Published in IET Systems Biology
Received on 10th July 2014
Revised on 25th August 2014
Accepted on 8th September 2014
doi: 10.1049/iet-syb.2014.0025

Special Issue celebrating the 10th anniversary of *IET Systems Biology*

ISSN 1751-8849

A technique for determining the signs of sensitivities of steady states in chemical reaction networks

Eduardo D. Sontag

Department of Mathematics, Rutgers University, Piscataway, NJ 08854-8019, USA
E-mail: sontag@math.rutgers.edu

Abstract: This paper studies the direction of change of steady states to parameter perturbations in chemical reaction networks, and, in particular, to changes in conserved quantities. Theoretical considerations lead to the formulation of a computational procedure that provides a set of possible signs of such sensitivities. The procedure is purely algebraic and combinatorial, only using information on stoichiometry, and is independent of the values of kinetic constants. Three examples of important intracellular signal transduction models are worked out as an illustration. In these examples, the set of signs found is minimal, but there is no general guarantee that the set found will always be minimal in other examples. The paper also briefly discusses the relationship of the sign problem to the question of uniqueness of steady states in stoichiometry classes.

1 Introduction

A key question in the mathematical analysis of chemical reaction networks is the characterisation of sensitivities of steady states to parameter perturbations [1–7]. In the time scale of cellular signalling, assuming no turn-over due to expression and degradation or dilution, one such parameter could be, for example, the total concentration of a certain enzyme in its various activity states. The value of this parameter might be manipulated experimentally in various forms in order to achieve knock-downs or up-regulation. Often, especially in the context of inhibitors for therapeutic purposes, it is desirable to be able to predict the *sign* of the effect of such perturbations on states, in a manner that depends only on the structure of the network of reactions and not on the actual values of other parameters, such as kinetic constants, which are typically very poorly characterised.

1.1 An example

We introduce the problem to be studied through an example, an enzymatic network consisting of a cascade of two reversible covalent modifications, see Fig. 1.

Specifically, we consider the following reaction network:

$$\begin{array}{rcl} M_0 + E \rightleftharpoons A & \rightarrow & M_1 + E \\ M_1 + G \rightleftharpoons B & \rightarrow & M_0 + G \\ N_0 + M_1 \rightleftharpoons C & \rightarrow & N_1 + M_1 \\ N_1 + F \rightleftharpoons D & \rightarrow & N_0 + F \end{array} \tag{1}$$

Here E is a constitutively active kinase which drives a phosphorylation reaction in which a substrate M_0 is converted to an active form M_1, which can be dephosphorylated back into inactive form by a constitutively active phosphatase G. There are two intermediate enzyme–substrate complexes, $A = M_0E$ and $B = M_1G$, for these enzymatic reactions. The active form M_1 is itself a kinase which drives a phosphorylation reaction in which a second substrate N_0 is converted to an active form N_1, which can be dephosphorylated back into inactive form by a constitutively active phosphatase F. There are also two intermediate enzyme–substrate complexes, $C = N_0M_1$ and $D = N_1F$, for these last enzymatic reactions. In cell signalling, one typically views (1) as a cascade of a subsystem described by the first group of reactions, involving the enzyme M in its various forms, and a second subsystem described by N in its various forms, as diagrammed in Fig. 1.

An instance of great biological interest is provided by the proteins from MAPK/ERK pathways. There are several different MAPK ("mitogen-activated protein kinase") pathways, in each cell of a given organism, as well as in cells of different organisms, but they all share the same basic architecture, comprising a set of phosphorylation/dephosphorylation covalent modification cycles (sometimes with multiple phosphorylation steps in each subsystem). They are found in all eukaryotes [8–11], and are key participants in the regulation of some of the most important cell processes, from cell division and gene expression to differentiation and apoptosis. The targeting of MAPK/ERK components is the focus of current-generation drugs to treat advanced melanomas and a wide range of other tumours, including lung and thyroid cancers [12]. Normally, there are three, rather than two, components to MAPK cascades, corresponding to proteins generically called MAPK, MAPKK ("MAPK kinase"), and MAPKKK ("MAPKK kinase"), a typical example being given by Erk, Mek, and Ras respectively. Fig. 2 shows several typical MAPK pathways in mammalian cells. In our example, "*M*" and "*N*" could be MAPKK and MAPK respectively, or MAPKKK and MAPKK respectively.

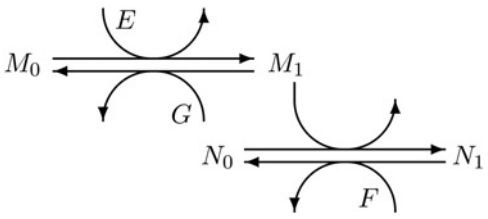

Fig. 1 *An enzymatic cascade*

Let us denote the concentrations of the various species in (1) using the corresponding lower case letters

$$(e, m_0, a, m_1, g, b, n_0, c, n_1, f, d)$$

We assume mass action kinetics for each reaction, and an ordinary differential equation (ODE) model. For example, the forward reaction $M_0 + E \to A$ will proceed at a rate km_0a, where $k > 0$ is a kinetic constant, so a differential equation for the state component e will have a term "$-km_0a$". A set of independent conservation laws (corresponding to a basis of the left nullspace of the stoichiometry matrix in the usual sense, see Appendix) for this system is given by:

$$e + a = E_T \tag{2}$$

$$g + b = G_T \tag{3}$$

$$f + d = F_T \tag{4}$$

$$m_0 + a + m_1 + b + c = M_T \tag{5}$$

and

$$n_0 + c + n_1 + d = N_T \tag{6}$$

We may think of E_T as total amount of constitutively active enzyme (bound or in complex), G_T and F_T as total amount of phosphatases, M_T as total amount of the first substrate in all free and bound forms, and N_T as total amount of the second substrate in all free and bound forms.

The question that we wish to study is: *how do steady states of the system change upon a change in one of these conserved quantities?* Our interest is especially in understanding how, for example, a variation in the total amount of the second phosphatase F_T "backwards" affects the steady-state of the first component of the system. Specifically, we wish to determine the direction of change (increase of decrease) in individual steady-state components when such a parameter is perturbed. Moreover, we would like to find information that is robust to the actual values of kinetic constants in each reaction. Experimental perturbations of quantities such as F_T are implemented, in practice, through genetic, biochemical, and physical methods, including small-molecule kinase inhibitors, changes in gene expression, repression of transcription by siRNA's, or laser trapping with optical tweezers.

Of course, there are no true "forward" and "backward" directions: the system is tightly connected, and the input/output formalism of control theory is inadequate as a paradigm (a point that was much emphasised by Willems in his work on behavioural foundations of systems theory [13]). Nonetheless, the idea of unidirectional information flow in MAPK and other cascades is well-established and has biological substance, through the ultimate transfer of information from cell surface receptors to gene expression. This question of "backward propagation" of effects has been the subject of considerable research in the context of modularity of biological systems [14, 15] and, specifically, in the context of the "retroactivity" phenomenon [16–23]. Retroactivity is a fundamental systems-engineering issue that arises when interconnecting biological subsystems, just as with electrical or mechanical systems: the effect of "loads" on the "output" of a system in effect creates biochemical "impedance" connections that are not obvious from a unidirectional signal-flow view of information processing.

When we apply the theory to be developed in this paper, we find that perturbations of the total second substrate, N_T, and perturbations of the second phosphatase, F_T, both lead to

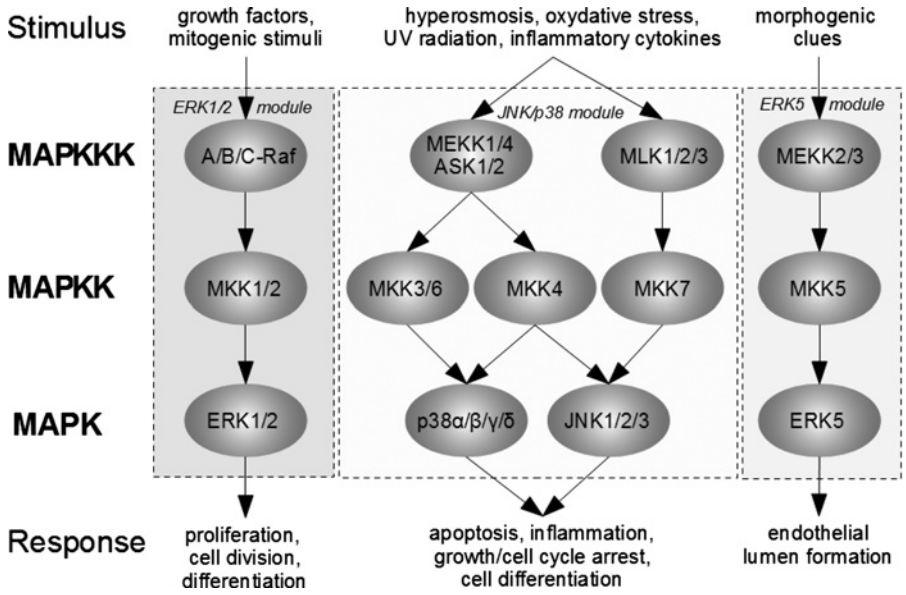

Fig. 2 *MAPK pathways in mammalian cells*

Image freely licensed from creative commons media file repository

doi: 10.1049/iet-syb.2014.0025

changes in "upstream" steady states. This is an instance of the retroactivity phenomenon. More interestingly, these two types of perturbations of the "downstream" layer have opposite effects on steady-state concentrations. This prediction has been tested experimentally and found to be correct [7].

We now turn to a precise problem statement, theoretical developments, and the description of an algorithm that addresses the question of directionality of changes in steady states upon parameter perturbations. We have developed a MATLAB® script, "CRNSESI" (Chemical Reaction Network SEnsitivity SIgns) that implements our procedure. After this, we return to the motivating example and display the signs of state changes for perturbations in each of the conserved quantities, as obtained from the use of CRNSESI. While in this example it turns out that the signs of state variations are unambiguously determined, such is not the case with other examples. To illustrate this lack of uniqueness, we provide a second example, a simple model of a phosphotransfer system, that exhibits ambiguity in one of the state components.

2 Preliminaries: general systems

We start with arbitrary systems of ODEs

$$\dot{x}(t) = f(x(t)) \tag{7}$$

The vectors x are assumed to lie in the positive orthant $\mathbb{R}^{n_s}_+$ of $\mathbb{R}^{n_s}$, that is, $x = (x_1, \ldots, x_{n_s})^T$ with each $x_i > 0$, and f is a differentiable vector field, mapping $\mathbb{R}^{n_s}_+$ into $\mathbb{R}^{n_s}$. We later specialise to ODEs that describe chemical reaction networks (CRNs), for which the abstract procedure to be described next can be made computationally explicit. In the latter context, we think of the coordinates $x_i(t)$ of x as describing the concentrations of various chemical species S_i, $i = 1, \ldots, n_s$.

Suppose that x^λ describes a λ-parametrised smooth curve of steady states for the system (7), where λ is a scalar parameter ranging over some open interval Λ. The steady-state condition amounts to asking that

$$f(x^\lambda) = 0 \tag{8}$$

for all values of the parameter $\lambda \in \Lambda$.

In addition to (8), we also assume that the steady states of interest are constrained by a set of algebraic equations

$$g_1(x^\lambda) = 0,\ g_2(x^\lambda) = 0, \ldots, g_{n_c}(x^\lambda) = 0 \tag{9}$$

where n_c is some positive integer (which we take to be zero when there are no additional constraints). We write simply $g(x^\lambda) = 0$, where $g : \mathbb{R}^{n_s}_+ \to \mathbb{R}^{n_c}$ is a differentiable mapping whose components are the g_i's. Some or all g_i might be linear functions, representing moities or stochiometric constraints, but non-linear constraints will be useful when treating certain examples, as will be discussed later.

Let us denote by

$$\xi^\lambda := \frac{dx^\lambda}{d\lambda} \in \mathbb{R}^{n_s \times 1}$$

the derivative of the vector function x^λ with respect to λ, viewed as a function $\Lambda \to \mathbb{R}^{n_s \times 1}$.

We are interested in answering the following question:

what are the signs of the entries of ξ^λ?

Obviously, the answer to this question will, typically, depend on the chosen value of λ. The computation of the steady state x^λ as a function of λ will ordinarily involve the numerical approximate solution of non-linear algebraic equations, or simulation of differential equations, and has to be repeated for each individual parameter λ. Our aim is, instead, to provide conditions that allow one to put constraints on these signs *independently of the specific* λ, *and even independently of other parameters that might appear in the specification of* f *and of* g, *such as kinetic constants*, and to do so using only linear algebraic and logical operations, with no recourse to numerical approximations.

Proceeding in complete generality, we take the derivative with respect to λ in (8), so that, by the chain rule, we have that $f'(x^\lambda)\xi^\lambda = 0$, where $f'(x)$ denotes the Jacobian matrix of f evaluated at a state x. In other words,

$$\xi^\lambda \in \mathcal{N}(f'(x^\lambda)) \tag{10}$$

where $\mathcal{N}(f'(x))$ denotes the nullspace of the matrix $f'(x)$. Similarly, we have that

$$\xi^\lambda \in \mathcal{N}(g'(x^\lambda)) \tag{11}$$

The reason for introducing separately f and g will become apparent later: we will be asking that each of the $n_c \times n_s$ entries of the Jacobian matrix of g should not change sign over the state space (which happens, in particular, when g is linear, as is the case with stoichiometric constraints). No similar requirement will be made of f, but instead, we will study the special case in which f represents the dynamics of a CRN.

2.1 Notations for signs of vectors and of subspaces

We use the following sign notations. For any (row or column) vector u with real entries, the vector of signs of entries of u, denoted sign u, is the (row or column) vector with entries in the set $\{-1, 0, 1\}$ whose ith coordinate satisfies:

$$(\operatorname{sign} u)_i = \begin{cases} -1, & \text{if } u_i < 0 \\ 1, & \text{if } u_i > 0 \\ 0, & \text{if } u_i = 0 \end{cases}$$

(The function sign is sometimes called the "signature function" when viewed as a map $\mathbb{R}^m \to \{-1, 0, 1\}^n$.) More generally, for any subspace $\mathcal{W}$ of vectors with real entries, we define

$$\operatorname{sign} \mathcal{W} = \{\operatorname{sign} v \mid v \in \mathcal{W}\}$$

Computing sign $\mathcal{W}$ amounts to determining which orthants are intersected by $\mathcal{W}$. This combinatorial problem is studied in the theory of oriented matroids: given a basis of $\mathcal{W}$, the signs of $\mathcal{W}$ represent the oriented matroid associated to a matrix that lists the basis as its columns, which is the set of "covectors" of this basis. See [24] for details and further theoretical discussion.

