Automatica 47 (2011) 1113–1122

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Inferring stable genetic networks from steady-state data☆

Michael M. Zavlanos [a,*], A. Agung Julius [b], Stephen P. Boyd [c], George J. Pappas [d]

[a] Department of Mechanical Engineering, Stevens Institute of Technology, Hoboken, NJ 07030, USA
[b] Department of Electrical, Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180, USA
[c] Department of Electrical Engineering, Stanford University, Stanford, CA 94305, USA
[d] Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104, USA

ARTICLE INFO

Article history:
Received 19 January 2010
Received in revised form
30 November 2010
Accepted 14 January 2011
Available online 3 March 2011

Keywords:
Genetic network identification
Convex optimization
Stability
Linear systems

ABSTRACT

Gene regulatory networks capture the interactions between genes and other cell substances, resulting from the fundamental biological process of transcription and translation. In some applications, the topology of the regulatory network is not known, and has to be inferred from experimental data. The experimental data consist of expression levels of the genes, which are typically measured as mRNA concentrations in micro-array experiments. In a so-called genetic perturbation experiment, small perturbations are applied to equilibrium states and the resulting changes in expression activity are measured. This paper develops novel algorithms that identify a sparse and stable genetic network that explains data obtained from noisy genetic perturbation experiments. Our identification algorithm is based on convex relaxations of the sparsity and stability constraints and can also incorporate a variety of prior knowledge of the network structure. Such knowledge can be either qualitative, specifying positive, negative or no interactions between genes, or quantitative, specifying a range of interaction strengths. Our approach is applied to both synthetic and experimental data, obtained for the SOS pathway in *Escherichia coli*, and the results show that the stability specification not only ensures consistency with the steady-state assumptions, but also significantly increases the identification performance. Since the method is based on convex optimization, it can be efficiently applied to large scale networks.

1. Introduction

Recent advances in systems biology have given rise to the need for a more systemic understanding of large scale quantitative experimental data. In particular, the use of RNA micro-arrays that enables gene expression measurements for large scale biological networks, has provided researchers with valuable data that can be used to identify gene interactions in large genetic networks. Besides promoting biological knowledge, identification of such networks is also important in drug discovery, where a systems-wide understanding of regulatory networks is crucial for identifying the targeted pathways (Schreiber, 2000).

Due to the significance of its potential applications, genetic network identification has recently received considerable attention. Depending on whether identification aims at relating the expression of a gene to the sequence motifs found in its promoter or to the expression of other genes in the cell, approaches can be characterized as *gene-to-sequence* or *gene-to-gene*, respectively (Bansal, Belcastro, Ambesi-Impiombato, & di Bernardo, 2007; Gardner & Faith, 2005). The ensemble of both classes form the so-called *genetic network identification* problem. Solution techniques can either ignore or explicitly consider the underlying gene dynamics.

Members of the former class are clustering algorithms (Amato et al., 2006; Eisen, Spellman, Brown, & Botstein, 1998) that group genes with similar expressions, due to the high probability that they are functionally, but not necessarily directly, related to each other. Alternatively, grouping of co-expressed genes may be achieved using information-theoretic methods (Steuer, Kurths, Daub, Weise, & Selbig, 2002). Both approaches, however, are restricted to identifying undirected networks and hence, lack causality. Causality may be recovered using Bayesian networks (Pe'er, Nachman, Linial, & Friedman, 2000), which can handle directed graphs. But Bayesian networks typically do not accommodate cycles and hence, cannot handle feedback motifs that are common in genetic regulatory networks. Both causality and feedback motifs are no longer an issue when the network is modeled as a set of differential equations (Amato, Cosentino, Curatola, & di Bernardo, 2007; August & Papachristodoulou, 2009;

☆ The material in this paper was partially presented at the 2008 American Control Conference, June 11–13, 2008, Seattle, Washington, USA. This paper was recommended for publication in revised form by Associate Editor Graziano Chesi under the direction of Editor Francis J. Doyle III. A preliminary version of this work can be found in Zavlanos, Julius, Boyd, and Pappas (2008).

* Corresponding author. Tel.: +1 201 261 8301; fax: +1 201 216 8315.
E-mail addresses: michael.zavlanos@stevens.edu (M.M. Zavlanos), agung@ecse.rpi.edu (A.A. Julius), boyd@stanford.edu (S.P. Boyd), pappasg@seas.upenn.edu (G.J. Pappas).

0005-1098/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2011.02.006

Bansal, Gatta, & di Bernardo, 2006; Cinquemani, Porreca, Lygeros, & Ferrari-Trecate, 2009; Gardner, di Bernardo, Lorenz, & Collins, 2003; Julius, Zavlanos, Boyd, & Pappas, 2009; Papachristodoulou & Recht, 2007; Porreca, Drulhe, de Jong, & Ferrari-Trecate, 2008; Sontag, Kiyatkin, & Kholodenko, 2004; Srividhy, Crampin, McSharry, & Schnell, 2007; Tegner, Yeung, Hasty, & Collins, 2003). Identification is then typically optimization based, while approaches depend on whether the data is obtained from steady-state measurements (Gardner et al., 2003; Julius et al., 2009; Tegner et al., 2003) or dynamic time-series (Amato et al., 2007; August & Papachristodoulou, 2009; Bansal et al., 2006; Cinquemani et al., 2009; Papachristodoulou & Recht, 2007; Porreca et al., 2008; Sontag et al., 2004; Srividhy et al., 2007). Although time-series data includes more information about the system dynamics, identification in this case is more difficult due to the high computational effort that is typically required.

The approach proposed in this paper falls under the latter class of networks modeled as differential equations and aims at obtaining a minimal model that explains given genetic perturbation data at steady-state. The minimality specification is due to the observation that biological networks exhibit loose connectivity (Arnone & Davidson, 1997; Thieffry, Huerta, Pérez-Rueda, & Collado-Vides, 1998) and in the present framework, it was first addressed in Gardner et al. (2003) in the form of *a priori* combinations of constraints on the connectivity of the network. On the other hand, the steady-state nature of the data implies stability of the underlying genetic networks, and to the best of our knowledge, this is a first attempt to formally address this specification.

To avoid the combinatorially hard nature of the problem, we employ a weighted ℓ_1 relaxation of the minimality constraint (Boyd, 0000; Candes, Wakin, & Boyd, 2008; Han, Yoon, & Cho, 2007; Hassibi, How, & Boyd, 1999; Julius et al., 2009; Tropp, 2006), which leads to much more scalable linear constraints. The convex optimization formulation in our approach is also preserved by the stability specification, which we capture by either linear or semidefinite constraints that arise from Geršgorin's and Lyapunov's theorems, respectively. Finally, we employ additional linear constraints so that our model best fits the given genetic perturbation data as well as satisfies *a priori* knowledge on the network structure. We show that in the absence of the stability specification, our approach performs well for sufficiently large data sets with low noise, while smaller and noisy data sets hinder its performance, partly due to identification of unstable networks. However, introducing the stability specification greatly improves the identification performance, and not only validates our model but also makes it promising for future research.

The rest of this paper is organized as follows. In Section 2 we describe the genetic network identification problem, while in Section 3 we develop the proposed ℓ_1 relaxation and discuss the aforementioned stability issues that could hinder its identification performance. In Section 4 we extend our algorithm to account for stability of the identified solutions. Finally, in Sections 5 and 6, we illustrate efficiency of our approach on artificial noisy data sets as well as on experimental data for the SOS pathway in *Escherichia coli*.

