Invalidation of the structure of genetic network dynamics: a geometric approach

Ricardo Porreca, Eugenio Cinquemani, John Lygeros, Giancarlo Ferrari-Trecate

INTERNATIONAL JOURNAL OF ROBUST AND NONLINEAR CONTROL
Int. J. Robust. Nonlinear Control 0000; **00**:1–17
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/rnc

Invalidation of the structure of genetic network dynamics: A geometric approach

Riccardo Porreca[1*], Eugenio Cinquemani[2], John Lygeros[1] and Giancarlo Ferrari-Trecate[3]

[1]*Institut für Automatik, ETH Zürich, Physikstrasse 3, 8092 Zürich, Switzerland*
[2]*INRIA Grenoble-Rhône-Alpes, 655 avenue de l'Europe, Montbonnot, 38334 Saint Ismier cedex, France*
[3]*Dipartimento di Informatica e Sistemistica, Università di Pavia, via Ferrata 1, 27100 Pavia, Italy*

SUMMARY

This work concerns the identification of the structure of a genetic network model from measurements of gene product concentrations and synthesis rates. In earlier work, we developed a data preprocessing algorithm that is able to reject many hypotheses on the network structure by testing certain monotonicity properties for a wide family of network models. Here we develop a geometric interpretation of the method. Then, for a relevant subclass of genetic network models, we extend our approach to the combined testing of monotonicity and convexity-like properties associated with the network structures. The theoretical aspects and practical performance of the enhanced methods are illustrated by way of numerical results.

Received ...

KEY WORDS: systems biology; identification; quasiconvexity; unate functions; sigmoidal activation functions.

1. INTRODUCTION

Genetic networks govern the behavior of living cells in response to changes in the environment, and determine growth, replication, and death of cells. They are composed of genes, i.e. pieces of the DNA strand encoding a specific protein. Proteins are synthesized in several copies upon gene expression and participate in the regulation of the expression of other genes, thus giving rise to a complex network of biochemical interactions.

Modern technologies for the time-course measurement of gene expression, such as gene reporter systems, allow one to step from pure topological modelling of gene networks to the modelling of the interaction dynamics. However, this requires setting up a dynamical model of the network whose structure and parameters are typically unknown or uncertain. Data-based identification of an accurate model is challenging due to the size of the family of possible model alternatives. Yet, a priori biological knowledge may be exploited so as to ameliorate the complexity of the problem.

In [1], we developed an identification strategy for genetic network dynamics with a unate structure. These are ODE models built upon a family of Boolean network models which reportedly capture most of the experimentally observable gene regulation interactions [2]. In [1], we showed that unate models possess monotonicity properties that can be tested inexpensively on experimental data, so as to discard entire sets of model hypotheses and focus the search on model structures

*Correspondence to: Institut für Automatik, ETH Zürich, Physikstrasse 3, 8092 Zürich, Switzerland. E-mail: riccardo.porreca@control.ee.ethz.ch

consistent with the data. A similar approach to the identification of gene network topology using time-course data was recently developed in [3], relying on discrete-time models still having monotonicity properties and without taking explicitly into account noise affecting the data. Other relevant approaches to reverse engineering of gene networks, based on different modeling assumptions and inference techniques, can be found in [4, 5, 6, 7, 8, 9].

One question that arises naturally is whether additional properties of the models in the class (other than monotonicity) can be exploited so as to further narrow down the search for valid models. In this paper we address this question by considering a subclass of unate models. In particular, we leverage on the analysis in [10] showing that 87% of the 139 Boolean genes regulatory rules considered in [11] belong to a narrower class, which we refer to as $S_0 \cup S_1$, and show that ODE models with $S_0 \cup S_1$ structure possess convexity-like properties that can be used for checking the consistency of different model hypotheses with the experimental data. To this purpose, we introduce a geometric framework that also provides an alternative interpretation of the methods by [1]. In the economics literature, a similar approach was considered by [12] for testing hypotheses on production processes.

In Sections 2.1 and 2.2 we review our results from [1] on modelling and invalidation of genetic network dynamics with unate structure. In Section 2.3, we give these results a new geometrical interpretation. The same approach is exploited in Section 3 to analyze convexity-like properties of the models with $S_0 \cup S_1$ structure and set up new model invalidation strategies. In Section 4 we discuss efficient implementations of these methods as well as a strategy to handle noisy data. Theoretical and experimental results are discussed in Section 5 by way of illustrative simulations. Mathematical proofs of the results developed in the paper can be found in [13]. Sections 2.3 and 3–5 provide the original contributions of this work.

Notation: If $v \in \mathbb{R}^d$, $\mathrm{diag}(v)$ stands for the diagonal matrix $V \in \mathbb{R}^{d\times d}$ with $V_{ii} = v_i$, $i = 1, \ldots, d$. For a set $M \subset \mathbb{R}^n$, $\mathrm{Conv}(M)$ denotes its convex hull. The cardinality of a finite set P is $|P|$.

2. UNATE MODELS AND MONOTONICITY

Sections 2.1 and 2.2 are a concise review of our results in [1] and provide the context and the background for the results developed in this work. Section 2.3 reconsiders these results from a geometrical point of view and establishes the ground for the results presented in the later sections.

2.1. Genetic network models with unate structure

In the context of Boolean network modelling, the activation status of gene i, with $i = 1, \ldots, n$, in a network of n genes is encoded by a binary variable X_i that takes the value 1 if the gene is active and 0 otherwise. A Boolean rule $B_i(X) : \{0,1\}^n \to \{0,1\}$ describes the logics governing the activation of gene i as a function of $X = (X_1, \ldots, X_n)$. Depending on what genes regulate the expression of gene i, B_i will effectively depend on only some of the entries of X. Based on biochemical reaction modelling arguments [2], it can be argued that most regulatory interactions are well described by unate functions, i.e. Boolean functions that are monotone (either nondecreasing or nonincreasing) in each of the input variables, meaning that increasing the value of a single X_i from 0 to 1 can only force a specific trend of $B_i(X)$ (either increasing from 0 to 1 or decreasing from 1 to 0) or leave it unchanged, regardless of the value of the other variables. In conjunctive normal form, unate functions are given by

$$B_i(X) = \bigwedge_{l=1}^{h_i} \bigvee_{j\in J_{il}} \tilde{X}_j \ , \tag{1}$$

where "$\wedge$" and "$\vee$" stand for "and" and "or", respectively, h_i is a nonnegative integer and each J_{il} is a nonempty set of indices from $\{1, \ldots, n\}$. Furthermore, unate functions must fulfill the additional constraint that each variable $\tilde{X}_j$ appears in (1) exclusively as either X_j or $\neg X_j$, where "$\neg$" is negation. By convention, a conjunction of $h_i = 0$ terms is equal to 1. An example of function that is not in this class is the exclusive or ("xor"). Indeed, if $X_2 = 0$ then X_1 xor X_2 increases from 0 to 1 as X_1 goes from 0 to 1, while if $X_2 = 1$ the function decreases from 1 to 0. Moreover,

DOI: 10.1002/rnc

the conjunctive normal form X_1 xor $X_2 = (X_1 \vee X_2) \wedge (\neg X_1 \vee \neg X_2)$ includes e.g. both X_1 and its negated form $\neg X_1$. The class of unate models represents all networks where each regulator of a given gene acts unambiguously either as an activator or as a repressor of that gene, though it may promote expression of one gene and inhibit the expression of another gene. Unate functions include the so-called Hierarchically Canalizing Functions (HCFs) [14], which capture a large class of the known regulatory interactions among genes and are intimately related with the stability properties of the network [15, 16].

In several cases of interest, further assumptions can be made a priori on the structure of $B_i(X)$.

Example 1
It is shown in [10] that many gene activation rules fall into two subclasses of HCF, S_0 and S_1, having the following form:

$$B_i(X) = \begin{cases} \tilde{X}_{j_1} \wedge \tilde{X}_{j_2} \wedge \tilde{X}_{j_3} \wedge \cdots \wedge \tilde{X}_{j_\ell}, & \text{if } B_i \in S_0, \\ [\tilde{X}_{j_1} \vee \tilde{X}_{j_2}] \wedge \tilde{X}_{j_3} \wedge \cdots \wedge \tilde{X}_{j_\ell}, & \text{if } B_i \in S_1, \end{cases} \tag{2}$$

where ℓ is the number of effective inputs of $B_i(X)$ and $j_1, \ldots, j_\ell$ are pairwise different indices from the set $\{1, \ldots, n\}$.