We also introduce the positive and negative parts of a vector u, denoted by u^+ and u^- respectively, as follows:

$$(u^+)_i = \begin{cases} u_i, & \text{if } u_i > 0 \\ 0, & \text{if } u_i \leq 0 \end{cases} \quad (u^-)_i = \begin{cases} -u_i, & \text{if } u_i < 0 \\ 0, & \text{if } u_i \geq 0 \end{cases}$$

Note that $u = u^+ - u^-$, $\operatorname{sign} u = \operatorname{sign} u^+ - \operatorname{sign} u^-$, and:

$$(\operatorname{sign} u)^+ = \operatorname{sign}(u^+), \quad (\operatorname{sign} u)^- = \operatorname{sign}(u^-) \tag{12}$$

Suppose that $u \in \mathbb{R}^{1\times n}$ and $v \in \mathbb{R}^{n\times 1}$, for some positive integer n. The equality:

$$\operatorname{sign}(uv) \;=\; \operatorname{sign}(\operatorname{sign}(u)\operatorname{sign}(v)) \tag{13}$$

need not hold for arbitrary vectors, for example, if $u = (1,\ -1/4,\ -1/4,\ -1/4)$ and $v = (1, 1, 1, 1)^T$ then $\operatorname{sign}(uv) = \operatorname{sign}(1/4) = 1$, but, on the other hand,

$$\begin{aligned}\operatorname{sign}(\operatorname{sign}(u)\operatorname{sign}(v)) &= \operatorname{sign}((1,\ -1,\ -1,\ -1)(1, 1, 1, 1)^T) \\ &= \operatorname{sign}(-2) = -1,\end{aligned}$$

which is not equal to $\operatorname{sign}(uv)$. However, equality (13) is true provided that we assume that (a) $u^- = 0$ or $u^+ = 0$ (i.e. either $u_i \geq 0$ for all i, or $u_i \leq 0$ for all i, respectively), and also that (b) $v^- = 0$ or $v^+ = 0$. This is proved as follows. Take first the case $u^- = 0$ and $v^- = 0$. Each term in the sum $uv = \sum_{i=1}^n u_i v_i$ is non-negative. Thus, $uv > 0$, that is, $\operatorname{sign}(uv) = 1$, if and only if $u_i > 0$ and $v_i > 0$ for some common index i, and $uv = \operatorname{sign}(uv) = 0$ otherwise. Similarly, as $\operatorname{sign}(u)\operatorname{sign}(v) = \sum_{i=1}^n \operatorname{sign}(u_i)\operatorname{sign}(v_i)$, we know that $\operatorname{sign}(u)\operatorname{sign}(v) > 0$, that is,

$$\operatorname{sign}(\operatorname{sign}(u)\operatorname{sign}(v)) = 1,$$

if and only if $\operatorname{sign}(u_i) = \operatorname{sign}(v_i) = 1$ for some i, and $\operatorname{sign}(u)\operatorname{sign}(v) = 0$ otherwise. But, $\operatorname{sign}(u_i) = \operatorname{sign}(v_i) = 1$ is the same as $u_i > 0$ and $v_i > 0$. Thus (13) is true. The case $u^+ = 0$ and $v^- = 0$ can be reduced to $u^- = 0$ and $v^- = 0$ by considering $-u$ instead of u: $\operatorname{sign}(uv) = -\operatorname{sign}((-u)v) = -\operatorname{sign}(\operatorname{sign}(-u)\operatorname{sign}(v)) = \operatorname{sign}(\operatorname{sign}(u)\operatorname{sign}(v))$. Similarly for the remaining two cases.

2.2 A parameter-dependent constraint set

Denoting

$$\mathcal{W}(x^\lambda) = \mathcal{N}(f'(x^\lambda)) \cap \mathcal{N}(g'(x^\lambda))$$

we have that (10) and (11) implies, in terms of the sign notations just introduced:

$$\pi^\lambda \;:=\; \operatorname{sign}\xi^\lambda \;\in\; \operatorname{sign}\mathcal{W}(x^\lambda)$$

Therefore, one could in principle determine the possible values of π^λ once that $\mathcal{W}(x^\lambda)$ is known. However, in applications one typically does not know explicitly the curve x^λ, which makes the problem difficult because the subspace $\mathcal{W}(x^\lambda)$ depends on λ, and even computing the steady states x^λ is a hard problem. As discussed below, for the special case of ODE systems arising from CRNs, a more systematic procedure is possible. Before turning to CRNs, however, we discuss general facts true for all systems.

For every positive concentration vector x define:

$$\Sigma^f(x) \;:=\; \{\operatorname{sign}(\nu f'(x)) \mid \nu \in \mathbb{R}^{1\times n_s}\} \tag{14}$$

$$\Sigma^g(x) \;:=\; \{\operatorname{sign}(\nu g'(x)) \mid \nu \in \mathbb{R}^{1\times n_s}\} \tag{15}$$

$$\Sigma(x) \;:=\; \Sigma^f(x) \cup \Sigma^g(x) \;\subseteq\; \{-1, 0, 1\}^{1\times n_s} \tag{16}$$

The row vectors ν are used in order to generate arbitrary linear combinations of the rows of the Jacobian matrices of f and g, a set rich enough to, ideally, permit the unique determination of the sign of ξ^λ.

Since at a steady state $x = x^\lambda$, $f'(x^\lambda)\xi^\lambda = 0$ and $g'(x^\lambda)\xi^\lambda = 0$, we also have that:

$$v\,\xi^\lambda \;=\; 0 \tag{17}$$

for every linear combination $v = \nu f'(x^\lambda)$ and $v = \nu g'(x^\lambda)$.

We now prove an easy yet key result, which shows that the sign vectors in the set $\Sigma(x^\lambda)$ strongly constrain the possible signs $\pi^\lambda = \operatorname{sign}\xi^\lambda = \operatorname{sign}\frac{dx^\lambda}{d\lambda}$. For simplicity in notations, we drop λ in π^λ and in ξ^λ when λ is clear from the context, and write simply π or ξ, with coordinates π_i and ξ_i, respectively.

To state the result, we use formal logic notations. Let $p_{\sigma,\pi}$ and $q_{\sigma,\pi}$ be the following logical disjunctions:

$$p_{\sigma,\pi} = \exists i\ \sigma_i\pi_i > 0$$

$$q_{\sigma,\pi} = \exists j\ \sigma_j\pi_j < 0$$

Recall that the "XNOR(p, q)" binary function has value "true" if and only if p and q are simultaneously true or simultaneously false. Consider the following statement, for any given $\lambda \in \Lambda$, and with $\pi = \pi^\lambda$:

$$\operatorname{XNOR}(p_{\sigma,\pi},\, q_{\sigma,\pi}) \quad \forall\, \sigma \in \Sigma(x^\lambda) \tag{18}$$

This statement is true if and only if for every $\sigma \in \Sigma(x^\lambda)$ it holds that either:

$$\forall\, i\ \sigma_i\pi_i = 0 \tag{19}$$

or:

$$(\exists i\ \sigma_i\pi_i > 0) \quad \text{and} \quad (\exists j\ \sigma_j\pi_j < 0) \tag{20}$$

(where i and j range over $\{1, \ldots, n_s\}$ in all quantifiers). In other words, either all the coordinates of the vector

$$(\sigma_1\pi_1,\ \sigma_2\pi_2,\ \ldots,\ \sigma_{n_s}\pi_{n_s})$$

are zero, or the vector must have both positive and negative entries.

Lemma 1: For any $\lambda \in \Lambda$, let $\pi = \pi^\lambda$. Then (18) is true.

Proof: Pick $\sigma = \operatorname{sign} v \in \Sigma(x^\lambda)$, $\pi = \pi^\lambda$, $\xi = \xi^\lambda$. Suppose that (19) is false. Then, either there is some i such that $\sigma_i\pi_i > 0$ or there is some j such that $\sigma_j\pi_j < 0$. If $\sigma_i\pi_i > 0$ for some i, then also $v_i\xi_i > 0$. As (17) holds, $\sum_{i=1}^{n_s} v_i\xi_i = 0$, so that there must exist some other index j for which $v_j\xi_j < 0$, which means that $\sigma_j\pi_j < 0$. Similarly,

doi: 10.1049/iet-syb.2014.0025

if there is some j such that $\sigma_j \pi_j < 0$, necessarily there is some i such that $\sigma_i \pi_i > 0$, by the same argument. □

In terms of the original data, Lemma 1 can be rephrased as follows. For each parameter value $\lambda \in \Lambda$, and each vector $\nu \in \mathbb{R}^{1\times n_s}$, either $\text{sign}\, \nu \frac{\partial f}{\partial x_i} \text{sign} \frac{dx_i^\lambda}{d\lambda} = 0$ for all $i \in \{1, \ldots, n_s\}$ or there are both positive and negative numbers in this sequence; and similarly for the partial derivatives of g.

The condition (18) given in Lemma 1 is only necessary, not sufficient. It may well be the case that there are sensitivity signs that pass this test, yet are not realisable for a given set of kinetic constants. In our experience, however, and as shown by the worked out examples, (18) is enough to provide a minimal set of signs, and is tight in that sense.

Given any two sign vectors σ, π, testing property (18) is simple in any programming language. For example, in MATLAB® syntax, one may write:

$$\begin{aligned} \zeta &= \sigma .* \pi \\ p &= \text{sign}\,(\text{sum}\,(\zeta > 0)) \\ q &= \text{sign}\,(\text{sum}\,(\zeta < 0)) \\ \text{XNOR} &= \text{sign}\,(p * q + (1-p) * (1-q)) \end{aligned}$$

and the variable XNOR will have value 1 if $\text{XNOR}(p_{\sigma,\pi}, q_{\sigma,\pi})$ is true and value 0 otherwise.

The basis of our approach will be as follows. We will show how to obtain a state-independent set Σ_0 which is a subset of $\Sigma(x)$ for all states x. In particular, for all steady states x^λ, we will have:

$$\Sigma_0 \subseteq \bigcap_{\lambda\in\Lambda} \Sigma(x^\lambda) \tag{21}$$

Compared to the individual sets $\Sigma(x^\lambda)$, which depend on the particular steady state x^λ, the elements of this subset are obtained using only linear algebraic operations; the computation of Σ_0 does not entail solving non-linear equations nor simulating differential equations. Since $\Sigma_0 \subseteq \Sigma(x^\lambda)$ for all x^λ, it follows that

$$\text{XNOR}(p_{\sigma,\pi}, q_{\sigma,\pi})\ \forall \sigma \in \Sigma(x^\lambda) \Rightarrow \text{XNOR}(p_{\sigma,\pi}, q_{\sigma,\pi})\ \forall \sigma \in \mathcal{T}$$

for any subset $\mathcal{T} \subseteq \Sigma_0$. Thus, we have:

$$\begin{aligned} &\text{For every } \lambda \in \Lambda,\ \pi^\lambda \in \mathcal{P} \\ &= \left\{ \pi \,\middle|\, \bigwedge_{\sigma\in\mathcal{T}} \text{XNOR}(p_{\sigma,\pi}, q_{\sigma,\pi}) \text{ is true} \right\} \end{aligned} \tag{22}$$

because of Lemma 1. We will construct such subsets $\mathcal{T}$ in our procedure, and test, for each potential sign vector π, whether the "orthogonality" property $\text{XNOR}(p_{\sigma,\pi}, q_{\sigma,\pi})$ is true or not, with respect to elements of $\mathcal{T}$. Our procedure will provide the set $\mathcal{P}$. Often, our construction of $\mathcal{T}$ leads to a $\mathcal{P}$ that has just three elements, $\mathcal{P} = \{0, \pi, -\pi\}$. (Note that $\pi = 0$ is always a solution, and solutions always appear in pairs, since $\nu\xi = 0$ implies $\nu(-\xi) = 0$.)

To generate $\mathcal{P}$, we carry out a sieve procedure (for moderate number of species, this is easy and fast): we test for each π if the conjunction in (22) is true; if the test fails, the sign vector π is eliminated from the list. The surviving π's are the possible sign vectors. Of course, since the conjunction in (22) is only a necessary, and not a sufficient, condition, we are not guaranteed to find a minimal set of signs. Observe that even though questions about the set $\mathcal{P}$ are decidable using propositional logic (there are a finite number of possible sign vectors), they have high computational complexity; for example, asking whether $\text{card}(\mathcal{P}) = 3$ is NP-hard on the number of species. Good heuristics for CNF problems include the Davis–Putnam–Logemann–Loveland (DPLL) algorithm [25]. The high computational complexity of these problems means that, generally speaking, our approach will only work well for relatively small networks.

The key issue, then, is to find a way to explicitly generate a state-independent subset Σ_0 of $\Sigma(x^\lambda)$, and we turn to that problem next.

2.3 Sketch of idea

To provide some intuition, let us consider, for the motivating example, the differential equation for e, which takes the form:

$$\dot{e} = -k_1 m_0 e + k_2 a + k_3 a$$

for some positive constants k_1, k_2, and k_3. Along a curve of steady states, we must have

$$-k_1 m_0(\lambda) e(\lambda) + k_2 a(\lambda) + k_3 a(\lambda) \equiv 0$$

and therefore, taking derivatives with respect to λ,

$$-k_1 e(\lambda) m_0'(\lambda) - k_1 m_0(\lambda) e'(\lambda) + (k_2 + k_3) a'(\lambda) \equiv 0 \tag{23}$$

Since $e(\lambda) > 0$ and $m_0(\lambda) > 0$, this means that the following triplets of signs for m_0', e', and a':

$$(-1,\ -1,\ 1),\ (1,\ 1,\ -1)$$

can never appear, since they would lead to a contradiction, namely a strictly positive and a strictly negative left-hand side, respectively, in (23).

We were able to derive this conclusion because the signs of the coefficients of m_0', e', and a' are uniquely determined independently of the value of λ. That fact, in turn, follows from the fact that the gradient of the function $(m_0, e, a) \mapsto -k_1 m_0 e + (k_2 + k_3) a$ (which appears in the right-hand side of the differential equation) has a constant sign. In contrast, if we had, for example, a differential equation like

$$\dot{x}_1 = -k_1 x_1 x_2 + k_2 x_2 x_3$$

then we would derive, arguing in the same manner, the constraint

$$-k_1 x_2 x_1'(\lambda) + (k_2 x_3 - k_1 x_1) x_2'(\lambda) + k_2 x_2 x_3'(\lambda) \equiv 0$$

and here the sign of the coefficient of $x_2'(\lambda)$, $k_2 x_3(\lambda) - k_1 x_1(\lambda)$, cannot be determined unless the values of $x_1(\lambda)$ and $x_3(\lambda)$ are known.

In general, additional information can be obtained by using linear combinations of right-hand sides. For example, still for the same example, consider the equation:

$$\dot{m}_0 = -k_1 m_0 e + k_2 a + k_4 b$$

Arguing as earlier, this leads to the identity

$$-k_1 e(\lambda) m_0'(\lambda) - k_1 m_0(\lambda) e'(\lambda) + k_2 a'(\lambda) + k_4 b'(\lambda) \equiv 0 \tag{24}$$

Subtracting (24) from (23), we have that

$$k_3 a'(\lambda) - k_4 b'(\lambda) \equiv 0$$

from which we conclude that $a'(\lambda)$ and $b'(\lambda)$ must have the same sign. Thus, we may obtain more information by taking linear combinations, but, again, we must check that the obtained coefficients have constant sign (which, in this case, is clear because k_3 and k_4 are constants). Our procedure is based on identifying such constant-sign linear combinations, using only information from stoichiometry.

