	0.82	0.28	0.88	0	0.39	0.83	0.76
$t4 = 1.19$	0	0.21	0.78	0.87	0.74	0.89	0.82	0.44	0.81	0.94	0.36	0.72	0.74	0.81	0.93	0.92	0.76	0.19	0.88	0.85	0.74
$t5 = 1.79$	0.65	0.19	0.90	0.80	0.79	0.94	0.83	0.28	0.48	0.94	0.70	0.95	0.81	0.83	0.95	0.89	0.88	0.47	0.94	0.69	0.80
$t6 = 2.56$	0.81	0.35	0.73	0.64	0.18	0.92	0.74	0.59	0.83	0.92	0.90	0.46	0.86	0.86	0.91	0.70	0.48	0.89	0.90	0.72	0.89
$t7 = 3.58$	0.87	0.85	0.87	0.55	0.90	0.90	0.75	0.20	0.78	0.53	0.88	0.53	0.75	0.71	0.68	0.88	0.75	0.71	0.78	0.79	0.66
$t8 = 5.12$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$t9 = 7.16$	0.45	0.81	0.59	0.44	0.66	0.57	0.68	0	0.05	0.057	0.46	0.70	0.83	0.76	0.01	0.68	0.28	0.85	0.50	0.69	0.52

S8 Table. Residual sum of squares (RSS) from the Glycine HTR data at 180 °C over all model and time points. Model i ($1 \leq i \leq 21$) corresponds to the model where C_i is the response, respectively.

TPs (min)	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}	C_{17}	C_{18}	C_{19}	C_{20}	C_{21}
$t1 = 0.4$	0.03	0.01	0.03	0.01	0.02	0.01	0.02	0.01	0.04	0.01	0.04	0.03	0.01	0.01	0.01	0.02	0.01	0.01	0.03	0.01	0.03
$t2 = 0.6$	0.03	0.01	0.03	0.01	0.02	0.01	0.02	0.01	0.04	0.01	0.04	0.03	0.01	0.01	0.01	0.02	0.01	0.01	0.03	0.01	0.03
$t3 = 0.9$	1.4	0.04	0.07	0.01	0.20	0.04	0.93	0.01	0.11	0.11	0.07	0.52	0.074	0.48	0.07	0.93	0.076	0.07	0.06	0.29	0.52
$t4 = 1.19$	0.13	0.14	0.14	0.01	0.03	0.01	0.09	0.01	0.23	0.01	0.19	0.14	0.02	0.02	0.01	0.03	0.23	0.02	0.02	0.03	0.01
$t5 = 1.79$	0.15	0.02	0.06	0.01	0.15	0.03	0.09	0.02	0.06	0.09	0.04	0.05	0.03	0.06	0.09	0.15	0.06	0.06	0.02	0.09	0.04
$t6 = 2.56$	0.45	0.01	0.31	0.49	0.02	0.012	0.31	0.01	0.31	0.007	0.04	0.02	0.10	0.02	0.01	0.31	0.31	0.015	0.01	0.02	0.06
$t7 = 3.58$	0.11	0.06	0.04	0.03	0.01	0.01	0.01	0.03	0.04	0.10	0.029	0.03	0.11	0.09	0.10	0.01	0.17	0.06	0.01	0.11	0.11
$t8 = 5.12$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$t9 = 7.16$	0.39	0.14	0.39	0.25	0.08	0.39	0.07	0.06	0.18	0.15	0.39	0.39	0.25	0.39	0.39	0.12	0.39	0.14	0.25	0.39	0.25

S9 Table. Occurrences of every compounds in models with coefficient of determination greater or equal to 0.8 at early, intermediary and late time points for Glycine HTR data at 180 °C

Compounds	Compound classes	Early TPs ($t_1 - t_2$)	Intermediary TPs ($t_3 - t_4$)	Intermediary TPs ($t_5 - t_6$)	Late TPs ($t_7 - t_9$)	All TPs
C_1	Carboxylic acid derivative	12	0	1	1	14
C_2	Carboxylic acid derivative	0	0	0	2	2
C_3	Natural amino acid	8	15	23	7	53
C_4	not categorized	12	15	24	10	61
C_5	not categorized	2	3	3	5	13
C_6	carboxylic acid derivative	2	1	2	1	6
C_7	carboxylic acid derivative	4	1	1	0	6
C_8	carboxylic acid derivative	11	0	0	0	11
C_9	Natural amino acid	2	1	1	0	4
C_{10}	cyclic derivative	6	1	2	0	9
C_{11}	cyclic derivative	0	0	1	1	2
C_{12}	carboxylic acid derivative	0	0	1	0	1
C_{13}	amide derivative	2	0	2	1	5
C_{14}	amide derivative	2	1	2	0	5
C_{15}	cyclic derivative	2	2	2	0	6
C_{16}	carboxylic acid derivative	2	1	1	1	5
C_{17}	cyclic derivative	2	1	1	0	4
C_{18}	carboxylic acid derivative	2	0	1	1	4
C_{19}	cyclic derivative	2	1	2	0	5
C_{20}	cyclic derivative	4	2	0	0	6
C_{21}	carboxylic acid derivative	8	0	2	0	10