We are interested in quantitative models of gene expression. We consider models of the form [17]

$$\dot{x}_i = g_i(x) - \gamma_i(x) \ , \tag{3}$$

where $i = 1, \ldots, n$ denotes the ith of n genes, $x_i \geq 0$ denotes the concentration of the corresponding product, $x = (x_1, \ldots, x_n)$, and $g_i(x) \geq 0$ and $\gamma_i(x) \geq 0$ are synthesis and degradation rates, respectively. Functions $g_i(x)$ and $\gamma_i(x)$ are generally used to model the regulatory effects on the synthesis and degradation of the ith gene of the network. In view of the falsification approach developed in this work, we are especially interested in synthesis rate regulation functions. In particular, for $i = 1, \ldots, n$, we assume that

$$g_i(x) = \kappa_{0,i} + \kappa_{1,i} b_i(x) \ , \tag{4}$$

where $\kappa_{0,i}$ and $\kappa_{1,i}$ are nonnegative constants and the gene activation level $b_i(x)$ is typically a combination of switch-like (e.g. sigmoidal) regulatory functions describing the effect of protein j on the expression of gene i and the synthesis of the corresponding protein. In order to account for the discrete regulatory logics (1) in the quantitative model (3)–(4), we follow an approach inspired by [18]. Each variable X_i is replaced by a monotone, nondecreasing sigmoidal function $\sigma^+ : [0, +\infty) \to [0, 1]$ of the concentration x_i. Given any two functions $\tau(x)$ and $\tau'(x)$ representing the Boolean expressions $T(X)$ and $T'(X)$, $\neg T(X)$ is replaced by $1 - \tau(x)$ and $T(X) \wedge T'(X)$ by $\tau(x) \cdot \tau'(x)$. In particular $\neg X_i$ is represented by $\sigma^-(x_i) = 1 - \sigma^+(x_i)$. Applying these rules to (1) leads to the following class of models.

Definition 1
A unate model is given by (3) and (4) where, for some integer $h_i \geq 0$ and some sets of indices $J_{il} \subseteq \{1, \ldots, n\}$, with $l = 1, \ldots, h_i$,

$$b_i(x) = \prod_{l=1}^{h_i} \left(1 - \prod_{j \in J_{il}} \left(1 - \sigma^{\pm}(x_j) \right) \right) \ , \tag{5}$$

where either $\sigma^{\pm}(x_j) = \sigma^+(x_j)$ or $\sigma^{\pm}(x_j) = \sigma^-(x_j)$ and, by convention, products over an empty set return 1.

A typical choice of sigmoid is the Hill function $\sigma^+(x) = 1/[1 + (\eta/x)^d]$ [19, 20]. For any choice of the cooperativity parameter $d \geq 1$ and the threshold parameter $\eta \geq 0$, this function increases monotonically from 0 to 1, satisfies $\sigma^+(\eta) = 1/2$ and $\frac{d\sigma^+}{dx}(\eta) \geq 0$ increases with d. For $d = 1$, in

Prepared using rncauth.cls
DOI: 10.1002/rnc

particular, one recovers Michaelis-Menten kinetics. An alternative choice is the logistic function

$$\sigma^+(x') = \frac{1}{1+e^{-d(x'-\tilde{\eta})}} \ , \qquad x' \in \mathbb{R}_{\geq 0} \ , \tag{6}$$

with analogous interpretation for d and $\tilde{\eta}$. Note that in this case $\sigma^+(x') \neq 0$ when $x' = 0$. If one looks at (6) over the entire $\mathbb{R}$, it is immediate to see that Hill functions are in one-to-one correspondence with (6) via the transformations

$$x' = \log(x) \ , \tag{7}$$

$$\tilde{\eta} = \log(\eta) \ . \tag{8}$$

Therefore, we accept that b_i and g_i are generally defined over the entire $\mathbb{R}^n$ and assume without loss of generality that all sigmoids in (5) are logistic functions. The methods developed below also apply to models involving Hill functions, by taking logarithms of concentration variables and measurements, as above.

2.2. Invalidation of unate models: sign patterns

Consider the problem of identifying a unate model for gene i from the dataset $\mathscr{D} = \{(x^k, g_i^k) : \ k = 1, \ldots, m\}$, where each x^k is a vector of protein concentrations and $g_i^k = g_i(x^k)$ is the corresponding synthesis rate. In practice, (noisy versions of) measurements $\mathscr{D}$ can be obtained by perturbation experiments (see [21] and references therein) or time-course experiments [22]. In particular, methods proposed in this paper are well adapted to gene reporter systems where average promoter activities over a cell population are sampled with relatively high frequency. In fact, protein concentrations x and synthesis rates g_i can be inferred from coarse promoter activity data e.g. by means of the nonparametric estimation methods proposed in [22]. Since x and g_i are both observed, one faces a regression problem for the function g_i only, i.e. the specific form of $\gamma_i(x)$ in (3) is irrelevant.

A fundamental source of complexity is that the function $b_i(x)$ in (5) depends upon discrete quantities, i.e. the integers h_i, the sets J_{il} and the signs of the sigmoids. For realistic size problems, it is computationally prohibitive to search for the best fitting model by identifying values for all the parameters ($\kappa_{0,i}$, $\kappa_{1,i}$, thresholds, cooperativity parameters) for all possible combinations of discrete quantities. Hence, we focus on the problem of invalidating families of unate models on the basis of the dataset $\mathscr{D}$ independently of the value of continuous parameters.

For ease of exposition, we start by assuming that data are noiseless and return to the case where measurement noise is present in Section 4.1. Moreover, since the problem is the same for all genes, we drop the index i to simplify the notation. In [1] we addressed the invalidation problem by exploiting monotonicity properties of $g(x)$. Given a model in the form (4)–(5), we define its sign pattern $p = (p_1, \ldots, p_n) \in \{-1, 0, 1\}^n$ by

$$p_j = \begin{cases} 0, & \text{if } j \notin J_l, \ l = 1, \ldots, h, \\ 1, & \text{if } \sigma^{\pm}(x_j) = \sigma^+(x_j), \\ -1, & \text{if } \sigma^{\pm}(x_j) = \sigma^-(x_j). \end{cases}$$

Note that several alternative structures of (5) share the same sign pattern p, e.g. $b(x) = \sigma^+(x_1)\sigma^-(x_2)$ and $b(x) = 1 - (1 - \sigma^+(x_1))\,(1 - \sigma^-(x_2))$, corresponding to the Boolean formulas $X_1 \wedge \neg X_2$ and $X_1 \vee \neg X_2$, respectively, are both represented by $p = (1, -1)$. For this reason, we introduce the family $U(p)$ of unate models $g(x)$ given by (4)-(5) with sign pattern p. In light of Definition 1, $g \in U(p)$ (and, similarly, the corresponding function b) is nondecreasing (respectively, nonincreasing) in x_j if $p_j = 1$ (respectively, $p_j = -1$), and is independent of x_j if $p_j = 0$.

Proposition 1
For any $g \in U(p)$ and $l, k \in \{1, \ldots, m\}$ it holds that

$$\Big[p_j(x_j^k - x_j^l) \geq 0, \ j = 1, \ldots, n\Big] \Rightarrow \Big[g(x^k) - g(x^l) \geq 0\Big]$$

DOI: 10.1002/rnc

where x_j^k indicates the jth entry of x^k.

Based on Proposition 1, the family $U(p)$ is invalidated by $\mathscr{D}$ if two indices $l, k \in \{1, \ldots, m\}$ exist such that

$$\left[p_j(x_j^k - x_j^l) \geq 0,\ j = 1, \ldots, n\right] \text{ and } \left[g^k - g^l < 0\right] \ . \tag{9}$$

In this case, p is said to be *inconsistent* (with $\mathscr{D}$).

Remark 1
In [1] we further introduced the concept of subpattern, i.e. p' is a subpattern of p (and p is a superpattern of p') if all nonzero entries of p' are equal to the corresponding entries of p. We showed that if p is inconsistent (respectively, consistent) then every subpattern (respectively, superpattern) p' is also inconsistent (respectively, consistent). This partial ordering relationship allowed us to characterize entire hierarchies of consistent patterns by way of minimal elements, and devise efficient algorithms for the storage and the enumeration of all consistent patterns.