2. Genetic network identification

Genetic regulatory networks consisting of n genes can be modeled as n-dimensional dynamical systems (Gardner et al., 2003). In general, such models assume the form

$$\dot{\hat{x}} = f(\hat{x}, u), \tag{1}$$

where $\hat{x}(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^p$. Here $\hat{x}_i(t) \in \mathbb{R}$ denotes the transcription activity (typically measured as mRNA concentration) of gene i in the network, and u_i is the so called transcription perturbation.[1] Nonlinear genetic networks as in (1) can have multiple stable equilibria, each one typically corresponding to a phenotypical state of the system. Then, the dynamics in a neighborhood of any given equilibrium x_{eq} can be approximated by the set of linear differential equations

$$\dot{\tilde{x}} = A\tilde{x} + Bu, \tag{2}$$

where $\tilde{x} \triangleq \hat{x} - x_{eq}$ (Sontag et al., 2004). The matrix $A \in \mathbb{R}^{n\times n}$ encodes pairwise interactions between the individual genes in the network at the given equilibrium or phenotypical state, while the matrix $B \in \mathbb{R}^{n\times p}$ indicates which genes are affected by the transcriptional perturbations. Assuming the equilibrium $\tilde{x} = 0$ is stable and the perturbation u is sufficiently small and constant, the system (2) will restabilize at a new equilibrium $\tilde{x}$, at which

$$A\tilde{x} + Bu = 0. \tag{3}$$

Let m be the number of available transcription perturbations[2] and define the matrices $U = [u_1 \cdots u_m] \in \mathbb{R}^{p\times m}$ and $\tilde{X} = [\tilde{x}_1 \cdots \tilde{x}_m] \in \mathbb{R}^{n\times m}$ containing the transcription perturbations of all m experiments and their associated steady-state mRNA concentrations, respectively. Then, collecting all m experiments at steady-state, system (3) can be written as

$$A\tilde{X} + BU = 0. \tag{4}$$

Because of nonlinearity and measurement noise, the measured deviation of the mRNA concentrations can be different from the ones predicted by the linear model. If we denote these measured quantities as X, we can then write $X = \tilde{X} + \Delta X$. We then have the following relation

$$AX + BU = (A\tilde{X} + BU) + A\Delta X. \tag{5}$$

Here, $\eta \triangleq A\Delta X$ is the residual of the linear model. Finding the linear model that best fits the experimental data amounts to making η as small as possible (in some norm). Then, the network identification problem can be stated as follows.

Problem 1 (*Genetic Network Identification*)**.** Given steady-state transcription perturbation and mRNA concentration data U and X, determine the sparsest stable matrix A that results in *sufficiently small* residual η, while incorporating any *a priori* biological knowledge regarding the presence, absence, or nature of specific gene interactions.

The requirement that A is sparse is due to biological networks being sparse in nature (Arnone & Davidson, 1997; Thieffry et al., 1998), while the stability condition is necessary for the steady-state to be observed. Finally, accordance with *a priori* biological knowledge is both desired and naturally expected to result in improved identification performance.

Remark 2. Ultimately, the effect of the transcription perturbation in the model is characterized by BU, where the matrices B and U are typically unknown. However, if we assume controllable networks, i.e., networks where we can perturb each individual gene, then U can be chosen so that BU is a diagonal matrix, subject to scaling.

3. Linear programming formulation

Given any genetic network described by (2), the problem of identifying the sparsest matrix A that approximately satisfies

[1] In a transcription perturbation experiment, individual genes are over-expressed using an episomal expression plasmid. Then U can be quantified using a second strain with a reporter gene in place of the over-expressed gene on the plasmid. After the perturbation, cells grow under constant physiological conditions to the steady-state and the change in mRNA concentration, compared to cells in the same physiological conditions but unperturbed, is measured (DiBernardo, Gardner, & Collins, 2004). For large scale networks, we may assume that not all genes are affected by a given perturbation, resulting in $p \le n$.

[2] Typically, each transcription perturbation corresponds to a specific experiment.

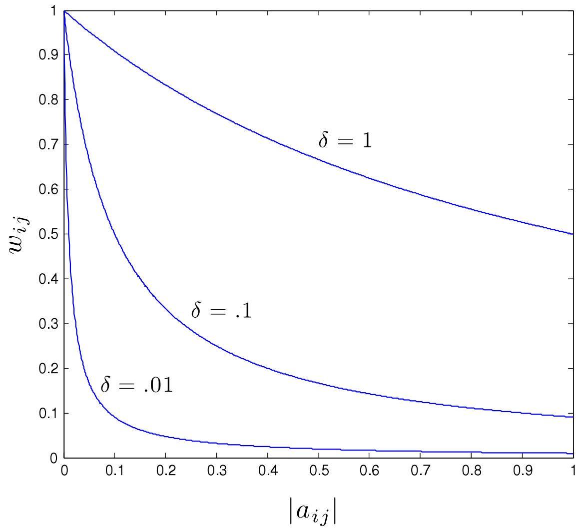

Fig. 1. Plot of the weights w_{ij} as a function of the entries $|a_{ij}|$, for different values of the parameter $\delta > 0$.

constraints (4), can be formulated as the following optimization problem

$$\begin{array}{ll} \text{minimize} & t\,\text{card}(A) + (1-t)\epsilon \\ \text{subject to} & \|AX + BU\|_1 \leq \epsilon, \quad \epsilon > 0 \end{array} \tag{6}$$

where card(A) denotes the number of nonzero entries in matrix A, and $\|A\|_1 = \sum_{i,j=1}^{n} |a_{ij}|$ denotes the (elementwise) ℓ_1 norm of a matrix A. Variables in problem (6) are the matrix A and fitting error ϵ, while the problem data are the matrices X, B, U and the parameter $0 \leq t \leq 1$, which is used to control the trade-off between sparsity, i.e., card(A), and best fit, i.e., ϵ. Note that any other norm could be used in the constraints here; we use the ℓ_1 norm since it handles outliers well.

When *a priori* knowledge about the network is also available, it is typically in the form of a partial sign pattern $S = (s_{ij}) \in \{0, +, -, ?\}^{n\times n}$, which encodes known positive interactions $(+)$, negative interactions $(-)$, no interactions (0), or no a priori knowledge regarding interactions $(?)$ between any two genes in the network. Such knowledge can be included in (6) by means of the set of linear constraints

$$A \in S \Leftrightarrow \begin{cases} a_{ij} \geq 0, & \text{if } s_{ij} = + \\ a_{ij} \leq 0, & \text{if } s_{ij} = - \\ a_{ij} = 0, & \text{if } s_{ij} = 0 \\ a_{ij} \in \mathbb{R}, & \text{if } s_{ij} = ? \end{cases} \tag{7}$$

resulting in the problem

$$\begin{array}{ll} \text{minimize} & t\ \text{card}(A) + (1-t)\epsilon \\ \text{subject to} & \|AX + BU\|_1 \leq \epsilon, \quad A \in S,\ \epsilon > 0. \end{array} \tag{8}$$

From a computational point of view, formulation (8) poses a significant challenge. Although both constraints are convex in the matrix A (Boyd & Vandenberghe, 2004), the cost function card(A) is not convex. Solving this problem globally can be done, for instance by branch-and-bound methods or directly by considering all possible 2^{n^2} sparsity patterns for A. Nevertheless, these methods are typically very slow, and cannot scale to networks with more than a handful of genes.

To obtain a method that can scale to large networks, we propose a convex relaxation of the cardinality cost function. In particular, we replace the card(A) objective with the weighted ℓ_1-norm $\sum_{i,j=1}^{n} w_{ij}|a_{ij}|$, resulting in the following *convex* program

$$\begin{array}{ll} \text{minimize} & t \displaystyle\sum_{i,j=1}^{n} w_{ij}|a_{ij}| + (1-t)\epsilon \\ \text{subject to} & \|AX + BU\|_1 \leq \epsilon, \quad A \in S,\ \epsilon > 0, \end{array} \tag{9}$$

where the weights w_{ij} are chosen such that (Fig. 1)

$$w_{ij} = \frac{\delta}{\delta + |a_{ij}|}, \quad \text{for all } i, j = 1, \ldots, n \tag{10}$$

for sufficiently small $\delta > 0$ (Boyd, 0000). The main idea behind the proposed heuristic is to uniformly initialize all weights by $w_{ij} = 1$ (this corresponds to the standard ℓ_1 relaxation of the cost function) and repeatedly solve problem (9), each time updating the weights using (10) (Algorithm 1). Then, large weights are always assigned to small matrix entries $|a_{ij}|$ and small weights to large entries $|a_{ij}|$, which can eliminate any weak genetic interactions in the final identified matrix A. In practice, Algorithm 1 requires no more than $J = 10$ iterations, regardless of the problem's size. We refer the reader to our earlier publication on this subject (Julius et al., 2009). Furthermore, recent theoretical results (Candes, Romberg, & Tao, 2006) show that, in some cases (not including the present application), minimizing the weighted ℓ_1 norm of a matrix A, in fact does minimizes card(A) with high probability.

Algorithm 1 Network ID (Ignoring Stability)

Require: Sign pattern S, experimental data X and U, and control parameter $0 \leq t \leq 1$,
Initialize weights $w_{ij} = 1$ for all $i, j = 1, \ldots, n$,
for $it = 1$ to J **do**
  Solve the linear program (9) for A and ϵ,
  Update the weights w_{ij} using (10),
end for

4. Incorporating stability

In Section 3 we developed an iterative procedure, based on the solution of linear programs, able to identify a sparse matrix that best fits possibly noisy network data, while satisfying a priori knowledge about the network. In this section, we propose two different ways of incorporating stability in Algorithm 1, both preserving its convex nature and hence, having the associated scalability and global optimality properties. Furthermore, we show that these modified approaches significantly increase the performance of our identification algorithm.

