BMC Systems Biology

Research article

Parameter optimization in S-system models

Marco Vilela*[1,5], I-Chun Chou[2], Susana Vinga[3], Ana Tereza R Vasconcelos[4], Eberhard O Voit[2] and Jonas S Almeida[1,5]

Address: [1]Dept. Bioinformatics and Computational Biology, University of Texas M.D. Anderson Cancer Center, 1515 Holcombe Blvd, Houston, TX 77030, USA, [2]Dept. Biomedical Engineering, Georgia Institute of Technology and Emory University, 313 Ferst Drive, Atlanta, GA 30332, USA, [3]Instituto de Engenharia de Sistemas e Computadores: Investigação e Desenvolvimento (INESC-ID), R. Alves Redol 9, 1000-029 Lisboa, Portugal, [4]Dept. Computatinal and Applied Mathematics, Laboratório Nacional de Computação Científica, Petrópolis, Rio de Janeiro, Brazil and [5]Instituto de Tecnologia Química e Biológica, Universidade Nova de Lisboa, Rua da Quinta Grande 6, Apartado 127, 2780-156 Oeiras, Portugal

Email: Marco Vilela* - mvilela@mdanderson.org; I-Chun Chou - bigjump@gatech.edu; Susana Vinga - svinga@kdbio.inesc-id.pt; Ana Tereza R Vasconcelos - atrv@lncc.br; Eberhard O Voit - eberhard.voit@bme.gatech.edu; Jonas S Almeida - jalmeida@mdanderson.org

* Corresponding author

Published: 16 April 2008

BMC Systems Biology 2008, **2**:35 doi:10.1186/1752-0509-2-35

Received: 14 January 2008
Accepted: 16 April 2008

This article is available from: http://www.biomedcentral.com/1752-0509/2/35

Abstract

Background: The inverse problem of identifying the topology of biological networks from their time series responses is a cornerstone challenge in systems biology. We tackle this challenge here through the parameterization of S-system models. It was previously shown that parameter identification can be performed as an optimization based on the decoupling of the differential S-system equations, which results in a set of algebraic equations.

Results: A novel parameterization solution is proposed for the identification of S-system models from time series when no information about the network topology is known. The method is based on eigenvector optimization of a matrix formed from multiple regression equations of the linearized decoupled S-system. Furthermore, the algorithm is extended to the optimization of network topologies with constraints on metabolites and fluxes. These constraints rejoin the system in cases where it had been fragmented by decoupling. We demonstrate with synthetic time series why the algorithm can be expected to converge in most cases.

Conclusion: A procedure was developed that facilitates automated reverse engineering tasks for biological networks using S-systems. The proposed method of eigenvector optimization constitutes an advancement over S-system parameter identification from time series using a recent method called *Alternating Regression*. The proposed method overcomes convergence issues encountered in alternate regression by identifying nonlinear constraints that restrict the search space to computationally feasible solutions. Because the parameter identification is still performed for each metabolite separately, the modularity and linear time characteristics of the alternating regression method are preserved. Simulation studies illustrate how the proposed algorithm identifies the correct network topology out of a collection of models which all fit the dynamical time series essentially equally well.

Background

Metabolic and genetic time series have arisen as important sources of information about biological processes. However, the quantitative characterization of these processes from their temporal responses is not a trivial problem due to the complexity of typical biological networks and the multi-fold interdependencies among their components. Any effective method for this task needs to be able to filter out all possible quantitative information from observed time series and convert it into mathematical features that reliably characterize the true topology of the network, as well as its regulation. Biochemical System Theory (BST) [1-3] has been shown to provide a consistent mathematical framework for representing and analyzing biological systems. The S-system variant of BST represents the biological network as a set of differential equations in the general format

$$\dot{X}_i = \boldsymbol{\alpha}_i \prod_{j=1}^{M} X_j^{g_{ij}} - \boldsymbol{\beta}_i \prod_{j=1}^{M} X_j^{h_{ij}}, i = 1, 2, \cdots, M. \quad (1)$$

Here, X_i represents the concentration of metabolite i, $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$ are non-negative rate constants, and g_{ij} and h_{ij} are real-valued kinetic orders for the production and degradation term, respectively. A considerable amount of information about S-systems can be found in [1-5]. A major advantage of this representation is that it uniquely maps dynamical and topological information onto its parameters; an illustration is given in Figure 1.

Several numerical techniques have been proposed in the literature to tackle the inverse problem of S-system parameterization from time series; most of them use computationally expensive meta-heuristics such as Genetic Algorithms (GA) [6-11], Simulated Annealing (SA) [12], artificial neural networks [13], function approximation [14,15], or global optimization methods [16]. Collectively, these studies have shown that any direct parameter estimations typically face grave problems. Major improvements in efficiency are found when the derivatives at a series of time points are replaced with estimated slopes [4-6] and [17]. This step at once replaces the differential equations with sets of algebraic equations and decouples these sets so that the parameters for each metabolite can be computed separately.

Differing from expensive direct estimation methodologies, alternating regression (AR) [18] was proposed as a fast deterministic method for S-system parameter estimation with low computational cost (see Methods Section). Its superb efficiency is due to the reduction of the nonlinear estimation problem into iterative steps of linear regression. Apparently its only disadvantage is the observation that the method does not converge for some systems, and that necessary and sufficient criteria for convergence are not known. Thus, given a new system and new data, it is *a priori* difficult to predict whether AR will or will not converge. If it converges, it converges very fast.

In this report, we propose a new method, inspired by AR and based on multiple linear regression and sequential quadratic programming (SQP) optimization, to address the S-system parameter identification problem when no information about the network topology is known. The algorithm accounts for the often observed quasi-redundancy among S-system parameters, where errors in kinetic orders can largely be compensated by adjustments in other kinetic orders and rate constants. In contrast to AR, the proposed method operates initially only on one term (production or degradation), whose constant rate ($\boldsymbol{\alpha}$ or $\boldsymbol{\beta}$) and kinetic orders (g's and h's) are optimized completely before the complementary term is estimated. In many cases, the method provides alternative candidate models that fit the time series both in the decoupled and the fully integrated forms.

Results

Synthetic time series

The proposed method was tested on synthetic time series generated by reference test models [11,18,19] of 2, 4, and 5 state variables (Equations 2, 3, and 4 respectively). Each system was simulated with different initial concentrations of its variables in order to imitate different biological stimulus-response experiments as described in [18]. All specifications of the simulations with different initial conditions can be found in Additional file 1.

In all three case studies, no knowledge about the pathway was assumed and all parameters were considered freely variable. Even so, the correct network topology was extracted in all cases, with a mean error magnitude of 10^{-5} for each numerically integrated state variable.