2.3. A geometric approach to the invalidation of unate models

For a real-valued function g and $\varepsilon \in \mathbb{R}$, define the super-level set $T_\varepsilon(g) = \{x :\ g(x) \geq \varepsilon\}$ and the sub-level set $B_\varepsilon(g) = \{x :\ g(x) \leq \varepsilon\}$. We will now show that testing if a family $U(p)$ is invalidated by $\mathscr{D}$ can be done using inner approximations of sets $T_{g^k}(g)$, $k \in \{1, \ldots, m\}$, computed on the basis of the dataset $\mathscr{D}$ only. For $x \in \mathbb{R}^n$ and $p \in \{-1, 0, 1\}^n$, define the cone

$$\square^{\geq}(x, p) = \{z \in \mathbb{R}^n :\ p_j z_j \geq p_j x_j,\ j = 1, \ldots, n\}$$

with vertex x (see Fig. 1), that is the orthant defined by the nonzero entries of p, translated to x. Intuitively, for any $g \in U(p)$, p determines the direction of growth of g, hence $\square^{\geq}(x^k, p)$ is a region where g must be no smaller than g^k.
For a nonempty set $\mathscr{D}' \subseteq \mathscr{D}$, let $\mathcal{K}(\mathscr{D}') \subseteq \{1, \ldots, m\}$ be the set of indices of the elements of $\mathscr{D}$ contained in $\mathscr{D}'$ and define

$$M(\mathscr{D}', p) = \bigcup_{k \in \mathcal{K}(\mathscr{D}')} \square^{\geq}(x^k, p) \ , \qquad \mu(\mathscr{D}') = \min_{k \in \mathcal{K}(\mathscr{D}')} g^k \ .$$

The following result, that is illustrated in Fig. 1, shows that $M(\mathscr{D}', p)$ provides a data-based inner approximation of the set $T_{\mu(\mathscr{D}')}(g)$.

Proposition 2
For all nonempty sets $\mathscr{D}' \subseteq \mathscr{D}$, if $g \in U(p)$ then $M(\mathscr{D}', p) \subseteq T_{\mu(\mathscr{D}')}(g)$.

It is important to notice that the set $M(\mathscr{D}', p)$ depends on p but not on the particular $g \in U(p)$. This allows us to redefine inconsistent sign patterns as follows.

Definition 2
A sign pattern p is *m-inconsistent*[†] if there is a nonempty set $\mathscr{D}' \subseteq \mathscr{D}$ and $(x^*, g^*) \in \mathscr{D} \setminus \mathscr{D}'$ such that $x^* \in M(\mathscr{D}', p)$ and $g^* < \mu(\mathscr{D}')$. Otherwise p is *m-consistent*.

It can be shown that Definition 2 is equivalent to the definition of inconsistent sign pattern of Section 2, which means that one can only consider singleton sets $\mathscr{D}'$. In particular, the construction of the falsification region $M(\mathscr{D}', p)$, relies only on the monotonicity properties of g. However, the added benefit of this geometric approach becomes apparent when one focuses on subclasses of unate models, where larger falsification regions can be constructed using additional model properties. This is the approach we will follow in the next section.

[†]"m-" stands for monotone.

Prepared using rncauth.cls
DOI: 10.1002/rnc

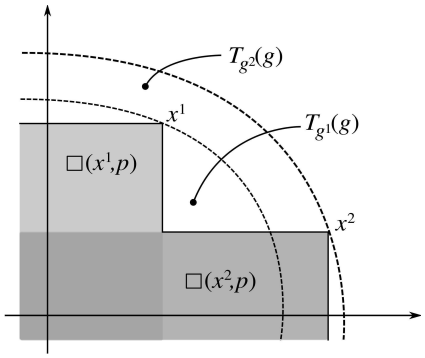


Figure 1. Cones $\square^{\geq}(x^1,p)$ and $\square^{\geq}(x^2,p)$ composing the set $M(\mathscr{D}')$ for $p=(-1,-1)$, $\mathscr{D}'=\{(x^1,g^1),(x^2,g^2)\}$ and $g^2=\mu(\mathscr{D}')$. Dashed lines show the boundaries of super-level sets $T_{g^1}(g)$ and $T_{g^2}(g)$.

3. QUASI-CONVEXITY ANALYSIS OF GENETIC NETWORK MODELS

Following on Example 1, we introduce subsets of unate models that are the algebraic counterpart of the $S_0 \cup S_1$ Boolean models.

Definition 3
An S_0 model is given by (3) and (4) where, for some $\ell \in \{0,\dots,n\}$ and some subset $\{j_1,\dots,j_\ell\}$ of $\{1,\dots,n\}$,

$$b(x) = \sigma^{\pm}(x_{j_1})\sigma^{\pm}(x_{j_2})\sigma^{\pm}(x_{j_3})\cdots\sigma^{\pm}(x_{j_\ell}) \ . \tag{10}$$

Similarly, an S_1 model is characterized by

$$b(x) = b_\vee(x)b_\wedge(x) \ , \tag{11a}$$
$$b_\vee(x) = \left[1-\left(1-\sigma^{\pm}(x_{j_1})\right)\left(1-\sigma^{\pm}(x_{j_2})\right)\right] \ , \tag{11b}$$
$$b_\wedge(x) = \sigma^{\pm}(x_{j_3})\sigma^{\pm}(x_{j_4})\dots\sigma^{\pm}(x_{j_\ell}) \ . \tag{11c}$$

Finally an $S_0 \cup S_1$ model is given by (3) and (4) if either (10) or (11) holds.

In the sequel, $S_0(p)$ will denote the family of S_0 models $g(x)$ given by (4) and (10) with sign pattern p. Note that p defines an S_0 model up to the values of the kinetic and sigmoid parameters. In the case of S_1 models, the structure is parametrized by triplets $(j_\vee, j_\wedge, p)$, where p is a sign pattern and $j_\vee = \{j_1, j_2\}$, $j_\wedge = \{j_3,\dots,j_\ell\}$ are sets of indices partitioning $\{i \in \{1,\dots,n\} : p_i \neq 0\}$. We denote by $S_1(j_\vee, j_\wedge, p)$ the family of S_1 models sharing the same structure $(j_\vee, j_\wedge, p)$, while $S_1(p)$ is the union of all families $S_1(j_\vee, j_\wedge, p)$ sharing the same sign pattern p.

Remark 2
Note that if we allow for $j_\vee = \emptyset$ (and $j_\wedge = \{i \in \{1,\dots,n\} : p_i \neq 0\}$) in S_1 models, we recover S_0 models. This suggests that one can focus on the invalidation of families of S_1 models only. However, invalidation methods for S_0 models are simpler and hence, for the sake of clarity, we will discuss the S_0 and S_1 cases separately.

Next, we will show that $S_0 \cup S_1$ models have quasi-convexity properties that can be used for invalidating entire families $S_0(p)$ or $S_1(j_\vee, j_\wedge, p)$ using data in $\mathscr{D}$ only. Let us begin with some basic definitions and results of convex analysis (see [23] for more details).

Definition 4
Let $D \subseteq \mathbb{R}^n$ be a convex set. A function $g : D \to \mathbb{R}$ is *quasi-convex* if the following equivalent conditions hold:

DOI: 10.1002/rnc

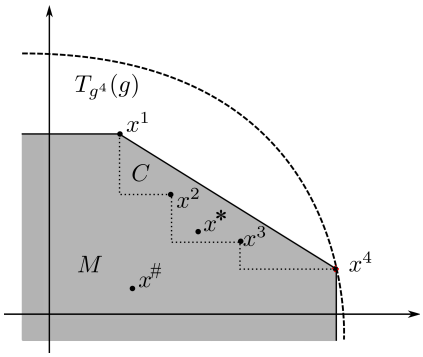


Figure 2. Sets $C(\mathscr{D}', p)$ (gray area) and $M(\mathscr{D}', p)$ (subset of gray area left of the dotted lines) for $p = (-1, -1)$, $\mathscr{D}' = \{(x^i, g^i), i = 1, \ldots, 4\}$ and $g^4 = \mu(\mathscr{D}')$. The dashed line shows the boundary of the super-level set $T_{g^4}(g)$. Point $x^\#$ represents a test point that allows both m- and c-inconsistency falsification, whereas x^* allows for c-inconsistency falsification only.

i. for every $\alpha \in [0, 1]$ and every $x, y \in D$,

$$g\big(\alpha x + (1-\alpha)y\big) \leq \max\{g(x), g(y)\} \ ; \tag{12}$$

ii. for every $\varepsilon \in \mathbb{R}$, the sub-level set $B_\varepsilon(g)$ is convex.

g is *quasi-concave* if $-g$ is quasi-convex, that is:

i'. for every $\alpha \in [0, 1]$ and every $x, y \in D$,

$$f\big(\alpha x + (1-\alpha)y\big) \geq \min\{g(x), g(y)\} \ ; \tag{13}$$

ii'. for every $\varepsilon \in \mathbb{R}$, the super-level set $T_\varepsilon(g)$ is convex.

The following proposition is the basis for new invalidation procedures.

Proposition 3
Function $b(x)$ in (10) is quasi-concave (with respect to $(x_{j_1} \ldots, x_{j_\ell})$). Function $b_\vee(x)$ in (11b) is quasi-convex (with respect to (x_{j_1}, x_{j_2})), while $b_\wedge(x)$ in (11c) is quasi-concave (with respect to $(x_{j_3} \ldots, x_{j_\ell})$).