4.1. Linear approximation

Incorporating stability of the identified matrix A as a linear constraint in Algorithm 1 relies on the following theorem by Geršgorin.

Theorem 3 (*Horn & Johnson, 1985*)**.** *Let* $A = (a_{ij}) \in \mathbb{R}^{n\times n}$ *and for all* $i = 1, \ldots, n$ *define the* deleted absolute row sums *of* A *by* $R_i(A) \triangleq \sum_{j\neq i} |a_{ij}|$. *Then, all eigenvalues of* A *are located in the union of* n *discs*

$$G(A) \triangleq \bigcup_{i=1}^{n} \{z \in \mathbb{C} \mid |z - a_{ii}| \leq R_i(A)\}.$$

Furthermore, if a union of k *of these* n *discs forms a connected region that is disjoint from all the remaining* $n-k$ *discs, then there are exactly* k *eigenvalues of* A *in this region.*

The region $G(A)$ is often called the *Geršgorin region* (for the rows) of A, the individual discs in $G(A)$ are called the *Geršgorin discs*, while the boundaries of these discs are called the *Geršgorin circles*. Since A and A^T have the same eigenvalues, one can also obtain a similar Geršgorin disc theorem for the columns of A. Clearly, if

$$a_{ii} \leq -\sum_{j\neq i} |a_{ij}|, \quad \text{for all } i = 1, \ldots, n \tag{11}$$

Algorithm 2 Network ID (Geršgorin Stability)

Require: Sign pattern S, experimental data X and U, and control parameter $0 \le t \le 1$,
Initialize weights $w_{ij} = 1$ for all $i, j = 1, \ldots, n$,
for $it = 1$ to J **do**
  Solve the linear program (13) for A and ϵ,
  Update the weights w_{ij} using (10),
  Update the weights v_i using (14),
end for

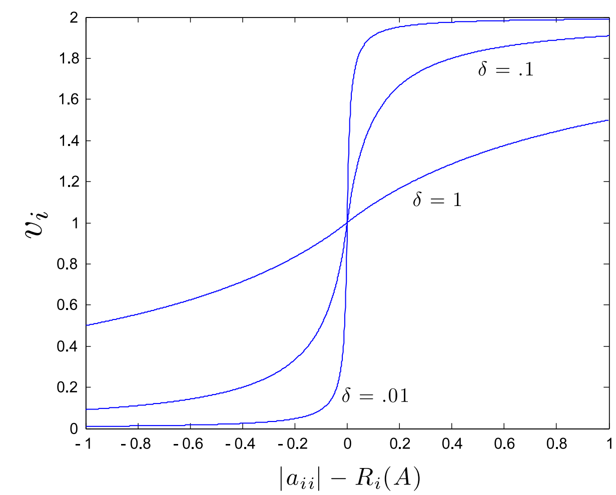

Fig. 2. Plot of the weights v_i as a function of the entries $|a_{ii}| - R_i(A)$, for average $\beta = 0$ and different values of the parameter $\delta > 0$.

then all discs $\{z \in \mathbb{C} \mid |z - a_{ii}| \le R_i(A)\}$ are in the left half plane $\mathbb{C}_-$ and Theorem 3 ensures that all eigenvalues of A are also in $\mathbb{C}_-$, which implies that A is stable. What is appealing about constraints (11) is that they are convex in the entries of A, and can be expressed as a set of linear inequalities; hence, they can be directly incorporated in the linear program (9) in Algorithm 1, rendering a stable matrix. However, constraints (11) also impose strict structural constraints on the entries of A. In particular, they restrict all diagonal entries of A to be non-positive and matrix A to be *diagonally dominant*, namely

$$|a_{ii}| \ge \sum_{j \ne i} |a_{ij}|, \quad \text{for all } i = 1, \ldots, n.$$

This later constraint can be relaxed by applying a similarity transformation on A. In particular, since $V^{-1}AV$ and A share the same eigenvalues for any invertible matrix V, we can apply Geršgorin's theorem to $V^{-1}AV$ and for a smart choice of V we can obtain sharper bounds on the eigenvalues. A particularly convenient choice is $V \triangleq \mathrm{diag}(v_1, \ldots, v_n)$, with $v_i > 0$ for all $i = 1, \ldots, n$. Then, $V^{-1}AV = (v_j a_{ij}/v_i)$ and Geršgorin's theorem states that all eigenvalues of A lie in the region

$$G(V^{-1}AV) \triangleq \bigcup_{i=1}^{n} \left\{ z \in \mathbb{C} \mid |z - a_{ii}| \le \frac{1}{v_i} \sum_{j \ne i} v_j |a_{ij}| \right\}.$$

Clearly, if we require that

$$a_{ii} \le -\frac{1}{v_i} \sum_{j \ne i} v_j |a_{ij}|, \quad i = 1, \ldots, n, \tag{12}$$

then $G(V^{-1}AV) \subset \mathbb{C}_-$, which implies that matrix A is stable, but not necessarily diagonally dominant. Constraints (12) are still linear in the entries of A and hence, can be directly incorporated in (9) resulting in the linear program

Algorithm 3 Network ID (Lyapunov Stability)

Require: Sign pattern S, experimental data X and U, and control parameter $0 \le t \le 1$,
Apply Algorithm 1 for matrix A,
if matrix A is unstable **then**
  Solve (17) for a Lyapunov matrix P,
  Initialize weights $w_{ij} = 1$ for all $i, j = 1, \ldots, n$,
  for $it = 1$ to J **do**
    Solve the semidefinite program (18) for A and ϵ,
    Update the weights w_{ij} using (10),
  end for
end if

$$\begin{aligned} \text{minimize} \quad & t \sum_{i,j=1}^{n} w_{ij} |a_{ij}| + (1 - t)\epsilon \\ \text{subject to} \quad & \|AX + BU\|_1 \le \epsilon, \quad A \in S, \ \epsilon > 0 \\ & a_{ii} \le -\frac{1}{v_i} \sum_{j \ne i} v_j |a_{ij}|, \quad i = 1, \ldots, n. \end{aligned} \tag{13}$$

The identification procedure is then described in Algorithm 2. Intuitively, the weights v_i, should penalize Geršgorin discs far in the left half plane and assign the remaining slack to discs close to (or intersecting) the imaginary axis, breaking in this way the diagonal dominance in the associated row. In particular, for $\beta \triangleq \frac{1}{n} \sum_{i=1}^{n} (|a_{ii}| - R_i(A))$ we choose the weights v_i by (Fig. 2)

$$v_i \triangleq \begin{cases} 1 + \dfrac{|a_{ii}| - R_i(A) - \beta}{\delta + (|a_{ii}| - R_i(A) - \beta)}, & \text{if } |a_{ii}| - R_i(A) > \beta \\ \dfrac{\delta}{\delta - (|a_{ii}| - R_i(A) - \beta)}, & \text{if } |a_{ii}| - R_i(A) \le \beta, \end{cases} \tag{14}$$

where $R_i(A)$ denotes the deleted absolute sum for row i, as in Theorem 3, and the quantity $|a_{ii}| - R_i(A) > 0$ indicates how far in the left half plane the associated Geršgorin disc is located. Convergence of Algorithm 2 is slower than that of Algorithm 1 and for certain ill-conditioned problem instances it may result in periodic solutions.

4.2. *Semidefinite approximation*

Let A be the matrix identified by Algorithm 1 which can possibly be unstable. The goal in this section is to characterize "small" perturbations to A that render it stable, while satisfying the desired sign pattern and maintaining its sparsity structure. For this, let $D \in \mathbb{R}^{n \times n}$ be the sought perturbation matrix and define the matrix $A' \triangleq A + D$. A necessary and sufficient condition for stability of A' is the existence of a symmetric positive definite Lyapunov matrix P such that

$$(A + D)^T P + P(A + D) \prec 0. \tag{15}$$

Letting $L \triangleq PD$, Eq. (15) becomes

$$A^T P + L^T + PA + L \prec 0, \tag{16}$$

which is a linear matrix inequality in both P and L and can be efficiently solved using semidefinite programming (Boyd & Vandenberghe, 2004). In particular, solving the following semidefinite program

$$\begin{aligned} \text{minimize} \quad & \|LX\|_2 \\ \text{subject to} \quad & A^T P + L^T + PA + L \prec 0, \quad P \succeq I, \end{aligned} \tag{17}$$

gives $D = P^{-1}L$ and the desired stable matrix A' becomes $A' = A + P^{-1}L$. This program formulation can be motivated by noticing that

$$\begin{aligned} \|(A'X + BU) - (AX + BU)\|_2 &= \|P^{-1}LX\|_2, \\ &\le \frac{\|LX\|_2}{\|P\|_2} \le \|LX\|_2, \end{aligned}$$

since $\|P\|_2 \geq 1$. Therefore, minimizing the objective $\|LX\|_2$ means minimizing an upper bound of the difference between $AX + BU$ and $A'X + BU$. Clearly, the matrix A' may no longer satisfy the desired sign pattern or sparsity specifications. Therefore, we need to further perturb A' to obtain a new matrix A (in a neighborhood of A') that is also stable. For this, we use the Lyapunov matrix P associated with A' and compute A by modifying problem (9) as

$$\begin{aligned} &\text{minimize} \quad t\sum_{i,j=1}^{n} w_{ij}|a_{ij}| + (1-t)\epsilon \\ &\text{subject to} \quad \|AX + BU\|_1 \leq \epsilon, \quad \epsilon > 0 \\ &\qquad\qquad\quad A^T P + PA \prec 0, \quad A \in S. \end{aligned} \tag{18}$$

We iterate until convergence, as in Algorithm 1. This procedure is described in Algorithm 3.