The 2-dimensional system

$$\begin{aligned} \dot{X}_1 &= 3X_2^{-2} - X_1^{0.5}X_2 \\ \dot{X}_2 &= X_1^{0.5}X_2 - X_2^{0.5} \end{aligned} \quad (2)$$

exhibits oscillatory behavior that is challenging for estimation purposes, leading to difficulties of standard algorithms in finding good solutions. The reason is that even small shifts in the oscillation phase between the dynamics of the estimated system and the true target system result in significant cumulative errors. By contrast, the 4-dimensional system

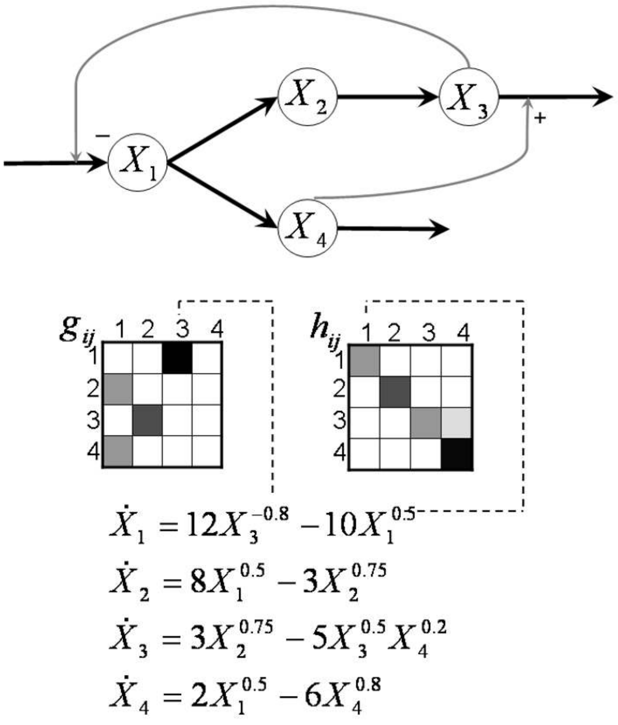

X_1
X_2
X_3
X_4
−
+
g_{ij}
1 2 3 4
1
2
3
4
h_{ij}
1 2 3 4
1
2
3
4
$\dot{X}_1 = 12X_3^{-0.8} - 10X_1^{0.5}$
$\dot{X}_2 = 8X_1^{0.5} - 3X_2^{0.75}$
$\dot{X}_3 = 3X_2^{0.75} - 5X_3^{0.5}X_4^{0.2}$
$\dot{X}_4 = 2X_1^{0.5} - 6X_4^{0.8}$

Figure 1

Topology mapping. Example of network topology mapping onto kinetic orders in an S-system [17]. The exponents in the equations directly correspond to effects of metabolites on processes (flux arrows) in the pathway diagram. As an example, the flux out of X_3 is affected by X_3 as substrate and by X_4 as activator. Both variables appear in the corresponding term with their respective kinetic orders. The gray-scale in the *g* and *h* matrices reflects the magnitudes of the exponents in the production and degradation terms of the S-system, respectively, with higher values shown in darker hues.

$$\begin{aligned}\dot{X}_1 &= 12X_3^{-0.8} - 10X_1^{0.5}\\ \dot{X}_2 &= 8X_1^{0.5} - 3X_2^{0.75}\\ \dot{X}_3 &= 3X_2^{0.75} - 5X_3^{0.5}X_4^{0.2}\\ \dot{X}_4 &= 2X_1^{0.5} - 6X_4^{0.8}\end{aligned} \qquad (3)$$

(see Figure 1 for the corresponding pathway) is relatively well behaved and will be used to identify problems that are likely to emerge even for the inference of less complicated dynamic models. The third system (Equation 4) describes an artificial genetic network and has been used as a benchmark [11,18,20] for S-system inference algorithms.

$$\begin{aligned}\dot{X}_1 &= 5X_3X_5^{-1} - 10X_1^2\\ \dot{X}_2 &= 10X_1^2 - 10X_2^2\\ \dot{X}_3 &= 10X_2^{-1} - 10X_2^{-1}X_3^2\\ \dot{X}_4 &= 8X_3^2X_5^{-1} - 10X_4^2\\ \dot{X}_5 &= 10X_4^2 - 10X_5^2\end{aligned} \qquad (4)$$

The results of the algorithm on the 2, 4 and 5-dimensional systems, presented in Additional file 1, demonstrate that the proposed method retrieves the correct parameter values for noise-free time series. Three different data sets were created for each test systems (Equations 2, 3 and 4) using different initial conditions in the system's numerical integration (see Additional file 1). These three data sets allowed us to assess the ability of the algorithm to deal with different time series dynamics. Using each data set, we performed 10 trials for each system's variables (X_i). The runs differed in the random initial guess for $\boldsymbol{\beta}$ (see *Initial parameter guesses* section for the initialization of the kinetic order values) which was chosen from the range [0.1, 12]. The search space for kinetic orders was limited to a reasonable range of [-2, 3], which is consistent with collective experience in the field (see Chapter 5 in [4]). As an example result, the experiment with the 5-dimensional system performed on the first data set illustrates the success rate of the algorithm: the exact parameter values were found for all variables in all trails except for variable X_5 in one of the trials. The procedure is computationally efficient, requiring 3 minutes to perform 40 optimizations for the 4-dimensional system (10 optimizations for each state variable corresponding to approximately 5 seconds per case), on a personal computer with a 2.00 GHz processor and 1 GB RAM. Thanks to the numerical decoupling, the complexity of the algorithm is of the order *O(M*N)* where *M* is the number of state variables and *N* is the number of data points used in the optimization. All experiments were performed with 100 data points. For the 5-dimensional system the proposed algorithm found the correct parameter set, overcoming the problematic identification of the kinetic orders g_{32} and h_{32} of the state variable X_3 presented by most algorithms in the literature. If a stop criterion is defined as a value of 1e-12 for the sum of the squared errors between the slopes of the optimized system and the true slopes, the time required to identify the system parameters for the 5-dimensional system was 23 sec on the machine described above. An experiment with a 10-dimensional system was also performed and the total time consumed was 75 sec (see Additional file 1).

Similar results were achieved with the optimization of the 2-dimensional system. Importantly, the correct parameter set was found, although not with the same regularity as in the 4- and 5-dimensional system optimizations. Issues encountered in finding the correct solutions appeared to be caused by a combination of different features of the system, such as the position of the optimal point within the feasible parameter space, which in the 5-variable case is situated right on the border of the infeasible region within the parameter space (see Figure 1 of the Additional file 1), multiple local minima, as well as the particular choice of initial parameter guesses. These peculiarities of the algorithm and the problem itself lead to different parameter values, although the errors of the decoupled and integrated system are still small (typically about at the order of 1e-5; for instance, see Tables 23, 29 and 30 in the Additional file 1).

The proposed algorithm calculates the initial guesses for the kinetic orders as close to zero as possible, given an initial β value (see section *Initial parameter guesses*). However, in this specific case study, near-zero values of the kinetic orders h_{11} and h_{12} for the constant rate $\beta_1 = 1$ fall into the infeasible parameter region, which complicates the parameter optimization. For instance, the smallest feasible value for h_{12} is 0.8636. The proposed algorithm overcomes this initial problem by adjusting itself and subsequently returns correct solutions when the system is rescaled in time [21]. This is most easily achieved by multiplying the alphas (α_1 and α_2) and betas (β_1 and β_2) with a positive factor (see example in Additional file 1), which increases the feasible parameter space. This step is, in fact, equivalent to multiplying the slope vector by a positive number. Thanks to the modularity of the decoupled system, this scaling can be performed separately for each state variable without affecting the kinetic order values. Only the values of the rate constants are changed, but they are easily recovered by dividing them by the positive number used for scaling. It was observed that this strategy often, but not always, enhances the algorithmic performance. It appears to improve performance most if the rate constants have small values.

