Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **1**, No. 6, pp. 711-728

DOI: 10.4208/aamm.09-m09S04
December 2009

Inverse Eigenvalue Problems for Exploring the Dynamics of Systems Biology Models

James Lu[1,*]

[1] *Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Altenbergerstrasse 69 Linz 4010, Austria*

Received 05 April 2009; Accepted (in revised version) 01 September 2009
Available online 18 November 2009

Abstract. This paper describes inverse eigenvalue problems that arise in studying qualitative dynamics in systems biology models. An algorithm based on lift-and-project iterations is proposed, where the lifting step entails solving a constrained matrix inverse eigenvalue problem. In particular, prior to carrying out the iterative steps, *a-priori* bounds on the entries of the Jacobian matrix are computed by relying on the reaction network structure as well as the form of the rate law expressions for the model under consideration. Numerical results on a number of models show that the proposed algorithm can be used to computationally explore the possible dynamical scenarios while identifying the important mechanisms via the use of sparsity-promoting regularization.

AMS subject classifications: 65F18, 93B55, 65P30, 37N25, 15A29

Key words: Inverse eigenvalue problems, dynamical systems, bifurcation, biology, sparsity.

1 Introduction

Over the past decade, there has been much focus on the field of *systems biology*, with the fundamental goal being to understand how genes act together to bring about the wide-ranging regulatory functions within cells [1]. Various processes are controlled by networks of genes, including the cell division cycle and the circadian rhythm clock [12]; moreover, gene regulatory networks possess robust dynamical properties such that they are able to withstand fluctuating environmental conditions and the imprecision of the underlying biochemical components [2]. To shed light on the many questions that arise, various modeling paradigms have been developed, ranging from boolean models, ODE, PDE, delay-differential equations to stochastic models [12].

*Corresponding author.
URL: `http://www.ricam.oeaw.ac.at/people/page/jameslu/`
Email: james.lu@oeaw.ac.at (J. Lu)

In this paper, we focus exclusively on ODE models of the form,

$$\dot{x}(t) = f(x(t), q), \tag{1.1}$$

where $x(t)$ denotes the n-dimensional state vector and q the m-dimensional parameter vector. In systems biology modeling, one rarely has a detailed knowledge of the parameter values and sometimes even the knowledge of the network topology is incomplete. However, there often exist a direct relation between the qualitative dynamics of system (1.1) as captured in the bifurcation diagram with the cell physiology [33]. In this paper, we examine an inverse problem associated with the qualitative dynamics of (1.1) which commonly arise at the initial stages of modeling gene regulatory networks, namely: given a set of genes as well as their known and hypothetical interactions, can the network be bistable or oscillatory for some choice of parameter values? If so, what are the minimal sets of parameters that one could vary in order to obtain these dynamical phenotypes? Such questions are of practical relevance to the modelers, who may wish to explore or eliminate different hypothetical reaction network topologies and mechanisms. It needs to be emphasized that the dependence of qualitative dynamics on the choice of parameters cannot be neglected: it has been shown that even with a fixed reaction network topology and biochemical mechanism, different choices of parameters can result in oscillators of various qualitative types [7]. At the initial modeling stage, there is no prior knowledge on the influence of its parameters and inverse eigenvalue analysis could be a useful first step by bringing the system to a relevant parameter regime where a more complete picture of its dynamical characteristics would be obtained by carrying out a (forward) bifurcation analysis [19] followed by possible inverse bifurcation analysis [22, 23].

This paper is organized as follows: In Section 2, we describe the underlying methodology and the proposed lift-and-project algorithm. In particular, we discuss in detail the matrix inverse eigenvalue problem that arises and give an illustrative example using a MAP kinase model. In Section 3, we demonstrate the proposed algorithm by applying to a number of systems biology models.The ODE systems for the numerical examples are given in the Appendix.

2 Methodology and algorithms

Let (df/dx) be the Jacobian matrix of the ODE system (1.1). Motivated by such a need to explore qualitative dynamics of gene networks, a computational method has been developed in the systems biology context [5] to find parameters that give rise to limit-point (LP) and Hopf (H) bifurcations [19], which satisfy the spectral conditions $0 \in \sigma(df/dx)$ and $\sigma(df/dx) \supset \{\pm i\omega\}$ respectively.

A minimization problem is formulated, with the objective being functions of eigenvalues of (df/dx) whose minimum are attained by limit-point and Hopf bifurcations [5]. To locate parameters that could bring about the corresponding bifurcations, genetic algorithms are applied. The proposed method has been applied to a number

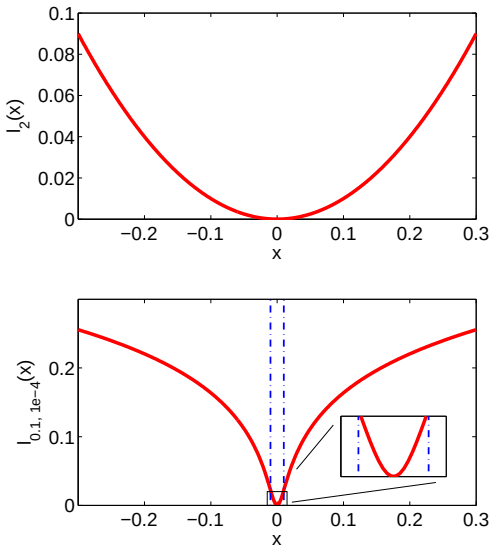


Figure 1: Comparison of standard l_2 with the sparsity-promoting regularization term.

of ODE models known to exhibit bistability or oscillations, where the algorithm has been demonstrated to find parameters resulting in the desired qualitative behaviors from random initial parameter values.

Our focus differs somewhat from [5] in that we not only wish to explore the possibility for bifurcations but also identify (in a stable manner) the important biochemical mechanisms that could give rise to the specified changes in the dynamics. Hence, in solving these inverse eigenvalue problems we consider *regularization strategies* [11]. While stabilizing ill-posed problems, regularization strategies also bring a bias to the identified solutions. In biological applications, one typically wish to identify the important governing mechanisms, out of the possibly many alternatives. For instance, in analyzing gene expression data, sparse principal component analysis has been proposed as way to attribute clustering to a few genes, hence making biological interpretations easier [8] . In our case, we wish to identify a few important "knobs" that control the qualitative dynamical features of the system. The use of regularization terms that promote the sparsity of identified solution has been studied and used in the field of inverse problems [9, 27]. Here, we employ a non-convex, sparsity-promoting regularization function as has been previously used in analyzing inverse bifurcation problems [22]. In particular, we consider the following $l_{p,\epsilon}$-functional for $0 \leq p < 1$,

$$l_{p,\epsilon}(x) = \sum_i (x_i^2 + \epsilon)^{\frac{p}{2}}, \tag{2.1}$$

where $0<\epsilon\ll 1$ ensures differentiability of the function. For $p<1$, the function is clearly non-convex; in fact, as illustrated in Fig. 2, it is only convex within the small box $\{x : |x_i| \leq \sqrt{\epsilon/(1-p)}\}$. In the numerical examples, we choose the value of ϵ=0.01^2,

hence any parameter that changes >1% lies within the concave region of the penalty function and could be classified as having been identified by the algorithm.

While there is ongoing work on developing a mathematical theory for its regularization properties of such $l_{p,\epsilon}$ functional [37], it has been computationally shown to be effective for obtaining sparse solutions in the context of inverse bifurcation problems [22].