Note that Proposition 3 strongly relies on the standing assumption that sigmoids are logistic functions (see Section 2) but is completely independent of the signs of the sigmoids and the values of the cooperativity and threshold parameters. Moreover, quasi-convexity is not affected by multiplication by and addition of nonnegative scalars and hence $g(x)$ is quasi-convex if and only if $b(x)$ has the same property. In practice, this will allow us to infer properties of $b(x)$ from data generated by the function $g(x)$. We now apply these results to the invalidation of S_0 models (Section 3.1) and S_1 models (Section 3.2).

3.1. Invalidation of S_0 models

For any $\mathscr{D}' \subseteq \mathscr{D}$ and any sign pattern p let $C(\mathscr{D}', p) = \text{Conv}(M(\mathscr{D}', p))$. The following result, that is illustrated in Fig. 2, shows that $C(\mathscr{D}', p)$ provides a data-based inner approximation of the set $T_{\mu(\mathscr{D}')}(g)$ and hence leads to a new definition of inconsistent S_0 models.

Proposition 4
If $g \in S_0(p)$, then $C(\mathscr{D}', p) \subseteq T_{\mu(\mathscr{D}')}(g)$.

DOI: 10.1002/rnc

Definition 5
A sign pattern p is *c-inconsistent*[‡] if there is a (nonempty) set $\mathscr{D}' \subseteq \mathscr{D}$ and $(x^*, g^*) \in \mathscr{D} \setminus \mathscr{D}'$ such that $x^* \in C(\mathscr{D}', p)$ and $g^* < \mu(\mathscr{D}')$. Otherwise p is *c-consistent*.

Definition 5 strengthens Definition 2 for S_0 models since $C(\mathscr{D}', p) \supseteq M(\mathscr{D}', p)$. In other words, if p is m-inconsistent, no $g \in U(p)$ (and hence no $g \in S_0(p)$) can generate the dataset $\mathscr{D}$. However, if p is m-consistent the whole family $S_0(p)$ is still invalidated if p is c-inconsistent. Relations between m- and c-inconsistency are illustrated in the following example.

Example 2
With reference to Fig. 2, consider the dataset $\mathscr{D} = \mathscr{D}' \cup \{(x^\#, g^*)\}$, $\mathscr{D}' = \{(x^i, g^i), i = 1, \dots, 4\}$ and the sign pattern $p = (-1, -1)$. If $g^* < \min_{i=1,\dots,4} g^i$ one has $x^\# \in M(\mathscr{D}', p)$ and $g^* < \mu(\mathscr{D}')$ and therefore, according to Definition 2, the sign pattern p is m-inconsistent. This means that all models in $U(p)$ must be rejected, including $k_0 + k_1\sigma^-(x_1)\sigma^-(x_2)$ that is the only model in $S_0(p)$. If instead $\mathscr{D} = \mathscr{D}' \cup \{(x^*, g^*)\}$ because $x^* \in C(\mathscr{D}', p)$, according to Definition 5, the sign pattern p is c-inconsistent. This means that the model in $S_0(p)$ has to be rejected. However $x^* \notin M(\mathscr{D}', p)$ and therefore no model in $U(p)$ can be invalidated based on m-inconsistency of p.

3.2. Invalidation of S_1 models

For S_1 models a convexity-like property does not globally hold. Hence the goal is to combine the different properties of (11b) and of (11c). There are different ways to do so, each leading to different conditions for the invalidation of model structures. For a generic $z \in \mathbb{R}^d$ and $p \in \{-1, 0, 1\}^d$, recall the definition of the cone

$$\square^{\geq}(z, p) = \{z' \in \mathbb{R}^d : \; p_j z'_j \geq p_j z_j, \; \forall j = 1, \dots, d\} \tag{14a}$$

and define the cone

$$\square^{\leq}(z, p) = \square^{\geq}(z, -p) = \{z' \in \mathbb{R}^d : \; p_j z'_j \leq p_j z_j, \; \forall j = 1, \dots, d\} \; . \tag{14b}$$

For sets of indices $j_\vee = \{j_1, j_2\}$, $j_\wedge = \{j_3, \dots, j_\ell\}$ and a sign pattern p, let $p_\vee = (p_{j_1}, p_{j_2})$ and $p_\wedge = (p_{j_3}, \dots, p_{j_\ell})$. Similarly, for any vector $x \in \mathbb{R}^n$, let $x_\vee = (x_{j_1}, x_{j_2})$ and $x_\wedge = (x_{j_3}, \dots, x_{j_\ell})$. To emphasize that $b_\vee$ and $b_\wedge$ depend only on $x_\vee$ and $x_\wedge$, respectively, with an abuse of notation we will write $b_\vee(x_\vee)$ and $b_\wedge(x_\wedge)$ in place of $b_\vee(x)$ and $b_\wedge(x)$. For any nonempty subset $\mathscr{D}'$ of $\mathscr{D}$, define the sets

$$L_{\max,\vee}(\mathscr{D}', p_\vee) = \text{Conv}\left(\bigcup_{k \in \mathcal{K}(\mathscr{D}')} \square^{\leq}(x^k_\vee, p_\vee)\right), \tag{15a}$$

$$U_{\max,\vee}(\mathscr{D}', p_\vee) = \bigcap_{k \in \mathcal{K}(\mathscr{D}')} \square^{\geq}(x^k_\vee, p_\vee) \; , \tag{15b}$$

$$L_{\min,\wedge}(\mathscr{D}', p_\wedge) = \bigcap_{k \in \mathcal{K}(\mathscr{D}')} \square^{\leq}(x^k_\wedge, p_\wedge) \; , \tag{15c}$$

$$U_{\min,\wedge}(\mathscr{D}', p_\wedge) = \text{Conv}\left(\bigcup_{k \in \mathcal{K}(\mathscr{D}')} \square^{\geq}(x^k_\wedge, p_\wedge)\right). \tag{15d}$$

The next proposition clarifies the approximation properties of the various sets L and U in (15).

[‡]"c-" stands for convex.

Prepared using rncauth.cls
DOI: 10.1002/rnc

Proposition 5
Let $\mathscr{M}_\vee(\mathscr{D}') = \max\{b_\vee(x_\vee^k) : \ k \in \mathcal{K}(\mathscr{D}')\}$ and $\mu_\wedge(\mathscr{D}') = \min\{b_\wedge(x_\wedge^k) : \ k \in \mathcal{K}(\mathscr{D}')\}$. Then,

$$L_{\max,\vee}(\mathscr{D}', p_\vee) \subseteq B_{\mathscr{M}_\vee(\mathscr{D}')}(b_\vee) \ , \tag{16a}$$
$$U_{\max,\vee}(\mathscr{D}', p_\vee) \subseteq T_{\mathscr{M}_\vee(\mathscr{D}')}(b_\vee) \ , \tag{16b}$$
$$L_{\min,\wedge}(\mathscr{D}', p_\wedge) \subseteq B_{\mu_\wedge(\mathscr{D}')}(b_\wedge) \ , \tag{16c}$$
$$U_{\min,\wedge}(\mathscr{D}', p_\wedge) \subseteq T_{\mu_\wedge(\mathscr{D}')}(b_\wedge) \ , \tag{16d}$$

or, equivalently,

$$x_\vee \in L_{\max,\vee}(\mathscr{D}', p_\vee) \implies b_\vee(x_\vee) \leq \mathscr{M}_\vee(\mathscr{D}') \ , \tag{17a}$$
$$x_\vee \in U_{\max,\vee}(\mathscr{D}', p_\vee) \implies b_\vee(x_\vee) \geq \mathscr{M}_\vee(\mathscr{D}') \ , \tag{17b}$$
$$x_\wedge \in L_{\min,\wedge}(\mathscr{D}', p_\wedge) \implies b_\wedge(x_\wedge) \leq \mu_\wedge(\mathscr{D}') \ , \tag{17c}$$
$$x_\wedge \in U_{\min,\wedge}(\mathscr{D}', p_\wedge) \implies b_\wedge(x_\wedge) \geq \mu_\wedge(\mathscr{D}') \ . \tag{17d}$$

According to (17), points in the various sets U and L provide upperbounds and lowerbounds to minima and maxima of $b_\vee$ and $b_\wedge$ over $\mathscr{D}'$. For example, from (17a) one has that $b_\vee(x_\vee)$ is a Lowerbound (L) to the maximum (max) $\mathscr{M}_\vee(\mathscr{D}')$, hence the notation $L_{\max,\vee}$. Similarly, $b_\wedge(x_\wedge)$ is an Upperbound (U) to the minimum (min) $\mu_\wedge(\mathscr{D}')$, whence $U_{\min,\wedge}$. Equivalently, sets U and L provide inner approximations for the various sets T and B. The idea is now to combine these results to establish inequalities for $b_\vee(x_\vee)b_\wedge(x_\wedge)$, and hence for the measured values of $g(x) = \kappa_0 + \kappa_1 b_\vee(x_\vee)b_\wedge(x_\wedge)$. Recall that $\mu(\mathscr{D}') = \min\{g^k : \ k \in \mathcal{K}(\mathscr{D}')\}$ and let $\mathscr{M}(\mathscr{D}') = \max\{g^k : \ k \in \mathcal{K}(\mathscr{D}')\}$.