S10 Table. Identified substrate complexes for the models obtained from Glycine HTR data at 180°C, with coefficient of determination (R^2) greater or equal to 0.8 for time point t_1 and t_2. The different values of m for each time point correspond to those where an elbow is detected on the plot of number of reactions versus $\theta(m)$. The cross mark (✗) in row i signifies the absence of a $i-$th reaction at a particular time point.

	Time point 1			Time point 2		
Reactions	$m = 2$	$m = 3$	$m = 14$	$m = 5$	$m = 15$	$m = 20$
r_1	$0.65C_8 + 0.28C_{10} + 0.05C_{21}$	$0.77C_8 + 0.22C_{21}$	C_{15}	C_{14}	C_9	$0.44C_7 + 0.55C_{12}$
r_2	$0.05C_{21}0.21C_1 + 0.78C_{21}$	$0.22C_{10} + 0.77C_{21}$	C_5	$0.29C_1 + 0.25C_7$	C_4	C_5
r_3	✗	$0.47C_{10} + 0.52C_1$	C_9	$0.44C_3 + 0.55C_4$	C_{15}	C_{16}
r_4	✗	✗	C_4	$0.15C_3 + 0.27C_{17}$	C_{13}	$0.55C_1 + 0.44C_{12}$
r_5	✗	✗	$0.51C_1 + 0.48C_9$	C_{16}	C_6	C_{20}
r_6	✗	✗	C_{21}	✗	$0.37C_5 + 0.62C_{19}$	C_{11}
r_7	✗	✗	$0.48C_6 + 0.51C_{19}$	✗	$0.37C1 + 0.25C_7 + 0.37C_{19}$	C_9
r_8	✗	✗	C_7	✗	C_8	C_{19}
r_9	✗	✗	$0.22C_1 + 0.77C_8$	✗	C_{10}	C_2
r_{10}	✗	✗	$0.51C_6 + 0.48C_{18}$	✗	$0.62C_5 + 0.37C_{18}$	C_6
r_{11}	✗	✗	C_{16}	✗	$0.05C_{18} + 0.94C_{21}$	C_3
r_{12}	✗	✗	C_{14}	✗	C_3	C_{15}
r_{13}	✗	✗	C_{13}	✗	C_{16}	C_{14}
r_{14}	✗	✗	C_{10}	✗	C_{14}	C_{21}
r_{15}	✗	✗	✗	✗	$0.32C17 + 0.11C_{18} + 0.55C_{20}$	C_{18}
r_{16}	✗	✗	✗	✗	✗	C_4
r_{17}	✗	✗	✗	✗	✗	C_8
r_{17}	✗	✗	✗	✗		C_{13}
r_{19}	✗	✗	✗	✗	✗	C_{17}
r_{20}	✗	✗	✗	✗	✗	C_{10}