Initially, all experiments were performed with noise-free time series, but in a second set of experiments, we added noise. Because the proposed algorithm uses the decou-

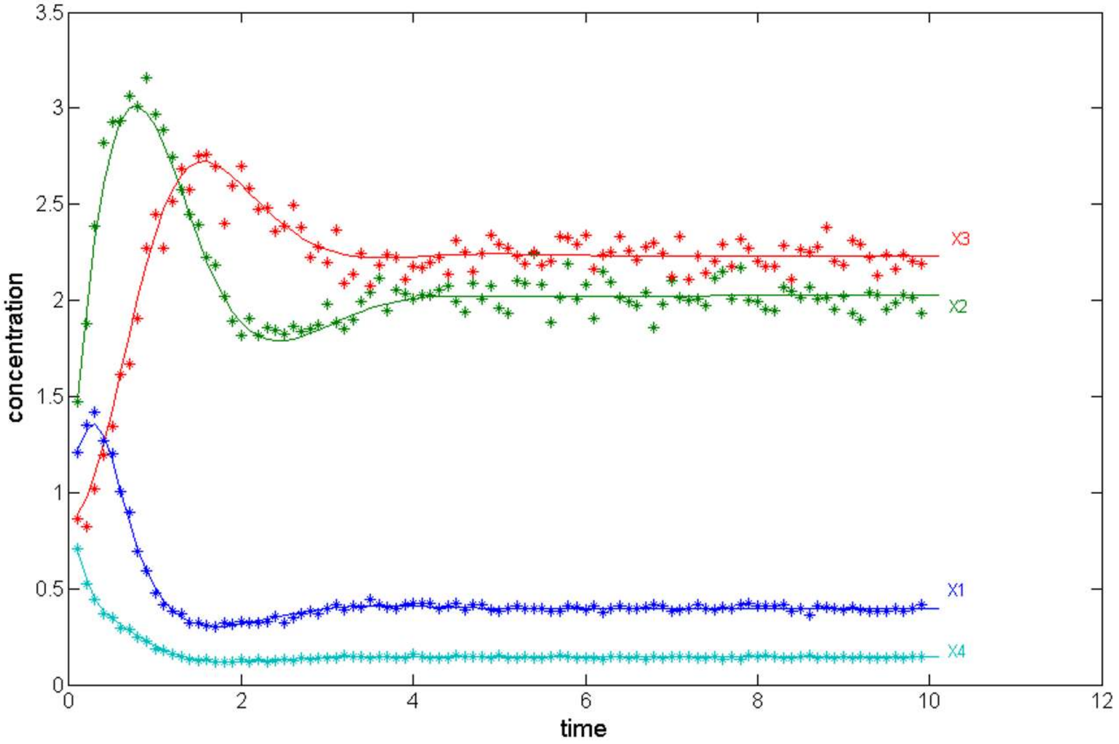


Figure 2

Noisy time series. Noisy time series data (symbols) and results of the numerical integration of the estimated model (solid lines; *cf*. Eq. (3)). In spite of slight numerical discrepancies between the estimated parameters and their target values (see Additional file 1), the estimated model accurately predicts the dynamics of the target system, indicating quasi-redundancy [*e.g*., [27] and [25]] or "sloppiness" [26] among the parameters.

pled, algebraic form, a signal extraction procedure was employed for the noisy data to provide smooth time series and slopes [22]. The results show that combining the two strategies (smoother and proposed algorithm) generate accurate dynamical responses for the case studies used in this report (Figure 2).

Error surfaces of decoupled S-systems

To explore the results of the proposed algorithm visually and to investigate patterns of convergence, we performed a grid search on the parameters of the 2-dimensional system (Equation 2). Specifically, we searched a 100 × 100 grid where each point represented the kinetic orders h_{11} and h_{12} over the range [-2.5, 2.0]. Correspondingly, 100 time points for X_1 and X_2 and its correspondent slopes S_1 and S_2 were generated by numerical integration of the 2-dimensional system (Equation 2) with $X_1(0) = 3$ and $X_2(0) = 1$ as initial conditions. Methods described in a later section were used on time series of X_1 and X_2 to calculate the regression matrix L, and for each given initial value of the rate constant $\boldsymbol{\beta}_1$ (uniformly spaced over the interval [1,6]) and for each point of the grid, the error surface for the variable X_1 was constructed. The algorithm started with the degradation term $DT_1 = \boldsymbol{\beta}_1 X_1^{h_{11}} X_2^{h_{12}}$ for the first grid point using a given value for $\boldsymbol{\beta}_1$ and the time series points for X_1 and X_2. Subsequently, the production vector (Vp_1 = [log($\boldsymbol{\alpha}_1$) g_{11} g_{12}]) was obtained from the slope vector S_1, the regression matrix L, and the degradation term DT_1 in Equations (7)–(10). Once all parameter values for variable X_1 in the production and degradation vectors were determined, the estimated slopes were calculated ($\hat{S}_1 = PT_1 - DT_1$) and the logarithm of the sum of the squared errors between these slopes and the target solutions was computed as $error = \log\left(\sum(S_1 - S_1)^2\right)$. This process was repeated for all points on the grid such that an error surface resulted for each $\boldsymbol{\beta}_1$ value. In this manner, ten surfaces were constructed using different $\boldsymbol{\beta}$ values; they are shown superimposed in Figure 3.

The first observation is that most of the search region is not feasible (unfilled *X-Y* space), even though there is *a priori* no hint that solutions in the open range should not converge. It turns out in retrospect that these are regions where the argument of the logarithm on right side of Equation 7 is negative, due to negative slope values. Also worth noting is that for each $\boldsymbol{\beta}$ a similarly shaped surface ("bowl") was found, but that not all surfaces have the same minimal point (Figures 3 and 4). This information will be of critical importance in the discussion of the convergence profile of the proposed method.

The same strategy was applied to noisy time series resulting in a new set of surfaces (data not shown). Gaussian noise with 15% variance was added to the X_1 and X_2 time series and a refined Whitaker's filter [22] was used to smooth the data and estimate slopes.

The error surfaces obtained using noisy data (Figure 5) present the same shapes as seen for the noise-free data except that the error average is higher and points to a different global minimum, which however is essentially indistinguishable in value from the local optima (see Additional file 1 for details).

Convergence problems

It would be unreasonable to assume that the algorithm converges to the global optimum under all imaginable conditions and initial settings: no estimation algorithm for nonlinear systems can – or should be expected to – measure up to such high a standard. For instance, if the ranges of initial guesses are changed or if the number of initial guesses is reduced, the algorithm may converge to an acceptable local minimum which, however, is not global. This is not surprising, given the complicated nature of the error surface of realistic systems and the fact that nonlinear systems often exhibit almost flat, banana-shaped or ellipsoid valleys in which the minimum is centered [23-27]. At this point, a comprehensive picture of potential obstacles to convergence is not available. One prominent reason for lacking or faulty convergence is that some problems are ill-posed, for instance, because of collinearity between columns of the regression matrix *L*. This situation occurs when two or more metabolites have similar dynamics [25] or when at least one variable is essentially constant and is therefore collinear with the first column of the *L* matrix. In these and some other cases, the regression matrix *L* has a high condition number, which the proposed procedure flags. It might be possible to remedy some of these ill-posed problems with a regularization algorithm for multiple linear regression and through redesigning the algorithm with the regularized solution. It seems advisable in any event to remove model redundancies, for instance by pooling or eliminating collinear variables or merging essentially constant variables with the rate constants of the term.