Remark 4 (*Connection to Linear Systems Theory*)**.** Assume that the left kernel $\mathcal{C}$ of the data matrix X is nontrivial, i.e., $c \triangleq \dim(\mathcal{C}) > 0$, and define a basis matrix $C \in \mathbb{R}^{c\times n}$ of $\mathcal{C}$, such that $\operatorname{rank}(C) = c$ and

$$v \in \mathcal{C} \Leftrightarrow \exists k \in \mathbb{R}^{1\times c} \quad \text{s.t. } v = kC.$$

Then, for any matrix $K \in \mathbb{R}^{n\times c}$, let $A' \triangleq A + KC$, where $K \in \mathbb{R}^{n\times c}$. Notice that $(A + KC)X + BU = AX + BU$, due to the fact that $CX = 0$. The matrix C parameterizes all models A' that result in the same residual as A. Obtaining a matrix K that renders A' stable is equivalent to the *observer design problem* in linear systems theory (Rugh, 1996). A well-known condition for the existence of such a K is the *detectability* of the pair (A, C). In particular, the pair (A, C) is called detectable if

$$\operatorname{rank}\begin{bmatrix} \lambda I - A \\ C \end{bmatrix} = n,$$

for all $\lambda \in \mathbb{C}_+$ (the closed right half plane). Then, K can be obtained by the solution of the Lyapunov equation $(A + KC)^T P + P(A + KC) \prec 0$, where P is a symmetric positive definite Lyapunov matrix. Setting $L \triangleq PK$ we get a linear matrix inequality in P and L, similar to the one in (16).

5. Synthetic data and discussion

5.1. Sensitivity to parameter selection

In this section we study how the parameter $0 \leq t \leq 1$ that regulates the tradeoff between sparsity and best fit in problems (9), (13) and (18) affects the performance of our identification methods. As the measure of performance, we use the Receiver Operating Characteristic (ROC) curve. The ROC curve, plots the sensitivity of the prediction results against (1-specificity). These quantities are given by the formula (De Muth, 2006)

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{and} \quad \text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}},$$

where T = True, F = False, P = Positives, and N = Negatives. Since, the parameter t regulates the weight put on sparsity, i.e., number of zeros, vs. best fit, the terms "Positives" and "Negatives" here refer to non-zero and zero interactions between genes, respectively.[3] The best possible prediction, will give a point in the upper left corner of the plot, representing 100% sensitivity, i.e., no false zero identifications, and 100% specificity, i.e., no false non-zero identifications. A completely random guess will give a point along the diagonal line (line of no discrimination).

To evaluate the performance of our algorithms, we created ROC plots for networks of size $n = 20$ genes with $c = 20\%$ connectivity, and for different values of sign knowledge σ, data size m, and noise levels ν. We applied our algorithms to a set of 20 *stable*, *random* and *well-conditioned* (otherwise, preconditioning would be required) interconnection matrices A that were generated to be identified. The sample matrices A were obtained as the solution of the following program:

$$\begin{aligned} &\text{minimize} \quad \|D\|_2 \\ &\text{subject to} \quad \gamma I \preceq \frac{1}{2}((\tilde{A} + D) + (\tilde{A} + D)^T) \preceq \epsilon I, \\ &\qquad\qquad\quad \Gamma < \gamma < \epsilon < E < 0, \\ &\qquad\qquad\quad D_{ij} = 0 \quad \text{if } A_{ij} = 0, \ \forall i, j = 1, \ldots, n, \end{aligned}$$

where $\tilde{A}$ is a random, not necessarily stable, interconnection matrix that satisfies a 20% sparsity specification, and D is a perturbation added to $\tilde{A}$ to obtain a stable matrix $A = \tilde{A} + D$. If the (i, j)th entry of $\tilde{A}$ is zero, so is the (i, j)th entry of D, by construction. The constants $\Gamma, E < 0$ regulate the condition number of $A = \tilde{A} + D$ (more accurately, its eigenvalues). The above optimization problem is based on the observation that an asymmetric matrix A is negative definite if and only if its symmetric part $\frac{1}{2}(A + A^T)$ is negative definite. Then, A will be stable with Lyapunov matrix I. The data sets associated with matrix A are obtained by $X = -A^{-1}BU + \nu N$, where $BU \in \mathbb{R}^{n\times m}$ is the identity matrix (see Remark 2) and $N \in \mathbb{R}^{n\times m}$ is a zero mean and unit variance normally distributed random matrix (entry-wise). All algorithms were implemented in MATLAB using the `cvx` toolbox for convex optimization problems (Grant, Boyd, & Ye, 0000) and run on an Intel Core 2 Duo 3.06 GHz processor with 8 GB RAM. For problems of size $n = 20$, each iteration of Algorithms 1, 2 and 3 took approximately 2, 5 and 8 s, respectively, while no more than 15 iterations are in general required for Algorithms 1 and 2, and 25 iterations for Algorithm 3.

Figs. 3 and 4 contain the ROC plots for parameters $\sigma = 30\%$, $\nu = 10\%$, $m = n$ (full data), and $\sigma = 0\%$, $\nu = 50\%$, $m = \lceil \frac{n}{3} \rceil$ (partial data), respectively. These cases correspond to the two "extremes" in terms of possible identification performance, i.e., many high quality vs. few low quality available data. Every point in the plots corresponds to a different value of t. As expected, high quality data gives better identifications, i.e., many points are clustered close to the upper left corner of the plot (Fig. 3). This also means that the value of t does not affect much the quality of identification. This is not the case with few low quality data, as shown in Fig. 4. Although the parameter t still does not affect much the quality of identification, now most points are clustered in the bottom left corner of the plot close to the line of no discrimination, which implies much worse identification. In particular, Algorithm 1 (Unstable) does not perform any better than a random prediction. For data quality in-between these two extremes, the identification performance depends on the parameter t, as shown in Fig. 5.

We observe that Algorithms 2 (Geršgorin) and 3 (SDP) always perform better than Algorithm 1 (Unstable).[4] This observation is an indication that stability is important, not only for consistency with the problem assumptions, but also for identification performance. Additionally, from Figs. 3–5, we see that Algorithm 3 (SDP) performs slightly better than Algorithm 2 (Geršgorin). This is reasonable, since it does not impose any hard constraints on

[3] "Precision" and "Recall" are metrics that are often also used to measure performance. "Recall" is defined by the ratio TP/(TP + FN) and is, therefore, the same as "Sensitivity". Nevertheless, "Precision" is defined by TP/(TP + FP) and is a different metric, that is sometimes also referred to as Positive Predictive Value (PPV).

[4] Due to space limitations, ROC plots for other parameter combinations (σ, m, ν) are not contained in this paper.

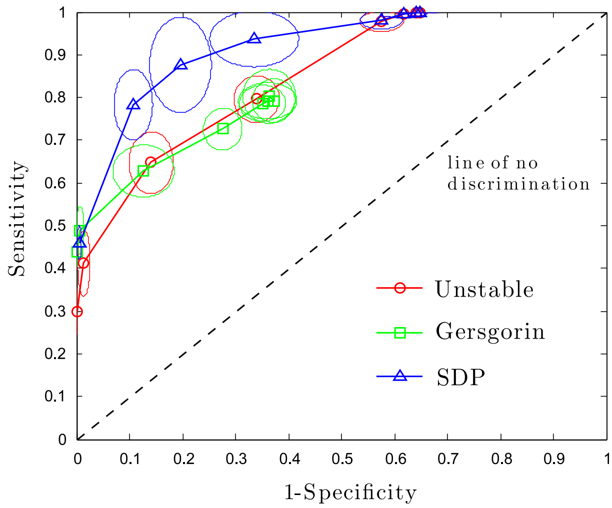

Fig. 3. ROC plots of Algorithms 1 (Unstable), 2 (Geršgorin) and 3 (SDP) for network size $n = 20$ and connectivity $c = 20\%$. Shown are the curves (mean and standard deviation) for $\sigma = 30\%$, $\nu = 10\%$ and $m = n$ (full data). This is an ideal case for identification, with many high quality data. It is expected that predictions should be good and trusted.