2.1 Lift-and-project iterations

For the above mentioned class of inverse eigenvalue problems, we propose an algorithm based on carrying out a series of alternating Lift-and-Project iterations. This iterative solution method for finding the common intersection of sets has been analyzed in the convex setting [4] but has also found applications in non-convex settings including pole placement [35] as well as in solving least squares formulation of partially-prescribed inverse eigenvalue problems [6]. We believe that the proposed strategy of dividing the problem into lift and projection steps while using as much structural information of the underlying reaction network as possible, can provide an attractive alternative to the existing method of globally minimizing an objective function involving the eigenvalues of the Jacobian matrix [5], the computational effort of which could become intensive in nonlinear problems of high dimensions.

Here, we consider the case of trying to locate a Hopf bifurcation with oscillation frequency ω. In the proposed Algorithm 2.1, the first step consists of deriving linear constraints on the lifting matrix $\tilde{A}$: entry-wise bounds of the form

$$LB_{ij} \leq \tilde{A}_{ij} \leq UB_{ij},$$

and linear inequality constraints of the form $C : \tilde{A} \geq 0$; note that this bounding procedure is done only once, the result of which is subsequently used in all the iterative steps. These upper and lower bounds are derived via algebraic computation and the linear constraints are obtained by identifying relationships between Jacobian entries; this is illustrated by a specific example in Section 2.2.1. The main part of Algorithm 2.1 consists of the Lift-and-Project steps: the Lifting step entails the solution of an matrix inverse eigenvalue problem, which is formulated as a bilinear program involving the norm

$$\|x\|_{l_1} = \sum_i |x_i|,$$

and is described in detail in Section 2.2. The Projection step is formulated as a nonlinear, constrained optimization problem where one tries to find parameter and state solution, q and x, such that the mismatch of the Jacobian matrix (df/dx) to the lifted matrix $\tilde{A}$ is minimized while satisfying the equilibrium constraint $f(x,q)=0$. To regularize the inverse problem, a sparsity-promoting penalty term is added, $\mu l_{p,\epsilon}\big((q - q^*)/q^*\big)$ where μ is the regularization parameter, which has the effect of identifying parameter solutions that entail changing as few entries as possible from the nominal values of q^*. We note that the *a-priori* derived bounds and linear constraints placed on

$\tilde{A}$ go towards narrowing the possible gap between the lifted matrix and the space of Jacobian matrices that can be realized by the system under consideration.

Algorithm: Find Oscillations (frequency ω, parameter box $q_L \leq q \leq q_U$)

Step 1. Obtain *a-priori* computable bounds and linear constraints on the matrix entries:

$$\{LB, UB, C\} \leftarrow \text{DeriveConstraints}\Big(\big(\frac{df}{dx}\big), q_L \leq q \leq q_U\Big).$$

Step 2. Iterate:

(1) *Lifting step*: for details, refer to Section 2.2

$$A \leftarrow \left(\frac{df}{dx}\right), \qquad \tilde{A} \leftarrow \arg\min_{\tilde{A}, v} \left\|A - \tilde{A}\right\|_{l_1},$$
$$\text{s.t.} \quad \begin{pmatrix} \tilde{A} - \omega I \\ \omega I \tilde{A} \end{pmatrix} \cdot v = 0, \qquad LB_{ij} \leq \tilde{A}_{ij} \leq UB_{ij}, \qquad C : \tilde{A} \geq 0, \text{ etc...}$$

(2) *Projection step*:

$$q \leftarrow \arg\min_{x, q} \left\| \left(\frac{df}{dx}\right) - A \right\|_{l_1} + \mu l_{p,\epsilon} \left(\frac{q - q^*}{q^*}\right),$$
$$\text{s.t.} \quad f(x, q) = 0.$$

The presented algorithm has been implemented in *Mathematica* [28] as an add-on package to `MathSBML` [30], which reads and simulates models specified in the Systems Biology Markup Language (SBML) format. The bilinear programming problem in the Lifting step is solved via either the interior point method provided as a built-in routine of *Mathematica*, or the branch-and-bound algorithm of MathOptimizer Professional [15]; the nonlinear optimization problem that arise in the Projection Step is solved with the interior point algorithm of *Mathematica*.

2.2 Matrix inverse eigenvalue problem

Over the years, many types of inverse eigenvalue problems have been studied: issues with regard to the existence and uniqueness of solutions have been addressed and many algorithms have been developed; for a general overview, refer to the book by Chu and Golub [6]. There continues to be work developing algorithms tailored to the various situations that arise in applications. In [13], the authors describe a method for solving structured inverse eigenvalue problems under the assumption that a few eigenpairs are completely prescribed. While the assumption is satisfied for problems of model updating, where a number of oscillatory modes as well as their associated frequencies are observed, for application in the qualitative analysis of biological models this underlying assumption is not valid since the oscillatory modes are typically not known *a-priori*. Alternatively, if the complete spectrum of the matrix is known or specified but no eigenvector conditions are imposed, there also exist algorithms for

these problems [16]. However, this assumption is also in vast contrast to our application, where we would like to impose only a small number of eigenvalue conditions (or the co-dimension of bifurcation) ranging from 1 up to 4: saddle-node and Hopf bifurcations have co-dimension 1, Bogdanov-Takens and Double Hopf have co-dimension 2 [19].

In the field of controls, algorithms have also been proposed for the related, so-called pole placement problems [14,18,25,36], which arise from the need for the stabilization of dynamic systems. Such pole placement problems are typically of the form: given the state space matrices $A\in\mathcal{R}^n, B\in\mathcal{R}^{n\times m}$, find the static output feedback matrix $K\in\mathcal{R}^m$ such that spectrum of $A+BKB^T$ lies in some desired, closed convex sets, $\lambda_i(A+BKB^T)\in\mathcal{C}_i$. For instance, in [35] the following problem is considered: the location of a complex conjugate pair of eigenvalues in the left half plane is specified, $\mathcal{C}_1=\{c\}, \mathcal{C}_2=\{\bar{c}\}$, while requiring the remaining eigenvalues to lie within some convex domain $\mathcal{C}_3=\cdots=\mathcal{C}_n=\mathcal{C}$. For solving such problems, iterative algorithms based on alternating projection have been developed [35].

Although many algorithms for various problem classes have been proposed in the fields of control and model identification, inverse eigenvalue problems that arise in systems biology applications differ in a number of ways. Firstly, since the goal is to infer important mechanisms that can give rise to different dynamics, we would like to find sparse solutions: namely, those that involve changing as few entries of the input matrix as possible. This motivates the use of l_1 minimization, rather than l_2 based formulation considered in almost all inverse eigenvalues so far. We note that l_1 based formulation has found applications in fields such as image processing [9] and compressive sensing [10]. Some theoretical results under restricted assumptions are available on the effectiveness of l_1 minimization for sparsity: for instance, it has been proved that under linear constraints, the minimum l_1 objective can recover the true solution provided it is sufficiently sparse [10]. Other than relying on a sparsity-promoting objective, in order to narrow down the set of candidate matrices we utilize as much structural information of the biochemical model as possible. For instance, in many biological applications the entries of the Jacobian matrix are often rational functions of the state variables. Hence, if one is searching over bounded parameter boxes and if all values of chemical concentrations are assumed to be positive, non-trivial entry-wise lower and upper bounds of the form

$$\tilde{A}_{ij}\geq LB_{ij}, \qquad \tilde{A}_{ij}\leq UB_{ij},$$

can be computed via algebraic computation algorithms, for instance the cylindrical algebraic decomposition [3]; we illustrate this method via a numerical example in Section 2.2.1. Another useful piece of structural information which one can use is that there is typically a reaction network graph underlying the ODE system, namely which chemical species take part in what reactions [29]. By using patterns arising from the graph structure, linear equalities and inequalities between matrix entries of the form $C:\tilde{A}\geq 0$ can be obtained. For instance, from chemical conservation laws stating the sum of certain species is constant in time, one can derive that row sums of certain

Jacobian entries should be zero. An algorithm has been implemented in *Mathematica* that extract such linear relational bounds from the Jacobian entries; for an illustration, refer to Section 2.2.1.