Proposition 6
If $g \in S_1(j_\vee, j_\wedge, p)$, for any nonempty set $\mathscr{D}' \subseteq \mathscr{D}$ one has

$$\{x \in \mathbb{R}^n : x_\vee \in L_{\max,\vee}(\mathscr{D}', p_\vee), x_\wedge \in L_{\min,\wedge}(\mathscr{D}', p_\wedge)\} \subseteq B_{\mathscr{M}(\mathscr{D}')}(g) \ ,$$
$$\{x \in \mathbb{R}^n : x_\vee \in U_{\max,\vee}(\mathscr{D}', p_\vee), x_\wedge \in U_{\min,\wedge}(\mathscr{D}', p_\wedge)\} \subseteq T_{\mu(\mathscr{D}')}(g) \ .$$

The inner approximations provided by Proposition 6 lead to the following criteria for the invalidation of the family $S_1(j_\vee, j_\wedge, p)$.

Definition 6
The structure $(j_\vee, j_\wedge, p)$ is *c-inconsistent* if there exists a nonempty set $\mathscr{D}' \subseteq \mathscr{D}$ and a data point $(x^*, g^*) \in \mathscr{D} \setminus \mathscr{D}'$ such that either of the following conditions applies:

(I) $x_\vee^* \in L_{\max,\vee}(\mathscr{D}', p_\vee)$, $x_\wedge^* \in L_{\min,\wedge}(\mathscr{D}', p_\wedge)$, $g^* > \mathscr{M}(\mathscr{D}')$

(II) $x_\vee^* \in U_{\max,\vee}(\mathscr{D}', p_\vee)$, $x_\wedge^* \in U_{\min,\wedge}(\mathscr{D}', p_\wedge)$, $g^* < \mu(\mathscr{D}')$

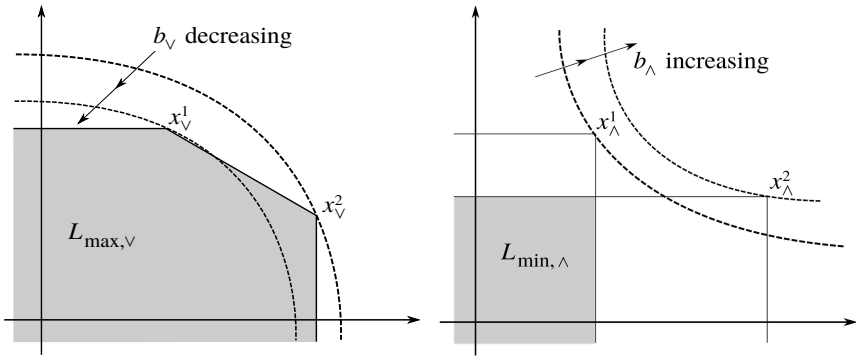
The structure $(j_\vee, j_\wedge, p)$ is *c-consistent* if it is not c-inconsistent.

Definition 6, that is illustrated in Fig. 3, is a strengthening of Definition 2 to S_1 models. In particular, when $|\mathscr{D}'| = 1$, condition (II) is equivalent to the condition in Definition 2. Therefore, if p is m-inconsistent, no $g \in U(p)$ (and hence no $g \in S_1(p)$) can generate the dataset $\mathscr{D}$. However, if p is m-consistent the family $S_1(j_\vee, j_\wedge, p)$ is still invalidated if the structure $(j_\vee, j_\wedge, p)$ is c-inconsistent.

Remark 3
Following on Remark 2, if one sets $j_\vee = \emptyset$, conditions $x_\vee^* \in L_{\max,\vee}(\mathscr{D}', p_\vee)$ and $x_\vee^* \in U_{\max,\vee}(\mathscr{D}', p_\vee)$ become empty. In this case, condition (II) in Definition 6 coincides with the condition in Definition 5. Moreover, with $j_\vee = \emptyset$, it is possible to show that condition (I) in Definition 6 is encompassed by m-inconsistency conditions and hence by condition (II). Therefore, c-inconsistency of the structure $(\emptyset, j_\wedge, p)$ is equivalent to c-inconsistency of the sign pattern p.

We highlight that inconsistency conditions in Definition 6 could be easily adapted to address the more general case of models with $b(x) = \hat{b}(\hat{x})\check{b}(\check{x})$, where $\hat{x}$ and $\check{x}$ are distinct subvectors of x, $\hat{b}(\hat{x})$ is quasi-concave and $\check{b}(\check{x})$ is quasi-convex.

DOI: 10.1002/rnc



(a) Set $L_{\max,\vee}(\mathscr{D}', p_\vee)$ in the $x_\vee$-plane. (b) Set $L_{\min,\wedge}(\mathscr{D}', p_\wedge)$ in the $x_\wedge$-plane.

Figure 3. Sets in condition (I) of Definition 6 for $x \in \mathbb{R}^4$, $\mathscr{D}' = \{(x^1, g^1), (x^2, g^2)\}$ and $p_\vee = p_\wedge = (1, 1)$. Dashed lines indicate the boundaries of level sets of $b_\vee$ (left) and $b_\wedge$ (right).

3.3. Hierarchical properties of c-inconsistency

Following on Remark 1, for $S_0 \cup S_1$ models, it is possible to establish hierarchical relationships among the *model structures* invalidated by the data. To this purpose, we allow for $j_\vee = \emptyset$ and introduce a partial order relating nested model structures. The structure $s' = (j'_\vee, j'_\wedge, p')$ is a *substructure* of $s = (j_\vee, j_\wedge, p)$ (and s is a *superstructure* of s') if p' is a subpattern of p (see Remark 1), $j'_\vee \subseteq j_\vee$ and $j'_\wedge \subseteq j_\wedge$.

Using arguments similar to [1] for the m-inconsistency analysis, it is possible to show that the following properties hold:

- if a structure is c-inconsistent, then all its substructures are c-inconsistent;
- if a structure is c-consistent, then all its superstructures are c-consistent.

Such properties have two important consequences. First, they allow one to avoid testing c-consistency of a structure if a substructure (respectively, a superstructure) is already found to be c-consistent (respectively, c-inconsistent). Second, it is possible to provide a compact description of the hierarchy of c-consistent structures by means of its minimal elements (with respect to the substructure partial order). These features are exploited to set up an efficient exploration of all possible structures, as shown in the next section.

4. ALGORITHMS AND IMPLEMENTATION

An efficient method for testing m-inconsistency was proposed in [1]. The procedure is based on Proposition 1 and hierarchical properties of sign patterns (see Remark 1) rather than the geometric approach discussed in Section 2.3. Here we are concerned with the practical use of Definitions 5–6 for testing c-inconsistency of $S_0 \cup S_1$ models. In the following discussion, in view of Remark 3, we will allow $j_\vee$ to be empty and focus on Definition 6 only.

A direct application of Definition 6 is impractical since conditions (I) and (II) must be checked for all subsets of $\mathscr{D}$. In the sequel we show that, without loss of generality, it is possible to check inconsistency of a structure by constructing only two subsets of $\mathscr{D}$ for each $(x^*, g^*) \in \mathscr{D}$.