3 Sensitivities for CRNs

From now on, we assume that we have a system of differential equations associated to a chemical reaction network:

$$\frac{dx}{dt} = f(x) = \Gamma R(x) \tag{25}$$

(see Appendix). Observe that $f'(x) = \Gamma R'(x)$, where $R'(x)$ is the Jacobian matrix of R, which is the matrix whose (k, j)th entry is $\frac{\partial R_k}{\partial x_j}(x)$.

We will assume from now on also specified a differentiable mapping

$$g : \mathbb{R}^{n_s}_+ \to \mathbb{R}^{n_c}$$

where n_c is some positive integer (possibly zero, to indicate the case where there are no additional constraints), and g has the property that

$$\text{all } n_c \times n_s \text{ entries of the Jacobian } g'(x) \text{ have constant sign} \tag{26}$$

In other words, the gradients $\nabla g_i(x)$ of the components $\{g_i, i = 1, \ldots, n_c\}$ of g, must have signs that do not depend on the state x.

We use g in order to incorporate, in particular, stoichiometric conservation laws, which are linear functions, and thus have constant gradients and therefore gradients whose signs do not depend on x. Recall that stoichiometric constraints are obtained from the matrix Γ as follows: one considers the vectors in the left nullspace of Γ, that is, the row vectors $\rho \in \mathbb{R}^{1\times n_s}$ such that $\rho\Gamma = 0$. The linear functions $x \mapsto \rho x$ are called conserved moities or stochiometric constraints; the time derivative of $\rho x(t)$ is constant along solutions of (25), since $\frac{d(\rho x)}{dt} = \rho\Gamma R(x) = 0$. Without loss of generality, one may take the vector ρ to have rational components or (clearing denominators) integer components, because the matrix Γ is rational. We emphasise that we do *not* include as components of g all stoichiometric constraints, or even all elements of a basis of the left nullspace of Γ. Indeed, in most examples of chemical reaction networks, this would lead to a unique steady state, or at most a discrete set of states. Our objective is precisely to study how steady states vary when one parameter varies, and hence a continuum of steady states is of interest.

The example in the introduction, for example, has five independent constraints, and one may show (see Appendix) that when all constraints are imposed, the steady state (given a specified set of kinetic reaction parameters) is unique. However, if, for instance, we keep G_T, F_T, M_T, N_T fixed but not impose a constant value on E_T, a continuum of steady states exists, as E_T is allowed to vary.

Observe that a non-linear function g may sometimes also have the constant sign property. For example, suppose that $n_s = 5$, $n_c = 1$, and

$$g(x) = k_1 x_1 x_3 - k_2 x_2^2$$

where k_1 and k_2 are positive constants. Then the Jacobian matrix (gradient, since $n_c = 1$) is:

$$g'(x) = \nabla g(x) = (k_1 x_3,\ -2k_2 x_2,\ k_1 x_1,\ 0,\ 0)$$

which has constant sign $(1, -1, 1, 0, 0)$.

For chemical reaction networks, it is not necessary for the entries of $f'(x)$, and much less the entries of the products $vf'(x)$ for vectors v, to have constant sign. Our next task will be to introduce algebraic conditions that allow one to check if the sign is constant, for any given vector v.

Before proceeding, however, we give an example of non-constant sign. Take the following CRN, with $n_s = 4$ and $n_R = 2$:

$$\mathcal{R}_1 : X_1 + X_2 \to X_4, \qquad \mathcal{R}_2 : X_2 + X_3 \to X_1 \tag{27}$$

which is formally specified, assuming mass-action kinetics, as follows:

$$A = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad \Gamma = \begin{pmatrix} -1 & 1 \\ -1 & -1 \\ 0 & -1 \\ 1 & 0 \end{pmatrix}$$

$$R(x) = (k_1 x_1 x_2,\ k_2 x_2 x_3)^T$$

Thus the ODE set $\dot{x} = f(x) = \Gamma R(x)$ corresponding to this CRN has:

$$f(x) = \begin{pmatrix} -k_1 x_1 x_2 + k_2 x_2 x_3 \\ -k_1 x_1 x_2 - k_2 x_2 x_3 \\ -k_2 x_2 x_3 \\ k_1 x_1 x_2 \end{pmatrix}$$

Let $v = e_1^T$, where, in general e_i is the canonical row vector $(0, \ldots, 0, 1, 0, \ldots, 0)$ with a "1" in the ith position and zeroes elsewhere. Observe that $vf'(x) = (-k_1 x_2, -k_1 x_1 + k_2 x_3, k_2 x_2, 0)$ does not have constant sign, because its second entry, which is the same as the (1, 2) entry of $f'(x)$, is the function $-k_1 x_1 + k_2 x_3$, which changes sign depending on whether $x_1 > k_2 x_3 / k_1$ or $x_1 < k_2 x_3 / k_1$. Ruling out vectors v that lead to such ambiguous signs is the purpose of our algorithm to be described next.

3.1 A first space

Introduce the following space:

$$\mathbf{V} := \text{row span of } \Gamma = \{\nu\Gamma \mid \nu \in \mathbb{R}^{1\times n_s}\} \subseteq \mathbb{R}^{1\times n_R}$$

doi: 10.1049/iet-syb.2014.0025

Since $f'(x) = \Gamma R'(x)$, the definition (14) of $\Sigma^f(x)$ becomes:

$$\Sigma^f(x) \ := \ \{\text{sign}(\nu R'(x)) \,|\, \nu \in \mathbf{V}\}$$

when specialised to CRN. Later on, we will explain how Property (26) allows us to obtain sign vectors induced by $g(x)$ that are independent of x. On the other hand, the sign vectors $\sigma = \text{sign}\, \nu R'(x)$ generally depend on the particular x. The following lemma shows that, for vectors ρ with non-negative entries, the sign of the vector $\rho R'(x)$ is the same, no matter what the state x is, and moreover, this sign can be explicitly computed using only stoichiometry information. We denote by

$$A_j = (a_{j1}, \ \ldots, a_{jn_{\mathrm{R}}})^T \ \in \ \mathbb{R}^{n_{\mathrm{R}} \times 1}$$

the jth column of the transpose A^T, i.e.. the transpose of the jth row of A.

Lemma 2: For any positive concentration vector x, any non-negative row vector ρ of size n_{R}, and any species index $j \in \{1, \ \ldots, n_{\mathrm{S}}\}$:

$$\rho A_j = 0 \Leftrightarrow \rho \frac{\partial R}{\partial x_j}(x) = 0 \tag{28}$$

Thus, also

$$\rho A_j > 0 \Leftrightarrow \rho \frac{\partial R}{\partial x_j}(x) > 0 \tag{29}$$

since the expressions in each side of (28) can only be zero or positive.

Proof: We have that

$$\rho A_j = \sum_{k \in K_\rho} \rho_k a_{jk}$$

where $K_\rho := \{k | \rho_k > 0\}$. Since every $a_{jk} \geq 0$, the equality $\rho A_j = 0$ holds if and only if $a_{jk} = 0$ for all $k \in K_\rho$. Similarly, from

$$\rho \frac{\partial R}{\partial x_j}(x) = \sum_{k \in K_\rho} \rho_k \frac{\partial R_k}{\partial x_j}(x)$$

and $\frac{\partial R_k}{\partial x_j}(x) \geq 0$ we have that $\rho \frac{\partial R}{\partial x_j}(x) = 0$ if and only if $\frac{\partial R_k}{\partial x_j}(x) = 0$ for all $k \in K_\rho$. From (47), in the Appendix on CRN, we conclude (28). □

Lemma 2 is valid for all non-negative ρ. When specialised to $\nu = \nu\Gamma \in \mathbf{V}$, and defining $\sigma = \text{sign}\, \nu R'(x)$, it says that σ does not depend on x. However, elements of the form $\nu = \nu\Gamma \in \mathbf{V}$ will generally not be non-negative (nor non-positive), so the lemma cannot be applied to them. Instead, we will apply Lemma 1 to the positive and negative parts of such a vector, but only when such positive and negative parts satisfy a certain "orthogonality" property, as defined by the subset of $\mathbf{V}$ introduced below.

3.2 A state-independent subset of Σ

For any $\nu \in \mathbf{V}$, consider the sign vector $\widetilde{\mu}_\nu := \text{sign}\, \nu A^T \in \{-1, 0, 1\}^{1 \times n_{\mathrm{S}}}$, whose jth entry is $\nu A_j = \nu\Gamma A_j$ if $\nu = \nu\Gamma$ with $\nu \in \mathbb{R}^{1 \times n_{\mathrm{S}}}$, as well as the positive and negative parts of ν, ν^+ and ν^-. Define the following set of vectors ("G" for "good"):

$$\mathbf{V}_G \ := \ \{\nu \in \mathbf{V} \,|\, \text{for each } j \in \{1, \ \ldots, n_{\mathrm{S}}\}$$
$$\text{either } \nu^+ A_j = 0 \text{ or } \nu^- A_j = 0\}$$

Observe that, if $\nu \in \mathbf{V}_G$, then, from $\nu A_j = (\nu^+ - \nu^-)A_j = \nu^+ A_j - \nu^- A_j$, it follows that

$$\nu A_j = \begin{cases} \nu^+ A_j, & \text{if } \nu^- A_j = 0 \\ -\nu^- A_j, & \text{if } \nu^+ A_j = 0 \\ 0, & \text{if } \nu^+ A_j = \nu^- A_j = 0 \end{cases} \tag{30}$$

Consider the following set of sign vectors $\widetilde{\mu}_\nu$ parametrised by elements of $\mathbf{V}_G$:

$$\widetilde{\Sigma}_0 \ := \ \{\widetilde{\mu}_\nu = \text{sign}(\nu A^T) \,|\, \nu \in \mathbf{V}_G\} \ \subseteq \ \{-1, 0, 1\}^{1 \times n_{\mathrm{S}}} \tag{31}$$

The key fact is that this is a subset of $\Sigma(x)$, for all x:

Lemma 3: For every positive concentration vector x,

$$\widetilde{\Sigma}_0 \subseteq \Sigma(x)$$

A proof is provided in Section 6.

Remark 1: To interpret the set $\mathbf{V}_G$, it is helpful to study the special case in which ν is simply a row of Γ, that is, $\nu = \nu\Gamma$ and $\nu = e_i^T$. Since

$$e_i^T B - e_i^T A = e_i^T (B - A) = e_i^T \Gamma = \nu^+ - \nu^-$$

and the vectors $e_i^T B$ and $e_i^T A$ have non-overlapping positive entries (by the non-autocatalysis assumption), we have that $\nu^+ = e_i^T B$ and $\nu^- = e_i^T A$. Since $e_i^T B A_j = \sum_k b_{ik} a_{jk}$, asking that this number be positive amounts to asking that

$$i \text{ is a product of a reaction } \mathcal{R}_k \text{ which has } j \text{ as a reactant} \tag{32}$$

Since $e_i^T A A_j = \sum_k a_{ik} a_{jk}$, asking that this number is positive amounts to asking that

$$i \text{ and } j \text{ are both reactants in some reaction } \mathcal{R}_{k'} \tag{33}$$

Thus, if the network in question has the property that (32) and (33) cannot both hold simultaneously for any pair of species i, j, then we cannot have that both $e_i^T B A_j > 0$ and $e_i^T A A_j > 0$ hold. In other words, $e_i^T \in \mathbf{V}_G$ for all i.

As an illustration, take the CRN $\mathcal{R}_1$: $X_1 + X_2 \to X_4$ and $\mathcal{R}_2$: $X_2 + X_3 \to X_1$ treated in (27). We claim that $e_1^T \notin \mathbf{V}_G$, which reflects the fact that $e_1^T f'(x)$ does not have constant sign. Indeed, in this case we have that, with $i = 1$ and $j = 2$, X_1 and X_2 are reactants in $\mathcal{R}_1$ but X_1 is also a

product of reaction $\mathcal{R}_2$, which has X_2 as a reactant. Algebraically, $e_1^T\Gamma = (-1, 1) = (0, 1) - (1, 0) = v^+ - v^-$ and $A_2 = (1, 1)^T$, so $v^+A_2 = 1$ and $v^-A_2 = 1$. This means that $v = e_1^T \notin \mathbf{V}_G$, since the property defining $\mathbf{V}_G$ would require that at least one of v^+A_2 or v^-A_2 should vanish. We have re-derived, in a purely algebraic manner, the fact that $-k_1x_1 + k_2x_3$ changes sign.

Testing whether a given vector $v \in \mathbf{V}$, $v = \nu\Gamma$ with $\nu \in \mathbb{R}^{1\times n_s}$, belongs to $\mathbf{V}_G$ is easy to do. For example, in MATLAB®-like syntax, one may write:

$$\begin{aligned} v &= \nu * \Gamma \\ v^+ &= (v > 0). * v \\ v^- &= -(v < 0). * v \\ v_A^+ &= \text{sign}(v^+ * A') \\ v_A^- &= \text{sign}(v^- * A') \end{aligned}$$

and we need to verify that the vectors v_A^+ and v_A^- have disjoint supports, which can be done with the command

$$\text{sum}(v_A^+. * v_A^-) == 0$$

which returns 1 (true) if and only if $v \in \mathbf{V}_G$, in which case we accept v and we may use $\sigma = \text{sign}(vA^T)$ to test the conditions in Lemma 1.

3.3 Explicit generation of elements of $\widetilde{\Sigma}_0$

The set $\widetilde{\Sigma}_0$ defined in (31) is constructed in such a way as to be independent of states x, which makes it more useful than the sets $\Sigma(x)$ from a computational standpoint. Yet, in principle, computing this set potentially involves the testing of the conditions "$v^+A_j = 0$ or $v^-A_j = 0$" that define the set $\mathbf{V}_G$, for every $v = \nu\Gamma$, that is, for every possible real-valued vector $\nu \in \mathbb{R}^{1\times n_s}$ (and each j). We describe next a more combinatorial way to generate the elements of $\widetilde{\Sigma}_0$.

We introduce the set of signs associated to the row span $\mathbf{V}$ of Γ:

$$\mathbf{S} := \text{sign}\,\mathbf{V} \subseteq \{-1, 0, 1\}^{1\times n_R} \tag{34}$$

Denote:

$$\alpha := \text{sign}\,A^T \in \{0, 1\}^{n_R\times n_S}$$

so that the jth column of α is $\alpha_j = \text{sign}\,A_j \in \{0, 1\}^{n_R\times 1}$.

Lemma 4: Pick any $s \in \mathbf{S}$, $s = \text{sign}\,v$, where $v \in \mathbf{V}$. Then, for each $j \in \{1, \ldots, n_s\}$:

$$\text{sign}(v^+A_j) = \text{sign}(s^+\alpha_j), \quad \text{sign}(v^-A_j) = \text{sign}(s^-\alpha_j)$$

Proof: By (13), applied with $u = v^+$ and $v = A_j$, $\text{sign}(v^+A_j) = \text{sign}(\text{sign}(v^+)\alpha_j)$. By (13) applied with $u = v^-$ and $v = A_j$, $\text{sign}(v^-A_j) = \text{sign}(\text{sign}(v^-)\alpha_j)$. Since, by (12) applied with $u = v$, $s^+ = \text{sign}(v^+)$ and $s^- = \text{sign}(v^-)$, the conclusion follows.