We formulate the eigenvalue conditions as bilinear constraints involving the matrix to be identified, $\tilde{A}$, as well as the singular vector v: for the case of limit-point and Hopf bifurcations, the constraints are

- Limit-point bifurcation: $\tilde{A} \cdot v = 0$
- Hopf bifurcation: $\begin{pmatrix} \tilde{A} & -\omega I \\ \omega I & \tilde{A} \end{pmatrix} \cdot v = 0,$

where ω specifies the frequency of oscillation that one seeks; it could also be freed as a variable while preserving the bilinear problem structure. With the constraints stated above, to locate Hopf bifurcation for instance, we solve the following bilinear program: given the input matrix A, the *a-priori* computed lower and upper bounds on matrix entries LB, UB as well as the linear inequality constraints on entries via the constraint matrix C, we minimize the l_1 norm of mis-fit between A and $\tilde{A}$ via the auxiliary variables t_{ij},

$$\begin{aligned} &\min_{\tilde{A}\in\mathcal{R}^{n\times n},\ t\in\mathcal{R}^{n\times n},\ v\in\mathcal{R}^n,\|v\|=1} \sum_{i,j} t_{ij}, \\ s.t.\quad & -t_{ij} \le A_{ij} - \tilde{A}_{ij} \le t_{ij}, \\ & \begin{pmatrix} \tilde{A} & -\omega I \\ \omega I & \tilde{A} \end{pmatrix} \cdot v = 0, \\ & \tilde{A}_{ij} \le UB_{ij}, \quad \forall ij \in I_{UB}, \\ & \tilde{A}_{ij} \ge LB_{ij}, \quad \forall ij \in I_{LB}, \\ & C : \tilde{A} \ge 0. \end{aligned} \tag{2.2}$$

In some applications, in addition to the constraints as stated in (2.2), linear constraints of the form $K \cdot v \ge 0$ for some matrix K may be placed on the critical eigenvector v. As we illustate via biological examples in Section 3, these constraints may arise from wishing to specify desired phase relationships in the oscillatory species or the type of switching behavior in genes.

2.2.1 Numerical example: MAP Kinase (MAPK) cascade

In this example, we look at the Mitogen-Activated Protein Kinase (MAPK) cascade, involved in signal transduction, relaying extracellular stimuli from the cell membrane to cytoplasm and nucleus [17]. The MAPK pathway consists of several levels, where the *kinase* at each level transfers phosphate groups to the kinase that lies at the subsequent level. More specifically, the phosphorylated, active form of the kinase at each level phosphorylates the kinase at the next level down the cascade, thereby turning it active. For instance, MAPK is phosphorylated to MAPK-P by the doubly-phosphorylated form of MAPK kinase (MKK-PP). The schematic of the model described in [17] is given in Fig. 2. Note that the doubly phosphorylated MAPK-PP negatively feedbacks to the

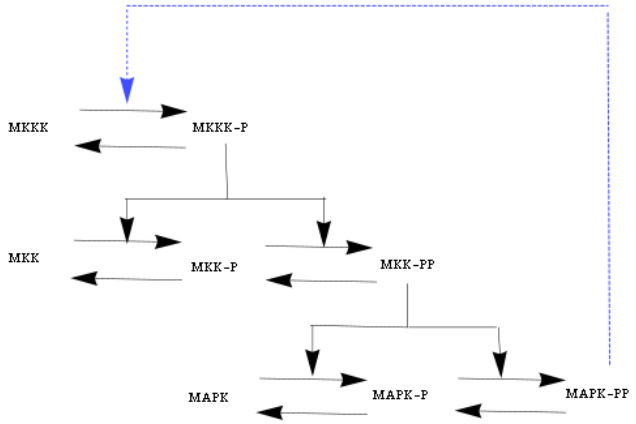


Figure 2: Schematic of MAP kinase model.

phosphorylation of MKKK; this feedback is shown as a blue arrow in Fig. 2. The ultrasensitivity brought about by the cascade together with the negative feedback can bring about oscillations. In [17], Michaelis-Menten rate rule is used in the modeling of phosphorylation steps, leading to the following ODE system [17]:

$$
\begin{aligned}
\mathrm{MAPK}'(t) &= \frac{V_{10}\mathrm{MAPK}_P(t)}{\mathrm{KK}_{10}+\mathrm{MAPK}_P(t)} - \frac{k_7\mathrm{MAPK}(t)\mathrm{MKK}_{\mathrm{PP}}(t)}{\mathrm{KK}_7+\mathrm{MAPK}(t)},\\
\mathrm{MAPK}'_P(t) &= -\frac{k_8\mathrm{MAPK}_P(t)\mathrm{MKK}_{\mathrm{PP}}(t)}{\mathrm{KK}_8+\mathrm{MAPK}_P(t)} + \frac{k_7\mathrm{MAPK}(t)\mathrm{MKK}_{\mathrm{PP}}(t)}{\mathrm{KK}_7+\mathrm{MAPK}(t)}\\
&\quad - \frac{V_{10}\mathrm{MAPK}_P(t)}{\mathrm{KK}_{10}+\mathrm{MAPK}_P(t)} + \frac{V_9\mathrm{MAPK}_{\mathrm{PP}}(t)}{\mathrm{KK}_9+\mathrm{MAPK}_{\mathrm{PP}}(t)},\\
\mathrm{MAPK}'_{\mathrm{PP}}(t) &= \frac{k_8\mathrm{MAPK}_P(t)\mathrm{MKK}_{\mathrm{PP}}(t)}{\mathrm{KK}_8+\mathrm{MAPK}_P(t)} - \frac{V_9\mathrm{MAPK}_{\mathrm{PP}}(t)}{\mathrm{KK}_9+\mathrm{MAPK}_{\mathrm{PP}}(t)},\\
\mathrm{MKK}'(t) &= \frac{V_6\mathrm{MKK}_P(t)}{\mathrm{KK}_6+\mathrm{MKK}_P(t)} - \frac{k_3\mathrm{MKK}(t)\mathrm{MKKK}_P(t)}{\mathrm{KK}_3+\mathrm{MKK}(t)},\\
\mathrm{MKK}'_P(t) &= -\frac{k_4\mathrm{MKK}_P(t)\mathrm{MKKK}_P(t)}{\mathrm{KK}_4+\mathrm{MKK}_P(t)} + \frac{k_3\mathrm{MKK}(t)\mathrm{MKKK}_P(t)}{\mathrm{KK}_3+\mathrm{MKK}(t)}\\
&\quad - \frac{V_6\mathrm{MKK}_P(t)}{\mathrm{KK}_6+\mathrm{MKK}_P(t)} + \frac{V_5\mathrm{MKK}_{\mathrm{PP}}(t)}{\mathrm{KK}_5+\mathrm{MKK}_{\mathrm{PP}}(t)},\\
\mathrm{MKK}'_{\mathrm{PP}}(t) &= \frac{k_4\mathrm{MKK}_P(t)\mathrm{MKKK}_P(t)}{\mathrm{KK}_4+\mathrm{MKK}_P(t)} - \frac{V_5\mathrm{MKK}_{\mathrm{PP}}(t)}{\mathrm{KK}_5+\mathrm{MKK}_{\mathrm{PP}}(t)},\\
\mathrm{MKKK}'(t) &= \frac{V_2\mathrm{MKKK}_P(t)}{\mathrm{KK}_2+\mathrm{MKKK}_P(t)} - \frac{V_1\mathrm{MKKK}(t)}{(K_1+\mathrm{MKKK}(t))\left(\left(\frac{\mathrm{MAPK}_{\mathrm{PP}}(t)}{\mathrm{Ki}}\right)^n+1\right)},
\end{aligned}
$$

$$\mathrm{MKKK}_P'(t) = \frac{V_1 \mathrm{MKKK}(t)}{(K_1 + \mathrm{MKKK}(t))\left(\left(\frac{\mathrm{MAPK_{PP}}(t)}{\mathrm{Ki}}\right)^n + 1\right)} - \frac{V_2 \mathrm{MKKK}_P(t)}{\mathrm{KK}_2 + \mathrm{MKKK}_P(t)},$$

where the term describing negative feedback mechanism is shown in blue in Fig. 2. At the nominal parameter values given as,