Proposition 7
A structure $(j_\vee, j_\wedge, p)$ is c-inconsistent if and only if there exists $(x^*, g^*) \in \mathscr{D}$ such that either of the following conditions apply:

(I') $x_\vee^* \in L_{\max,\vee}(\mathscr{D}_L, p_\vee)$ where

$$\mathscr{D}_L = \{(x, g) \in \mathscr{D} \setminus \{(x^*, g^*)\} : g < g^*,\ \mathrm{diag}(p_\wedge)(x_\wedge - x_\wedge^*) \geq 0\} \quad ; \tag{18}$$

DOI: 10.1002/rnc

Algorithm 1 c-inconsistency test for a structure $s = (j_\vee, j_\wedge, p)$

1: label s as c-consistent
2: **for** all $(x^*, g^*) \in \mathscr{D}$ **do**
3: compute $\mathscr{D}_L$ as in (18). If $x^*_\vee \in L_{\max,\vee}(\mathscr{D}_L, p_\vee)$ label s as c-inconsistent and exit.
4: compute $\mathscr{D}_U$ as in (19). If $x^*_\wedge \in U_{\min,\wedge}(\mathscr{D}_U, p_\wedge)$ label s as c-inconsistent and exit.
5: **end for**

(II') $x^*_\wedge \in U_{\min,\wedge}(\mathscr{D}_U, p_\wedge)$ where

$$\mathscr{D}_U = \{(x, g) \in \mathscr{D} \setminus \{(x^*, g^*)\} : g > g^*,\ \mathrm{diag}(p_\vee)(x_\vee - x^*_\vee) \leq 0\} \quad . \tag{19}$$

The rightmost inequalities in (18) and (19) are interpreted componentwise.

The complete method for testing c-inconsistency of a given structure is summarized in Algorithm 1. Note that, when $j_\vee = \emptyset$ (respectively, $j_\wedge = \emptyset$), condition (I') (respectively, condition (II')) in Proposition 7 is not of interest hence one can ignore line 3 (respectively, line 4) of Algorithm 1.

For the efficient implementation of Algorithm 1, it is crucial to have a computationally efficient method for verifying conditions $x^*_\vee \in L_{\max,\vee}(\mathscr{D}_L, p_\vee)$ and $x^*_\wedge \in U_{\min,\wedge}(\mathscr{D}_U, p_\wedge)$. In particular, we would like to avoid computing convex hulls in (15a) and (15d). Inspired by [12], we propose a solution based on Linear Programs (LPs). It is easy to see that conditions (I') and (II') are both instances of the following problem: given points $z^1, \ldots, z^K, z^* \in \mathbb{R}^d$ and $p \in \{-1, 1\}^d$, check if

$$z^* \in \mathcal{Z} = \mathrm{Conv}\left(\bigcup_{k=1}^{K} \{z \in \mathbb{R}^d :\ p_j z_j \geq p_j z^k_j,\ \forall j = 1, \ldots, d\}\right) \quad . \tag{20}$$

In an equivalent way, (20) is false if and only if there exists a hyperplane $h^T z = h_0$, for some $h \in \mathbb{R}^d$ and $h_0 \in \mathbb{R}$, separating z^* from the polyhedron $\mathcal{Z}$. It is easily seen that if such a hyperplane exists, then one exists with normal direction aligned with p, i.e. fulfilling $p_i h_i \geq 0$ for all $i = 1, \ldots, d$. Under this condition, one can just seek a hyperplane passing through z^* and such that $h^T z^k > h_0$ for all points z^k. Testing condition (20) then amounts to solve the LP

$$\begin{aligned} \max_{\delta \in \mathbb{R}, h \in \mathbb{R}^d} \quad & \delta \\ \text{s.t.} \quad & p_i h_i \geq 0, \forall i = 1, \ldots, d \\ & h^T (z^k - z^*) - \delta \geq 0, \forall k = 1, \ldots, K \\ & \sum_{i=1}^{n} p_i h_i = 1 \end{aligned} \tag{21}$$

and then check if the optimal cost is nonpositive. It is possible to show that (21) is always feasible, with the last constraint ensuring boundedness. By the use of the LP (21), one execution of Algorithm 1 amounts in the worst case to solving $|\mathscr{D}|$ LPs in $|j_\vee| + 1$ variables and $|\mathscr{D}|$ LPs in $|j_\wedge| + 1$ variables, with one equality constraint and at most $|j_\vee| + |\mathscr{D}| - 1$ and $|j_\wedge| + |\mathscr{D}| - 1$ inequality constraints, respectively.

As already mentioned, if the pattern p is m-inconsistent, all structures $(j_\vee, j_\wedge, p)$ are also c-inconsistent. However, Algorithm 1 requires the solution of LPs, making c-inconsistency tests more computationally demanding than m-inconsistency tests, for which the efficient algorithm in [1] can be applied. Therefore, substantial computational savings can be achieved using the procedure reported in Algorithm 2 for computing the set $S_{\min}$ of minimal (with respect to the partial order on structures) c-consistent structures, which exploits the m-inconsistency analysis and the hierarchical properties discussed in Section 3.3.

The correctness of Algorithm 2 (i.e. the fact that the output $S_{\min}$ is indeed the set of all minimal consistent structures) can be proven by the same techniques used in [1] for the computation of

DOI: 10.1002/rnc

Algorithm 2 Computation of the set $S_{\min}$ of minimal c-consistent structures

```
1: initialize S_min = ∅;
2: assess m-inconsistency of sign patterns by means of the algorithm in [1];
3: for ℓ = 1, ..., n do
4:    for all structures s = (j_∨, j_∧, p) such that p is m-consistent and has ℓ nonzero entries do
5:       if there is no structure s' ∈ S_min such that s' is a substructure of s then
6:          if s is found c-consistent by Algorithm 1 then
7:             S_min = {s} ∪ S_min;
8:          end if
9:       end if
10:   end for
11: end for
```

minimal consistent sign patterns. In particular, ℓ in line 3 is the number of regulating genes, and hence Algorithm 2 tests c-consistency of simpler structures first. Algorithm 2 was implemented in Matlab 7.10 (R2010a), resorting to the free solver CDD [24] and its Matlab interface CDDmex [25] for solving the LP (21).

4.1. Handling noisy data

To deal with noisy measurements of (x^k, g^k) in $\mathscr{D}$, we follow a *robust* approach. We assume lower and upper bounds $l(\cdot)$ and $u(\cdot)$ to be available for the true values of g^k and x^k_j, for all $k = 1, \dots, m$ and $j = 1, \dots, n$. This means that every x^k is surrounded by an uncertainty box. The example in the next section shows that this approach is still viable in the case of Gaussian (unbounded) noise affecting the data. The idea is to robustify all inconsistency conditions by defining worst-case scenarios that take bounded uncertainty into account. For what concerns g, conditions $g^* < \mu(\mathscr{D}')$ and $g^* > \mathscr{M}(\mathscr{D}')$ are replaced by

$$u(g^*) < \min_{k \in \mathcal{K}(\mathscr{D}')} l(g^k) \quad \text{and} \quad l(g^*) > \max_{k \in \mathcal{K}(\mathscr{D}')} u(g^k) \ , \tag{22}$$

respectively. Conditions on points x all involve sets computed as combinations (union, intersection, convex hull) of cones. In this case we consider worst-case inner approximations of such sets obtained by replacing $\square^{\geq}(z,p)$, $\square^{\leq}(z,p)$ with

$$\begin{aligned} \widetilde{\square}^{\geq}(z,p) = \{z' \in \mathbb{R}^d : \ & z'_j \geq u(z_j), \ \forall j \text{ such that } p_j = 1, \\ & z'_j \leq l(z_j), \ \forall j \text{ such that } p_j = -1\} \ , \end{aligned} \tag{23a}$$

$$\begin{aligned} \widetilde{\square}^{\leq}(z,p) = \widetilde{\square}^{\geq}(z,-p) = \{z' \in \mathbb{R}^d : \ & z'_j \leq l(z_j), \ \forall j \text{ such that } p_j = 1, \\ & z'_j \geq u(z_j), \ \forall j \text{ such that } p_j = -1\} \ . \end{aligned} \tag{23b}$$

respectively. Moreover, the uncertainty of the test point x^* is also taken into account by considering the point $\tilde{x}^*$ instead, with

$$\tilde{x}^*_j = \begin{cases} l(x^*_j), & j \text{ such that } p_j = 1, \\ u(x^*_j), & j \text{ such that } p_j = -1, \end{cases} \tag{24a}$$

or

$$\tilde{x}^*_j = \begin{cases} u(x^*_j), & j \text{ such that } p_j = 1, \\ l(x^*_j), & j \text{ such that } p_j = -1, \end{cases} \tag{24b}$$

DOI: 10.1002/rnc

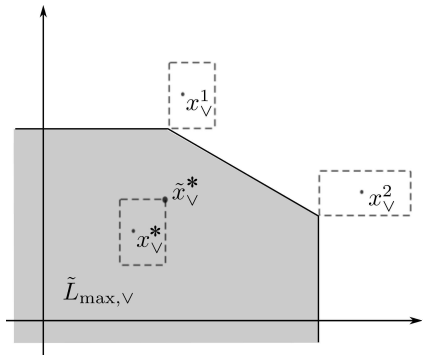


Figure 4. Uncertainty boxes (dashed), set $\tilde{L}_{\max,\vee}(\mathscr{D}', p_\vee)$ in (25) and robustified test point $\tilde{x}^*_\vee$ for $\mathscr{D}' = \{x^1, x^2\}$ and $p_\vee = (1, 1)$.

for conditions involving $\square^{\geq}$ or $\square^{\leq}$ cones, respectively. As an example, condition $x^*_\vee \in L_{\max,\vee}(\mathscr{D}', p_\vee)$ for $p_\vee = (1, 1)$ is replaced by testing if $\tilde{x}^*_\vee = (u(x^*_{j_1}), u(x^*_{j_2}))$ belongs to the set

$$\tilde{L}_{\max,\vee}(\mathscr{D}', p_\vee) = \text{Conv}\left(\bigcup_{k \in \mathcal{K}(\mathscr{D}')} \widetilde{\square}^{\leq}(x^k_\vee, p_\vee) \right) , \tag{25}$$

as represented in Fig. 4, to be compared to Fig. 3(a).