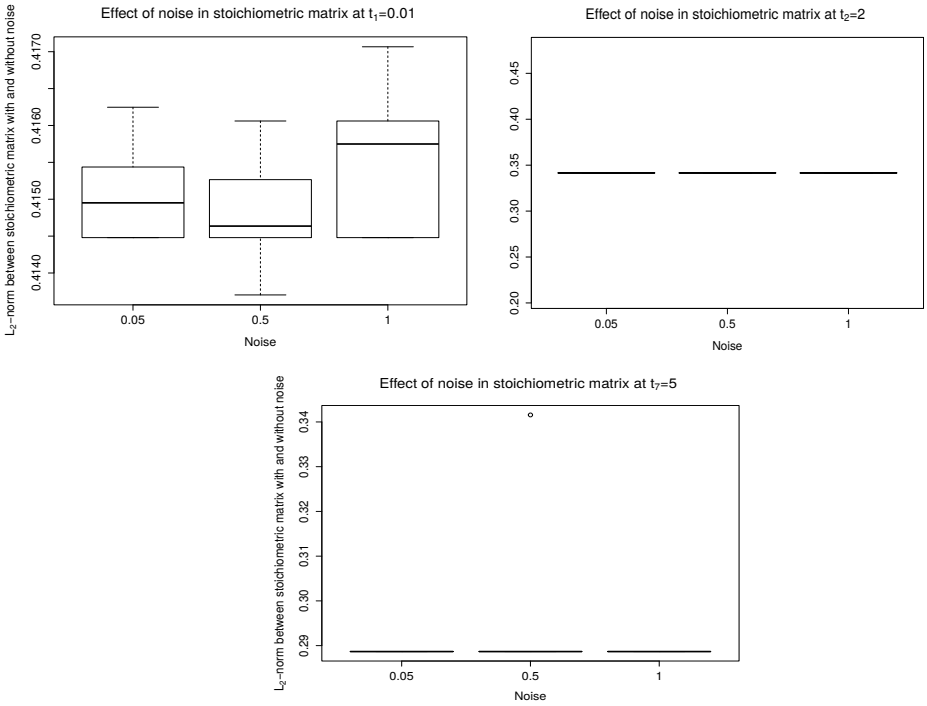
S11 Table. Identified substrate complexes for the models obtained from Glycine HTR data at 180°C, with coefficient of determination (R^2) greater or equal to 0.8 for time point t_3, t_4, t_5 and t_6. The different values of m for each time point correspond to those where an elbow is detected on the plot of number of reactions versus $\theta(m)$. The cross mark (✗) in row i signifies the absence of a $i-$th reaction at a particular time point.

	Time point 3	Time point 4	Time point 5	Time point 6	
Reactions	$m = 5$	$m = 11$	$m = 13$	$m = 10$	$m = 11$
r_1	$0.95C_{15} + 0.42C_{17}$	$0.09C_7 + 0.99C_{16}$	$0.99C_7 + 0.008C_{19}$	$0.002C_5 + 0.2C_{11}$	$0.01C_5 + 0.03C_{10}$
r_2	$0.97C_4 + 0.02C_{15}$	$0.02C_{10} + 0.97C_{19}$	$0.99C_{10} + 0.003C_{13} + 0.005C_{21}$	$0.31C_9 + 0.68C_{11}$	$0.05C_3 + 0.94C_{18}$
r_3	$0.93C_{17} + 0.06C_{20}$	$0.99C_4 + 0.009C_6$	$0.021C_5 + 0.97C_6$	$0.51C_6 + 0.48C_9$	$0.05C3 + 0.0002C_5 + 0.94C_{14}$
r_4	$0.03C3 + 0.05C_5 + 0.91C_{20}$	$0.04C_9 + 0.95C_{14}$	$0.98C_4 + 0.019C_{15}$	$0.1C_9 + 0.89C_{19}$	$0.05C_1 + 0.91C_9$
r_5	C_3	$0.009C_3 + 0.99C_{20}$	$0.01C_6 + 0.98C_{21}$	$0.72C_3 + 0.27C_6$	$0.05C_3 + 0.94C_{13}$
r_6	✗	$0.96C_{10} + 0.03C_{14}$	$0.008C_{12} + 0.99C_{17}$	$0.1C_3 + 0.89C_{21}$	$0.84C_3 + 0.15C_4$
r_7	✗	$0.058C_6 + 0.94C_9$	$0.008C_{12} + 0.99C_{16}$	$0.04C_1 + 0.007C_3$	$0.05C_4 + 0.94C_{21}$
r_8	✗	$0.98C_7 + 0.01C_{19}$	$0.008C_5 + 0.99C_{14}$	$0.89C_{10} + 0.1C_{15}$	$0.13C_1 + 0.86C_{15}$
r_9	✗	C_3	$0.02C_5 + 0.97C_{12}$	$0.1C_3 + 0.89C_{14}$	$0.05C_1 + 0.94C_6$
r_{10}	✗	$0.009C_6 + 0.99C_{15}$	$0.01C_4 + 0.98C_{13}$	$0.1C_6 + 0.89C_{13}$	$0.05C_4 + 0.94C_{19}$
r_{11}	✗	$0.03C3 + 0.05C_5 + 0.91C_6$	$0.02C_3 + 0.97C_{15}$	✗	$0.91C_{10} + 0.08C_{15}$
r_{12}	✗	✗	$0.016C_3 + 0.98C_{19}$	✗	✗
r_{13}	✗	✗	C_3	✗	✗

S12 Table. Identified substrate complexes for the models obtained from Glycine HTR data at 180°C, with coefficient of determination (R^2) greater or equal to 0.8 for time point t_7 and t_9.The different values of m for each time point correspond to those where an elbow is detected on the plot of number of reactions versus $\theta(m)$. The time point-Ignored (t_8) is the one that was not giving any robust ($R^2 \geq 0.8$) statistical information. The compounds ($C_i, 1 \leq i \leq 21$) can be seen in Table. The cross mark (✗) in row i signifies the absence of a $i-$th reaction at a particular time point.