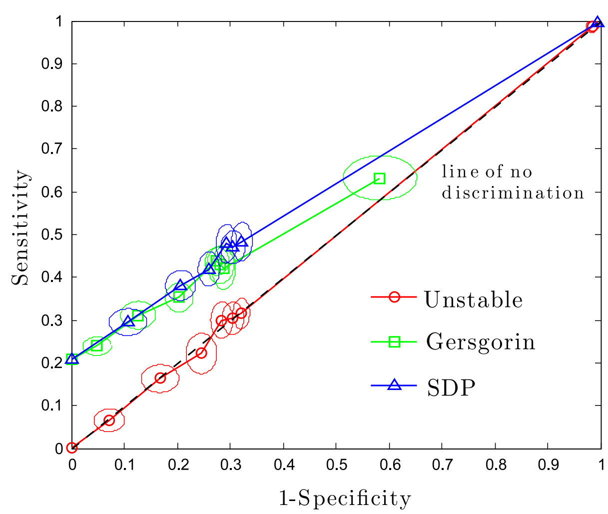

Fig. 4. ROC plots of Algorithms 1 (Unstable), 2 (Geršgorin) and 3 (SDP) for network size $n = 20$ and connectivity $c = 20\%$. Shown are the curves (mean and standard deviation) for $\sigma = 0\%$, $\nu = 50\%$ and $m = \lceil \frac{n}{3} \rceil$ (partial data). This is challenging case for identification, with few low quality data. It is expected that predictions are not so good and possibly should not be trusted much.

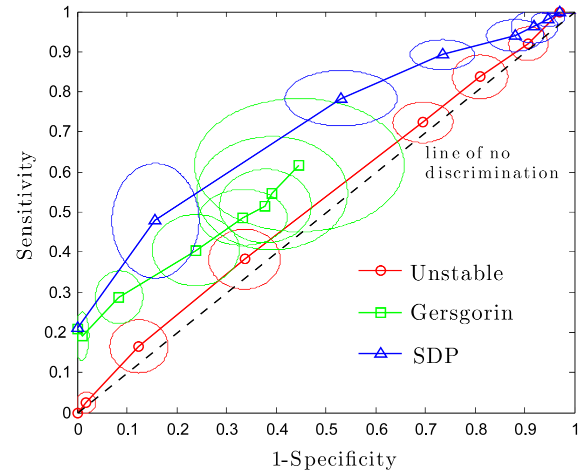

Fig. 5. ROC plots of Algorithms 1 (Unstable), 2 (Geršgorin) and 3 (SDP) for network size $n = 20$ and connectivity $c = 20\%$. Shown are the curves (mean and standard deviation) for $\sigma = 0\%$, $\nu = 50\%$ and $m = n$ (full data). The identification performance depends on the parameter t.

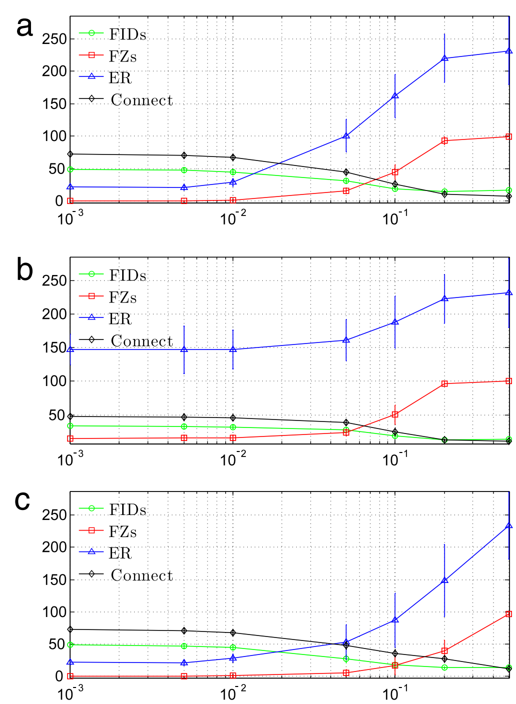

Fig. 6. Identification performance (y-axis) as a function of the parameter $0 \le t \le 1$ (x-axis), for networks of size $n = 20$, connectivity $c = 20\%$, sign knowledge $\sigma = 30\%$, noise $\nu = 10\%$ and $m = n$ (full data). (a) Algorithm 1 (Unstable), (b) Algorithm 2 (Geršgorin), (c) Algorithm 3 (SDP).

the edge weights of the network. Nevertheless, Algorithm 2 (Geršgorin) has a simple linear formulation and scales better with the problem size.

5.2. Identification performance

In this section, we study the performance of our algorithms in terms of the total false identifications. The performance metrics of interest are the total number of false identifications (FIDs), the fitting error (ER) compared to the best fit (ER*) obtained if the identified network was the sought one, and the number of false zero identifications (FZs) as a function of the total false identifications. The ratio FZs/FIDs captures sparsity of the network and ER/ER* indicates how close the identification is to the sought one. Too high or low FZs/FIDs are undesirable, since they correspond to very dense or sparse networks that do not capture reality. Similarly, ER/ER* that is far away from 1, possibly indicates low quality identification, either qualitatively (signs) or quantitatively (edge weights' values). The ratio FZs/FIDs is also related to the connectivity (Connect) of the identified networks.

As in Section 5.1, we focus on networks of size $n = 20$ with connectivity $c = 20\%$, generated as before. Fig. 6(a), (b) and (c) show the performance of Algorithms 1 (Unstable), 2 (Geršgorin) and 3 (SDP), respectively, for parameters $\sigma = 30\%$, $\nu = 10\%$ and $m = n$ (full data). We observe that as t increases, FIDs and connectivity decrease, while FZs/FIDs and ER/ER* increase. In fact, very large values of t result in the lowest FIDs that are also the worst in quality, since then FZs/FIDs $= 1$. To address this tradeoff, we select t so that it results in a network with equal/similar connectivity to the desired one ($c = 20\%$). This, results in identification performances as shown in Tables 1–3 for Algorithms 1, 2 and 3, respectively. Table 1

Table 1
Algorithm 1 (unstable): selection of the parameter $0 \le t \le 1$ for networks with $n = 20$ genes and $c = 20\%$ connectivity.

	$\sigma = 0\%$		$\sigma = 30\%$	
	$\nu = 10\%$	$\nu = 50\%$	$\nu = 10\%$	$\nu = 50\%$
$m = 20$	$t = 0.0962$ 32% FIDs 58% $\frac{\text{FZs}}{\text{FIDs}}$ 180% $\frac{\text{ER}}{\text{ER}^*}$ 100% StIDs	$t = 0.0854$ 33% FIDs 59% $\frac{\text{FZs}}{\text{FIDs}}$ 36% $\frac{\text{ER}}{\text{ER}^*}$ 92% StIDs	$t = 0.1407$ 17% FIDs 64% $\frac{\text{FZs}}{\text{FIDs}}$ 185% $\frac{\text{ER}}{\text{ER}^*}$ 59% StIDs	$t = 0.1362$ 20% FIDs 63% $\frac{\text{FZs}}{\text{FIDs}}$ 39% $\frac{\text{ER}}{\text{ER}^*}$ 48% StIDs
$m = 7$	$t = 0.0808$ 34% FIDs 55% $\frac{\text{FZs}}{\text{FIDs}}$ 145% $\frac{\text{ER}}{\text{ER}^*}$ 93% StIDs	$t = 0.0834$ 35% FIDs 55% $\frac{\text{FZs}}{\text{FIDs}}$ 27% $\frac{\text{ER}}{\text{ER}^*}$ 85% StIDs	$t = 0.1303$ 22% FIDs 59% $\frac{\text{FZs}}{\text{FIDs}}$ 220% $\frac{\text{ER}}{\text{ER}^*}$ 5% StIDs	$t = 0.1460$ 23% FIDs 59% $\frac{\text{FZs}}{\text{FIDs}}$ 40% $\frac{\text{ER}}{\text{ER}^*}$ 2% StIDs

Table 2
Algorithm 2 (Geršgorin): Selection of the parameter $0 \le t \le 1$ for networks with $n = 20$ genes and $c = 20\%$ connectivity.