In analogy to the definition of the set $\mathbf{V}_G$, we define ("G" for "good"):

$$\begin{aligned} \mathbf{S}_G := \{&s \in \mathbf{S} \mid \text{for each } j \in \{1, \ldots, n_s\} \\ &\text{either } s^+\alpha_j = 0 \text{ or } s^-\alpha_j = 0\} \end{aligned}$$

Observe that, if $s \in \mathbf{S}_G$, then, since $s\alpha_j = (s^+ - s^-)\alpha_j = s^+a_j - s^-a_j$,

$$s\alpha_j = \begin{cases} s^+\alpha_j, & \text{if } s^-\alpha_j = 0 \\ -s^-\alpha_j, & \text{if } s^+\alpha_j = 0 \\ 0, & \text{if } s^+\alpha_j = s^-\alpha_j = 0 \end{cases} \tag{35}$$

Consider the following set of sign vectors parametrised by elements of $\mathbf{S}_G$:

$$\Sigma_0 := \{\mu_s = \text{sign}(s\alpha) \mid s \in \mathbf{S}_G\} \subseteq \{-1, 0, 1\}^{1\times n_S} \tag{36}$$

Proposition 1: Pick any $s \in \mathbf{S}$, $s = \text{sign}\,v$, where $v \in \mathbf{V}$. Then

$$s \in \mathbf{S}_G \text{ if and only if } v \in \mathbf{V}_G$$

and for such s and v,

$$\text{sign}(vA^T) = \text{sign}(s\alpha) \tag{37}$$

A proof is provided in Section 6.

Corollary 1: $\widetilde{\Sigma}_0 = \Sigma_0$.

Proof: Pick any element of $\widetilde{\Sigma}_0$, $\widetilde{\mu}_v = \text{sign}(vA^T)$, $v \in \mathbf{V}_G$. By Corollary 1, $s = \text{sign}\,v \in \mathbf{S}_G$. Moreover, also by Corollary 1, $\widetilde{\mu}_v = \text{sign}(s\alpha)$, so we know that $\widetilde{\mu}_v \in \Sigma_0$. Conversely, take an element $\mu_s \in \Sigma_0$. This means that $\mu_s = \text{sign}(s\alpha)$ for some $s \in \mathbf{S}_G \subseteq \mathbf{S} = \text{sign}\,\mathbf{V}$. Let $v \in \mathbf{V}$ be such that $s = \text{sign}\,v$. By Corollary 1, $v \in \mathbf{V}_G$, and also $\mu_s = \text{sign}(vA^T)$. By definition of $\widetilde{\Sigma}_0$, this means that $\mu_s \in \widetilde{\Sigma}_0$. □

We can simplify the definition of Σ_0 a bit further, by noticing that the finite subset $\mathbf{S}$ can be in fact be generated using only *integer* vectors. The definition in (34) says that:

$$\mathbf{S} = \{\text{sign}\,(v\Gamma) \mid v \in \mathbb{R}^{1\times n_S}\} \subseteq \{-1, 0, 1\}^{1\times n_R}$$

Lemma 5:

$$\mathbf{S} = \{\text{sign}\,(v\Gamma) \mid v \in \mathbb{Z}^{1\times n_S}\} \subseteq \{-1, 0, 1\}^{1\times n_R}$$

A proof is provided in Section 6.

3.4 Adding rows to g by linear combinations of linear components

Recall that we made the assumption [Property (26)], that the n_c components of g have gradients of constant sign. This

doi: 10.1049/iet-syb.2014.0025

means that the elements in following subset of $\Sigma^g(x)$, for all x:

$$\Sigma_1^g := \{\text{sign}(e_i^T g'(x)) \mid i \in \{1, \ldots, n_c\}\} \quad (38)$$

where e_i^T denotes the canonical row vector $(0, \ldots, 0, 1, 0, \ldots, 0)$ with a "1" in the ith position and zeroes elsewhere, have constant sign, independently of the particular state x. We will also consider the following subset of $\Sigma^g(x)$, for all x:

$$\Sigma_2^g := \{\text{sign}(\nu g'(x)) \mid \nu \in \mathbb{R}_I^{1\times n_s}\} \quad (39)$$

where $I \subseteq \{1, \ldots, n_c\}$ denotes the set of indices of rows of g that are linear functions, and $\nu \in \mathbb{R}_I^{1\times n_s}$ means that ν is supported in I, that is, $\nu_j = 0$ whenever $j \notin I$. Since a linear combination of linear functions is again linear, the elements of Σ_2^g also have constant sign. Thus, we will only use elements of $\Sigma_1^g \cup \Sigma_2^g$ in our procedure, instead of arbitrary elements of $\Sigma^g(x)$. As part of our algorithm, we add selected combinations of such constraints as new components of g – ideally the whole sign space of the span of the rows, but in practice just a few sparse linear combinations suffice. If the coefficients of these linear functions are rational numbers (as is the case with coordinates of g that represent stoichiometric constraints), we may, without loss of generality, take integer combinations, as justified in the same manner as Lemma 5.

Let us explain, through an example, why this procedure is necessary. Suppose that the following are two rows of g

$$g_1 = x_1 + x_2 + x_3 - c_1$$
$$g_2 = 2x_2 + x_3 - c_2$$

which might represent the conservation of two quantities. If x^λ is a curve of steady states, and denoting derivatives with respect to λ by primes, we have therefore that

$$x_1'(\lambda) + x_2'(\lambda) + x_3'(\lambda) = 0 \quad \text{and} \quad 2x_2'(\lambda) + x_3'(\lambda) = 0$$

The first of these tells us that the sign vector $s = \pi^\lambda = (s_1, s_2, s_3)$ is either zero or must have two components of opposite signs, and the second one implies that $s_2 = -s_3$. The conjunction of these two constraints gives the following set of possible signs:

$$\{(0, 0, 0), (1, 1, -1), (0, 1, -1), (-1, 1, -1)\}$$

(and the negatives of the last three). However, notice that, if we add to the rows of g also the difference $g_3 = g_1 - g_2 = x_1 - x_2 - c_1 + c_2$, then we also know that $x_1'(\lambda) - x_2'(\lambda) = 0$, so that, in fact, we should also have that $s_1 = s_2$. Adding this constraint serves to eliminate the last two possibilities (as well as their negatives), giving the unique non-zero solution $(1, 1, -1)$ [and its negative $(-1, -1, 1)$]. Thus, adding the linear combination g_3, even if it is redundant from a purely linear-algebraic point of view, provides additional information when looking for signs.

3.5 Addition of "virtual constraints" to g

We have also found, when working out examples, that the following heuristic is useful. Consider the set $\mathcal{I}$ consisting of all state-dependent linear combinations

$$h(x) = \sum_{i=1}^{n_s} r_i(x)\Gamma_i R(x)$$

of the rows of the right-hand side of the dynamics (25), where Γ_i denotes the ith row of Γ, and the r_i's are scalar functions. In abstract algebra terminology, when the reactions R_i's are polynomials (as with mass action kinetics), and if we restrict to polynomial coefficients r_i, then $\mathcal{I}$ is the ideal generated by the functions $\Gamma_i R$. Take any $h \in \mathcal{I}$, and a parametrised set of positive steady states x^λ. Since $\Gamma R(x^\lambda) = 0$, it follows that also $h(x^\lambda) = 0$ for every $\lambda \in \Lambda$. Now, suppose that one is able to find a function h of this form with the property that $h(x) = m(x)g(x)$, where $m(x)$ is a monomial and $g(x)$ has a gradient of constant sign. Then $g(x^\lambda) = 0$ for every $\lambda \in \Lambda$, because $m(x) \neq 0$ at all positive x. This means that we may add g to the set of constraints. We call a function g of this form a "virtual constraint."

Testing for the existence of such elements is in principle a difficult computational algebra problem. However, in many or even most natural examples of CRN's, the reaction functions R_i are either linear or quadratic. If we consider only linear functions r_i, then the combination elements h obtained by the above construction are at most polynomials of order three. Suppose that we look for factorisations of the form $h(x) = x_i g(x)$, where g is a polynomial of order at most two, and is so that the monomials in g all involve different variables. Such a g has constant-sign gradient (because $\nabla g(x)$'s coordinates are all either constants or single variables x_i). Testing for such a factorisation, for each fixed variable x_i as "$m(x)$" and any fixed group of monomials for g, becomes a linear algebraic problem on the coefficients of the functions r_i. We do not discuss this further in general, but only mention an example which will be useful when analysing a particular network below.

Suppose that some two rows of $f = \Gamma R(x)$ are as follows:

$$f_1(x) = k_1 x_0 y_1 - k_{-1} x_1 y_0$$
$$f_2(x) = k_2 x_1 y_1 - k_{-2} x_2 y_0$$

where we are denoting the coordinates of x as $(x_0, x_1, x_2, y_0, y_1)$ for reasons that will be clear when we discuss the network where this example appears. Taking $r_1(x) = k_{-2}x_2$ and $r_2(x) = k_{-1}x_1$, we have that

$$h(x) = k_{-2}x_2 f_1(x) - k_{-1}x_1 f_2(x) = m(x)g(x)$$

where $m(x) = y_1$ is a monomial, and:

$$g(x) = k_1 k_{-2} x_0 x_2 - k_{-1} k_2 x_1^2$$

has the gradient:

$$g'(x) = \nabla g(x) = (k_1 k_{-2} x_2,\ -2k_{-1}k_2 x_1,\ k_1 k_{-2} x_0,\ 0,\ 0)$$

which has constant sign $(1, -1, 1, 0, 0)$.

3.6 Remarks on global properties

We do not directly address in this study the issue of uniqueness of steady states in each stoichiometry class. In those examples in which the space of fixed conservation laws has codimension one, as in our example when we fix all except one of the values E_T and so on, it is possible in principle that for each value of the remaining conserved

quantity there may exist several equilibria. This is a well-studied question for CRNs, see for instance [26–35]. A routine argument on CRNs can be used to prove that for our motivating example (1), steady states are unique once that all conservation laws are taken into account (see Appendix).

However, in this work our concern has been with the determination of signs of sensitivities, and not their actual values. These are different questions. Indeed, signs might be unique even when values are not: different steady states may well "move" in the same direction upon a perturbation of parameters. For a completely trivial illustration, take any one-dimensional (1D) differential equation $\dot{x} = f(x)$. Even if f has multiple roots, leading to multiple steady states, $\mathcal{N}(f'(x))$ is either equal to $\{0\}$ or $\mathbb{R}$ at each steady state. This means that the signs of the elements in $\mathcal{N}(f'(x))$ are unique (zero in the first case) or, at worst, unique up to sign reversals (in the second case). Note that any f which has the property that $f(0) \geq 0$ arises from some CRN, $f = \Gamma R(x)$. Indeed, a representing CRN for $f(x) = \sum_{i=0}^{n} a_i x^i$, with $a_0 \geq 0$, can be obtained as follows. For $i = 0$, we include a reaction $0 \to X$ with rate constant a_0. For $i > 0$ and $a_i \leq 0$, we introduce a reaction $iX \to 0$ with rate constant $-a_i/i$. For $i > 0$ and $a_i > 0$, we introduce a reaction $iX \to (i+1)X$ with rate constant a_i. Then $\Gamma = [1, \gamma_1, \ldots, \gamma_n]$, with $\gamma_i = -i$ if $k_i \leq 0$ and $\gamma_i = 1$ if $k_i \geq 0$, and $R(x) = (k_0, k_1, \ldots, k_n)^T$ with $k_i = -(a_i/i)x^i$ if $a_i \leq 0$ and $k_i = a_i x^i$ if $a_i \geq 0$. (This network has autocatalytic reactions, but adding additional species turns it into one that does not.)

Another example is given by the 2D system that has vector field $f(x) = ((x_1 - x_2)(x_1 - 2x_2)(x_1 - 3x_2), 0)^T$. At steady states of the form (x_2, x_2), $(2x_2, x_2)$, and $(3x_2, x_2)$, the first row of the Jacobian matrix f' is (ax_2^2, bx_2^2) (and the second row is zero), where $a = 2, -1, 2$ and $b = -2, 2, -6$, respectively. Thus, the nullspace $\mathcal{N}(f'(x^\lambda)) = \{(u_1, u_2) \in \mathbb{R}^2 \mid au_1 + bu_2 = 0\}$ is the span of (1, 1), (2, 1), or (3, 1). These are three different subspaces, yet they all have the common sign (1, 1) (plus its negative, and zero). In summary, even though the tangent vectors are not unique, in this example signs are.

Remark 2: Suppose that signs of sensitivities are unique up to sign reversals and zero, that is, for some $\pi \in \{-1, 0, 1\}^{n_s \times 1}$ and all parameter values $\lambda \in \Lambda$, $\pi^\lambda \in \{\pi, -\pi, 0\}$. Then a global result along any smooth non-singular ($\xi^\lambda \neq 0$ for all λ) curve connecting steady states follows as a corollary. In other words, the conclusion from infinitesimal perturbations extends to global perturbations. Indeed, suppose that we want to compare the values of the steady-state concentrations x^{λ_1} and x^{λ_2} at two parameter values λ_1, λ_2. We have:

$$\operatorname{sign}(x^{\lambda_2} - x^{\lambda_1}) = \operatorname{sign}\left(\int_{\lambda_1}^{\lambda_2} \xi^\lambda \, d\lambda\right) = \pm\pi$$

the sign depending on whether $\pi^\lambda = \pi$ or $\pi^\lambda = -\pi$ for all λ (no change of sign is possible, by nonsingularity).

4 Summary and implementations

Our procedure for finding the set $\mathcal{P}$ in (22), which contains all possible signs π^λ of derivatives ξ^λ, consists of the following steps:

1. Construct a subset $\mathcal{S} \subseteq \mathbf{S}$ (see below).
2. For each element $s \in \mathcal{S}$, test the property $(s^+\alpha_j) \cdot (s^-\alpha_j) = 0$, which defines $\mathbf{S}_G$. The s's that pass this test are collected into a set $\mathcal{S}_G$, which is known to be a subset of $\mathbf{S}_G$.
3. Take the set of elements of the form $\mu_s = \operatorname{sign}(s\alpha)$, for s in $\mathcal{S}_G$, and add to these the signs of the rows of the Jacobian g' of g, as well as a subset of combinations of linear components of g (by assumption, these sign vectors are independent of x). Let us call this set $\mathcal{T}$.
4. Optionally, add to $\mathcal{T}$ sign vectors from "virtual constraints" as explained earlier.
5. Now apply the sieve procedure, testing the conjunction in (22). The elements π that pass this test are reported as possible signs of derivatives of steady states with respect to the parameter λ, in the sense that they have not been eliminated. These are the elements of $\mathcal{P}$.
6. If a unique (after eliminating 0 as well as one element of each pair $\{\pi, -\pi\}$) solution remains, we stop. If there is more than one sign that passed all tests, and if $\mathcal{S}$ was a proper subset of $\mathbf{S}$, we may generate a larger set $\mathcal{S}$, and hence a potentially larger $\mathcal{T}$, and repeat the subsequent steps for the larger subset.