	Time point 8 (ignored)	Time point 7		Time point 9	
Reactions	✗	$m = 7$	$m = 8$	$m = 3$	$m = 5$
r_1	✗	$0.26C_3 + 0.73C_{16}$	$0.03C_3 + 0.96C_{16}$	$0.002C_5 + 0.049C_{13} + 0.94C_{18}$	$0.096C_2 + 0.003C_3$
r_2	✗	$0.15C_4 + 0.84C_5$	C_{11}	$0.049 + C_2 + 0.051C_3 + 0.89C_{13}$	$0.008C_3 + 0.04C_4 + 0.94C_{13}$
r_3	✗	$0.78C_4 + 0.21C_6$	$0.02C_4 + 0.97C_5$	$0.94C_3 + 0.05C_4$	$0.05C_3 + 0.94C_{18}$
r_4	✗	$0.15C_1 + 0.84C_{11}$	C_6	✗	$0.94C_3 + 0.005C_5$
r_5	✗	$0.62C_3 + 0.37C_1$	C_2	✗	C_4
r_6	✗	$0.36C_1 + 0.63C_6$	$0.96C_3 + 0.03C_4$	✗	✗
r_7	✗	$0.89C_2 + 0.1C_{16}$	C_{10}	✗	✗
r_8	✗	✗	C_4	✗	✗

Supplementary Figures



S1 Fig. Effect of the time profile errors in the stoichiometric matrix at time points $t_1 = 0.01, t_2 = 2$ and $t_7 = 5$ for the toy example 1. The time profiles were perturbed with some random values following a normal distribution of zero mean and 0.05, 0.5 and 1 variances. For many instances of each noise, the euclidean distance between the stoichiometric matrices with and without noise was computed.