	$\sigma = 0\%$		$\sigma = 30\%$	
	$\nu = 10\%$	$\nu = 50\%$	$\nu = 10\%$	$\nu = 50\%$
$m = 20$	$t = 0.0933$ 26% FIDs 61% $\frac{\text{FZs}}{\text{FIDs}}$ 190% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.0777$ 28% FIDs 61% $\frac{\text{FZs}}{\text{FIDs}}$ 38% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.1373$ 16% FIDs 67% $\frac{\text{FZs}}{\text{FIDs}}$ 201% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.1215$ 17% FIDs 65% $\frac{\text{FZs}}{\text{FIDs}}$ 41% $\frac{\text{ER}}{\text{ER}^*}$
$m = 7$	$t = 0.0809$ 27% F.IDs 57% $\frac{\text{FZs}}{\text{FIDs}}$ 175% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.0820$ 29% F.IDs 57% $\frac{\text{FZs}}{\text{FIDs}}$ 33% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.1341$ 18% F.IDs 62% $\frac{\text{FZs}}{\text{FIDs}}$ 234% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.1544$ 19% F.IDs 62% $\frac{\text{FZs}}{\text{FIDs}}$ 45% $\frac{\text{ER}}{\text{ER}^*}$

Table 3
Algorithm 3 (SDP): selection of the parameter $0 \le t \le 1$ for networks with $n = 20$ genes and $c = 20\%$ connectivity.

	$\sigma = 0\%$		$\sigma = 30\%$	
	$\nu = 10\%$	$\nu = 50\%$	$\nu = 10\%$	$\nu = 50\%$
$m = 20$	$t = 0.1485$ 25% FIDs 65% $\frac{\text{FZs}}{\text{FIDs}}$ 183% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.2575$ 24% FIDs 62% $\frac{\text{FZs}}{\text{FIDs}}$ 42% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.3334$ 13% FIDs 65% $\frac{\text{FZs}}{\text{FIDs}}$ 185% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.3126$ 16% FIDs 66% $\frac{\text{FZs}}{\text{FIDs}}$ 42% $\frac{\text{ER}}{\text{ER}^*}$
$m = 7$	$t = 0.1394$ 27% F.IDs 58% $\frac{\text{FZs}}{\text{FIDs}}$ 164% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.1546$ 28% F.IDs 57% $\frac{\text{FZs}}{\text{FIDs}}$ 34% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.2562$ 17% F.IDs 63% $\frac{\text{FZs}}{\text{FIDs}}$ 203% $\frac{\text{ER}}{\text{ER}^*}$	$t = 0.3086$ 19% F.IDs 65% $\frac{\text{FZs}}{\text{FIDs}}$ 43% $\frac{\text{ER}}{\text{ER}^*}$

also shows the percent of stable identifications (StIDs) returned by Algorithm 1. Note that this decreases significantly as the noise increases or the size of data set and sign knowledge decreases. In all cases, the ratio FZs/FIDs is approximately 60%, while the error ER ranges from a fraction to a multiple of the best one ER*. What is noteworthy is that Algorithms 2 (Geršgorin) and 3 (SDP) have comparable performance in terms of FIDs, which can get as low as 16% for high quality data (low noise, high sign knowledge and full data). In all cases, Algorithms 2 (Geršgorin) and 3 (SDP) perform better than Algorithm 1 in terms of FIDs.

5.3. Discussion

In Sections 5.1 and 5.2 we discussed two ways of choosing the parameter t. The first depending on proximity to the upper left corner of the ROC plot and the second depending on the desired connectivity of the identified network. In this section we show consistency of these two methods. In other words, we show that a parameter t that gives an identification with desired connectivity, lies as close as possible to the upper left corner of the ROC plot. For this, we check the locations in the ROC plot of the identifications contained in Tables 1–3. For illustration purposes, we focus on the parameters $(m, \sigma, \nu) = \{(20\%, 30\%, 10\%), (7\%, 0\%, 50\%), (20\%, 0\%, 50\%)\}$ in order to compare with Figs. 3–5, respectively. For these data sets, we get sensitivity and (1-specificity) values, as shown in Table 4. Locating these values in Figs. 3–5 we see that they lie at least as close to the upper left corner compared to other points in these ROC plots and, therefore, they correspond to better identification performance. Although network connectivity is typically unknown, its is easier to get an estimate of it from biological knowledge, than construct ROC plots that depend on identification performance.

Table 4
Sensitivity and (1-specificity) values for selected identifications in Tables 1–3.

(m, σ, ν) and Alg.		1-specificity	Sensitivity
(20%, 30%, 10%)	Alg. 1	0.13 ± 0.04	0.61 ± 0.07
	Alg. 2	0.11 ± 0.05	0.61 ± 0.06
	Alg. 3	0.08 ± 0.03	0.63 ± 0.07
(7%, 0%, 50%)	Alg. 1	0.19 ± 0.03	0.19 ± 0.04
	Alg. 2	0.16 ± 0.03	0.33 ± 0.03
	Alg. 3	0.18 ± 0.02	0.36 ± 0.03
(20%, 0%, 50%)	Alg. 1	0.21 ± 0.06	0.25 ± 0.07
	Alg. 2	0.18 ± 0.07	0.36 ± 0.08
	Alg. 3	0.33 ± 0.07	0.57 ± 0.04

6. SOS pathway in *E. coli*

We further applied the proposed identification algorithms to a subnetwork of the SOS pathway in *E. coli*, using the genetic perturbation experimental data set

$$X = 10^{-3}\begin{bmatrix} 906 & -132 & -139 & 187 & 291 & -61 & -77 & -17 & -25 \\ 212 & 383 & -117 & 64 & 169 & -87 & 39 & 125 & 84 \\ 18 & -107 & 10\,524 & 61 & 80 & 13 & 64 & 89 & -70 \\ 104 & -50 & -273 & 139 & 180 & 146 & 69 & -4 & 275 \\ 119 & -97 & 56 & 315 & 2147 & 142 & -68 & 135 & 113 \\ 76 & -189 & -214 & 250 & 347 & 2017 & -67 & -172 & -22 \\ -122 & -47 & -102 & -107 & -11 & 104 & 3068 & 365 & 217 \\ 178 & -183 & 36 & -70 & -34 & -155 & 8 & 26\,633 & 87 \\ 72 & -128 & 73 & 81 & 305 & 51 & -61 & 274 & 672 \end{bmatrix}$$

provided in Gardner et al. (2003). Since there was no explicit mention to U, we assumed that $BU = I_9$.[5] The *a priori* knowledge we used is depicted in Table 5 and has been obtained based on the diagram of Fig. 7.

The subnetwork that we considered consists of nine genes and several transcription factors and metabolites (Fig. 7). The main pathway featured in this network is the pathway between the single-stranded DNA (ssDNA) and the protein LexA that acts as a repressor to several other genes (*recA*, *ssb*, *dinI*, *umuDC*, and *rpoD*). The protein RecA, which is activated by the single-stranded DNA, cleaves LexA and thus upregulates the above-mentioned genes. Other key regulators in the network are the sigma factors $\sigma 70$, $\sigma 32$, and $\sigma 38$. These sigma factors play an important role in initiating transcription in heat shock and starvation responses.

[5] Note that this is a reasonable assumption, since different values of BU would only result in scaling of the model. See also Remark 2.

Table 5
A summary of *a priori* knowledge for the SOS pathway in *E. coli*. A "+" sign indicates known activation, a "−" sign indicates known inhibition, "0" indicates the absence of connection, and "?" indicates an unknown connection. In brackets are known gene interactions that are considered unknown for the purposes of identification.

Genes	*recA*	*lexA*	*ssb*	*recF*	*dinI*	*umuDC*	*rpoD*	*rpoH*	*rpoS*
recA	?	−	?(−)	?(+)	?(+)	?(−)	+	?(0)	?(0)
lexA	+	−	?(−)	?(+)	?(+)	?(−)	+	?(0)	?(0)
ssb	+	−	?(−)	?(+)	?(+)	?(−)	+	?(0)	?(0)
recF	?(0)	?(0)	?(0)	?(−)	?(0)	?(0)	+	?(0)	+
dinI	+	−	?(−)	?(+)	?	?(−)	+	?(0)	?(0)
umuDC	+	−	?(−)	?(+)	?(+)	?(−)	+	?(0)	?(0)
rpoD	+	−	?(−)	?(+)	?(+)	?(−)	?	+	?(0)
rpoH	?(0)	?(0)	?(0)	?(0)	?(0)	?(0)	+	?	?(0)
rpoS	?(0)	?(0)	?(0)	?(0)	?(0)	?(0)	+	?(0)	?