The theory guarantees that our procedure will eliminate all impossible sign vectors, thus providing a set $\mathcal{P}$ of possible sign vectors. As is typically the case with heuristics for computationally intractable problems, there is no *a priori* guarantee that the set $\mathcal{P}$ obtained by steps 1–5 should be a minimal such set, and this is why step 6 is included for further search.

The first step, constructing $\mathbf{S}$, or a large subset $\mathcal{S}$ of it, can be done in various ways. Since, by Lemma 5, we can generate $\mathbf{S}$ using integer vectors, the elements of $\mathbf{S}$ have the form sign v where we may assume, without loss of generality, that each entry of $v = \nu\Gamma$ is either zero or, if non-zero, is either ≥ 1 or ≤ -1. Thus, testing whether a sign vector s belongs to $\mathbf{S}$ amounts to testing the feasibility of a linear program (LP): we need that $\nu\Gamma e_i = 0$ for those indices i for which $s_i = 0$, that $\nu\Gamma e_i \leq -1$ for those indices i for which $s_i = -1$, and that $\nu\Gamma e_i \geq 1$ for those indices i for which $s_i = 1$. (These are closed, not strict, conditions, as needed for an LP formulation.) This means that one can check each of the 3^n possible sign vectors efficiently.

One can combine the testing of LP feasibility with the search over the 3^n possible sign vectors into a mixed integer linear programming (MILP) formulation, by means of the technique called in the MILP field a "big M" approximation [36]. This is a routine reduction: one first fixes a large positive number M, and then formulates the following inequalities:

$$\nu\Gamma e_i - ML_i + U_i \leq 0, \quad -\nu\Gamma e_i - MU_i + L_i \leq 0, \quad L_i + U_i \leq 1$$

where the vector ν is required to be real and the variables L_i, U_i binary ($\{0, 1\}$). Given any solution, we have that $-M \leq \nu\Gamma e_i \leq -1$ (so $s = -1$) for those i for which $(L_i, U_i) = (0, 1)$, $1 \leq \nu\Gamma e_i \leq M$ (so $s = 1$) for indices for which $(L_i, U_i) = (1, 0)$, and $\nu\Gamma e_i = 0$ (i.e. $s_i = 0$) when $(L_i, U_i) = (0, 0)$. (This trick will miss any solutions for which $\nu\Gamma e_i \leq -1$ but M was not taken large enough that $-M \leq \nu\Gamma e_i$, or $\nu\Gamma e_i \geq 1$ but M was not taken large enough that $\nu\Gamma e_i \leq M$.) The resulting MILP can be solved using relaxation-based cutting plane methods, branch and bound approaches, or heuristics such as simulated annealing

doi: 10.1049/iet-syb.2014.0025

[37, 38]. Such mixed-integer techniques have been used for the related but very different problem of parameter identification for biochemical networks, see for instance [39].

Often, however, simply testing sparse integer vectors in the integer-generating form in Lemma 5 works well. In practice, we find that linear combinations with small coefficients of pairs of canonical basis vectors $\nu = e_i^T$, and similarly for the appropriate conservation laws, is typically enough to obtain the set of all possible sign vectors π (up to all signs being reversed, and except for the trivial solution $\pi = 0$).

We have developed a MATLAB® script, "CRNSESI" (Chemical Reaction Network SEnsitivity SIgns) that implements our procedure. The examples given in the next section were worked out using this software. (Actual output from the program is shown in the Supplementary Materials.)

5 Three worked-out examples

5.1 Kinase cascade

In particular, the example given in the introduction was worked out using CRNSESI. Specifically, we introduced stoichiometric constraints to keep all but one conservation law fixed, and analysed the signs of the resulting sensitivities for any curve, obtaining in each case a unique solution (up to sign reversals or the identically zero solution). The output of CRNSESI, for the concrete example given by reactions (1), can be summarised as shown below. In each case, "−1" or "1" means that the respective component of the state vector changes negatively or positively, respectively, under the corresponding perturbation.

if the first kinase, E_T, *decreases* (keeping G_T, F_T, M_T, N_T fixed):

−1	1	−1	−1	1	−1	1	−1	−1	1	−1
e	m_0	a	m_1	g	b	n_0	c	n_1	f	d

if the first substrate, M_T, *increases* (keeping E_T, G_T, F_T, N_T fixed):

−1	1	1	1	−1	1	−1	1	1	−1	1
e	m_0	a	m_1	g	b	n_0	c	n_1	f	d

if the first phosphatase, G_T, *increases* (keeping E_T, F_T, M_T, N_T fixed):

−1	1	1	−1	1	1	1	−1	−1	1	−1
e	m_0	a	m_1	g	b	n_0	c	n_1	f	d

if the second substrate, N_T, *decreases* (keeping E_T, G_T, F_T, M_T fixed):

−1	1	1	1	−1	1	−1	−1	−1	1	−1
e	m_0	a	m_1	g	b	n_0	c	n_1	f	d

if the second phosphatase, F_T, *decreases* (keeping E_T, G_T, M_T, N_T fixed):

−1	1	1	1	−1	1	−1	−1	1	−1	−1
e	m_0	a	m_1	g	b	n_0	c	n_1	f	d

If the opposite change is made on a total amount, then the signs get reversed. For example, if the second substrate, N_T, increases, then we obtain:

1	−1	−1	−1	1	−1	1	1	1	−1	1
e	m_0	a	m_1	g	b	n_0	c	n_1	f	d

Typically, one is also interested the effect of perturbations on the total concentration of active kinase, free or bound, $X = M_1 + B + C$ and the total concentration of product, free or bound, $Y = N_1 + D$. Experimentally, these quantities are far easier to quantify using Western blots or mass spec techniques [7]. In order to study changes in X and Y, we introduce "virtual" variables x and y and artificial stoichiometric constraints $m_1 + b + c - x = 0$ and $n_1 + d - y = 0$, and re-apply our algorithm. Results are as follows (using the same sign conventions as above):

if the first kinase, E_T, *decreases:* $x, y = -1, -1$
if the first substrate, M_T, *increases:* $x, y = 1, 1$
if the first phosphatase, G_T, *increases:* $x, y = -1, -1$
if the second substrate, N_T, *decreases:* $x, y = -1, -1$
if the second phosphatase, F_T, *decreases:* $x, y = -1, 1$

Notice the following remarkable phenomenon: when the total second substrate, N_T, is perturbed, we see that x and y, the total amounts of active enzymes, both vary in the same direction. A network identification procedure that employs these experimental perturbations will infer a positive correlation between measured activity of these enzymes. On the other hand, an experiment in which the second phosphatase, F_T, is perturbed, will lead to an inference of a graph "repression" edge. Indeed, when decreasing the second phosphatase, a "local" perturbation in the second layer, the total amount of active enzyme y increases, as it should, but the effect on the "upstream" layer quantified by x is negative, which suggests a repression of x by y. These issues, including the apparently paradoxical effect of two different perturbations leading to opposite conclusions, are extensively discussed in [7], which conducted an experimental validation of this idea.

In order to obtain the additional information, about total active kinase X and product Y, we proceeded as follows. We first add two artificial variables, x and y, so that the full state is now $(e, m_0, a, m_1, g, b, n_0, c, n_1, f, d, x, y)$. The definitions of x and y are incorporated into two new "stoichiometric constraints" corresponding to these vectors in Σ:

$$(0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, -1, 0),$$

$$(0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, -1)$$

respectively. No change is made to the original stoichiometry matrix and original stoichiometric constraints, except for adding zeroes in the positions of x and y. The original algorithm can be run on this extended set. However, when adding artificial variables, such as x and y, which do not participate in reactions nor the original set of stoichiometric constraints, it is more efficient to first obtain solutions for the original problem, in which x and y have not yet been added, and only as a second step to add the "stoichiometric constraints" corresponding to the added variables. This typically results in a substantial savings of computing time. With this modified procedure, we obtained the following results.

E_T, G_T, F_T, M_T fixed, so that only the first kinase, E_T, is allowed to vary:

−1	1	−1	−1	1	−1	1	−1	−1	1	−1	−1	−1
e	$m0$	a	$m1$	g	b	$n0$	c	$n1$	f	d	x	y

E_T, G_T, F_T, N_T fixed, so that only the first substrate, M_T, is allowed to vary:

−1	1	1	1	−1	1	−1	1	1	−1	1	1	1
e	$m0$	a	$m1$	g	b	$n0$	c	$n1$	f	d	x	y

E_T, F_T, M_T, N_T fixed, so that only the first phosphatase, G_T, is allowed to vary:

−1	1	1	−1	1	1	1	−1	−1	1	−1	−1	−1
e	$m0$	a	$m1$	g	b	$n0$	c	$n1$	f	d	x	y

E_T, G_T, F_T, M_T fixed, so that only the second substrate, N_T, is allowed to vary:

−1	1	1	1	−1	1	−1	−1	−1	1	−1	−1	−1
e	$m0$	a	$m1$	g	b	$n0$	c	$n1$	f	d	x	y

E_T, G_T, M_T, N_T fixed, so that only the second phosphatase, F_T, is allowed to vary:

−1	1	1	1	−1	1	−1	−1	1	−1	−1	−1	1
e	$m0$	a	$m1$	g	b	$n0$	c	$n1$	f	d	x	y

Let us interpret these solutions. Take for example the solution obtained when only the last substrate, N_T, was allowed to vary. Both zero and the negative of this sign vector, namely:

1	−1	−1	−1	1	−1	1	1	1	−1	1	1	1
e	$m0$	a	$m1$	g	b	$n0$	c	$n1$	f	d	x	y

are solutions. This negative version is easier to interpret: since the changes in n_0, c, n_1, d are all positive and, by the definition (6), $N_T = n_0 + c + n_1 + d$, these are the signs of changes in steady states when N_T is experimentally *increased*. In this second form of the solution, we can read-out the changes (positive for x and y, negative for b, and so forth) under such a perturbation.

5.2 A phosphotransfer model

(We thank Domitilla del Vecchio for suggesting that we study this example.) Consider the two reversible reactions

$$X_0 + Y_1 \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} X_1 + Y_0$$

$$X_1 + Y_1 \underset{k_{-2}}{\overset{k_2}{\rightleftharpoons}} X_2 + Y_0$$

(we display rate constants because they play a role in the virtual constraints described later). This network can be thought to describe a phosphotransferase Y which, when in active (phosphorylated) form Y_1 transfers a phosphate group to X_0 (and hence becomes inactivated, denoted by Y_0, while X_0 becomes X_1), and which when active can also transfer a second phosphate group to X_1 (and hence becomes inactivated, while X_1 becomes X_2). We write coordinates of states as $x = (x_0, x_1, x_2, y_0, y_1)$. Two conservation laws are as follows:

$$x_0 + x_1 + x_2 = X_T$$
$$x_1 + 2x_2 + y_1 = P_T$$

representing the conservation of total X and total number of phosphate groups.

Two rows of $f = \Gamma R(x)$ are $f_1(x) = k_1 x_0 y_1 - k_{-1} x_1 y_0$ and $f_2(x) = k_2 x_1 y_1 - k_{-2} x_2 y_0$, so, as discussed earlier, using the virtual constraint obtained from $k_{-2} x_2 f_1(x) - k_{-1} x_1 f_2(x)$, we may add to $\mathcal{T}$ the following sign vector:

$$(1, -1, 1, 0, 0)$$

We ask now what happens if the total amount of kinase, $y_0 + y_1 = Y_T$, is allowed to vary, but keeping X_T and P_T constant.

CRNSESI returns this output:

−1	★	1	−1	−1
x_0	x_1	x_2	y_0	y_1

(all signs could be reversed and that would also be a solution). This means that x_0, y_0, and y_1 change in the same direction, but x_2 in the opposite direction, and x_1 is undetermined (star). Since $y_0 + y_1 = Y_T$, an increase in Y_T means that both y_0 and y_1 increase, and thus we conclude that x_0 increases and x_2 decreases when the kinase amount is up-regulated.

Is the fact that our theory cannot unambiguously predict the actual change in x_1 at steady state, under kinase perturbations, a reflection of an incomplete search by our algorithm, or an intrinsic property of this system? To answer this question, we simulated the system, taking for concreteness all parameters $k_i = 1$.

First, let us simulate a system in which $X_T = P_T = 10$ and we study a 10% up-regulation from $Y_T = 1$. We start from these the following two initial states:

$$(1, 9, 0, 0, 1)^T \quad (1, 9, 0, 0.1, 1)^T$$

which correspond to $Y_T = 1$ and $Y_T = 1.1$, respectively. The steady states reached from here are as shown in the first and second rows, respectively, of the following matrix:

$$\begin{pmatrix} 3.5772 & 3.3275 & 3.0953 & 0.5181 & 0.4819 \\ 3.6009 & 3.3264 & 3.0727 & 0.5718 & 0.5282 \end{pmatrix}$$

which means that the sign changes are:

$$1 \quad -1 \quad -1 \quad 1 \quad 1$$

consistently with our theoretical prediction.

Next, let us simulate a system in which $X_T = 8$, $P_T = 10$, and we study a 10% up-regulation from $Y_T = 3$, which is achieved by taking these two initial states:

$$(1, 7, 0, 0, 3)^T \quad (1, 7, 0, 0.3, 3)^T$$

which correspond to $Y_T = 3$ and $Y_T = 3.3$, respectively. The steady states reached from here are as shown in the first and

doi: 10.1049/iet-syb.2014.0025

second rows, respectively, of the following matrix:

$$\begin{pmatrix} 2.4505 & 2.6607 & 2.8888 & 1.4383 & 1.5617 \\ 2.5166 & 2.6638 & 2.8197 & 1.6031 & 1.6969 \end{pmatrix}$$

which means that the sign changes are now:

$$1 \quad 1 \quad -1 \quad 1 \quad 1$$

again consistently with our theoretical prediction.

These simulations explain why the actual change in x_1 at steady state, under kinase perturbations, cannot be unambiguously predicted from our algorithm, which does not take into the numerical values of the conserved quantities (nor, for that matter, of the kinetic constants k_i's). It is remarkable, however, that the sign of the perturbation in the "active" form x_2 can be unambiguously predicted (and perhaps counter-intuitive that the change is negative).

We also run CRNSESI on two other scenarios: (1) keeping X_T and Y_T constant gives these signs:

$$\begin{matrix} -1 & \star & 1 & -1 & 1 \\ x_0 & x_1 & x_2 & y_0 & y_1 \end{matrix}$$

and (2) keeping P_T and Y_T constant results in:

$$\begin{matrix} -1 & -1 & \star & -1 & 1 \\ x_0 & x_1 & x_2 & y_0 & y_1 \end{matrix}$$

in which case the signs of perturbations in the variable x_2 are not uniquely defined.