Codes snippets

Determination of beta coefficients

```
Betas=function(Met, Time ){
  # The parameter "Met" corresponds to the N times P matrix of
   compounds concentrations at different time points given by
  the entries of the vector "Time".
  # N=S*x, where x corresponds to the number of initial conditions
   (replicates) for which the profiles where computed.
    S is the number of time points.
  # P is the number of compounds.

  N=dim(Met)[1]
  P=dim(Met)[2]
  N11=3
  S=length(Time)

  # a- Replace the "0"s by a very small value
  for (i in 1:N){
    for (j in 1:P){
      if (Met[i,j]==0){
        Met[i,j]=runif(1,0,0.1)
      }
    }
  }

  NMet=matrix(0, 2*N,P)
  NNMet1=matrix(0, 2*(N11),P)
  NM1=Met[((N-2):N),]
  nMet1=matrix(0,2,P)
  a=1
  x=c(1:3)[-a]
  while (a<4){
    for (i in 1:2){
      nMet1[i,]=log(NM1[a,]/NM1[x[i],])
    }
    NNMet1[(((a-1)*2+1):(2*a)),]=nMet1
    a=a+1
    x=c(1:3)[-a]

  }
  NMet[((2*N-5):(2*N)),]=NNMet1

  NNMet=matrix(0, 2*(N11),P)

  NM=Met[1:N11,]
  a1=1
  while (a1<(S)) {

    b=1
    x=c(1:3)[-b]
    nMet=matrix(0, 2,P)
```

```
  while (b<4){
    for (i in 1:2){
      nMet[i,]=log(NM[b,]/NM[x[i],])
    }
    NNMet[(((b-1)*2+1):(2*b)),]=nMet
    b=b+1
    x=c(1:3)[-b]

  }
  NMet[(((a1-1)*6+1):(6*a1)),(1:P)]=NNMet
  NM=Met[(((a1)*3+1):(3*(a1+1))),]
  a1=a1+1

}

# 2- Let us standardize the obtained data

AnalMet=scale(NMet, center=TRUE, scale=TRUE)

# 3-Regression Analysis

N1=2*N11
x0=AnalMet[1:N1, ]
x1=AnalMet[(N1+1):(2*N1), ]
CD=c()
Save0=matrix(0,P-1,P)
Save1=matrix(0,P-1,P)
AllTP=matrix(0,S,P-1)
AllMod=matrix(0, P-1, P*S)
CoefDet=matrix(0,S,P)
Resid= c()
RSS=matrix(0,S,P)

# a-Now the compuation of coefficients at t0
library("chemometrics")
library("penalized")
library("survival")
for (c in 1:P){
  ny0=(-1)*x0[,c]
  nx0=x0[,-c]
  ny1=(-1)*x1[,c]
  nx1=x1[,-c]
  time0=data.frame(ny0,nx0)
  time1=data.frame(ny1,nx1)
  CV0=lassoCV(ny0~nx0, data=time0, K=3,
  fraction = seq(0.1, 0.5, by = 0.1))
  dev.off()
  Pen=penalized(ny0, nx0, lambda1=CV0$sopt , lambda2=0, positive=TRUE,
  data=time0, model="linear")
  Coef=coefficients(Pen,"penalized")
  Save0[,c]=Coef
  Rsq=var(nx0%*%Coef)/var(ny0)
```

```
CD[[length(CD)+1]]=Rsq
Resid[[length(Resid)+1]]=sum((ny0-(nx0%*%Coef))**2)

# b-At time point 2(t1)
# Compute the new variables

Kappa=c(6,10,2,7,11,4,19,13,8,20,14,9,18,3,17,5,12,16,1,15)

y11 = matrix(0, N1+P-1,1)
X=matrix(0,(N1+P-1),2*(P-1))
KapLam=matrix(0, length(Kappa), P+1)
for (i in 1:N1){
  y11[i]=ny1[i]
}

for (i in 1:N1){
  for (j in 1:(P-1)){
    X[i,j]=nx1[i,j]
  }
}
for (i in 1:(P-1)){
  X[N1+i,P-1+i]=1
}
for (i in 1:(length(Kappa))){

  # c-Enter the new Beta11, then the new data frame

  Beta00=matrix(0, (2*(P-1)), 1)
  for (j in P:(2*(P-1))){
    Beta00[j]=sqrt(Kappa[i])*(Coef[j-P+1])
  }
  Y=y11+X%*%Beta00
  ntime=data.frame(Y,X)
  CV1=lassoCV(Y~X,data=ntime, K=10,
  fraction = seq(0.1, 0.5, by = 0.1))
  Pen1=penalized(Y, X,lambda1=CV1$sopt, lambda2=0,
  positive=TRUE, data=ntime, model="linear")
  Coef1=coefficients(Pen1,"penalized")