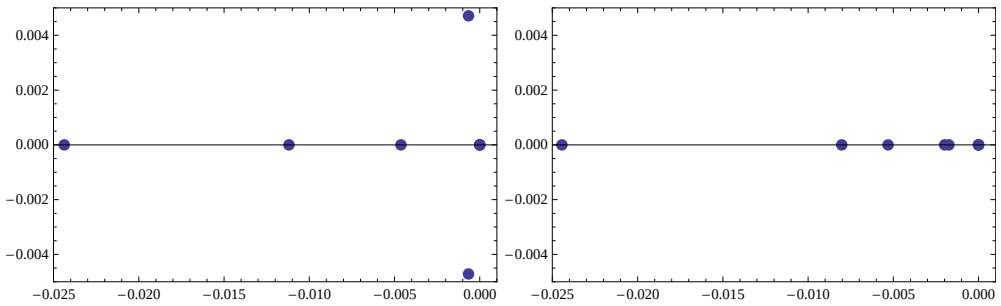
$$\begin{array}{lllll} K_1 = 10., & k_3 = 0.025, & k_4 = 0.025, & k_7 = 0.025, & k_8 = 0.025, \\ \mathrm{Ki} = 25., & \mathrm{KK}_{10} = 15., & \mathrm{KK}_2 = 8., & \mathrm{KK}_3 = 15., & \mathrm{KK}_4 = 15., \\ \mathrm{KK}_5 = 15., & \mathrm{KK}_6 = 15., & \mathrm{KK}_7 = 15., & \mathrm{KK}_8 = 15., & \mathrm{KK}_9 = 15., \\ V_1 = 0.215835, & V_{10} = 0.5, & V_2 = 0.25, & V_5 = 0.75, & V_6 = 0.75, \\ V_9 = 0.5, & & & & \end{array}$$

the Jacobian matrix at the equilibrium is,

$$A = \begin{pmatrix} -0.0176 & 0.00475 & 0 & 0 & 0 & 0 & 0 & -0.0102 \\ 0.0176 & -0.0119 & 0.0000957 & 0 & 0 & 0 & 0 & -0.00532 \\ 0 & 0.00723 & -0.0000957 & 0 & 0 & 0 & 0 & 0.0155 \\ 0 & 0 & 0 & -0.000183 & 0 & -0.0233 & 0.00218 & 0 \\ 0 & 0 & 0.000655 & 0 & -0.000301 & 0.00179 & 0 & 0 \\ 0 & 0 & -0.000655 & 0 & 0.000301 & -0.00179 & 0 & 0 \\ 0 & 0 & 0 & 0.000183 & 0 & 0.00358 & -0.00403 & 0.00545 \\ 0 & 0 & 0 & 0 & 0 & 0.0197 & 0.00185 & -0.00545 \end{pmatrix}.$$

The spectrum of the above matrix contains a pair of eigenvalues lying close to the imaginary axis, as shown in Fig. 3 (a). However, if the negative feedback is weakened (by setting Ki : 25 $\rightarrow$ 25 $\times$ 10^4 while decreasing V_1 : 2.5 $\rightarrow$ 0.2158), the system loses its oscillatory potential and all the eigevalues subsequently lie in the left-half-plane, as indicated in Fig. 3 (b). Weakening the negative feedback via the 2 parameters Ki and V_1 affects only 2 values of the Jacobian matrix, the entries (5,3) and (6,3): $\pm 0.000655 \rightarrow \pm 7 \times 10^{-7}$. This example provides us with the following matrix inverse eigenvalue problem: can we identify the matrix entries, corresponding to the indicated negative feedback, so as to bring about an imaginary pair of eigenvalues as shown in Fig. 3 (a)?

Prior to solving the stated matrix inverse eigenvalue problem, we show the linear equality and inequality constraints on the entries of the lifted matrix $\tilde{A}$ that can be



(a) With nominal negative feedback (b) Negligible negative feedback

Figure 3: Spectrum of Jacobian matrices.

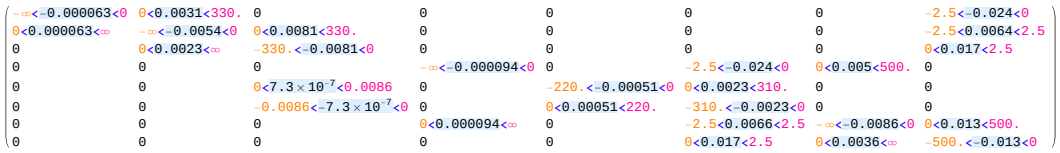


Figure 4: Computed bounds on Jacobian entries.

derived. For the rate equations given above, assuming positivity of state variables and kinetic constants lie within some bounded box, many of the Jacobian terms are bounded. For instance, let us look at the expression of Jacobian entry $(5,3)$ describing the negative feedback of MAPK-PP on the phosphorylation of MKKK,

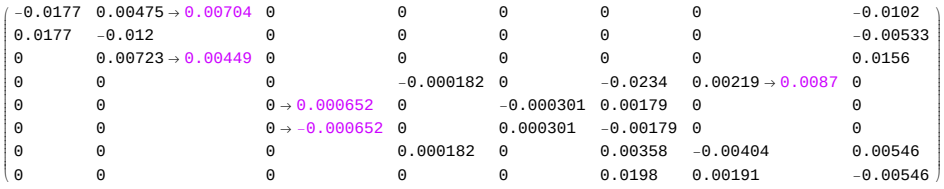
$$\left(\frac{df}{dx}\right)_{(5,3)} = \frac{\text{MKKK}(t)V_1}{\text{Ki}\left(\text{MKKK}(t)+K_1\right)\left(\frac{\text{MAPK}_{\text{PP}}(t)}{\text{Ki}}+1\right)^2}.$$

If the positivity of the chemical concentrations is assumed, $\text{MAPK}_{\text{PP}}(t) \geq 0, \text{MKKK}(t) \geq 0$, and the parameters are allowed to vary 2 orders of magnitude up and below its nominal values,

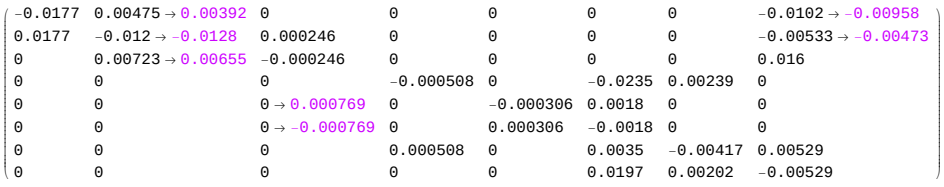
$$\frac{1}{10} \leq K_1 \leq 1000, \quad 2500 \leq \text{Ki} \leq 25000000, \quad \frac{199}{92200} \leq V_1 \leq \frac{9950}{461},$$

using algebraic computation software (algorithm based on cylindrical algebra decomposition as provided in *Mathematica* [34] is utilized in our current implementation) it is readily calculated that

$$0 \leq \frac{\text{MKKK}(t)V_1}{\text{Ki}\left(\text{MKKK}(t)+K_1\right)\left(\frac{\text{MAPK}_{\text{PP}}(t)}{\text{Ki}}+1\right)^2} \leq \frac{199}{23050}.$$



(a) Minimization of l_1-norm



(b) Minimization of l_2-norm

Figure 5: Identified matrix entries obtained by minimizing l_1 and l_2 objectives.