5.3 A ligand/receptor/antagonist/trap example

(We thank Gilles Gnacadja for suggesting that we try CRNSESI on this example.) The paper [40] studied a system that models the binding of interleukin-1 (IL-1) ligand to IL-1 type I receptor (IL-1RI), under competitive binding to the same receptor by human IL-1 receptor antagonist (IL-1Ra). IL-1Ra is used as a therapeutic agent in order to block IL-1 binding (which causes undesirable physiological responses). In addition, the model included the presence of a decoy (or "trap") receptor that binds to both IL-1 and IL-1Ra. A key question addressed in that paper was the determination of how the equilibrium concentration of the receptor–ligand complex depends on initial concentrations of the various players (reflected in variations in stoichiometrically conserved quantities), and specifically the determination of the direction of the changes in concentrations. We show here how CRNSESI recovers conclusions from that paper, which were obtained there through very ingenious and lengthy *ad-hoc* computations.

We will employ the same notations as in [40]: the species X_i, $i = 1, 2, 3, 4$ are, respectively, the ligand IL-1, receptor IL-1RI, antagonist IL-1Ra, and trap; and the species Y_i, $i = 1, 2, 3, 4$ are, respectively, the complexes X_1X_2, X_2X_3, X_3X_4, and X_4X_1. Thus, the reaction network is

$$\begin{aligned} X_1 + X_2 &\rightleftharpoons Y_1 \\ X_2 + X_3 &\rightleftharpoons Y_2 \\ X_3 + X_4 &\rightleftharpoons Y_3 \\ X_4 + X_1 &\rightleftharpoons Y_4 \end{aligned}$$

We use lower case letters to denote concentrations. There are four independent conservation laws:

$$\begin{aligned} x_1 + y_4 + y_1 &= b_1 \\ x_2 + y_1 + y_2 &= b_2 \\ x_3 + y_2 + y_3 &= b_3 \\ x_4 + y_3 + y_4 &= b_4 \end{aligned}$$

We will fix b_2, b_3, and b_4, and ask how steady states change in sign when b_1 is perturbed. The other cases (perturb b_2, etc.) are of course similar.

It is easy to see that $\alpha y_1 y_3 = \beta y_2 y_4$, for some positive constants α, β, at all steady states, and this allows one to introduce an additional virtual constraint obtained from $\alpha y_1 y_3 - \beta y_2 y_4$, meaning that we may add the following sign vector:

$$(0, 0, 0, 0, 1, \ -1, 1, \ -1)$$

to $\mathcal{T}$. Indeed, four rows of the vector field are: $f_1 = k_1 x_1 x_2 - \ell_1 y_1$, $f_2 = k_2 x_2 x_3 - \ell_2 y_2$, $f_3 = k_3 x_3 x_4 - \ell_3 y_3$, $f_4 = k_4 x_4 x_1 - \ell_4 y_4$ (for appropriate positive constants k_i and ℓ_i). So, at steady states, y_1 is a multiple of $x_1 x_2$, and similarly for the other y_i's, which gives that $y_1 y_3$ and $y_2 y_4$ are both multiples of $x_1 x_2 x_3 x_4$. Another way to say this is to note that the linear combination

$$k_1 k_3 k_4 x_1 x_4 f_2 + k_1 k_3 \ell_2 y_2 f_4 - k_2 k_3 k_4 x_3 x_4 f_1 - k_2 k_4 \ell_1 y_1 f_3$$

gives

$$k_2 k_4 \ell_1 \ell_3 y_1 y_3 - k_1 k_3 \ell_2 \ell_4 y_2 y_4$$

With this virtual constraint added, CRNSESI returns

$$\begin{matrix} 1 & -1 & 1 & -1 & 1 & * & * & 1 \\ x_1 & x_2 & x_3 & x_4 & y_1 & y_2 & y_3 & y_4 \end{matrix}$$

for the signs of derivatives with respect to b_1. Note that two variables are undetermined in sign. (To be more precise, CRNSESI also returns the negatives of these signs. However, since $b_1 = x_1 + y_4 + y_1$, and since all three of x_1, y_4, y_1 change with the same sign, the negative corresponds to the derivative with respect to $-b_1$.) This is exactly what is proved in [40] (see the first columns of the matrices in (10) and (12) in that paper). Notably, CRNSESI gave slightly more, namely that these particular signs of $(dy_2/db_1, dy_3/db_1)$ can never appear:

$$(1, 1), (1, 0), (0, 1), (0, 0).$$

In other words, it cannot be the case that both y_2 and y_3 increase.

6 Some technical proofs

We collect here some of the longer proofs.

6.1 Proof of Lemma 3

Proof: Pick any $\widetilde{\mu}_v \in \widetilde{\Sigma}_0$, where $v \in \mathbf{V}_G \subseteq \mathbf{V}$, and fix any positive concentration vector x. We must prove that $\widetilde{\mu}_v \in \Sigma(x)$. As $\Sigma(x)$ includes all expressions of the form $\text{sign}(vR'(x))$, for $v \in \mathbf{V}$, it will suffice to show that, for this

same vector v,

$$\operatorname{sign}\left(v\frac{\partial R}{\partial x_j}(x)\right) = \operatorname{sign}(vA_j) \tag{40}$$

for each species index $j \in \{1, \ldots, n_s\}$. For each $j \in \{1, \ldots, n_r\}$, we will show the following three statements:

$$v^- A_j > 0 \text{ (so } v^+ A_j = 0) \;\Rightarrow\; v\frac{\partial R}{\partial x_j}(x) = -v^- \frac{\partial R}{\partial x_j}(x) < 0 \tag{41}$$

$$v^+ A_j > 0 \text{ (so } v^- A_j = 0) \;\Rightarrow\; v\frac{\partial R}{\partial x_j}(x) = v^+ \frac{\partial R}{\partial x_j}(x) > 0 \tag{42}$$

and

$$v^- A_j = v^+ A_j = 0 \;\Rightarrow\; v\frac{\partial R}{\partial x_j}(x) = 0 \tag{43}$$

Suppose first that $v^- A_j > 0$. Applying (28) with $\rho = v^+$, we have that $v^+ \frac{\partial R}{\partial x_j}(x) = 0$. Applying (29) with $\rho = v^-$, we have that $v^- \frac{\partial R}{\partial x_j}(x) > 0$. Therefore,

$$v\frac{\partial R}{\partial x_j}(x) = (v^+ - v^-)\frac{\partial R}{\partial x_j}(x) = v^+ \frac{\partial R}{\partial x_j}(x) - v^- \frac{\partial R}{\partial x_j}(x) = -v^- \frac{\partial R}{\partial x_j}(x) < 0$$

thus proving (41). If, instead, $v^- A_j = 0$ and $v^+ A_j > 0$, a similar argument shows that (42) holds. Finally, suppose that $v^+ A_j = v^- A_j = 0$. Then, again by (28), applied to $\rho = v^+$ and $\rho = v^-$

$$v\frac{\partial R}{\partial x_j}(x) = (v^+ - v^-)\frac{\partial R}{\partial x_j}(x) = 0$$

and so (43) holds. The desired equality (40) follows from (41)–(43). Indeed, we consider three cases: (a) $vA_j < 0$, (b) $vA_j > 0$, and (c) $vA_j = 0$. In case (a), (30) shows that $vA_j = -v^- A_j$ (because the first and third cases would give a non-negative value), and therefore $-v^- A_j < 0$, that is, $v^- A_j > 0$, so (41) gives that $v\frac{\partial R}{\partial x_j}(x)$ is also negative. In case (b), similarly $v^+ A_j = vA_j > 0$, and so (42) shows (40). Finally, consider case (c), $vA_j = 0$. If it were the case that $v^+ A_j$ is non-zero, then, since $v \in \mathbf{V}_G$, $v^- A_j = 0$, and therefore (30) gives that $vA_j = v^+ A_j > 0$, a contradiction; similarly, $v^- A_j$ must also be zero. So, (43) gives that $v\frac{\partial R}{\partial x_j}(x) = 0$ as well. □

6.2 Proof of Proposition 1

Proof: Let $s = \operatorname{sign} v$, $v \in \mathbf{V}$, and pick any $j \in \{1, \ldots, n_s\}$. We claim that $s^{\pm}\alpha_j = 0$ if and only if $v^{\pm} A_j = 0$. Since j is arbitrary, this shows that $s \in \mathbf{S}_G$ if and only if $v \in \mathbf{V}_G$. Indeed, suppose that $s^+\alpha_j = 0$. By Lemma 4, $\operatorname{sign}(v^+ A_j) = \operatorname{sign}(s^+\alpha_j) = 0$, so $v^+ A_j = 0$. Conversely, if $v^+ A_j = 0$ then $s^+\alpha_j = 0$, for the same reason. Similarly, $s^-\alpha_j = 0$ is equivalent to $v^- A_j = 0$.

Suppose now that $s \in \mathbf{S}_G$ and $v \in \mathbf{V}_G$, and pick any $j \in \{1, \ldots, n_s\}$. Assume that $s^+\alpha_j = 0$. Since, by (35) and (30), $s\alpha_j = -s^-\alpha_j$ and $vA_j = -v^- A_j$, we have, again by Lemma 4, that

$$\operatorname{sign}(s\alpha_j) = -\operatorname{sign}(s^-\alpha_j) = -\operatorname{sign}(v^- A_j) = \operatorname{sign}(vA_j)$$

If, instead, $s^-\alpha_j = 0$ (and thus $v^- A_j = 0$)

$$\operatorname{sign}(s\alpha_j) = \operatorname{sign}(s^+\alpha_j) = \operatorname{sign}(v^+ A_j) = \operatorname{sign}(vA_j)$$

As j was arbitrary, and we proved that the jth coordinates of the two vectors in (37) are the same, the vectors must be the same. □

6.3 Proof of Lemma 5

Proof: Pick any $s \in \mathbf{S}$. Thus $s = \operatorname{sign} v$, where $v = \nu\Gamma$ for some $\nu \in \mathbb{R}^{1\times n_s}$. Consider the set of indices of the coordinates of v that vanish (equivalently, $s_i = 0$),

$$I = \{i \in \{1, \ldots n_s\} \mid v_i = 0\}.$$

Suppose that $I = \{i_1, \ldots, i_p\}$. Let e_i denote the canonical column vector $(0, \ldots 0, 1, 0, \ldots 0)^T$ with a "1" in the ith position and zeroes elsewhere, and introduce the $n_s \times p$ matrix $E_I = (e_{i_1}, e_{i_2}, \ldots, e_{i_p})$. The definition of I means that $\nu\Gamma E_I = vE_I = 0$ and $\nu\Gamma e_j = ve_j = v_j \neq 0$ for all $j \notin I$. The matrix $D = \Gamma E_I$ has integer, and in particular rational, entries. Thus, the left nullspace of D has a rational basis, that is, there is a set of rational vectors $\{u_1, \ldots, u_q\}$, where q is the dimension of this nullspace, such that $u_i D = 0$ and $uD = 0$ if and only if u is a linear combination of the u_i's. In particular, since $\nu D = 0$, there are real numbers $r_1, \ldots, r_q$ such that $\nu = \sum_i r_i u_i$. Now pick sequences of rational numbers $r_i^{(k)} \to r_i$ as $k \to \infty$ and define $\nu^{(k)} := \sum_i r_i^{(k)} u_i$. This sequence converges to ν, and, being combinations of the u_i's, $\nu^{(k)} D = 0$ for all k. Let $v^{(k)} := \nu^{(k)}\Gamma$, so we have that $v^{(k)} \to v$ as $k \to \infty$, and $v^{(k)} E_I = 0$ for all k. On the other hand, for each $j \notin I$, as $ve_j \neq 0$, for all large enough k, $(v^{(k)})_j$, the jth coordinate of $v^{(k)}$, has the same sign as v_j. In conclusion, for large enough k, $\operatorname{sign} v^{(k)} = \operatorname{sign} v = s$. Multiplying the rational vector $\nu^{(k)}$ by the least common denominator of its coordinates, the sign does not change, but now we have an integer vector with the same sign. □

7 Acknowledgments

This research was supported in part by NIH Grant 1R01GM100473, ONR Grant N00014-13-1-0074, and AFOSR Grant FA9550-14-1-0060.

8 References

1 Kholodenko, B.N., Kiyatkin, A., Bruggeman, F., Sontag, E.D., Westerhoff, H., Hoek, J. 'Untangling the wires: a novel strategy to trace functional interactions in signalling and gene networks'. Proc. National Academy of Sciences USA, 2002, 99, 12841–12846

2 de la Fuente, A., Brazhnik, P., Mendes, P.: 'Linking the genes: inferring quantitative gene networks from microarray data', *Trends Genet.*, 2002, **18**, (8), 395–398.