  dev.off()
  for (l in 3:(P+1)){
    KapLam[i,1]=(CV1$sopt)*(1+sqrt(Kappa[i]))
    KapLam[i,2]=Kappa[i]
    KapLam[i,l]=Coef1[l-2]
  }
}
CV11=lassoCV(ny1~nx1, data=time1, K=3,
fraction = seq(0.1, 0.5, by = 0.1))
dev.off()
L=numeric(length(Kappa))
for (n in 1:length(Kappa)){
  L[n]=abs(CV11$sopt - KapLam[n,2])
}
```

```
  ind1=which.min(L)
  Save1[,c]=KapLam[ind1,][3:(P+1)]
}
AllMod[(1:(P-1)),(1:P)]=Save0
AllMod[(1:(P-1)),((P+1):(2*P))]=Save1

CoefDet[1,]=CD
CoefDet[2,]=CD
RSS[1,]=Resid
RSS[2,]=Resid
ind2=which.max(CD)
AllTP[1,]=Save0[,ind2]
AllTP[2,]=Save1[,ind2]
NM=P
SCD=sort(CD)
NMHCD=SCD[(length(SCD)-NM+1):(length(SCD))]
MHCD0=matrix(0, P+1, NM)
MHCD1=matrix(0, P+1, NM)
ALLTPMHCD=matrix(0, P+1,NM*(length(Time)) )
for (i in 1:length(NMHCD)){
  s=which(CD==(NMHCD[i]))
  MHCD0[1,i]=sample(s,1)
  MHCD1[1,i]=sample(s,1)
  MHCD1[2,i]=NMHCD[i]
  MHCD0[2,i]=NMHCD[i]
  MHCD0[(3:(P+1)),(1:(NM))][,i]=Save0[,sample(s,1)]
  MHCD1[(3:(P+1)),(1:(NM))][,i]=Save1[,sample(s,1)]
}
ALLTPMHCD[(1:(P+1)),(1:NM)]=MHCD0
ALLTPMHCD[(1:(P+1)),((NM+1):(2*NM))]=MHCD1

# Now, let us compute the coefficients for S-2 remaining time points

Incr=2
RespMet=c(ind2,ind2)
x2=AnalMet[(Incr*N1+1):((Incr +1)*N1),]
while (Incr<(S-1)){
  nCD=c()
  nResid=c()
  nSave1=matrix(0,P-1,P)
  nSave2=matrix(0,P-1,P)
  for (c in 1:P){
    nny1=(-1)*x1[,c]
    nnx1=x1[,-c]
    ny2=(-1)*x2[,c]
    nx2=x2[,-c]
    time11=data.frame(nny1,nnx1)
    time2=data.frame(ny2,nx2)
    nCV1=lassoCV(nny1~nnx1, data=time11, K=3,
    fraction = seq(0.1, 0.5, by = 0.1))
    dev.off()
    nPen1=penalized(nny1, nnx1, lambda1=nCV1$sopt, lambda2=0,
```

```
positive=TRUE, data=time11, model="linear")
nCoef1=coefficients(nPen1,"penalized")

nRsq=var(nnx1%*%nCoef1)/var(nny1)
nCD[[length(nCD)+1]]=nRsq
nResid[[length(nResid)+1]]=sum((nny1-(nnx1%*%nCoef1))**2)

# At the next time point
# Compute the new variables


y22 = matrix(0, N1+P-1,1)
nX=matrix(0,(N1+P-1),2*(P-1))
nKapLam=matrix(0, length(Kappa), P+1)

for (i in 1:N1){
  y22[i]=ny2[i]
}

for (i in 1:N1){
  for (j in 1:(P-1)){
    nX[i,j]=nx2[i,j]
  }
}
for (i in 1:(P-1)){
  nX[N1+i,P-1+i]=1
}

for (i in 1:(length(Kappa))){

  #Enter the new Beta11, then the new data frame

  Beta11=matrix(0, 2*(P-1), 1)
  for (j in P:(2*(P-1))){
    Beta11[j]=sqrt(Kappa[i])*(nCoef1[j-P+1])
  }
  nY=y22+nX%*%Beta11
  nntime=data.frame(nY,nX)

  CV2=lassoCV(nY~nX,data=nntime, K=10,
  fraction = seq(0.1, 0.5, by = 0.1))
  dev.off()

  nPen2=penalized(nY, nX, lambda1=CV2$sopt, lambda2=0,
  positive=TRUE, data=nntime, model="linear")
  nCoef2=coefficients(nPen2,"penalized")


  for (l in 3:(P+1)){
    nKapLam[i,1]=(CV2$sopt)*(1+sqrt(Kappa[i]))
    nKapLam[i,2]=Kappa[i]
    nKapLam[i,l]=nCoef2[l-2]
```

```
    }
  }


  CV22=lassoCV(ny2~nx2, data=time2, K=3,
  fraction = seq(0.1, 0.5, by = 0.1))
  dev.off()
  nL=numeric(length(Kappa))
  for (n in 1:length(Kappa)){
    nL[n]=abs(CV22$sopt - nKapLam[n,2])
  }
  ind11=which.min(nL)
  nSave2[,c]=nKapLam[ind11,][3:(P+1)]


}
AllMod[(1:(P-1)),(Incr*P+1):((Incr+1)*P)]=nSave2
nSCD=sort(nCD)