Parameter estimation of constrained networks

The proposed method was extended to address the parameter identification for systems with topological constraints. This extension allows the algorithm to account for precursor-product relationships problems, which mandate that the degradation term of the precursor is

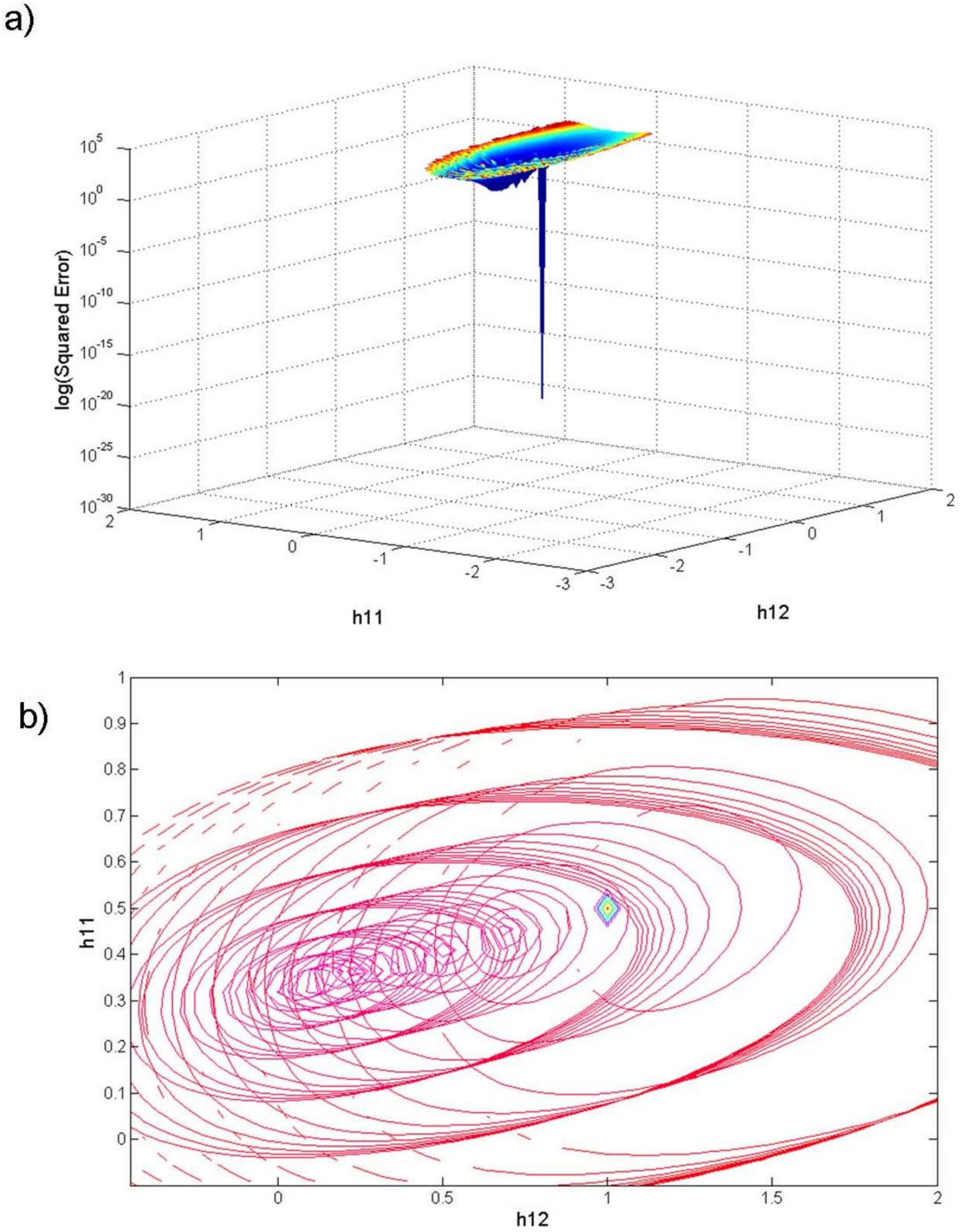


Figure 3

Error surfaces. a) Ten error surfaces associated with variable X_1 of the 2-dimensional system were obtained using an exhaustive grid search covering 10 different initial guesses. b) Zooming in shows the composite contour map (level sets) of the error surfaces.

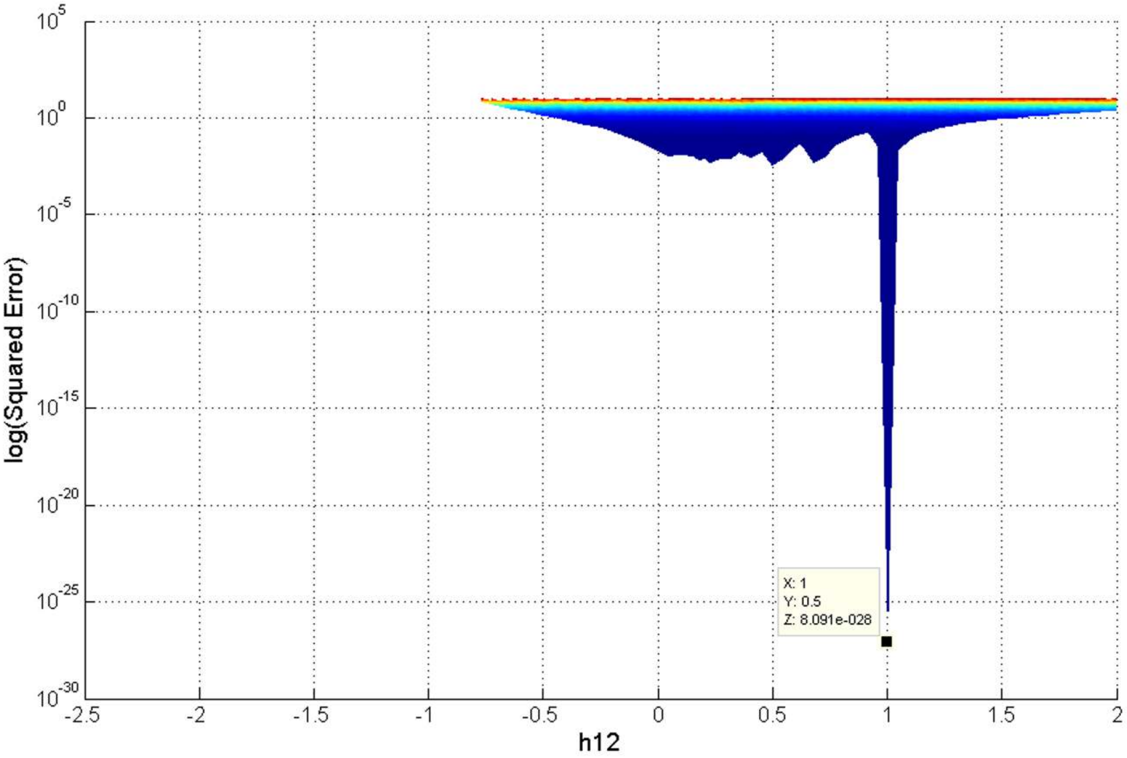


Figure 4
Multiple minima. *Z-Y* projection of the error surfaces in Figure 3a. Different minima are found for different ***β*** values.

equivalent to the production term of the product [28]. Thus, instead of optimizing the parameters for each metabolite separately, a set of terms is optimized simultaneously, consisting of one of the parameter vectors (production or degradation vector) of each metabolite. As an illustrative, simple example, consider a linear pathway with feedback, where we have to account for constraints between the production and degradation terms of subsequent metabolites (Figure 6). Specifically in the example system, the efflux from X_1 is identical to the influx into X_2, and the efflux from X_2 is identical to the influx into X_3. Consequently, the degradation term of X_1 is exactly the same as the production term of X_2, and the degradation term of X_2 must be the same as the production term of X_3. The amendment of the proposed method toward simultaneous estimation readily satisfies these types of constraints.

The extended algorithm was applied to the 3-dimensional linear pathway system in Figure 6, and some of the results are shown in Additional file 1. The algorithm found the correct parameter set, and all 10 optimizations, in which the algorithm now performs a single, combined optimization for all variables simultaneously, thereby accounting for constraints, were completed in 37 sec on a 2.00 GHz processor with 1 GB RAM.

Graphical user interface

An open source MATLAB toolbox and a stand-alone compiled Graphical User Interface (GUI) application were developed as an exploratory tool (see Section *Availability and requirements*). The application was developed as a modular extension of our previous work and constitutes a critical component within our long-term effort of advancing a data processing pipeline for S-system estimation from metabolomic time series [13,22]. A snapshot of the GUI is shown in Figure 7. All computational results and graphics described in this report can be reproduced using this application.