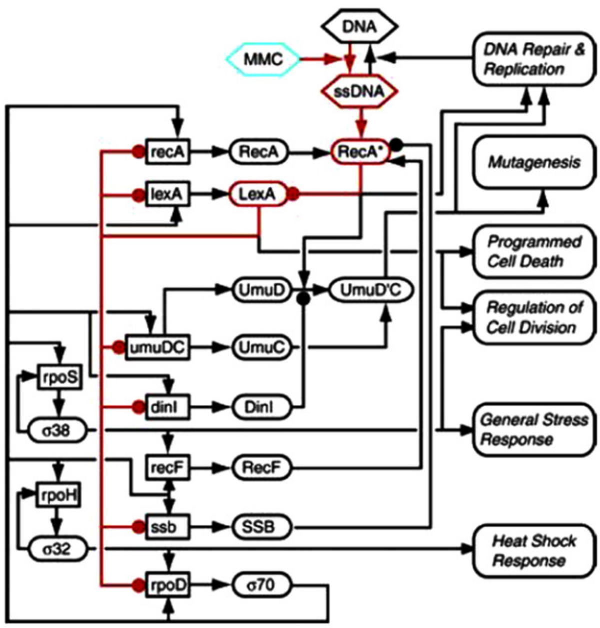

Fig. 7. Diagram of interactions in the SOS network. DNA lesions caused by mitomycin C (MMC) (blue hexagon) are converted to single-stranded DNA during chromosomal replication. Upon binding to ssDNA, the RecA protein is activated (RecA*) and serves as a coprotease for the LexA protein. The LexA protein is cleaved, thereby diminishing the repression of genes that mediate multiple protective responses. Boxes denote genes, ellipses denote proteins, hexagons indicate metabolites, arrows denote positive regulation, filled circles denote negative regulation. Red emphasis denotes the primary pathway by which the network is activated after DNA damage. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
Source: Taken from Gardner et al. (2003).

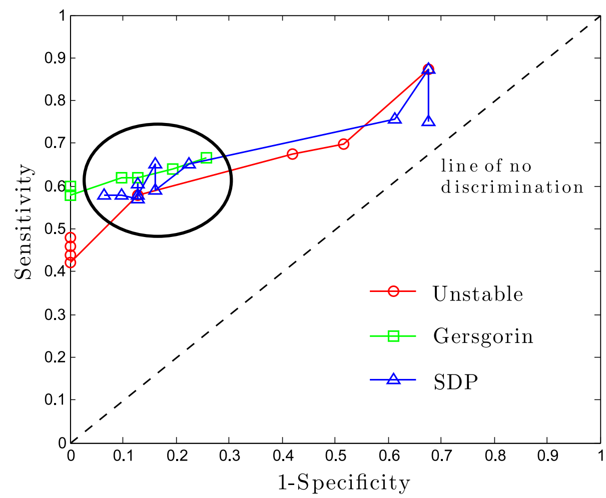

Fig. 8. ROC plots of Algorithms 1 (Unstable), 2 (Geršgorin) and 3 (SDP) for the SOS pathway shown in Fig. 7 and different values of the parameter t. The best identifications are contained in the ellipse close to the upper left corner.

We applied Algorithms 1–3 on the data set X for different values of the parameter t. The corresponding ROC plots are shown in Fig. 8. As discussed in Section 5, the best identifications will correspond to values of the parameter t that give points in the ellipse in Fig. 8. For Algorithm 1 these points correspond to $t \in [0.01, 0.1]$, for Algorithm 2 they correspond to $t \in [0, 0.1]$, and for Algorithm 3 they correspond to $t \in [0.05, 0.5]$. In particular, we choose $t = 0.01$ for Algorithm 1, $t = 0.01$ for Algorithm 2 and $t = 0.1$ for Algorithm 3. These parameters result in 37%, 31% and 31% false identifications, respectively. Therefore, Algorithms 2 and 3 still perform better than Algorithm 1, demonstrating the importance of the stability specification.

All identifications obtained from Algorithm 1 are unstable, while the obtained networks have connectivity approximately equal to 50%. Note that this identification performance is worse than the one shown in Tables 1–3 for full data and 30% sign knowledge. This is expected since the SOS pathway is much denser (its connectivity is approximately 60%), which conflicts with the sparsity objective.[6] Following we present the interconnection matrix for the SOS pathway in *E. coli* returned by Algorithm 2 for $t = 0.01$:

$$A = 10^{-3}\begin{bmatrix} -33 & -2 & 0 & 0 & 5 & 0 & 2 & 0 & 0 \\ 9 & -21 & -1 & -44 & 1 & 2 & 2 & 0 & 20 \\ 2 & -2 & -29 & 0 & 0 & 0 & 2 & 0 & 0 \\ 10 & 0 & -2 & -123 & 2 & 8 & 4 & 0 & 37 \\ 2 & -2 & 0 & 0 & -30 & 0 & 2 & 0 & 0 \\ 2 & -2 & 0 & 0 & 0 & -31 & 2 & 0 & 0 \\ 2 & -2 & 0 & 0 & 0 & 0 & -38 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & 0 \\ 2 & -2 & 0 & 0 & 2 & 0 & 2 & 0 & -15 \end{bmatrix}.$$

Matrix A has 7 false positives, 3 false negatives, 16 false zeros, and 26 false identifications in total, while it is also stable and satisfies the desired sparsity pattern. The matrix A returned by Algorithm 3 for $t = 0.1$ is:

[6] Note that we are analyzing a part of the SOS response mechanism in *E. coli* in which central role plays the protein LexA. This protein regulates the expression activity of a large number of other genes, which explains the particularly high connectivity observed here. In fact, it is known that LexA directly regulates, i.e., binds to the promoters, of 31 other genes (Fernandez De Henestrosa et al., 2000). The connectivity of genes like LexA is atypical.

$$A = 10^{-3}\begin{bmatrix} -10 & -3 & 0 & -1 & 2 & 0 & 2 & 0 & 0 \\ 5 & -23 & 0 & 0 & 0 & -1 & 2 & 0 & 2 \\ 2 & -2 & -1 & -4 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & -4 & 0 & 0 & 2 & 0 & 2 \\ 2 & -2 & 0 & 0 & -5 & 0 & 2 & 0 & 0 \\ 2 & -2 & 0 & -1 & 0 & -5 & 2 & 0 & 0 \\ 2 & -2 & 0 & -4 & 0 & 0 & -3 & 2 & 0 \\ 0 & 0 & 0 & -5 & 0 & 0 & 2 & 0 & 0 \\ 2 & -4 & 0 & 0 & 2 & 0 & 2 & 0 & -15 \end{bmatrix}.$$

Matrix A has 3 false positives, 6 false negatives, 16 false zeros, and 25 false identifications in total, while it is also stable and satisfies the desired sparsity pattern.

7. Conclusions

In this paper, we considered the problem of identifying a minimal model that best explains genetic perturbation data obtained at the network's equilibrium state. We relaxed the combinatorially hard cardinality optimization specification by employing its weighted ℓ_1 approximation and extended our formulation to account for *a priori* knowledge on the network structure, as well as stability of the derived solutions. We tested performance and sensitivity of our algorithms to parameter selection, for various sizes of data sets, sign knowledge and noise levels. We concluded that stability is not only necessary for consistency with the problem assumptions, but also for better identification performance. The strength of our approach lies in its convex nature that can handle large scale identification problems. Its efficiency was also demonstrated on real experimental data obtained for the SOS pathway in *E. coli*.

References

Amato, R., Ciaramella, A., Deniskina, N., Del Mondo, C., di Bernardo, D., Donalek, C., et al. (2006). A multi-step approach to time series analysis and gene expression clustering. *Bioinformatics*, *22*, 589–596.

Amato, F., Cosentino, C., Curatola, W., & di Bernardo, D. (2007). LMI-based algorithm for the reconstruction of biological networks. In *Proc. American control conference. New York, NY* (pp. 2720–2725).

Arnone, M., & Davidson, E. (1997). The hardwiring of development: organization and function of genomic regulatory systems. *Development*, *124*, 1851–1864.

August, E., & Papachristodoulou, A. (2009). Efficient, sparse biological network determination. *BMC Systems Biology*, *3*(25).

Bansal, M., Belcastro, V., Ambesi-Impiombato, A., & di Bernardo, D. (2007). How to infer gene networks from expression profiles. *Molecular Systems Biology*, *3*, doi:10.1038/msb4100120.