nNMHCD=nSCD[(length(nSCD)-NM+1):(length(nSCD))]
nMHCD=matrix(0, P+1, NM)
for (i in 1:(NM)){
  ns=which(nCD==(nNMHCD[i]))
  nMHCD[1,i]=ns[1]
  nMHCD[2,i]=nNMHCD[i]
  nMHCD[(3:(P+1)),(1:(NM))][,i]=nSave2[,ns[1]]
}
ALLTPMHCD[(1:(P+1)),(((Incr)*NM+1):((Incr+1)*NM))]=nMHCD

CoefDet[Incr+1,]=nCD
RSS[Incr+1,]=nResid
ind3=which.max(nCD)
RespMet[length(RespMet)+1]=ind3
AllTP[Incr+1,]=nSave2[,ind3]
x1=x2
Incr=Incr+1
x2=AnalMet[((Incr*N1+1):((Incr+1)*N1)),]
}
x8=AnalMet[((N1*(S-2)+1):(N1*(S-1))), ]
x9=AnalMet[((N1*(S-1)+1):(N1*(S))), ]
NCD=c()
NResid=c()
nSave3=matrix(0, P-1, P)

for (c in 1:P){
  nny1=(-1)*x8[,c]
  nnx1=x8[,-c]
  ny2=(-1)*x9[,c]
  nx2=x9[,-c]
  time11=data.frame(nny1,nnx1)
  time2=data.frame(ny2,nx2)
  nCV1=lassoCV(nny1~nnx1, data=time11, K=3,
  fraction = seq(0.1, 0.5, by = 0.1))
```

```
dev.off()

nPen1=penalized(nny1, nnx1, lambda1=nCV1$sopt, lambda2=0,
positive=TRUE, data=time11, model="linear")
nCoef1=coefficients(nPen1,"penalized")

nRsq=var(nnx1%*%nCoef1)/var(nny1)
NCD[[length(NCD)+1]]=nRsq
NResid[[length(NResid)+1]]=sum((nny1-(nnx1%*%nCoef1))**2)

# At the next time point
# Compute the new variables


y22 = matrix(0, N1+P-1,1)
nX=matrix(0,(N1+P-1),2*(P-1))
nnKapLam=matrix(0, length(Kappa), P+1)

for (i in 1:N1){
  y22[i]=ny2[i]
}

for (i in 1:N1){
  for (j in 1:(P-1)){
    nX[i,j]=nx2[i,j]
  }
}
for (i in 1:(P-1)){
  nX[N1+i,P-1+i]=1
}

for (i in 1:(length(Kappa))){

  #Enter the new Beta11, then the new data frame

  Beta11=matrix(0, 2*(P-1), 1)
  for (j in P:(2*(P-1))){
    Beta11[j]=sqrt(Kappa[i])*(nCoef1[j-P+1])
  }
  nY=y22+nX%*%Beta11
  nntime=data.frame(nY,nX)

  CV2=lassoCV(nY~nX,data=nntime, K=10,
  fraction = seq(0.1, 0.5, by = 0.1))
  dev.off()
  nPen2=penalized(nY, nX,lambda1=CV2$sopt, lambda2=0,
  positive=TRUE, data=nntime, model="linear")
  nCoef2=coefficients(nPen2,"penalized")


  for (l in 3:(P+1)){
    nnKapLam[i,1]=(CV2$sopt)*(1+sqrt(Kappa[i]))
```

```
      nnKapLam[i,2]=Kappa[i]
      nnKapLam[i,l]=nCoef2[l-2]
    }
  }


  CV22=lassoCV(ny2~nx2, data=time2, K=3,
  fraction = seq(0.1, 0.5, by = 0.1))
  dev.off()
  nL=numeric(length(Kappa))
  for (n in 1:length(Kappa)){
    nL[n]=abs(CV22$sopt - nnKapLam[n,2])
  }
  ind11=which.min(nL)
  nSave3[,c]=nnKapLam[ind11,][3:(P+1)]


}
AllMod[(1:(P-1)),((S-1)*P+1):(S*P)]=nSave3
nSCD1=sort(NCD)
nNMHCD1=nSCD1[(length(nSCD1)-NM+1):(length(nSCD1))]
nMHCD1=matrix(0, P+1, NM)
for (i in 1:(NM)){
  ns=which(NCD==(nNMHCD1[i]))
  nMHCD1[1,i]=ns[1]
  nMHCD1[2,i]=nNMHCD1[i]
  nMHCD1[(3:(P+1)),(1:(NM))][,i]=nSave3[,ns[1]]
}
ALLTPMHCD[(1:(P+1)),(((S-1)*NM+1):(S*NM))]=nMHCD1

CoefDet[S,]=NCD
RSS[S,]=NResid
ind3=which.max(NCD)
RespMet[length(RespMet)+1]=ind3
AllTP[S,]=nSave3[,ind3]
Results=list(Mat1=AllTP, Mat2=AllMod, )

# "AllTP" is a S times (P-1) matrix of the best performing
 (highest coefficient of determination) beta coefficients
 at each time points.
 Note that each compounds is consider as a response for each model.

# "AllMod" is a (P-1) times S*P matrix  of all models
(beta coefficients) at all time points.
return(Results)
}
```