For the MAP kinase model under consideration, out of the 26 non-zero Jacobian entries, 22 non-trivial lower and upper bounds can be derived; see Fig. 4 for the bounds that have been derived. Furthermore, 7 equality constraints have been extracted from algebraically examining relationships between Jacobian entries. This is apparent if one looks at the numerical entries of the matrix in (2.2.1): for instance, entry $(1,1)$ is the negative that of $(2,1)$; similarly, between entries $(4,4)$ and $(7,4)$. Deriving these relational constraints entail pattern matching across Jacobian entries and can be implemented in symbolic environments such as *Mathematica*. After taking into account these linear constraints, the bilinear programming problem (2.2) is solved, the matrix obtained is shown in Fig. 5(a). The identified matrix entries are shown with an arrow in the figure: we see that 5 entries (out of the 26 non-zero entries) have been changed. This set of 5 entries includes the elements $(5,3)$ and $(6,3)$ that have been brought essentially to 0 after weakening the negative feedback as previously described. As a comparison, we also solve the problem using l_2 objective rather than the proposed l_1 and the solution is shown in Fig. 5(b): we see that 7 entries are identified. This demonstrates that the use of l_1 objective in the lifting step can result in more sparse solutions.

3 Example systems

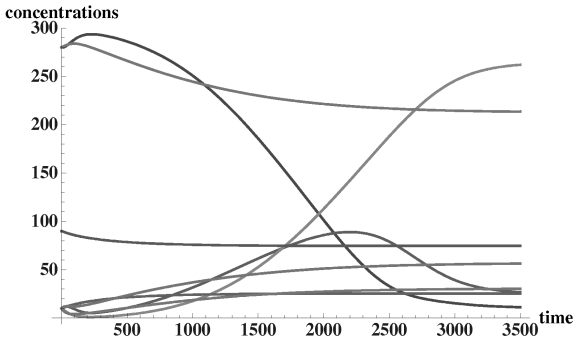
3.1 MAPK cascade

The first example that we show is the same MAP kinase model as used in Section 2.2. We start off from parameter values where the constant Ki, whose inverse describes the negative feedback strength, has been increased by 4 orders of magnitude from that of the published model [17]. The question, can the algorithm bring oscillations back into the model and identify the negative feedback as the crucial mechanism?

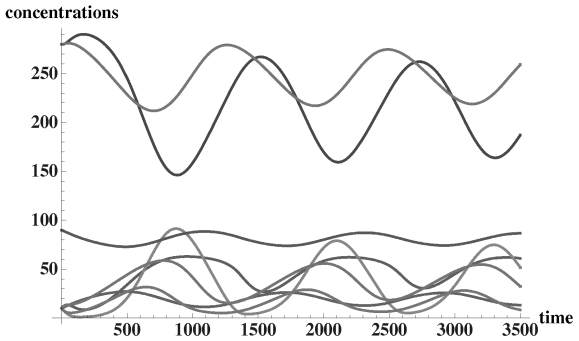
To solve this problem, we set regularization parameter μ=10^{-4}. After carrying out 8 lift-and-project iterations, an oscillatory solution is achieved by varying 7 out of the 23 parameters (defined as those changing more than 5% from their initial values), namely: Ki : $2.5\times10^5 \rightarrow 100$, KK_9 : $15 \rightarrow 5.816$, KK_4 : $15 \rightarrow 11.2$, KK_8 : $15 \rightarrow 13.76$, V_2 : $0.25 \rightarrow 0.207$, V_6 : $0.75 \rightarrow 0.660$, V_9 : $0.5 \rightarrow 0.403$. Using the parameter values as identified by the algorithm gives the results shown in Fig. 6. Clearly, the parameter that has been identified to be most crucial is that of Ki, which has been changed by more than 4 orders of magnitude. Due to sparsity-promoting penalty, out of the total number of 23 parameters in the model, 16 were essentially unchanged, thereby allowing one to narrow down the list of possible mechanisms for inducing oscillations.

3.2 Circadian rhythm

Circadian rhythms are free-running biological clocks that orchestrate the daily rhythmic activity of many organisms. Via the coupling of theory and experiments, models



(a) Using starting parameter set



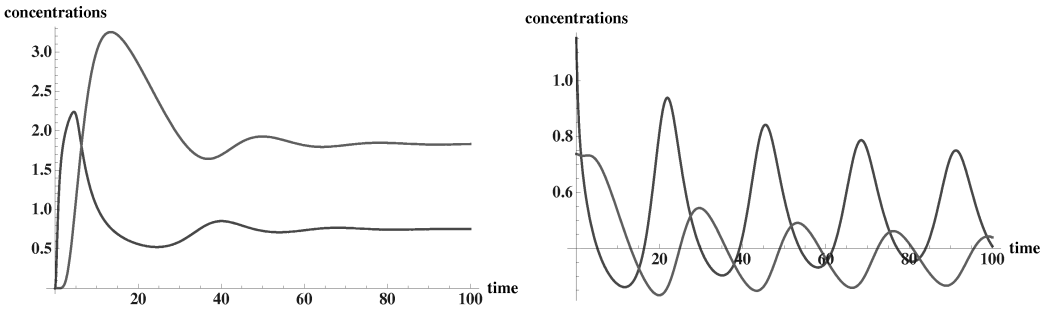
(b) Using identified parameter set

Figure 6: MAP kinase model: time-course plots.

have been built that capture qualitative aspects of these clocks in organisms ranging from *Arabidopsis* to *Drosophila* [20, 21]. In this example, we illustrate how inverse eigenvalue analysis might shed light on whether a hypothetical model could be consistent with some of the qualitative features in the data. We consider a tentative 10-dimensional ODE model with 25 parameters developed by Oxsana Sorokina [31, 32]; the full system of equations is given in the Appendix. With nominal parameter values, this model exhibit the damped oscillatory behavior shown in Fig. 7 (a). Given this solution, our question is: how to vary as few parameters as possible while getting it to oscillate? Furthermore, we would like to see if it is possible for the species rtAuc(t) and Ssn6uc(t) to oscillate with approximately π-radian phase difference between them. Hence, in addition to the eigenvalue constraint, in the lifting step we place the following additional constraints on the critical eigenvector components: for the specified phase angle difference ϕ, we impose:

$$\begin{aligned} v_{\text{rtAuc, Re}} &= \cos(\phi)v_{\text{Ssn6uc, Re}} - \sin(\phi)v_{\text{Ssn6uc, Im}}, \\ v_{\text{rtAuc, Im}} &= \sin(\phi)v_{\text{Ssn6uc, Re}} + \cos(\phi)v_{\text{Ssn6uc, Im}}. \end{aligned}$$

It turns out that the minimization algorithms attempted failed to find a solution when we requested $\phi{=}\pi$. In this case, one could try to verify the non-existence of solution to