We highlight that, for conditions involving convex hulls, this approach is equivalent to replacing the LP (21) with

$$\begin{aligned} \max_{\delta \in \mathbb{R}, h, z^1, \ldots, z^K, z^* \in \mathbb{R}^d} \quad & \delta \\ \text{s.t.} \quad & p_i h_i \geq 0, \forall i = 1, \ldots, d \\ & h^T (z^k - z^*) - \delta \geq 0, \forall k = 1, \ldots, K \\ & \sum_{i=1}^{n} p_i h_i = 1 \\ & l(z^*) \leq z^* \leq u(z^*) \\ & l(z^k) \leq z^k \leq u(z^k), \forall k = 1, \ldots, K \end{aligned}$$

Roughly speaking, larger uncertainties, that correspond to higher $u(x^k_j) - l(x^k_j)$ and $u(g^k) - l(g^k)$, result in a lower chance that a structure is declared c-inconsistent. In particular, (23)–(24) shrink sets L and U in (15) and shift the test point x^*, resulting in a smaller number of test points that can be used for falsifying a structure.

5. EXAMPLES AND DISCUSSION OF THE RESULTS

In order to assess the falsification capability provided by quasi-convexity properties, we considered the same artificial network introduced in [1] for evaluating the performance of the m-inconsistency analysis. The network, represented in Fig. 5, comprises 6 genes and several interactions. In particular, genes 1–3 represent the core oscillating part of the system and correspond to the repressilator network developed and synthesized in *Escherichia coli* [26]. The remaining three genes are those of interest in our study and are regulated by the three core genes according to different

DOI: 10.1002/rnc

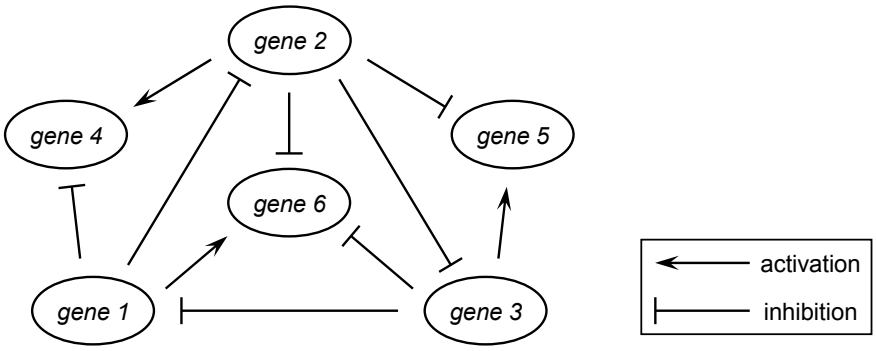


Figure 5. Artificial regulatory network: repressilator loop (genes 1–3) plus controlled genes 4–6.

logical rules. The dynamics of this part of the network is modeled by

$$\dot{x}_4 = \kappa_{0,4} + \kappa_{1,4}\sigma^-(x_1)\sigma^+(x_2) - \gamma_4 x_4 \ , \tag{26a}$$

$$\dot{x}_5 = \kappa_{0,5} + \kappa_{1,5}[1 - \sigma^+(x_2)\sigma^-(x_3)] - \gamma_5 x_5 \ , \tag{26b}$$

$$\dot{x}_6 = \kappa_{0,6} + \kappa_{1,6}[1 - \sigma^+(x_2)\sigma^+(x_3)]\sigma^+(x_1) - \gamma_6 x_6 \ . \tag{26c}$$

where x_i, $i = 1, \ldots, 6$, denotes the concentration of the ith gene product and $\sigma^{\pm}(\cdot)$ are Hill functions. We defer the reader to the supplementary material of [1] for the details about the complete model and the parameter values. It is easy to recognize regulation functions in both S_0 (gene 4) and S_1 (genes 5 and 6).

We have tested the performance of both the m-inconsistency analysis introduced in [1] and its combined use with c-inconsistency analysis developed in this work on the following datasets. The model was simulated for 15 time units[§], i.e. until 3 full oscillations were completed. To evaluate the sensitivity of our approach to the amount of data, we produced three datasets comprising $m = 45$, 23 and 12 equally spaced data points. In order to assess the impact of measurement errors, noisy synthesis rate and concentration data $\tilde{g}^k$ and $\tilde{x}^k$ were obtained by corrupting g^k and x^k with multiplicative noise, i.e., for $i = 1, \ldots, n$,

$$\tilde{x}_i^k = x_i^k(1 + s_e e_i^k) \ , \tag{27}$$

$$\tilde{g}_i^k = g_i^k(1 + s_\epsilon \epsilon_i^k) \ , \tag{28}$$

with e_i^k and ϵ_i^k mutually uncorrelated Gaussian random variables with zero mean and unit variance [1]. The scaling factors s_e, s_ϵ were chosen in the set $\{0.03, 0.05, 0.07\}$. Recalling that, approximately, noise samples $s_e e_i^k$ fall within $\pm 3 s_e$ with 0.99 probability (and similarly for $s_\epsilon \epsilon_i^k$), we considered noise contributions ranging from the 9% to the 21% of the noiseless data values. In order to establish lower and upper bounds to the data, as assumed in Section 4.1, we used 95% confidence intervals resulting in

$$l(\tilde{x}_i^k) = \tilde{x}_i^k(1 - 2s_e) \ , \qquad u(\tilde{x}_i^k) = \tilde{x}_i^k(1 + 2s_e) \ , \tag{29}$$

$$l(\tilde{g}_i^k) = \tilde{g}_i^k(1 - 2s_\epsilon) \ , \qquad u(\tilde{g}_i^k) = \tilde{g}_i^k(1 + 2s_\epsilon) \ . \tag{30}$$

Since this gene network model involves Hill functions, the log-transformation (7) was applied to the data prior to the execution of Algorithm 1.

We are interested in comparing the falsification performance of the c-consistency analysis to the case when only m-inconsistency is used, i.e. when only the method of [1] is applied. To this purpose, let $N_{\text{c-inc}}$ be the number of c-inconsistent structures and $N_{\text{m-inc}}$ the number of m-inconsistent structures. The performance index $I_{\%} \geq 0$ is defined as

$$I_{\%} = \frac{N_{\text{c-inc}} - N_{\text{m-inc}}}{N_{\text{m-inc}}} \cdot 100 \tag{31}$$

[§]The definition of the time unit is unimportant in our study.

DOI: 10.1002/rnc

Table I. Performance results on the example network.

m	gene	s_e,s_ϵ 0.03	s_e,s_ϵ 0.05	s_e,s_ϵ 0.07
45	4	$S_\%$=47.60 $I_\%$= 5.07	$S_\%$=42.30 $I_\%$= 5.35	$S_\%$=37.06 $I_\%$= 6.47
	5	$S_\%$=41.46 $I_\%$= 6.65	$S_\%$=31.23 $I_\%$= 8.54	$S_\%$=23.84 $I_\%$=12.08
	6	$S_\%$=40.19 $I_\%$= 6.01	$S_\%$=36.36 $I_\%$= 8.30	$S_\%$=31.25 $I_\%$=10.53
23	4	$S_\%$=45.00 $I_\%$= 7.92	$S_\%$=38.57 $I_\%$= 8.73	$S_\%$=32.86 $I_\%$= 9.28
	5	$S_\%$=34.81 $I_\%$= 8.65	$S_\%$=26.39 $I_\%$=12.13	$S_\%$=19.03 $I_\%$=13.03
	6	$S_\%$=37.66 $I_\%$=10.09	$S_\%$=32.86 $I_\%$=12.97	$S_\%$=27.29 $I_\%$=14.18
12	4	$S_\%$=41.78 $I_\%$= 9.63	$S_\%$=33.37 $I_\%$=10.84	$S_\%$=25.28 $I_\%$=10.99
	5	$S_\%$=29.44 $I_\%$=11.54	$S_\%$=22.37 $I_\%$=12.76	$S_\%$=16.62 $I_\%$=12.19
	6	$S_\%$=32.82 $I_\%$=10.88	$S_\%$=27.88 $I_\%$=12.87	$S_\%$=23.44 $I_\%$=12.45

and the larger the $I_\%$, the larger the percentage of $S_0 \cup S_1$ models invalidated by c-inconsistency but not by m-inconsistency that is, the larger the increase in performance with respect to the method in [1]. In order to quantify the fraction of all structures that are falsified, we also introduce the selectivity index

$$S_\% = \frac{N_{\text{c-inc}}}{|S_0 \cup S_1|} \cdot 100 \ , \tag{32}$$

where the total number of structures is $|S_0 \cup S_1| = 5588$ for the considered network. The larger $S_\%$ the larger the portion of $S_0 \cup S_1$ structures invalidated by c-inconsistency. Average values of the performance indices are reported in Table I for varying values of dataset size m and noise scaling factors s_e, s_ϵ. They were obtained from 100 Monte Carlo experiments, each characterized by different noise realizations.