doi: 10.1049/iet-syb.2014.0025

3 Gardner, T.S., di Bernardo, D., Lorenz, D., Collins, J.J.: 'Inferring genetic networks and identifying compound mode of action via expression profiling', *Science*, 2003, **301**, (5629), pp. 102–105
4 Andrec, M., Kholodenko, B.N., Levy, R.M., Sontag, E.D.: 'Inference of signalling and gene regulatory networks by steady-state perturbation experiments: structure and accuracy', *J. Theoret. Biol.*, 2005, **232**, (3), pp. 427–441
5 Santos, S.D.M., Verveer, P.J., Bastiaens, P.I.H.: 'Growth factor induced MAPK network topology shapes Erk response determining PC-12 cell fate', *Nat. Cell Biol.*, 2007, **9**, pp. 324–330
6 Kholodenko, B., Yaffe, M.B., Kolch, W.: 'Computational approaches for analyzing information flow in biological networks', *Sci. Signal.*, 2012, **5**, (220), re1.
7 Prabakaran, S., Gunawardena, J., Sontag, E.D.: 'Paradoxical results in perturbation-based signalling network reconstruction', *Biophys. J.*, 2014, **106**, pp. 2720–2728
8 Pearson, G., Robinson, F., Beers Gibson, T., *et al.*: 'Mitogen-activated protein (MAP) kinase pathways: regulation and physiological functions', *Endocr. Rev.*, April 2001, **22**, (2), pp. 153–183
9 Huang, C.-Y.F., Ferrell, J.E.: 'Ultrasensitivity in the mitogen-activated protein kinase cascade', *Proc. Natl. Acad. Sci. USA*, 1996, **93**, pp. 10078–10083
10 Shaul, Y.D., Seger, R.: 'The MEK/ERK cascade: from signalling specificity to diverse functions', *Biochim. Biophys. Acta*, 2007, **1773**, (8), pp. 1213–1226
11 Bardwell, L., Zou, X., Nie, Q., Komarova, N.L.: 'Mathematical models of specificity in cell signalling', *Biophys. J.*, 2007, **92**, (10), pp. 3425–3441
12 Ledford, H.. First-in-class cancer drug approved to fight melanoma. http://blogs.nature.com/news/2013/05/first-in-class-cancer-drug-approved-to-fight-melanoma.html, May 2013. Nature news blog
13 Willems, J.C.: 'Behaviors, latent variables, and interconnections', *Syst., Control Inf.*, 1999, **43**, (9), pp. 453–464
14 Lauffenburger, D.A.: 'Cell signalling pathways as control modules: complexity for simplicity? *Proc. Natl. Acad. Sci. USA*, 2000, **97**, pp. 5031–5033
15 Hartwell, L.H., Hopfield, J.J., Leibler, S., Murray, A.W.: 'From molecular to modular cell biology', *Nature*, 1999, **402**, pp. 47–52
16 Saez-Rodriguez, J., Kremling, A., Gilles, E.D.: 'Dissecting the puzzle of life: modularization of signal transduction networks', *Comput. Chem. Eng.*, 2005, pp. 619–629
17 Del Vecchio, D., Ninfa, A.J., Sontag, E.D.: 'Modular cell biology: Retroactivity and insulation', *Nat. Mol. Syst. Biol.*, 2008, **4**, p. 161
18 Kim, K.H., Sauro, H.M.: 'Measuring retroactivity from noise in gene regulatory networks', *Biophys. J.*, March 2011, **100**, (5), pp. 1167–1177
19 Alexander, R.P., Kim, P.M., Emonet, T., Gerstein, M.B.: 'Understanding modularity in molecular networks requires dynamics', *Sci. Signal.*, 2009, **2**, (81), p. 44
20 Jiang, P., Ventura, A.C., Sontag, E.D., Merajver, S.D., Ninfa, A.J., Del Vecchio, D.: 'Load-induced modulation of signal transduction networks', *Sci. Signal.*, 2011, **4**, (194), ra67
21 Kim, Y., Paroush, Z., Nairz, K., Hafen, E., Jimenez, G., Shvartsman, S. Y.: 'Substrate-dependent control of MAPK phosphorylation in vivo', *Mol. Syst. Biol.*, 2011, **7**, p. 467.
22 Jayanthi, S., Nilgiriwala, K.S., Del Vecchio, D.: 'Retroactivity controls the temporal dynamics of gene transcription', *ACS Synth. Biol.*, August 2013, **2**, (8), pp. 431–441
23 Barton, J., Sontag, E.D.: 'The energy costs of insulators in biochemical networks', *Biophys. J.*, 2013, **104**, pp. 1390–1380
24 Björner, A., Las Vergnas, M., Sturmfels, B., White, N., Ziegler, G.M., 'Oriented Matroids' (Cambridge University Press, 1999, 2nd edn.)
25 Harrison, J., 'Handbook of practical logic and automated reasoning' (Cambridge University Press, New York, NY, USA, 2009, 1st edn.)
26 Feinberg, M.: 'Chemical reaction network structure and the stability of complex isothermal reactors – i. the deficiency zero and deficiency one theorems', *Chem. Eng. Sci.*, 1987, **42**, pp. 2229–2268.
27 Feinberg, M.: 'The existence and uniqueness of steady states for a class of chemical reaction networks', *Archive for Rational Mechanics and Analysis*, 1995, **132**, pp. 311–370.
28 Conradi, C., Saez-Rodriguez, J., Gilles, E.-D., Raisch, J.: 'Using chemical reaction network theory to discard a kinetic mechanism hypothesis', *IEE Proc. Systems Biology*, 2005, **152**, pp. 243–248
29 Sontag, E.D.: 'Structure and stability of certain chemical networks and applications to the kinetic proofreading model of T-cell receptor signal transduction', *IEEE Trans. Autom.. Control*, 2001, **46**, (7), pp. 1028–1047
30 Craciun, G, Feinberg, M: 'Multiple equilibria in complex chemical reaction networks: extensions to entrapped species models', *IEE Proc Syst. Biol*, 2006, **153**, (4), pp. 179–186
31 Craciun, G., Tang, Y., Feinberg, M.: 'Understanding bistability in complex enzyme-driven reaction networks', *PNAS*, 2006, **103**, (23), pp. 8697–8702
32 Wang, L., Sontag, E.D.: 'On the number of steady states in a multiple futile cycle', *J. Math. Biol.*, 2008, **57**, pp. 29–52
33 Siegal-Gaskins, D., Grotewold, E., Smith, G.D.: 'The capacity for multistability in small gene regulatory networks', *BMC Syst. Biol.*, 2009, **3**, p. 96
34 Wilhelm, T.: 'The smallest chemical reaction system with bistability', *BMC Syst. Biol.*, 2009, **3**, p. 90
35 Halasz, A.M., Lai, H-J., McCabe Pryor, M., Radhakrishnan, K., Edwards, J.S.: 'Analytical solution of steady-state equations for chemical reaction networks with bilinear rate laws', *IEEE/ACM Trans. Comput. Biol. Bioinfor.*, 2013, **10**, (4), pp. 957–969.
36 Griva, I., Nash, S.G., Sofer, A., 'Linear and nonlinear optimization' (Society for Industrial Mathematics, Philadelphia, 2008, 2nd ed.)
37 Papadimitriou, C.H., Steiglitz, K., 'Combinatorial optimization: algorithms and complexity' (Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1982)
38 Williams, H.P., 'Logic and integer programming' (Springer Publishing Company, Incorporated, 2009, 1st edn.)
39 Guillen-Gosalbez, G., Miro, A., Alves, R., Sorribas, A., Jimenez, L.: 'Identification of regulatory structure and kinetic parameters of biochemical networks via mixed-integer dynamic optimization', *BMC Syst. Biol.*, October 2013, **7**, (1), p. 113.
40 Gnacadja, G., Shoshitaishvili, A., Gresser, M.J., *et al.*: 'Monotonicity of interleukin-1 receptor-ligand binding with respect to antagonist in the presence of decoy receptor', *J. Theor. Biol.*, February 2007, **244**, (3), pp. 478–488
41 Sontag, E.D.. Online text: Lecture Notes on Mathematical Systems Biology. http://www.math.rutgers.edu/~sontag/systems_biology_notes.pdf, 2014
42 Ingalls, B., 'Mathematical modeling in systems biology' (MIT Press, 2013)

9 Appendix

9.1 A review of chemical reaction networks terminology

We review here some basic notions about chemical networks. See, for example [41, 42] for more details. We consider a collection of chemical reactions that involves a set of n_s "species":

$$S_i,\ i \in \{1, 2, \ldots n_s\}$$

The "species" might be ions, atoms, or large molecules, depending on the context. A CRN involving these species is a set of chemical reactions $\mathcal{R}_k,\ k \in \{1, 2, \ldots, n_r\}$, represented symbolically as:

$$\mathcal{R}_k: \quad \sum_{i=1}^{n_s} a_{ik} S_i \to \sum_{i=1}^{n_s} b_{ik} S_i \tag{44}$$

where the a_{ik} and b_{ik} are some non-negative integers that quantify the number of units of species S_i consumed, respectively, produced, by reaction $\mathcal{R}_k$. Thus, in reaction 1, a_{11} units of species S_1 combine with a_{21} units of species S_2 and so on, to produce b_{11} units of species S_1, b_{21} units of species S_2 and so on, and similarly for each of the other $n_r - 1$ reactions. (If there is a reverse reaction to (44), $\sum_{i=1}^{n_s} a'_{ik} S_i \to \sum_{i=1}^{n_s} b'_{ik} S_i$ with $b'_{ik} = a_{ik}$ and $a'_{ik} = b_{ik}$, one sometimes summarises both by a reversible arrow $\sum_{i=1}^{n_s} a_{ik} S_i \rightleftharpoons \sum_{i=1}^{n_s} b_{ik} S_i$. However, from a theoretical standpoint, we view each direction as a separate reaction.)

We will assume the following "non-autocatalysis" condition: no species S_i can appear on both sides of the same reaction. With this assumption, either $a_{ik} = 0$ or $b_{ik} = 0$ for each species S_i and each reaction $\mathcal{R}_k$ (both are zero if the species in question is neither consumed nor produced), Note that we are not excluding autocatalysis

which occurs through one ore more intermediate steps, such as the autocatalysis of S_1 in $S_1 + S_2 \to S_3 \to 2S_1 + S_4$, so this assumption is not as restrictive as it might at first appear.

Suppose that $a_{ik} > 0$ for some (i, k); then we say that species S_i is a *reactant* of reaction $\mathcal{R}_k$, and by the non-autocatalysis assumption, $b_{ik} = 0$ for this pair (i, k). If instead $b_{ik} > 0$, then we say that species S_i is a *product* of reaction $\mathcal{R}_k$, and again by the non autocatalysis assumption, $a_{ik} = 0$ for this pair (i, k).

It is convenient to arrange the a_{ik}'s and b_{ik}'s into two $n_s \times n_r$ matrices A, B, respectively, and introduce the *stoichiometry matrix* $\Gamma = B - A$. In other words,

$$\Gamma = (\gamma_{ij})_{ij} \in \mathbb{R}^{n_s \times n_r}$$

is defined by:

$$\gamma_{ij} \;=\; b_{ij} - a_{ij}\,, \quad i = 1, \ldots, n_s\,, \quad j = 1, \ldots, n_r \tag{45}$$

The matrix Γ has as many columns as there are reactions. Its kth column shows, for each species (ordered according to their index i), the net "produced–consumed" by reaction $\mathcal{R}_k$. The symbolic information given by the reactions (44) is summarised by the matrix Γ. Observe that $\gamma_{ik} = -a_{ik} < 0$ if S_i is a reactant of reaction $\mathcal{R}_k$, and $\gamma_{ik} = b_{ik} > 0$ if S_i is a product of reaction $\mathcal{R}_k$.

To describe how the state of the network evolves over time, one must provide in addition to Γ a rule for the evolution of the vector:

$$\begin{pmatrix} [S_1(t)] \\ [S_2(t)] \\ \vdots \\ [S_{n_s}(t)] \end{pmatrix}$$

where the notation $[S_i(t)]$ means the concentration of the species S_i at time t. We will denote the concentration of S_i simply as $x_i(t) = [S_i(t)]$ and let $x = (x_1, \ldots, x_{n_s})^T$. Observe that only non-negative concentrations make physical sense. A zero concentration means that a species is not present at all; we will be interested in *positive vectors* x of concentrations, those for which $x_i > 0$ for all i, meaning that all species are present.

Another ingredient that we require is a formula for the actual rate at which the individual reactions take place. We denote by $R_k(x)$ be algebraic form of the kth reaction. We postulate the following two axioms that the reaction rates $R_k(x)$, $k = 1, \ldots, n_r$ must satisfy:

- for each (i, k) such that species S_i is a reactant of $\mathcal{R}_k$, $\frac{\partial R_k}{\partial x_i}(x) > 0$ for all (positive) concentration vectors x;
- for each (i, k) such that species S_i is not a reactant of $\mathcal{R}_k$, $\frac{\partial R_k}{\partial x_i}(x) = 0$ for all (positive) concentration vectors x.

These axioms are natural, and are satisfied by every reasonable model, and specifically by mass-action kinetics, in which the reaction rate is proportional to the product of the concentrations of all the reactants:

$$R_k(x) = \kappa_k \prod_{i=1}^{n_s} x_i^{a_{ij}} \;\text{ for all } j = 1, \ldots, n_r$$

The positive coefficients κ_k are the called reaction, or kinetic, constants. By convention, $x_i^{a_{ij}} = 1$ when $a_{ij} = 0$.

Recall that $a_{ik} > 0$ and $b_{ik} = 0$ if and only if S_i is a reactant of $\mathcal{R}_k$. Therefore the above axioms state that, for every positive x,

$$\frac{\partial R_k}{\partial x_i}(x) > 0 \iff a_{ik} > 0 \tag{46}$$

and also

$$\frac{\partial R_k}{\partial x_i}(x) = 0 \iff a_{ik} = 0 \tag{47}$$

because the expressions on both sides are either zero or positive.

We arrange reactions into a column vector function $R(x) \in \mathbb{R}^{n_r}$:

$$R(x) := \begin{pmatrix} R_1(x) \\ R_2(x) \\ \vdots \\ R_{n_r}(x) \end{pmatrix}$$

With these conventions, the system of differential equations associated to the CRN is given as in (25), which we repeat here for convenience:

$$\frac{dx}{dt} \;=\; f(x) \;=\; \Gamma\, R(x)$$

9.2 Existence and uniqueness for steady states in the example

For our motivating example (1), steady states are unique once that *all* conservation laws are taken into account. Existence of steady states follows from the fact that states evolve in a compact convex set, as argued, for example, in [23] (Supplemental Material). Uniqueness is shown as follows. Steady states satisfy that the right-hand sides of the differential equations:

$$\begin{aligned}
\dot{e} &= -\alpha m_0 e + \beta a + \chi a \\
\dot{m}_0 &= -\alpha m_0 e + \beta a + \phi b \\
\dot{a} &= \alpha m_0 e - \beta a - \chi a \\
\dot{m}_1 &= \chi a - \delta m_1 g + \varepsilon b - \gamma n_0 m_1 + \eta c + \iota c \\
\dot{g} &= -\delta m_1 g + \varepsilon b + \phi b \\
\dot{b} &= \delta m_1 g - \varepsilon b - \phi b \\
\dot{n}_0 &= -\gamma n_0\, m_1 + \eta c + \lambda d \\
\dot{c} &= \gamma n_0\, m_1 - \eta c - \iota c \\
\dot{n}_1 &= \iota c - \varphi n_1 f + \kappa d \\
\dot{f} &= -\varphi n_1 f + \kappa d + \lambda d \\
\dot{d} &= \varphi n_1 f - \kappa d - \lambda d
\end{aligned}$$

(where $\alpha, \beta, \ldots$ are some positive constants) are set to zero, together with the conservation laws. We argue as follows, using the constraints to first express all variables in terms of e, seen as a parameter, and then pointing out that this forces

doi: 10.1049/iet-syb.2014.0025

e to be uniquely determined ("increasing" and "decreasing" functions always means strictly so):

1. the conservation law for E_T gives that a is a decreasing function of e;
2. substituting $a = E_T - e$ into $\dot{e} = 0$ and solving for m_0 gives that m_0 is a decreasing function of e;
3. from $\dot{e} - \dot{m}_0 = 0$, b is an increasing function of a, and therefore b is a decreasing function of e;
4. substituting $g = G_T - b$ into $\dot{g} = 0$ and solving for m_1 gives that m_1 is an increasing function of b, and thus m_1 is a decreasing function of e;
5. the conservation law for M_T gives that c is a decreasing function of m_0, a, m_1, b, so c is an increasing function of e;
6. from $\dot{n}_1 - \dot{f} = 0$, d is an increasing function of c, so d is an increasing function of e;
7. solving $\dot{c} = 0$ for n_0 gives that n_0 is increasing in c and decreasing in m_1, so n_0 is an increasing function of e and an increasing function of c, and thus n_0 is an increasing function of e;
8. substituting $f = F_T - d$ into $\dot{f} = 0$ and solving for n_1 gives that n_1 is an increasing function of d, so n_1 is an increasing function of c, and thus n_1 is an increasing function of e.

In conclusion, the sum of concentrations $n_0 + c + n_1 + d$ is a strictly increasing function $\theta(e)$ of concentration of e. Thus, the constraint $N_T = \theta(e)$ provides a unique possible value for e. Substituting back, (unique) values are obtained for all other concentrations.

Supplementary Material

A technique for determining the signs of sensitivities of steady states in chemical reaction networks

Eduardo D. Sontag, IET Systems Biology, 2014

Contents:

Program output for these three examples:

S1. Kinase/substrate example

(Omitting preliminary output, which lists linear combinations of stoichiometric constraints. In every case, generating combinations $1-1$, $1+1$, $1+2$, $1-2$, $2-1$, and $2+1$ of rows.)