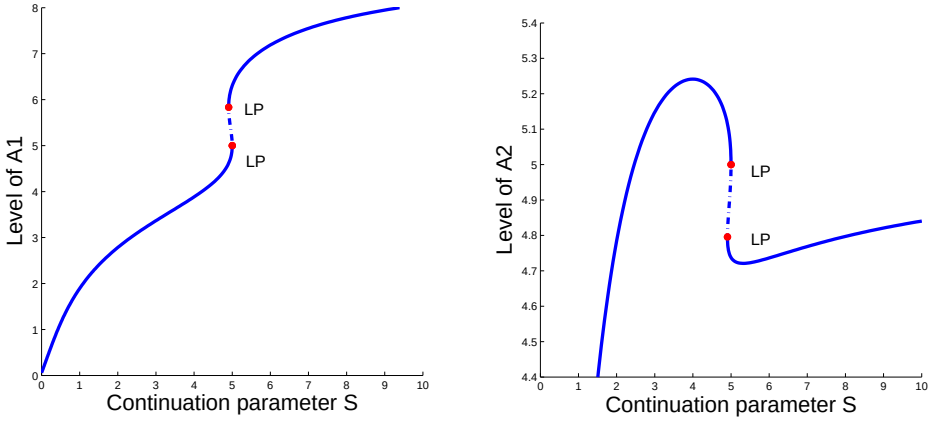
(a) Using starting parameter set (b) Using identified parameter set

Figure 7: Circadian rhythm modeling: time-course plots for species rtTAc(t) and Ssn6c(t).

the bilinear constraints via polynomial programming methods such as SOSTOOLS [26]. However, for smaller values of ϕ, solutions can be found; in particular, for ϕ=$6\pi/7$, the solution shown in Fig. 7(b) is obtained after 10 lift-and-project iterations. The algorithm shows that only 6 out of the 24 parameters in the model need to be changed in order to obtain the nearly-out-of-phase oscillation as shown.

3.3 Activator-inhibitor pair system

In the following example, we show how inverse eigenvalue analysis can be used to explore the possibility of a single model to function either as a switch or an oscillator, depending on the values of the parameters. Furthermore, we show how imposing additional constraints on the critical eigenvector can be used to computationally explore different scenarios of qualitative behaviors.



(a) Gene A_1 (b) Gene A_2

Figure 8: Reversed switching of genes A_1 and A_2 over a range of signal values S.

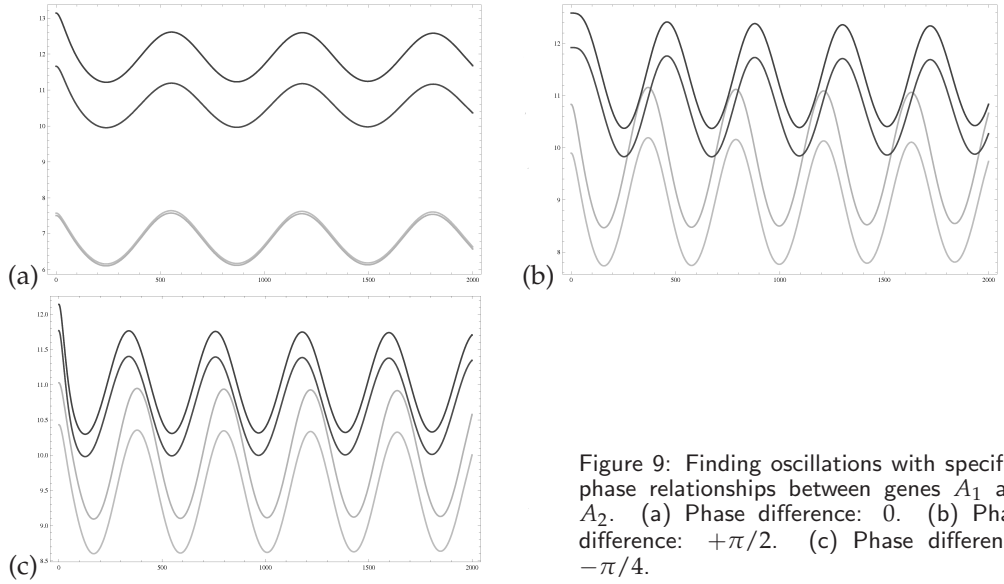


Figure 9: Finding oscillations with specified phase relationships between genes A_1 and A_2. (a) Phase difference: 0. (b) Phase difference: $+\pi/2$. (c) Phase difference: $-\pi/4$.

We consider a 14-dimensional system of equations for an activator-inhibitor gene model provided by Rainer Machné [24], shown in Appendix. The model consists of 2 genes, denoted as $A1$ and $A2$, which have opposite influences: gene $A1$ activates itself as well as $A2$, while gene $A2$ inhibits itself as well as $A1$. Due to the self-activation of $A1$, by intuitive reasoning one might correctly guess that there are parameter combinations such that the system undergoes a limit-point bifurcation, where both genes $A1$ and $A2$ undergo a discontinuous jump upwards as the input signal S increases over a certain threshold. Instead, we ask a question that is difficult to be addressed by intuition alone: are there parameter values such that $A1$ jumps up but $A2$ jumps down as S increases over a given threshold? Hence, we solve the inverse eigenvalue problem with the following critical eigenvector constraint,

$$v_{\mathrm{A1}} = -v_{\mathrm{A2}}.$$

The algorithm identifies a change of 8 out of the 44 parameters, giving rise to the solution with bifurcation diagram shown in Fig. 8: we see that while for most values of the signal the level of $A2$ increases with S, at the switching location (or limit-point on the bifurcation diagram) one can in fact have the opposite occurring. Thus, the algorithm can be used to bring the system to exhibit behaviors which might prove to be rather counter-intuitive.

Owing to the inhibitory effect of gene $A2$, it would seem that oscillations could be possible with this model. Indeed, oscillations can occur with various phase differences between the genes $A1$ and $A2$. By imposing additional eigenvector constraints as shown in Section 3.2, the algorithm identifies parameter values giving rise to these different possibilities. Fig. 8 shows solutions obtained with various phase differences imposed, from $A1$ and $A2$ being in-phase, ϕ=0, to out of phase by ϕ=$+\pi/2$ and

$\phi=-\pi/4$. It turns out the matrix inverse eigenvalue problem fails to find a solution for $\phi=-\pi/2$, but other methods of analysis would be necessary to confirm this finding. To sum up, we have computationally found parameter regimes that give rise to a variety of qualitative behaviors, which helps to develop insight into the model and could be used in experimental verifications.

4 Conclusions

A new algorithm is proposed for tackling qualitative inverse problems that arise in systems biology modeling. We have shown that the proposed method relying on l_1-based matrix inverse eigenvalue problem and l_p sparsity-promoting penalty can be successful in identifying sparse solutions that could be much more easily interpreted in the biological context. While further work regarding stability analysis awaits and several algorithmic extensions are possible, the proposed method has been shown to be promising in allowing one to computationally address the type of questions that arise in modeling gene networks.

Acknowledgments

The authors would like to acknowledge Vienna Science and Technology Fund (WWTF: MA07) for the funding support. We appreciate collaborators Rainer Machné of the University of Vienna as well as Oxsana Sorokina, Andrew Millar of the University of Edinburgh, for contributing the circadian rhythm model and originating the modeling questions that motivated the demonstrated inverse problems.