The true structure was never declared inconsistent, showing the robustness of the falsification procedure. This suggests that, despite the fact that bounds $l(\cdot), u(\cdot)$ do not take into account 5% of variability, the falsification conditions are rather robust for the dataset considered in our study. Concerning the selectivity index $S_\%$, one can notice a degradation of performance when either the noise level increases or the size of the dataset decreases. In both cases, this is due to the fact that less datapoints can be used for model falsification. While obvious for smaller datasets, with the increase of noise this behavior is explained as follows. Refer for instance to Fig. 4. Model invalidation relies on checking whether $\tilde{x}^*_\vee$ belongs to set $\tilde{L}_{\max,\vee}$. For larger values of noise, bounds are increased so that $x^*_\vee$ is robustly classified (with high probability) inside or outside $L_{\max,\vee}$ based on the noisy observations. This conservatism, which ensures that the true model is not invalidated (with high probability), comes at a price: less datapoints are appropriate for invalidation, whence the decrease of $S_\%$. The variability of $S_\%$ among the three genes also suggests that the considered datasets do not equally support structure falsification for different genes. The analysis of index $I_\%$ highlights an interesting behavior. Excluding the least favorable condition $s_e = s_\epsilon = 0.07$ and $m = 12$, the contribution of the c-inconsistency analysis increases when datasets become smaller and noisier. A more detailed analysis of the results reveals that the improvement is most significant (up to 40%) for candidate models with largest number of regulating genes. This means that c-inconsistency can play a key role when structures of high complexity need to be falsified on the basis of few noisy data.

Prepared using rncauth.cls
DOI: 10.1002/rnc

6. CONCLUSIONS

In this paper, we introduced and analyzed geometrical properties of a relevant class of gene network models. Under the assumption that measurements of gene product concentrations and synthesis rates are available, we exploited monotonicity and convexity-like properties to invalidate families of models that are inconsistent with the data. The proposed falsification techniques represent an extension of the approach presented in [1], where the concept of sign pattern was introduced to capture the monotone character of unate regulation functions. We highlight that the approach proposed in this work can be extended to deal with any regulation function in the form of a product of a quasi-concave and a quasi-convex function. Since this extension would allow for many more alternative model structures, an increase in the overall complexity of the falsification strategy is expected.

The performance of the method was evaluated by means of Monte Carlo simulation using an oscillating synthetic network model. The results demonstrate a substantial improvement with respect to the approach in [1], especially when a small, noisy dataset is used.

Future directions of this research include the application of the proposed method to real experimental data to confirm the results obtained *in silico*. On the theoretical side, we anticipate a detailed study of other model classes that can benefit from our invalidation methods as well as the development of criteria for the invalidation of more general subclasses of unate models.

ACKNOWLEDGEMENT

This work was partially supported by the European Commission under the Network of Excellence HYCON2, contract number FP7-ICT-257462, and by the SystemsX.ch research consortium under the project YeastX.

REFERENCES

1. Porreca R, Cinquemani E, Lygeros J, Ferrari-Trecate G. Identification of genetic network dynamics with unate structure. *Bioinformatics* 2010; **26**(9):1239–1245.
2. Grefenstette J, Kim S, Kauffman S. An analysis of the class of gene regulatory functions implied by a biochemical model. *BioSystems* 2006; **84**(2):81–90.
3. Julius A, Belta C. Genetic regulatory network identification using monotone functions decomposition. *Proceedings of the 18th IFAC World Congress*, Milano, Italy, 2011; 11 785–11 790.
4. Bansal M, Belcastro V, Ambesi-Impiombato A, di Bernardo D. How to infer gene networks from expression profiles. *Molecular Systems Biology* 2007; **3**:78.
5. Cantone I, Marucci L, Iorio F, Ricci MA, Belcastro V, Bansal M, Santini S, di Bernardo M, di Bernardo D, Cosma MP. A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches. *Cell* 2009; **137**(1):172–181.
6. Anderson J, Papachristodoulou A. On validation and invalidation of biological models. *BMC Bioinformatics* 2009; **10**:132.
7. Julius A, Zavlanos M, Boyd S, Pappas G. Genetic network identification using convex programming. *IET Systems Biology* 2009; **3**(3):155–166.
8. August E, Papachristodoulou A. Efficient, sparse biological network determination. *BMC Systems Biology* 2009; **3**:25.
9. Zavlanos M, Julius A, Boyd S, Pappas G. Inferring stable genetic networks from steady-state data. *Automatica* 2011; **47**(6):1113–1122.
10. Nikolajewa S, Friedel M, Wilhelm T. Boolean networks with biologically relevant rules show ordered behavior. *BioSystems* 2007; **90**(1):40–47.
11. Harris S, Sawhill B, Wuensche A, Kauffman S. A model of transcriptional regulatory networks based on biases in the observed regulation rules. *Complexity* 2002; **7**(4):23–40.
12. Hanoch G, Rothschild M. Testing the assumptions of production theory: A nonparametric approach. *Journal of Political Economy* 1972; **80**(2):256–275.
13. Porreca R, Cinquemani E, Lygeros J, Ferrari-Trecate G. Invalidation of the structure of genetic network dynamics: A geometric approach. *Technical Report AUT11-11*, ETH Zürich, Switzerland 2011. URL `http://control.ee.ethz.ch/`.
14. Aracena J. Maximum number of fixed points in regulatory boolean networks. *Bulletin of Mathematical Biology* 2008; **70**(5):1398–1409.
15. Kauffman S, Peterson C, Samuelsson B, Troein C. Genetic networks with canalyzing boolean rules are always stable. *Proceedings of the National Academy of Sciences* 2004; **101**(49):17 102–17 107.

DOI: 10.1002/rnc

16. Szallasi Z, Liang S. Modeling the normal and neoplastic cell cycle with "realistic boolean genetic networks": their application for understanding carcinogenesis and assessing therapeutic strategies. *Proceedings of the Pacific Symposium on Biocomputing*, Maui, Hawaii, 1998; 66–76.
17. de Jong H. Modeling and simulation of genetic regulatory systems: A literature review. *Journal of Computational Biology* 2002; **9**(1):69–105.
18. Plahte E, Mestl T, Omholt S. A methodological basis for description and analysis of systems with complex switch-like interactions. *Journal of Mathematical Biology* 1998; **36**(4):321–348.
19. Yang H, Hsu C, Hwang M. An analytical rate expression for the kinetics of gene transcription mediated by dimeric transcription factors. *Journal of Biochemistry* 2007; **142**(2):135–144.
20. Keller AD. Model genetic circuits encoding autoregulatory transcription factors. *Journal of Theoretical Biology* 1995; **172**(2):169–185.
21. Sontag E. Network reconstruction based on steady-state data. *Essays in Biochemistry* 2008; **45**:161–176.
22. de Jong H, Ranquet C, Ropers D, Pinel C, Geiselmann J. Experimental and computational validation of models of fluorescent and luminescent reporter genes in bacteria. *BMC Systems Biology* 2010; **4**:55.
23. Boyd S, Vandenberghe L. *Convex Optimization*. Cambridge University Press: Cambridge, 2004.
24. Fukuda K. C-library cddlib 2010. URL `http://www.ifor.math.ethz.ch/~fukuda/cdd_home/cdd.html`.
25. Kvasnica M, Grieder P, Baotić M. Cddmex - matlab interface for the cdd solver. URL `http://control.ee.ethz.ch/~hybrid/cdd.php`.
26. Elowitz M, Leibler S. A synthetic oscillatory network of transcriptional regulators. *Nature* 2000; **403**(6767):335–338.

DOI: 10.1002/rnc