Discussion

There are many reasons why it may be desirable to reverse engineer a biological network without making assumptions about the underlying processes. The most obvious reason is that no reliable information may be available about the processes. Another situation occurs when several network topologies are *a priori* possible and the reverse approach is employed to prioritize alternative

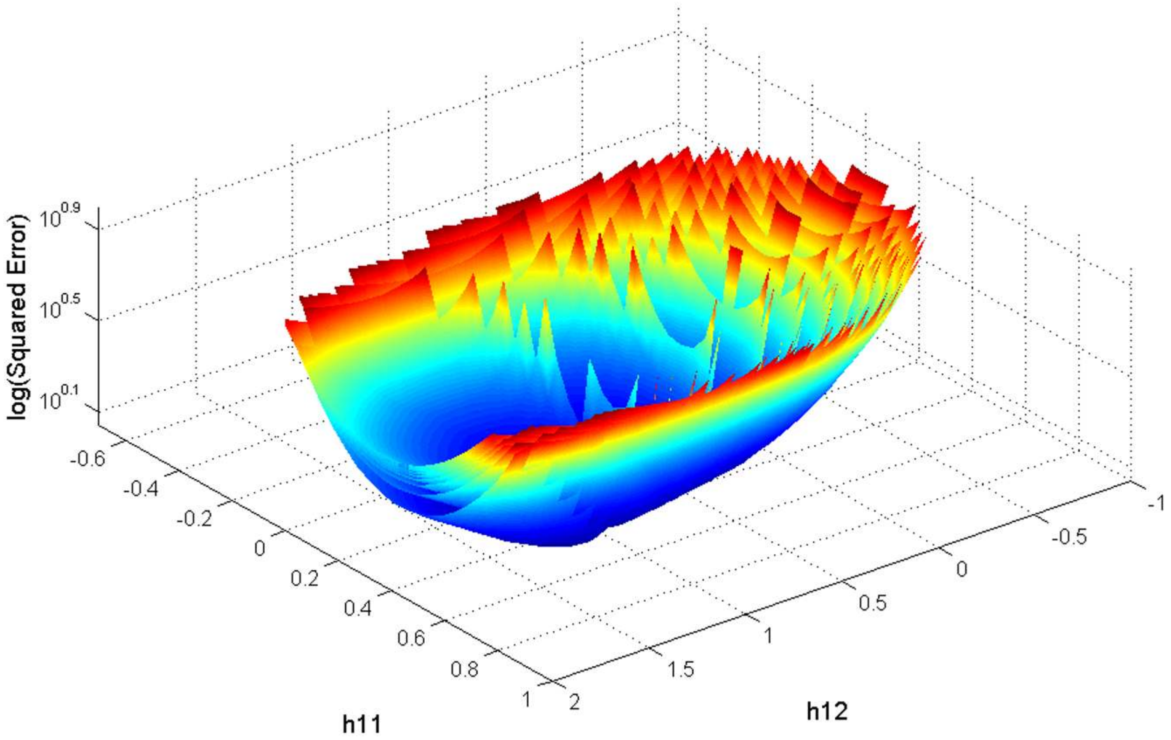
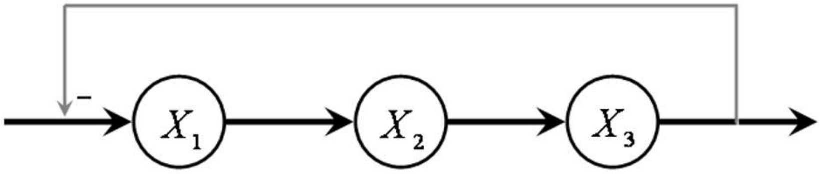


Figure 5
Error surfaces from noisy time series. Ten error surfaces of the variable X_1 of the 2-dimensional system obtained from noisy time series after signal extraction and slope estimation.

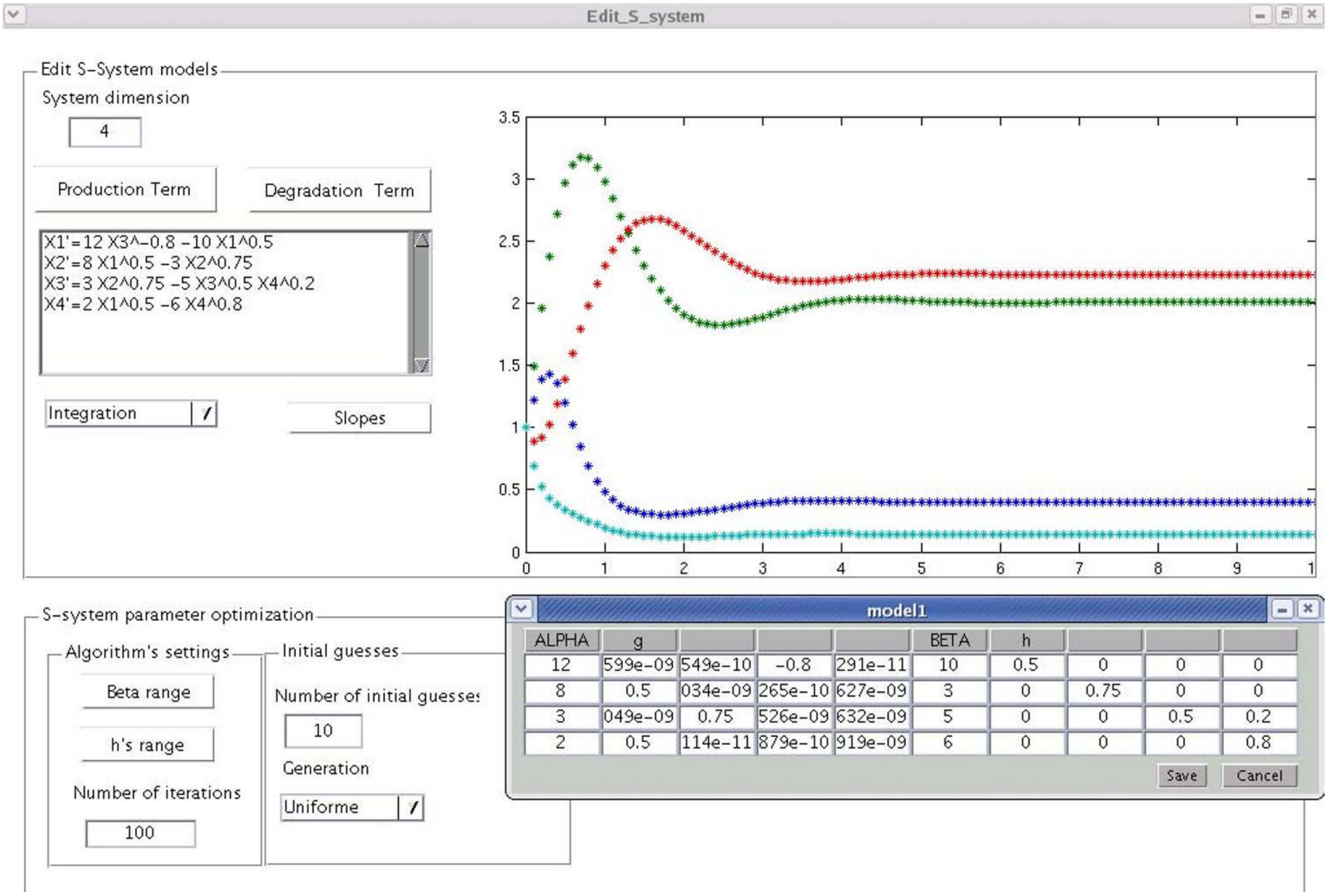
hypotheses. The algorithm proposed here is an extension of Alternating Regression (AR; [18]) that in many cases shows improved convergence behavior.