Bansal, M., Gatta, G. Della, & di Bernardo, D. (2006). Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics*, *22*(7), 815–822.

Boyd, S. ℓ_1-norm methods for convex cardinality problems. Lecture notes for EE364b. Stanford University. Available at: http://www.stanford.edu/class/ee364b/.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

Candes, E., Romberg, J., & Tao, T. (2006). Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, *52*(2), 489–509.

Candes, E. J., Wakin, M. B., & Boyd, S. (2008). Enhancing sparsity by reweighted ℓ_1 minimization. *The Journal of Fourier Analysis and Applications*, *14*(5), 877–905.

Cinquemani, E., Porreca, R., Lygeros, J., & Ferrari-Trecate, G. (2009). Canalizing structure of genetic network dynamics: modelling and identification via mixed-integer programming. In *Proc. IEEE conference on decision and control. Shangai, China*.

De Muth, J. E. (2006). *Basic statistics and pharmaceutical statistical applications* (2nd ed.) Chapman & Hall/CRC.

DiBernardo, D., Gardner, T. S., & Collins, J. J. (2004). Robust identification of large genetic networks. In *Pacific symposium on biocomputing, Vol. 9* (pp. 486–497).

Eisen, M., Spellman, P., Brown, P., & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences (PNAS)*, *95*(25), 14863–14868.

Fernandez De Henestrosa, A. R., Ogi, T., Aoyagi, S., Chafin, D., Hayes, J. J., Ohmori, H., et al. (2000). Identification of additional genes belonging to the Lexa regulation in *E. coli*. *Molecular Microbiology*, *35*, 1560–1572.

Gardner, T., di Bernardo, D., Lorenz, D., & Collins, J. (2003). Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, *301*, 102–105.

Gardner, T., & Faith, J. (2005). Reverse-engineering transcription control networks. *Physics of Life Reviews*, *2*, 65–88.

Grant, M., Boyd, S., & Ye, Y. `cvx` MATLAB software for disciplined convex programming. Available at: http://www.stanford.edu/~boyd/cvx/.

Han, S., Yoon, Y., & Cho, K. H. (2007). Inferring biomolecular interaction networks based on convex optimization. *Computational Biology and Chemistry*, *31*(5–6), 347–354.

Hassibi, A., How, J., & Boyd, S. (1999). Low-authority controller design via convex optimization. *Journal of Guidance, Control, and Dynamics*, *22*(6), 862–872.

Horn, R., & Johnson, C. (1985). *Matrix analysis*. Cambridge University Press.

Julius, A. A., Zavlanos, M. M., Boyd, S. P., & Pappas, G. J. (2009). Genetic network identification using convex programming. *IET Systems Biology*, *3*(3), 155–166.

Papachristodoulou, A., & Recht, B. (2007). Determining interconnections in chemical reaction networks. In *Proc. American control conference. New York, NY* (pp. 4872–4877).

Peér, D., Nachman, I., Linial, M., & Friedman, N. (2000). Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, *7*, 601–620.

Porreca, R., Drulhe, S., de Jong, H., & Ferrari-Trecate, G. (2008). Structural identification of piecewise-linear models of genetic regulatory networks. *Journal of Computational Biology*, *15*(10), 1365–1380.

Rugh, W. (1996). *Linear system theory*. Prentice Hall.

Schreiber, S. L. (2000). Target-oriented and diversity-oriented organic synthesis in drug discovery. *Science*, *287*(5460), 1964–1969.

Sontag, E., Kiyatkin, A., & Kholodenko, B. (2004). Inferring dynamic architecture of cellular networks using time series of gene expression, protein and metabolite data. *Bioinformatics*, *20*(12), 1877–1886.

Srividhy, J., Crampin, E. J., McSharry, P. E., & Schnell, S. (2007). Reconstructing biochemical pathways from time course data. *Proteomics*, *7*, 828–838.

Steuer, R., Kurths, J., Daub, C. O., Weise, J., & Selbig, J. (2002). The mutual information: detecting and evaluating dependencies between variables. *Bioinformatics*, *182*, 231–240.

Tegner, J., Yeung, M., Hasty, J., & Collins, J. (2003). Reverse engineering gene networks: integrating genetic perturbations with dynamical modeling. *Proceedings of the National Academy of Sciences (PNAS)*, *100*(10), 5944–5949.

Thieffry, D., Huerta, A., Pérez-Rueda, E., & Collado-Vides, J. (1998). From specific gene regulation to genomic networks: a global analysis of transcriptional regulation in *E. coli*. *Bioessays*, *20*(5), 433–440.

Tropp, J. A. (2006). Just relax: convex programming methods for identifying sparse signals. *IEEE Transactions on Information Theory*, *51*(3), 1030–1051.

Zavlanos, M. M., Julius, A. A., Boyd, S. P., & Pappas, G. J. (2008). Identification of stable genetic networks using convex programming. In *Proc. of the 2008 American control conference. Seattle, WA. June* (pp. 2755–2760).

Michael M. Zavlanos (S'05-M'09) received the Diploma in mechanical engineering from the National Technical University of Athens, Athens, Greece, in 2002, and the M.S.E. and Ph.D. degrees in electrical and systems engineering from the University of Pennsylvania, Philadelphia, PA, in 2005 and 2008, respectively.

He is currently an assistant professor with the department of mechanical engineering, Stevens Institute of Technology, Hoboken, NJ. His current research interests include a wide range of topics in the emerging discipline of networked systems and science, with applications in robotic, sensor, biomolecular and social networks. He is particularly interested in hybrid solution techniques, on the interface of control theory with the discrete science of networks and graphs.

Dr. Zavlanos was a recipient of the NSF Faculty Early Career Development (CAREER) Award in 2011 and a finalist for the Best Student Paper Award at the 45th IEEE Conference on Decision and Control, in 2006.

A. Agung Julius is an Assistant Professor at the Department of Electrical, Computer, and Systems Engineering at the Rensselaer Polytechnic Institute. He was previously a Postdoctoral Researcher at the GRASP Lab., University of Pennsylvania. He received the B.Eng. (cum laude) degree in electrical engineering from Institut Teknologi Bandung in 1998, and the M.Sc. (with distinction) and Ph.D. degrees in applied mathematics from the University of Twente, The Netherlands, in 2001 and 2005, respectively.

Dr. Julius' research interests include systems and control, systems biology, stochastic models in systems biology, control of biological systems, hybrid systems, reduction of complex systems, verification and robust testing of hybrid systems, and mathematical systems theory. He received an NSF CAREER award in 2010.

Stephen P. Boyd (S'82-M'85-SM'97-F'99) received the A.B. degree (summa cum laude) in mathematics from Harvard University, Cambridge, MA, in 1980, and the Ph.D. degree in electrical engineering and computer science from University of California, Berkeley, in 1985.

He is currently the Samsung Professor of Engineering, in the Information Systems Laboratory, Electrical Engineering Department, Stanford University, Stanford, CA. He is the author or coauthor of *Linear Controller Design—Limits of Performance* (Prentice-Hall, 1991), *Linear Matrix Inequalities in System and Control Theory* (SIAM 1994), and Convex Optimization (Cambridge University Press, 2004). His current research interests include convex programming applications in control, signal processing and circuit design.

Prof. Boyd was a recipient of an ONR Young Investigator Award, a Presidential Young Investigator Award, and the 1992 AACC Donald P. Eckman Award. He has received the Perrin Award for Outstanding Undergraduate Teaching in the School of Engineering, and an ASSU Graduate Teaching Award. In 2003, he received the AACC Ragazzini Education Award. He is a Distinguished Lecturer of the IEEE Control Systems Society.

George J. Pappas (S'90-M'91-SM'04-F'09) received the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, in 1998 where he received the Eliahu Jury Award for Excellence in Systems Research.

He is currently the Joseph Moore Professor of Electrical and Systems Engineering at the University of Pennsylvania. He is a member of the General Robotics, Automation, Sensing and Perception (GRASP) Laboratory and serves as the Deputy Dean for Research in the School of Engineering and Applied Science. His current research interests include hybrid and embedded systems, hierarchical control systems, distributed control systems, nonlinear control systems, with applications to robotics, unmanned aerial vehicles, biomolecular networks, and green buildings.

Prof. Pappas has received numerous awards, including the National Science Foundation (NSF) CAREER Award in 2002, the NSF Presidential Early Career Award for Scientists and Engineers in 2002, the 2009 George S. Axelby Outstanding Paper Award, and the 2010 Anotnio Ruberti Outstanding Young Researcher Prize. He is also a Fellow of IEEE.