Appendix

A1. Circadian rhythm model system

$$
\begin{aligned}
&\mathrm{Ace1Cu}'(t) = \mathrm{CTKCp}(\mathrm{Ace1tot} - \mathrm{Ace1Cu}(t)) - \mathrm{KCdAce1Cu}(t),\\
&\mathrm{rrtTA}'(t) = \frac{\mathrm{nrtTAAce1Cu}(t)^4}{\left(\mathrm{grtTA}^4 + \mathrm{Ace1Cu}(t)^4\right)\left(\frac{\mathrm{Ssn6n}(t)^4}{\mathrm{Ki}^4} + 1\right)} - \mathrm{tTArrtTA}(t),\\
&\mathrm{rSsn6}'(t) = \frac{\mathrm{nSsn6rA2d}(t)}{\mathrm{gSsn6} + \mathrm{rA2d}(t)} - \mathrm{msn6rSsn6}(t),\\
&\mathrm{rtTAc}'(t) = -\mathrm{KdcrtTAc}(t)^2 + \mathrm{prtTArrtTA}(t) + \mathrm{KddrtTAdimc}(t),\\
&\mathrm{rA2d}'(t) = \mathrm{dotKdocrAd}(t) - \mathrm{kdodrA2d}(t),\\
&\mathrm{rtTAdimc}'(t) = \mathrm{KdcrtTAc}(t)^2 + \mathrm{K2rA}(t) - \mathrm{K1rtTAdimc}(t) - \mathrm{KddrtTAdimc}(t),\\
&\mathrm{rAd}'(t) = \mathrm{dotKdocrA}(t) + \mathrm{kdodrA2d}(t) - \mathrm{dotKdocrAd}(t) - \mathrm{kdodrAd}(t),\\
&\mathrm{rA}'(t) = -\mathrm{K2rA}(t) - \mathrm{dotKdocrA}(t) - \mathrm{mTArA}(t) + \mathrm{kdodrAd}(t) + \mathrm{K1rtTAdimc}(t),\\
&\mathrm{Ssn6c}'(t) = \mathrm{pSsn6rSsn6}(t) - \mathrm{K3Ssn6c}(t) + \mathrm{K4Ssn6n}(t),
\end{aligned}
$$

$$\begin{aligned}
&\mathrm{Ssn6c}'(t) = \mathrm{pSsn6rSsn6}(t) - \mathrm{K3Ssn6c}(t) + \mathrm{K4Ssn6n}(t),\\
&\mathrm{Ssn6n}'(t) = \mathrm{K3Ssn6c}(t) - \mathrm{K4Ssn6n}(t) - \mathrm{mSsn6nSsn6n}(t).
\end{aligned}$$

A2. Activator-inhibitor pair model system

$$\begin{aligned}
&\mathrm{A1}'(t) = \frac{1}{3}\left(\mathrm{A1c}(t)\mathrm{kim}_{\mathrm{A1}} - \mathrm{A1}(t)\mathrm{kex}_{\mathrm{A1}}\right) - \mathrm{A1}(t)D_{\mathrm{A1}},\\
&\mathrm{A1c}'(t) = -\mathrm{A1c}(t)D_{\mathrm{A1c}} - U\mathrm{A1c}(t)\mathrm{kass}_{\mathrm{C1U}} + \mathrm{C1}(t)\mathrm{kdiss}_{\mathrm{C1U}}\\
&\qquad\qquad + \frac{1}{24}\left(\mathrm{A1}(t)\mathrm{kex}_{\mathrm{A1}} - \mathrm{A1c}(t)\mathrm{kim}_{\mathrm{A1}}\right) + \mathrm{mA1}(t)\mathrm{ktl}_{\mathrm{a1}},\\
&\mathrm{A2}'(t) = \frac{1}{3}\left(\mathrm{A2c}(t)\mathrm{kim}_{\mathrm{A2}} - \mathrm{A2}(t)\mathrm{kex}_{\mathrm{A2}}\right) - \mathrm{A2}(t)D_{\mathrm{A2}},\\
&\mathrm{A2c}'(t) = -\mathrm{A2c}(t)D_{\mathrm{A2c}} - U\mathrm{A2c}(t)\mathrm{kass}_{\mathrm{C2U}} + \mathrm{C2}(t)\mathrm{kdiss}_{\mathrm{C2U}}\\
&\qquad\qquad + \frac{1}{24}\left(\mathrm{A2}(t)\mathrm{kex}_{\mathrm{A2}} - \mathrm{A2c}(t)\mathrm{kim}_{\mathrm{A2}}\right) + \mathrm{mA2}(t)\mathrm{ktl}_{\mathrm{a2}},\\
&\mathrm{C1}'(t) = -\mathrm{C1}(t)D_{\mathrm{C1U}} + U\mathrm{A1c}(t)\mathrm{kass}_{\mathrm{C1U}} - \mathrm{C1}(t)\mathrm{kdiss}_{\mathrm{C1U}},\\
&\mathrm{C2}'(t) = -\mathrm{C2}(t)D_{\mathrm{C2U}} + U\mathrm{A2c}(t)\mathrm{kass}_{\mathrm{C2U}} - \mathrm{C2}(t)\mathrm{kdiss}_{\mathrm{C2U}},\\
&\mathrm{GLN}'(t) = -\mathrm{GLN}(t)D_{\mathrm{GLN}} + \frac{\mathrm{kim}_{\mathrm{GLN}}}{2} + \frac{M(t)\mathrm{Vmax}_{\mathrm{GLN}}}{M(t)+K_M},\\
&M'(t) = \mathrm{mM}(t)\mathrm{ktl}_m - M(t)D_M,\\
&\mathrm{mA1}'(t) = \frac{1}{24}\mathrm{rA1}(t)\mathrm{kex}_{\mathrm{a1}} - \frac{\mathrm{mA1}(t)\left(\mathrm{GLN}(t)D_{\mathrm{C1G}} + \frac{D_{\mathrm{mA1}}\mathrm{kdiss}_{\mathrm{C1G}}}{\mathrm{kass}_{\mathrm{C1G}}}\right)}{\mathrm{GLN}(t) + \frac{\mathrm{kdiss}_{\mathrm{C1G}}}{\mathrm{kass}_{\mathrm{C1G}}}},\\
&\mathrm{mA2}'(t) = \frac{1}{24}\mathrm{rA2}(t)\mathrm{kex}_{\mathrm{a2}} - \frac{\mathrm{mA2}(t)\left(\mathrm{GLN}(t)D_{\mathrm{C2G}} + \frac{D_{\mathrm{mA2}}\mathrm{kdiss}_{\mathrm{C2G}}}{\mathrm{kass}_{\mathrm{C2G}}}\right)}{\mathrm{GLN}(t) + \frac{\mathrm{kdiss}_{\mathrm{C2G}}}{\mathrm{kass}_{\mathrm{C2G}}}},\\
&\mathrm{mM}'(t) = \frac{1}{24}\mathrm{rM}(t)\mathrm{kex}_m - \mathrm{mM}(t)D_{\mathrm{mM}},\\
&\mathrm{rA1}'(t) = \frac{\mathrm{Va1}_A\left(\frac{\mathrm{A1}(t)}{\mathrm{ba1bA1Ka}_A} + \frac{\mathrm{actA2}(t)}{\mathrm{ba1bA2Ka}_A}\right)^2}{3\left(\frac{\mathrm{A1}(t)}{\mathrm{ba1bA1Ka}_A} + \frac{\mathrm{A2}(t)}{\mathrm{ba1bA2Ka}_A} + 1\right)^2} - \mathrm{rA1}(t)D_{\mathrm{rA1}}\\
&\qquad\qquad - \frac{1}{3}\mathrm{rA1}(t)\mathrm{kex}_{\mathrm{a1}} + \frac{\mathrm{Va1}_b}{3} + \frac{S\mathrm{Va1}_S}{3\left(S + \mathrm{ba1sKa}_S\right)},\\
&\mathrm{rA2}'(t) = \frac{\mathrm{Va2}_A\left(\frac{\mathrm{A1}(t)}{\mathrm{bA1ba2Ka}_A} + \frac{\mathrm{actA2}(t)}{\mathrm{ba2bA2Ka}_A}\right)^2}{3\left(\frac{\mathrm{A1}(t)}{\mathrm{bA1ba2Ka}_A} + \frac{\mathrm{A2}(t)}{\mathrm{ba2bA2Ka}_A} + 1\right)^2} - \mathrm{rA2}(t)D_{\mathrm{rA2}}\\
&\qquad\qquad - \frac{1}{3}\mathrm{rA2}(t)\mathrm{kex}_{\mathrm{a2}} + \frac{\mathrm{Va2}_b}{3} + \frac{S\mathrm{Va2}_S}{3\left(S + \mathrm{ba2sKa}_S\right)},\\
&\mathrm{rM}'(t) = \frac{\mathrm{Vm}_A\left(\frac{\mathrm{A1}(t)}{\mathrm{Km}_{\mathrm{A1}}} + \frac{\mathrm{actA2}(t)}{\mathrm{Km}_{\mathrm{A2}}}\right)^2}{3\left(\frac{\mathrm{A1}(t)}{\mathrm{Km}_{\mathrm{A1}}} + \frac{\mathrm{A2}(t)}{\mathrm{Km}_{\mathrm{A2}}} + 1\right)^2} - \mathrm{rM}(t)D_{\mathrm{rM}} - \frac{1}{3}\mathrm{rM}(t)\mathrm{kex}_m.
\end{aligned}$$

References

[1] Lilia Alberghina and Hans V. Westerhoff, editors. *Systems biology: definitions and perspectives*, volume 13 of *Topics in Current Genetics*. Springer, Berlin, 2005.