The proposed algorithm was exhaustively tested on diverse time series (see Text above and Additional File 1). In all of these tests, the convergence followed the same pattern: the error slowly decreased during the first few iterations and then suddenly dropped to a significant lower plateau, from where it gradually decreased again. This pattern repeated until one of the stop conditions (maximal number of iterations, minimal gradient value or minimal cost function value) was reached. The error drop points matched with significant changes in the beta gradient and appear to correspond to transitions to a "bowl" with a lower error surface (*cf*. Figures 3 and 5). As shown in Figures 3b and 5, most "bowls" have different minimal points, corresponding to good, yet local minima. Because



$$\dot{X}_1 = 12X_3^{-0.8} - 10X_1^{0.5}$$
$$\dot{X}_2 = 10X_1^{0.5} - 3X_2^{0.75}$$
$$\dot{X}_3 = 3X_2^{0.75} - 5X_3^{0.5}$$

Figure 6
Linear system topology. Linear pathway with precursor-product constraints.



Edit S-System models

System dimension: 4

```
X1'=12 X3^-0.8 -10 X1^0.5
X2'=8 X1^0.5 -3 X2^0.75
X3'=3 X2^0.75 -5 X3^0.5 X4^0.2
X4'=2 X1^0.5 -6 X4^0.8
```

Integration

S-system parameter optimization

Algorithm's settings

Number of iterations: 100

Initial guesses

Number of initial guesses: 10

Generation: Uniforme

ALPHA	g				BETA	h			
12	599e-09	549e-10	-0.8	291e-11	10	0.5	0	0	0
8	0.5	034e-09	265e-10	627e-09	3	0	0.75	0	0
3	049e-09	0.75	526e-09	632e-09	5	0	0	0.5	0.2
2	0.5	114e-11	879e-10	919e-09	6	0	0	0	0.8

Figure 7

Software application. Snapshot of the graphical user interface provided as a free stand-alone application.

the proposed algorithm is computationally very efficient, it allows the exploration of the parameter space in a reasonable amount of time (seconds to minutes). Such an exploration with new initial $\boldsymbol{\beta}$ values is recommended, if very precise solutions or alternative parameter sets are needed. Because alternative parameter combinations may correspond to different topological and regulatory structures [4], estimations with different initial values in fact constitute explorations of the structure and functionality of the biological space in which the pathway operates.

Conclusion

S-systems present a unique balance between proven biological relevance and validity on one hand, and mathematical convenience and tractability on the other. For this reason, the recent years have seen numerous methods for matching S-system models to measured biological time series data. In the relatively simpler scenario of this type, the topology and regulatory structure of the biological system is known, and the extraction of information from the data constitutes a parameter estimation task. In the more difficult situation, at least some of the structure is unknown, and in the extreme situation no information about the topology of the interactions between variables is available. In this article we propose a new algorithm that efficaciously identifies the correct topology of a system from time series. The only true assumptions made are that all important variables are accounted for and that the S-system model is capable of modeling the data. The first assumption is presently unavoidable, at least in the generality presented above. The second assumption has been found to be true in very many cases, as a rich body of publications on S-systems demonstrates. The proposed algorithm was conceived as a critical piece of an emerging data processing "pipeline" that will eventually accept time series and other data characterizing biological pathways and more or less automatically propose topological and regulatory structures that are consistent with the input data. This algorithm will be a valuable tool for analysis and hypothesis generation in systems biology.

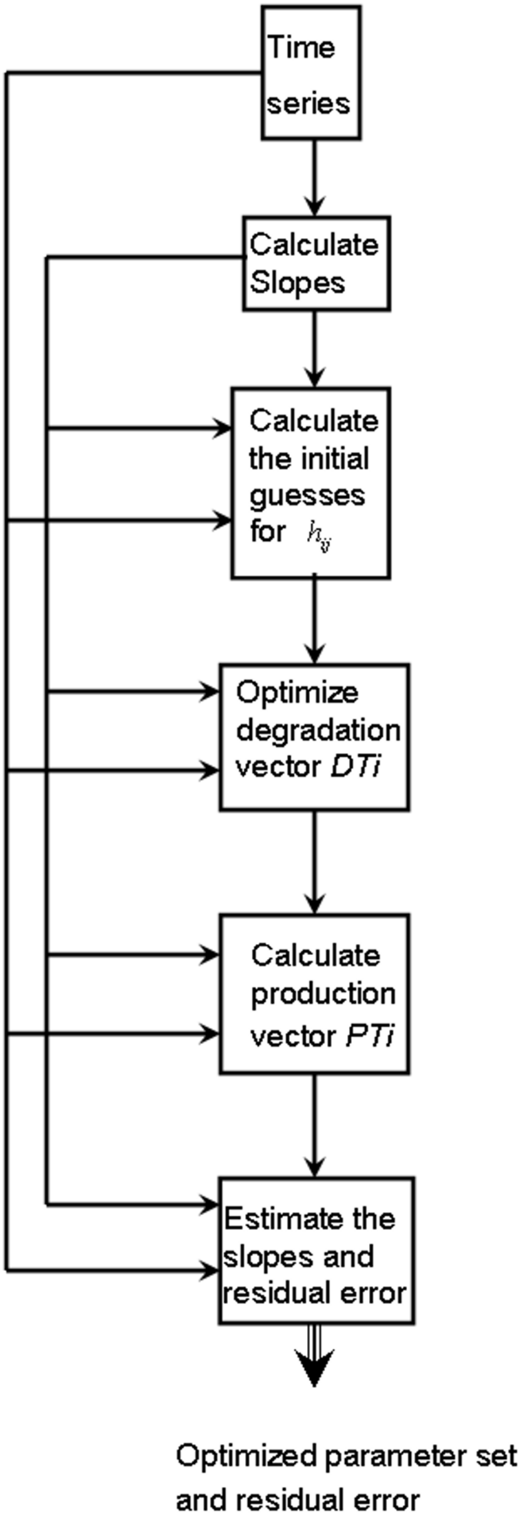

Time series
Calculate Slopes
Calculate the initial guesses for h_{ij}
Optimize degradation vector *DTi*
Calculate production vector *PTi*
Estimate the slopes and residual error
Optimized parameter set and residual error

Figure 8
Flowchart. Flowchart of the proposed algorithm. To perform the optimization process, the algorithm requires only the time series set and an initial $\boldsymbol{\beta}$ value as input.

Methods

Eigenvector optimization

The proposed method was inspired by Alternating Regression (AR [18]) and is based on the substitution of differentials with estimated slopes [4,5,17] and the minimization of the differences between two vectors obtained from multiple linear regression equations. In contrast to AR, the new algorithm estimates one term per equation with high accuracy and computes the other term through linear regression ensuring that the new term will fall into the feasible space. Specifically, the task is initially posed in relation to one of the two terms of an S-system equation with *M* species (*e.g.*, metabolites), either the production term vector $PT_i(t_n) = \boldsymbol{\alpha}_i \prod_{j=1}^{M} X_j(t_n)^{g_{ij}}$ or the degradation term vector $DT_i(t_n) = \boldsymbol{\beta}_i \prod_{j=1}^{M} X_j(t_n)^{h_{ij}}$, which are both defined for each metabolite *i* at a series of *N* time points t_n. Let $S_i(t_n)$ denote the estimated slope of metabolite *i* at time t_n. In simplified notation, $S_i(t_n)$ is given by

$$S_i(t_n) = PT_i(t_n) - DT_i(t_n),\ n = 1, 2, \cup, N \qquad (6)$$

Because PT_i must be positive, Equation 6 can be rewritten as

$$\log(PT_i) = \log(S_i + DT_i), \qquad (7)$$

or in matrix form as

$$L \cdot Vp_i = y_i, \qquad (8)$$

where the production parameter vector is given as $V_{pi} = [\log \boldsymbol{\alpha}_i\ g_{i1}\ g_{i2} \cup g_{iM}]$, $y_i = \log(S_i + DT_i)$, and the regression matrix L is

$$L = \begin{bmatrix} 1 & \log(X_1(t_1)) & \cdots & \log(X_i(t_1)) & \cdots & \log(X_M(t_1)) \\ 1 & \log(X_1(t_2)) & \cdots & \log(X_i(t_2)) & \cdots & \log(X_M(t_2)) \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ 1 & \log(X_1(t_N)) & \cdots & \log(X_i(t_N)) & \cdots & \log(X_M(t_N)) \end{bmatrix}. \qquad (9)$$