Linear program for finding the stoichiometric coefficients

```
Stoichcoeff=function(Met, Time, NewAllMod){
  # The parameter "Met" corresponds to the N times P matrix
    of compounds concentrations at different time points
    given by the entries of the vector "Time".
```

```
# n=S*x, where x corresponds to the number of initials conditions
 (replicates) for which the profiles where computed.
  S is the number of time points.
# P is the number of compounds.
# The parameter "NewAllMod" is the P times P*S matrix
  of all beta coefficients (including the responses
  with coefficients of 1) at all time points.
# m is the number of reactions
# c1 and c2 are as in the main paper
# Incr is the corresponding time point.
 To be incremented for each time point.

n=dim(Met)[2]
S=length(Time)

# III. Computation of the coefficients

Coeff=matrix(0, m,n)

RespMet=c(1:n)
FM=NewAllMod[(1:n),(((Incr-1)*n+1):(Incr*n))]
Num_NZ_PerMod=c()
ALL_NZ=c()
NZ=c()

# a. We identify the number of compounds (non-zero betas)
  per model for the supposed time point

for (i in 1:n){
  S1=which(FM[,i]!=0)
  S2=length(S1)
  NZ=append(NZ,S1)
  ALL_NZ=unique(sort(NZ))
  Num_NZ_PerMod[[length(Num_NZ_PerMod)+1]]=S2

}
LL=length(ALL_NZ)
Q=sum(Num_NZ_PerMod)

# b. We enter the constrained matrix

Amat=matrix(0, 2*m+2*LL+2*Q, m*LL+Q)
A=matrix(0, m,m*LL)
B=matrix(0,LL, m*LL)
for (i in 1:LL){
  A[(1:m),(((i-1)*m +1):(i*m))]=diag(m)
}
Amat[(1:m),(1:(m*LL))]=A
Amat[(m+1):(2*m),(1:(m*LL))]=-A
for (j in 1:LL){
  B[j,][((j-1)*m+1):(j*m)]=rep(1,m)
```

```
}
Amat[(2*m+1):(2*m+LL),(1:(m*LL))]=B
Amat[(2*m+LL+1):(2*m+2*LL),(1:(m*LL))]=-B
Mat=matrix(0, 2*Q, m*LL+Q)

s=1
T2=FM[,s]
a1=RespMet[s]
v1=which(T2!=0)
L=length(v1)
Ind=c()
for (k in 1:L){
  Ind=append(Ind, which(ALL_NZ==v1[k]))
}
e1=which(ALL_NZ==a1)
AA=matrix(0, 2*L, m*LL+Q)
D=matrix(0, L, m*LL+Q)
for (t in 1:L){
  if (v1[t]!=a1){
    D[t,][((Ind[t]-1)*m+1):(Ind[t]*m)]=rep(1,m)
    D[t,][((e1[1]-1)*m+1):(e1[1]*m)]=rep(-T2[v1[t]],m)
  }
  else {
    D[t,][((e1[1]-1)*m+1):(e1[1]*m)]=rep((1-T2[v1[t]]),m)
  }

}
D[(1:L),((m*LL+1):(m*LL+L))]=-diag(L)
AA[(1:L), (1:(m*LL+Q))]=D
AA[((L+1):(2*L)), (1:(m*LL+Q))]=-D
Mat[(1:(2*L)),(1:(m*LL+Q))]=AA




for (j in 2:n){
  T2=FM[,j]
  a1=RespMet[j]
  v1=which(T2!=0)
  L=length(v1)
  Ind=c()
  for (k in 1:L){
    Ind=append(Ind, which(ALL_NZ==v1[k]))
  }
  e1=which(ALL_NZ==a1)
  AA=matrix(0, 2*L, m*LL+Q)
  D=matrix(0, L, m*LL+Q)
  for (t in 1:L){
    if (v1[t]!=a1){
      D[t,][((Ind[t]-1)*m+1):(Ind[t]*m)]=rep(1,m)
      D[t,][((e1[1]-1)*m+1):(e1[1]*m)]=rep(-T2[v1[t]],m)
    }
```

```
      else {
        D[t,][((e1[1]-1)*m+1):(e1[1]*m)]=rep((1-T2[v1[t]]),m)
      }

    }
    x=sum(Num_NZ_PerMod[(1:(j-1))])
    D[(1:L),((m*LL+x+1):(m*LL+x+L))]=-diag(L)
    AA[(1:L), (1:(m*LL+Q))]=D
    AA[((L+1):(2*L)), (1:(m*LL+Q))]=-D
    Mat[((2*x+1):(2*x+2*L)),(1:(m*LL+Q))]=AA

  }
  Amat[(2*m+2*LL+1):(2*m+2*LL+2*Q), (1:(m*LL+Q))]=Mat
  bvec=c(rep(c1,m), rep(0,m), rep(c2,LL), rep(-1,LL), rep(0, 2*Q))

  cvec=c(rep(1, m*LL),rep(1,Q))

  lb=c(rep(0,m*LL), rep(0,Q))
  ub=rep(2,m*LL+Q)

  Amat1=rbind(-Amat, diag(dim(Amat)[2]), -diag(dim(Amat)[2]))
  bvec1=c(-bvec, lb, -ub)

  library("Rcplex")
  Sol1=Rcplex(cvec,Amat,bvec, Qmat=NULL,lb,ub,objsense="min",sense="L")

  #c. The solution
  Coeff=matrix(0,m, n)
  for (t in 1:LL){
    Coeff[,ALL_NZ[t]]=Sol1$xopt[((t-1)*m+1):(t*m)]
  }

  # "Coeff" is m times P matrix returning the stoichiometric
    coefficient of each compound in every reaction
  return(Coeff)


}
```