[2] Uri Alon. *An introduction to systems biology*. Chapman & Hall/CRC Mathematical and Computational Biology Series. Chapman & Hall/CRC, Boca Raton, FL, 2007. Design principles of biological circuits.

[3] Saugata Basu, Richard Pollack, and Marie-Françoise Roy. *Algorithms in real algebraic geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, second edition, 2006.

[4] Heinz H. Bauschke, Frank Deutsch, Hein Hundal, and Sung-Ho Park. Accelerating the convergence of the method of alternating projections. *Trans. Amer. Math. Soc.*, 355(9):3433–3461 (electronic), 2003.

[5] V. Chickarmane, S. R. Paladugu, F. Bergmann, and H. M. Sauro. Bifurcation discovery tool. *Bioinformatics*, 21:3688–3690, Sep 2005.

[6] Moody T. Chu and Gene H. Golub. *Inverse eigenvalue problems: theory, algorithms, and applications*. Numerical Mathematics and Scientific Computation. Oxford University Press, New York, 2005.

[7] E. Conrad, A. E. Mayo, A. J. Ninfa, and D. B. Forger. Rate constants rather than biochemical mechanism determine behaviour of genetic clocks. *J R Soc Interface*, 5 Suppl 1:9–15, Aug 2008.

[8] Alexandre d'Aspremont, Laurent El Ghaoui, Michael I. Jordan, and Gert R. G. Lanckriet. A direct formulation for sparse PCA using semidefinite programming. *SIAM Rev.*, 49(3):434–448 (electronic), 2007.

[9] Ingrid Daubechies, Michel Defrise, and Christine De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Comm. Pure Appl. Math.*, 57(11):1413–1457, 2004.

[10] David L. Donoho and Jared Tanner. Sparse nonnegative solution of underdetermined linear equations by linear programming. *Proc. Natl. Acad. Sci. USA*, 102(27):9446–9451 (electronic), 2005.

[11] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of inverse problems*, volume 375 of *Mathematics and its Applications*. Kluwer Academic Publishers Group, Dordrecht, 1996.

[12] Christopher P. Fall, Eric S. Marland, John M. Wagner, and John J. Tyson, editors. *Computational cell biology*, volume 20 of *Interdisciplinary Applied Mathematics*. Springer-Verlag, New York, 2002.

[13] Sau-Lon James Hu and Haujun Li. A systematic linear space approach to solving partially described inverse eigenvalue problems. *Inverse Problems*, 24(3):035014, 13, 2008.

[14] T. S. Hu, Z. L. Lin, and J. Lam. Unified gradient approach to performance optimization under a pole assignment constraint. *J. Optim. Theory Appl.*, 121(2):361–383, 2004.

[15] Frank J. Kampas Janos D. Pinter. MathOptimizer professional. `http://www.wolfram.com/products/applications/mathoptpro/`.

[16] Xingzhi Ji. On matrix inverse eigenvalue problems. *Inverse Problems*, 14(2):275–285, 1998.

[17] B. N. Kholodenko. Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activated protein kinase cascades. *Eur. J. Biochem.*, 267:1583–1588, Mar 2000.

[18] Meeyoung Kim, Joachim Rosenthal, and Xiaochang Alex Wang. Pole placement and matrix extension problems: a common point of view. *SIAM J. Control Optim.*, 42(6):2078–2093 (electronic), 2004.

[19] Yuri A. Kuznetsov. *Elements of applied bifurcation theory*, volume 112 of *Applied Mathemat-*

ical Sciences. Springer-Verlag, New York, third edition, 2004.
[20] J. C. Leloup and A. Goldbeter. Chaos and birhythmicity in a model for circadian oscillations of the PER and TIM proteins in drosophila. *J. Theor. Biol.*, 198:445–459, Jun 1999.
[21] J. C. Locke, L. Kozma-Bognr, P. D. Gould, B. Fehr, E. Kevei, F. Nagy, M. S. Turner, A. Hall, and A. J. Millar. Experimental validation of a predicted feedback loop in the multi-oscillator clock of Arabidopsis thaliana. *Mol. Syst. Biol.*, 2:59, 2006.
[22] J. Lu, H. W. Engl, R. Machné, and P. Schuster. Inverse bifurcation analysis of a model for mammalian G_1/S regulatory module. In *BIRD '07*, volume 4414 of *Lecture Notes in Bioinformatics*, pages 168–184. Springer, 2007.
[23] J. Lu, H. W. Engl, and P. Schuster. Inverse bifurcation analysis: application to simple gene systems. *Algorithms Mol Biol*, 1:11, 2006.
[24] Rainer Machné. Private communication, 2008.
[25] Robert Orsi. Numerical methods for solving inverse eigenvalue problems for nonnegative matrices. *SIAM J. Matrix Anal. Appl.*, 28(1):190–212 (electronic), 2006.
[26] S. Prajna, A. Papachristodoulou, P. Seiler, and P. A. Parrilo. *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*, 2004.
[27] Ronny Ramlau and Gerd Teschke. Tikhonov replacement functionals for iteratively solving nonlinear operator equations. *Inverse Problems*, 21(5):1571–1592, 2005.
[28] Wolfram Research. Mathematica. `http://www.wolfram.com/products/mathematica/`.
[29] C. H. Schilling and B. O. Palsson. The underlying pathway structure of biochemical reaction networks. *Proc. Natl. Acad. Sci. U.S.A.*, 95:4193–4198, Apr 1998.
[30] B. E. Shapiro, M. Hucka, A. Finney, and J. Doyle. MathSBML: a package for manipulating SBML-based biological models. *Bioinformatics*, 20:2829–2831, Nov 2004.
[31] Oxsana Sorokina. Private communication, 2008.
[32] Oxsana Sorokina. *Understanding biological timing by modelling simple circadian clocks*. PhD thesis, University of Edinburgh, 2009.
[33] J. J. Tyson, K. Chen, and B. Novak. Network dynamics and cell physiology. *Nat. Rev. Mol. Cell Biol.*, 2:908–916, Dec 2001.
[34] Stephen Wolfram. *The Mathematica® book*. Wolfram Media, Inc., Champaign, IL, fourth edition, 1999.
[35] Kaiyang Yang and Robert Orsi. Generalized pole placement via static output feedback: a methodology based on projections. *Automatica J. IFAC*, 42(12):2143–2150, 2006.
[36] Kaiyang Yang and Robert Orsi. Static output feedback pole placement via a trust region approach. *IEEE Trans. Automat. Control*, 52(11):2146–2150, 2007.
[37] C. A. Zarzer. On Tikhonov regularization with non-convex sparsity constraints. *Inverse Problems*, 25:1–13, 2009.