As is standard with multiple linear regression models, the production parameter vector Vp_i can be obtained as

$$Vp_i = (L^T L)^{-1} L^T y_i, \qquad (10)$$

as long as the inverse exists. Substituting this result in Equation 8 directly yields

$$L(L^T L)^{-1} L^T y_i = y_i. \qquad (11)$$

Recall that vector γ_i is a function of the degradation parameters ($\boldsymbol{\beta}_i$ and h_{ij}), which thus must satisfy Equation 11. Specifically, γ_i must be an eigenvector of the matrix $W = L(L^TL)^{-1}L^T$ with an eigenvalue equalling 1.

We used several standard algorithms to calculate the eigenvector of the matrix W directly, but none of them returned a satisfactory result. The presumed reason is that any vector which belongs to the eigenspace of W corresponding to eigenvalue 1 satisfies the Equation 11. We therefore forced the eigenvector γ_i to be in the form $\log(S_i+DT_i)$ and reformulated the task as a minimization problem for the logarithm of the squared residuals between the right and left side hands in Equation 11 and defined this problem in matrix form with the cost function

$$F = \log\Big(\left(\gamma_i - \gamma_i \right)^T \left(\gamma_i - \gamma_i \right) \Big), \tag{12}$$

where $\hat{\gamma}_i = W\gamma_i$. The gradients of this function with respect to the degradation parameters are given by Equations 13 and 14:

$$\frac{\partial F}{\partial \boldsymbol{\beta}_i} = \frac{2}{\boldsymbol{\phi}} \left[(W - I) \left(\left(\prod_{j=1}^{M} X_j^{h_{ij}} \right) \circ \left(S_i + DT_i \right)^{\circ -1} \right) \right]^T \left[(W - I) \log \left(S_i + DT_i \right) \right], \tag{13}$$

$$\frac{\partial F}{\partial h_{ij}} = \frac{2}{\boldsymbol{\phi}} \left[(W - I) \left(\left(\boldsymbol{\beta}_i \prod_{j=1}^{M} X_j^{h_{ij}} \circ \log \left(X_j \right) \right) \circ \left(S_i + DT_i \right)^{\circ -1} \right) \right]^T \left[(W - I) \log \left(S_i + DT_i \right) \right]. \tag{14}$$

Here, the symbol $\boldsymbol{o}$ represents the Hadamard product between vectors [29] and $\boldsymbol{\varphi}$ is the logarithm of the argument of the right hand side of the Equation 12. Analogous gradient equations are obtained for the production terms. The algorithm avoids unfeasible solutions by satisfying the feasibility constraints

$$S_i(t_n) + \boldsymbol{\beta}_i \prod_{j=1}^{M} X_j(t_n)^{h_{ij}} > 0, n = 1, 2, \cdots, N. \tag{15}$$

We used the **fmincon** routine in MATLAB® (MathWorks) with built-in Sequential Quadratic Programming to execute the cost function constrained minimization.

Initial parameters guesses

Like all numerical optimization algorithms, the proposed method requires initial guesses. Satisfying the constraints in Equation 15, the proposed algorithm calculates initial guesses for the kinetic order h_{ij}, given a user-supplied value $\boldsymbol{\beta}_i$; specifically, h_{ij} and a small buffer value $\boldsymbol{\varepsilon}$ are chosen such that

$$\boldsymbol{\beta}_i \prod_{j=1}^{M} X_j^{h_{ij}} = \boldsymbol{\varepsilon} - S_i^-, \tag{16}$$

where S_i^- represents all negative slope values from the time series of X_i. A simple linear regression step in logarithmic space thus suffices to determine admissible initial guesses for the kinetic orders h_{ij}. In this fashion, for a given $\boldsymbol{\beta}_i$, small values of kinetic orders h_{ij} are provided to the optimization algorithm. As a technical note, it is easier to keep a null parameter value than to bring it to zero during the optimization. If the slope vector contains no negative values, the procedure is performed without $\boldsymbol{\varepsilon}$. A flowchart of the proposed algorithm is shown in Figure 8.

Refining solutions

Differently parameterized S-systems can exhibit quite similar temporal dynamics. This behavior is due the fact that S-systems are composed of production and degradation terms that may compensate for each other through different kinetic orders and constant rates that ultimately produce very similar time courses. As one consequence, it is quite common that optimization schemes identify non-zero values for parameters that should in truth be zero. Moreover, it is unlikely that any algorithm based on gradients will obtain parameters values exactly equal to zero. For these reasons, our algorithm automatically checks parameter values and forces kinetics orders below a quite arbitrary threshold of (0.009) to be zero; a new optimization process is the initiated in which the parameter is constrained to be zero.

Extension to constrained topologies

To address linear pathway sections, constraints are imposed in accordance with the structure of the system when the parameter optimization is performed. For instance, for the linear system with precursor-product relationships (Figure 6), the optimization is performed with the degradation term of the precursor metabolite forced to be equal to the production terms of the product. In such a case, the Equation 11 is formulated for each state variable

$$\begin{array}{c} W\gamma_1(\boldsymbol{\beta}_1, h_{1j}, S_1) = \gamma_1(\boldsymbol{\beta}_1, h_{1j}, S_1) \\ W\gamma_2(\boldsymbol{\alpha}_2, g_{2j}, S_2) = \gamma_2(\boldsymbol{\alpha}_2, g_{2j}, S_2) \\ \vdots \\ W\gamma_M(\boldsymbol{\alpha}_M, g_{Mj}, S_M) = \gamma_M(\boldsymbol{\alpha}_M, g_{Mj}, S_M) \end{array}, \tag{17}$$

and the sum of the equations returns the eigenvector problem

$$W\left(\sum_{i=1}^{M} y_i\right) = \sum_{i=1}^{M} y_i. \tag{18}$$

A cost function similar to Equation 12 can be formulated using the Equation 18, and the same optimization procedure is used. To force flux conservation, the following constraints were imposed on the optimization algorithm

$$\begin{aligned} &\boldsymbol{\beta}_1 = \boldsymbol{\alpha}_2 \\ &h_{1j} = g_{2j},\ j = 1, 2, .., M \end{aligned} \tag{19}$$

to impose

$$DT_1 = PT_2, \tag{20}$$

and the degradation term of X_2 was forced to be equal the production term of X_3

$$\begin{aligned} &PT_3 = DT_2 \\ &\boldsymbol{\alpha}_3 \prod_{j=1}^{M} X_j^{g_{3j}} = PT_2 - S_2 \end{aligned}. \tag{21}$$

Applying logarithms on both sides of the Equation 21 and solving the equation by multiple linear regression, the final constraints are found as

$$\boldsymbol{\alpha}_3 = \prod_{n=1}^{N} \left(PT_2(t_n) - S_2(t_n) \right)^{C_{1n}}, \tag{22}$$

and

$$g_{3j} = \sum_{n=1}^{N} C_{j+1,n} \log\left(PT_2(t_n) - S_2(t_n) \right),\ j = 1, 2, .., M, \tag{23}$$

where $C = (L^T L)^{-1} L^T$. The constraints can be rewritten in a general form as

$$\boldsymbol{\alpha}_M = \prod_{n=1}^{N} \left(PT_{M-1}(t_n) - S_{M-1}(t_n) \right)^{C_{1n}}, \tag{24}$$

and

$$g_{Mj} = \sum_{n=1}^{N} C_{j+1,n} \log\left(PT_{M-1}(t_n) - S_{M-1}(t_n) \right),\ j = 2, .., M. \tag{25}$$

Analogous optimization routines were used for other constraints.

Availability and requirements

The implementation of the algorithm described in this report is made publicly (GNU GPL) available with open source as Matlab m-code (MathWorks Inc) at http://code.google.com/p/s-system-inference/. For the convenience of those without a Mathworks license we have also compiled the code as a stand-alone application made publicly available at the same site, or as a module ("Signal Extraction Toolbox") of the code distribution infrastructure of the Bioinformatics Station resource http://bioinformaticstation.org.

Competing interests

The author(s) declare that they have no competing interests.

Authors' contributions

MV conceived the core methods of eigenvector optimization. I-Chun Chou spearheaded Alternating Regression and tested the eigenvector method.

SV participated in the analysis and systematization of the method.

ATV supported the development of the proposed algorithm with funds and designed the computational tests.

EOV initiated the field of network identification with S-systems and supervised activities leading to this paper.

JSA conceived the ideas of automating the identification of S-systems and creating a model pipeline.

All authors contributed to in the preparation of the manuscript.

Additional material

Additional file 1
Supplementary material.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1752-0509-2-35-S1.doc]

Acknowledgements

This work was supported in part by a National Heart, Lung and Blood Institute Proteomics Initiative (Contract N01-HV-28181; D. Knapp, PI), a Molecular and Cellular Biosciences Grant (MCB-0517135; E.O. Voit, PI) from the National Science Foundation, project DynaMo (PTDC/EEA-ACR/69530/2006; S. Vinga, PI) from the Portuguese Science Foundation (FCT), and an endowment from the Georgia Research Alliance. M. Vilela also thanks the University of Texas M.D. Anderson Cancer Center for support. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsoring institutions.

References

1. Savageau MA: **Biochemical systems analysis. I. Some mathematical properties of the rate law for the component enzymatic reactions.** *J Theor Biol* 1969, **25(3):**365-369.
2. Savageau MA: **Biochemical systems analysis. II. The steady-state solutions for an n-pool system using a power-law approximation.** *J Theor Biol* 1969, **25(3):**370-379.
3. Savageau MA: **Biochemical systems analysis. 3. Dynamic solutions using a power-law approximation.** *J Theor Biol* 1970, **26(2):**215-226.
4. Voit EO: **Computational analysis of biochemical systems : a practical guide for biochemists and molecular biologists.** Cambridge ; New York , Cambridge University Press; 2000:xii, 531 p., [8] p. of plates.
5. Voit EO, Savageau MA: **Power-law approach to modeling biological systems; III. Methods of analysis.** *J Ferment Technol* 1982, **60(3):**233-241.
6. Sugimoto M, Kikuchi S, Tomita M: **Reverse engineering of biochemical equations from time-course data by means of genetic programming.** *Biosystems* 2005, **80(2):**155-164.
7. Nakatsui M Ueda, T., Ono, I., Okamoto, M.,: **Control Aspect of Common Interactions Extracted form Inferred Network Candidates of Gene Expression.** *Genome Informatics* 2004: P008.
8. Nakatsui M Ueda, T., Okamoto, M.: **Integrated System for Inference of Gene Expression Network.** *Genome Informatics* 2003, **14:**282-283.
9. Ueda T, Ono I, Okamoto M: **Development of system identification technique based on real-coded genetic algorithm.** *Genome Informatics* 2002, **13:**386-387.
10. Maki Y, Ueda T, Okamoto M, Uematsu N, Inamura K, Uchida K, Takahashi Y, Eguchi Y: **Inference of Genetic Network Using the Expression Profile Time Course Data of Mouse P19 Cells.** *Genome Informatics* 2002, **13:**382-383.
11. Kikuchi S, Tominaga D, Arita M, Takahashi K, Tomita M: **Dynamic modeling of genetic networks using genetic algorithm and S-system.** *Bioinformatics* 2003, **19(5):**643-650.
12. Gonzalez OR, Kuper C, Jung K, Naval PC Jr., Mendoza E: **Parameter estimation using Simulated Annealing for S-system models of biochemical networks.** *Bioinformatics* 2007, **23(4):**480-486.
13. Almeida JS, Voit EO: **Neural-network-based parameter estimation in S-system models of biological networks.** *Genome Inform* 2003, **14:**114-123.
14. Tsai KY, Wang FS: **Evolutionary optimization with data collocation for reverse engineering of biological networks.** *Bioinformatics* 2005, **21(7):**1180-1188.
15. Matsubara Y, Kikuchi S, Sugimoto M, Tomita M: **Parameter estimation for stiff equations of biosystems using radial basis function networks.** *BMC Bioinformatics* 2006, **7:**230.
16. Polisetty PK, Voit EO, Gatzke EP: **Identification of metabolic system parameters using global optimization methods.** *Theor Biol Med Model* 2006, **3:**4.
17. Voit EO, Almeida J: **Decoupling dynamical systems for pathway identification from metabolic profiles.** *Bioinformatics* 2004, **20(11):**1670-1681.
18. Chou IC, Martens H, Voit EO: **Parameter estimation in biochemical systems models with alternating regression.** *Theor Biol Med Model* 2006, **3(1):**25.
19. Kutalik Z, Tucker W, Moulton V: **S-system parameter estimation for noisy metabolic profiles using newton-flow analysis.** *IET Syst Biol* 2007, **1(3):**174-180.
20. Hlavacek WS, Savageau MA: **Rules for coupled expression of regulator and effector genes in inducible circuits.** *J Mol Biol* 1996, **255(1):**121-139.
21. Voit EO: **Symmetries of S-systems.** *Math Biosci* 1992, **109(1):**19-37.
22. Vilela M, Borges CC, Vinga S, Vanconcelos AT, Santos H, Voit EO, Almeida JS: **Automated smoother for the numerical decoupling of dynamics models.** *BMC Bioinformatics* 2007, **8(1):**305.
23. Bos A: **A class of small sample nonlinear least squares problems.** *Automatica* 1980, **16:**487-490.
24. Jorge JM, Burton SG, Kenneth EH: **Testing Unconstrained Optimization Software .** *Volume 7 .* ACM; 1981:17-41.
25. Sands PJ and E.O. Voit: **Flux-based estimation of parameters in S-systems.** *Ecol Modeling* 1996, **93:**75-88.
26. Gutenkunst RN, Waterfall JJ, Casey FP, Brown KS, Myers CR, Sethna JP: **Universally sloppy parameter sensitivities in systems biology models.** *PLoS Comput Biol* 2007, **3(10):**1871-1878.
27. Berg PH, Voit EO, White RL: **A pharmacodynamic model for the action of the antibiotic imipenem on Pseudomonas aeruginosa populations in vitro.** *Bull Math Biol* 1996, **58(5):**923-938.
28. Voit E, Neves AR, Santos H: **The intricate side of systems biology.** *Proc Natl Acad Sci U S A* 2006, **103(25):**9452-9457.
29. Magnus JR, Neudecker H: **Matrix differential calculus with applications in statistics and econometrics.** In *Wiley series in probability and mathematical statistics* Chichester [England] ; New York , Wiley; 1988:xvii, 393 p..