Kinase/substrate example, keep $G_{\mathrm{T}}, F_{\mathrm{T}}, M_{\mathrm{T}}, N_{\mathrm{T}}$ constant

```
how many possible signs: 88573
after using constraint     1, possible signs left: 42646
constraint used is:
    -1    -1     2     0     0     0     0     0     0     0     0
corresponding to this combination of species:
     1     0     0     0     0     0     0     0     0     0     0
after using constraint     2, possible signs left: 40459
constraint used is:
    -1    -1     1     0     0     1     0     0     0     0     0
corresponding to this combination of species:
     0     1     0     0     0     0     0     0     0     0     0
after using constraint     4, possible signs left: 34060
constraint used is:
     0     0     1    -2    -1     1    -1     2     0     0     0
corresponding to this combination of species:
     0     0     0     1     0     0     0     0     0     0     0
after using constraint     5, possible signs left: 18535
constraint used is:
     0     0     0    -1    -1     2     0     0     0     0     0
corresponding to this combination of species:
     0     0     0     0     1     0     0     0     0     0     0
after using constraint     7, possible signs left: 12334
constraint used is:
     0     0     0    -1     0     0    -1     1     0     0     1
corresponding to this combination of species:
     0     0     0     0     0     0     1     0     0     0     0
after using constraint     8, possible signs left:  9355
constraint used is:
     0     0     0     1     0     0     1    -2     0     0     0
corresponding to this combination of species:
     0     0     0     0     0     0     0     1     0     0     0
after using constraint     9, possible signs left:  6172
constraint used is:
     0     0     0     0     0     0     0     1    -1    -1     1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1     0     0
after using constraint    10, possible signs left:  4205
constraint used is:
     0     0     0     0     0     0     0     0    -1    -1     2
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     0     1     0
```

```
after using constraint    12, possible signs left:  1578
constraint used is:
     0     0     1     0     0    -1     0     0     0     0     0
corresponding to this combination of species:
     1    -1     0     0     0     0     0     0     0     0     0
after using constraint    40, possible signs left:  1557
constraint used is:
     0     0     1    -1    -1     1     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     1     0     0    -1     0     0     0     0
after using constraint    60, possible signs left:   613
constraint used is:
     0     0     0     0     0     0     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1    -1     0
after using constraint   305, possible signs left:   134
constraint used is:
     0     0     0     0     1     1     0     0     0     0     0
corresponding to this stoichiometry constraint:
     0     0     0     0     1     1     0     0     0     0     0
after using constraint   306, possible signs left:    29
constraint used is:
     0     0     0     0     0     0     0     0     0     1     1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     0     0     0     1     1
after using constraint   307, possible signs left:    10
constraint used is:
     0     1     1     1     0     1     0     1     0     0     0
corresponding to this stoichiometry constraint:
     0     1     1     1     0     1     0     1     0     0     0
after using constraint   308, possible signs left:     1
constraint used is:
     0     0     0     0     0     0     1     1     1     0     1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     1     1     1     0     1
ended search

possible_signs =
    -1     1    -1    -1     1    -1     1    -1    -1     1    -1
     e    m0     a    m1     g     b    n0     c    n1     f     d
```

Additional computations for "artificial variables" x and y:

```
how many possible signs:     9
after using constraint     5, possible signs left:     3
constraint used is:
     0     0     0     1     0     1     0     1     0     0     0    -1     0
corresponding to this stoichiometry constraint:
```

```
     0     0     0     1     0     1     0     1     0     0     0    -1     0
after using constraint     6, possible signs left:     1
constraint used is:
     0     0     0     0     0     0     0     0     1     0     1     0    -1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     0     0     1     0     1     0    -1
ended search

possible_signs =
    -1     1    -1    -1     1    -1     1    -1    -1     1    -1    -1    -1
     e    m0     a    m1     g     b    n0     c    n1     f     d     x     y
```

Kinase/substrate example, keep $E_{\mathrm{T}}, G_{\mathrm{T}}, F_{\mathrm{T}}, N_{\mathrm{T}}$ constant

```
how many possible signs: 88573
after using constraint     1, possible signs left: 42646
constraint used is:
    -1    -1     2     0     0     0     0     0     0     0     0
corresponding to this combination of species:
     1     0     0     0     0     0     0     0     0     0     0
after using constraint     2, possible signs left: 40459
constraint used is:
    -1    -1     1     0     0     1     0     0     0     0     0
corresponding to this combination of species:
     0     1     0     0     0     0     0     0     0     0     0
after using constraint     4, possible signs left: 34060
constraint used is:
     0     0     1    -2    -1     1    -1     2     0     0     0
corresponding to this combination of species:
     0     0     0     1     0     0     0     0     0     0     0
after using constraint     5, possible signs left: 18535
constraint used is:
     0     0     0    -1    -1     2     0     0     0     0     0
corresponding to this combination of species:
     0     0     0     0     1     0     0     0     0     0     0
after using constraint     7, possible signs left: 12334
constraint used is:
     0     0     0    -1     0     0    -1     1     0     0     1
corresponding to this combination of species:
     0     0     0     0     0     0     1     0     0     0     0
after using constraint     8, possible signs left:  9355
constraint used is:
     0     0     0     1     0     0     1    -2     0     0     0
corresponding to this combination of species:
     0     0     0     0     0     0     0     1     0     0     0
after using constraint     9, possible signs left:  6172
constraint used is:
```

```
     0     0     0     0     0     0     0     1    -1    -1     1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1     0     0
after using constraint    10, possible signs left:  4205
constraint used is:
     0     0     0     0     0     0     0     0    -1    -1     2
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     0     1     0
after using constraint    12, possible signs left:  1578
constraint used is:
     0     0     1     0     0    -1     0     0     0     0     0
corresponding to this combination of species:
     1    -1     0     0     0     0     0     0     0     0     0
after using constraint    40, possible signs left:  1557
constraint used is:
     0     0     1    -1    -1     1     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     1     0     0    -1     0     0     0     0
after using constraint    60, possible signs left:   613
constraint used is:
     0     0     0     0     0     0     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1    -1     0
after using constraint   305, possible signs left:   134
constraint used is:
     1     0     1     0     0     0     0     0     0     0     0
corresponding to this stoichiometry constraint:
     1     0     1     0     0     0     0     0     0     0     0
after using constraint   306, possible signs left:    29
constraint used is:
     0     0     0     0     1     1     0     0     0     0     0
corresponding to this stoichiometry constraint:
     0     0     0     0     1     1     0     0     0     0     0
after using constraint   307, possible signs left:     6
constraint used is:
     0     0     0     0     0     0     0     0     0     1     1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     0     0     0     1     1
after using constraint   308, possible signs left:     1
constraint used is:
     0     0     0     0     0     0     1     1     1     0     1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     1     1     1     0     1
ended search

possible_signs =
    -1     1     1     1    -1     1    -1     1     1    -1     1
     e    m0     a    m1     g     b    n0     c    n1     f     d
```

Additional computations for "artificial variables" x and y:

```
how many possible signs:     9
after using constraint     5, possible signs left:     3
constraint used is:
     0     0     0     1     0     1     0     1     0     0     0    -1     0
corresponding to this stoichiometry constraint:
     0     0     0     1     0     1     0     1     0     0     0    -1     0
after using constraint     6, possible signs left:     1
constraint used is:
     0     0     0     0     0     0     0     0     1     0     1     0    -1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     0     0     1     0     1     0    -1
ended search

possible_signs =
    -1     1     1     1    -1     1    -1     1     1    -1     1     1     1
     e    m0     a    m1     g     b    n0     c    n1     f     d     x     y
```

Kinase/substrate example, keep $E_{\mathrm{T}}, F_{\mathrm{T}}, M_{\mathrm{T}}, N_{\mathrm{T}}$ constant

```
how many possible signs: 88573
after using constraint     1, possible signs left: 42646
constraint used is:
    -1    -1     2     0     0     0     0     0     0     0     0
corresponding to this combination of species:
     1     0     0     0     0     0     0     0     0     0     0
after using constraint     2, possible signs left: 40459
constraint used is:
    -1    -1     1     0     0     1     0     0     0     0     0
corresponding to this combination of species:
     0     1     0     0     0     0     0     0     0     0     0
after using constraint     4, possible signs left: 34060
constraint used is:
     0     0     1    -2    -1     1    -1     2     0     0     0
corresponding to this combination of species:
     0     0     0     1     0     0     0     0     0     0     0
after using constraint     5, possible signs left: 18535
constraint used is:
     0     0     0    -1    -1     2     0     0     0     0     0
corresponding to this combination of species:
     0     0     0     0     1     0     0     0     0     0     0
after using constraint     7, possible signs left: 12334
constraint used is:
     0     0     0    -1     0     0    -1     1     0     0     1
corresponding to this combination of species:
     0     0     0     0     0     0     1     0     0     0     0
```

```
after using constraint     8, possible signs left:  9355
constraint used is:
     0     0     0     1     0     0     1    -2     0     0     0
corresponding to this combination of species:
     0     0     0     0     0     0     0     1     0     0     0
after using constraint     9, possible signs left:  6172
constraint used is:
     0     0     0     0     0     0     0     1    -1    -1     1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1     0     0
after using constraint    10, possible signs left:  4205
constraint used is:
     0     0     0     0     0     0     0     0    -1    -1     2
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     0     1     0
after using constraint    12, possible signs left:  1578
constraint used is:
     0     0     1     0     0    -1     0     0     0     0     0
corresponding to this combination of species:
     1    -1     0     0     0     0     0     0     0     0     0
after using constraint    40, possible signs left:  1557
constraint used is:
     0     0     1    -1    -1     1     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     1     0     0    -1     0     0     0     0
after using constraint    60, possible signs left:   613
constraint used is:
     0     0     0     0     0     0     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1    -1     0
after using constraint   305, possible signs left:   134
constraint used is:
     1     0     1     0     0     0     0     0     0     0     0
corresponding to this stoichiometry constraint:
     1     0     1     0     0     0     0     0     0     0     0
after using constraint   306, possible signs left:    29
constraint used is:
     0     0     0     0     0     0     0     0     0     1     1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     0     0     0     1     1
after using constraint   307, possible signs left:    10
constraint used is:
     0     1     1     1     0     1     0     1     0     0     0
corresponding to this stoichiometry constraint:
     0     1     1     1     0     1     0     1     0     0     0
after using constraint   308, possible signs left:     1
constraint used is:
     0     0     0     0     0     0     1     1     1     0     1
```

```
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     1     1     1     0     1
ended search

possible_signs =
    -1     1     1    -1     1     1     1    -1    -1     1    -1
     e    m0     a    m1     g     b    n0     c    n1     f     d
```

Additional computations for "artificial variables" x and y:

```
how many possible signs:     9
after using constraint     6, possible signs left:     3
constraint used is:
     0     0     0     0     0     0     0     0     1     0     1     0    -1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     0     0     1     0     1     0    -1
after using constraint    17, possible signs left:     1
constraint used is:
     0     1     1     0     0     0     0     0     0     0     0     1     0
corresponding to this stoichiometry constraint:
     0     1     1     0     0     0     0     0     0     0     0     1     0
ended search

possible_signs =
    -1     1     1    -1     1     1     1    -1    -1     1    -1    -1    -1
     e    m0     a    m1     g     b    n0     c    n1     f     d     x     y
```

Kinase/substrate example, keep $E_{\mathrm{T}}, G_{\mathrm{T}}, F_{\mathrm{T}}, M_{\mathrm{T}}$ constant

```
how many possible signs: 88573
after using constraint     1, possible signs left: 42646
constraint used is:
    -1    -1     2     0     0     0     0     0     0     0     0
corresponding to this combination of species:
     1     0     0     0     0     0     0     0     0     0     0
after using constraint     2, possible signs left: 40459
constraint used is:
    -1    -1     1     0     0     1     0     0     0     0     0
corresponding to this combination of species:
     0     1     0     0     0     0     0     0     0     0     0
after using constraint     4, possible signs left: 34060
constraint used is:
     0     0     1    -2    -1     1    -1     2     0     0     0
corresponding to this combination of species:
     0     0     0     1     0     0     0     0     0     0     0
after using constraint     5, possible signs left: 18535
constraint used is:
```

```
     0     0     0    -1    -1     2     0     0     0     0     0
corresponding to this combination of species:
     0     0     0     0     1     0     0     0     0     0     0
after using constraint     7, possible signs left: 12334
constraint used is:
     0     0     0    -1     0     0    -1     1     0     0     1
corresponding to this combination of species:
     0     0     0     0     0     0     1     0     0     0     0
after using constraint     8, possible signs left:  9355
constraint used is:
     0     0     0     1     0     0     1    -2     0     0     0
corresponding to this combination of species:
     0     0     0     0     0     0     0     1     0     0     0
after using constraint     9, possible signs left:  6172
constraint used is:
     0     0     0     0     0     0     0     1    -1    -1     1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1     0     0
after using constraint    10, possible signs left:  4205
constraint used is:
     0     0     0     0     0     0     0     0    -1    -1     2
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     0     1     0
after using constraint    12, possible signs left:  1578
constraint used is:
     0     0     1     0     0    -1     0     0     0     0     0
corresponding to this combination of species:
     1    -1     0     0     0     0     0     0     0     0     0
after using constraint    40, possible signs left:  1557
constraint used is:
     0     0     1    -1    -1     1     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     1     0     0    -1     0     0     0     0
after using constraint    60, possible signs left:   613
constraint used is:
     0     0     0     0     0     0     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1    -1     0
after using constraint   305, possible signs left:   134
constraint used is:
     1     0     1     0     0     0     0     0     0     0     0
corresponding to this stoichiometry constraint:
     1     0     1     0     0     0     0     0     0     0     0
after using constraint   306, possible signs left:    29
constraint used is:
     0     0     0     0     1     1     0     0     0     0     0
corresponding to this stoichiometry constraint:
     0     0     0     0     1     1     0     0     0     0     0
```

```
after using constraint   307, possible signs left:     6
constraint used is:
     0     0     0     0     0     0     0     0     0     1     1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     0     0     0     1     1
after using constraint   308, possible signs left:     1
constraint used is:
     0     1     1     1     0     1     0     1     0     0     0
corresponding to this stoichiometry constraint:
     0     1     1     1     0     1     0     1     0     0     0
ended search

possible_signs =
    -1     1     1     1    -1     1    -1    -1    -1     1    -1
     e    m0     a    m1     g     b    n0     c    n1     f     d
```

Additional computations for "artificial variables" x and y:

```
how many possible signs:     9
after using constraint     6, possible signs left:     3
constraint used is:
     0     0     0     0     0     0     0     0     1     0     1     0    -1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     0     0     1     0     1     0    -1
after using constraint    19, possible signs left:     1
constraint used is:
     0     1     1     0     0     0     0     0     0     0     0     1     0
corresponding to this stoichiometry constraint:
     0     1     1     0     0     0     0     0     0     0     0     1     0
ended search

possible_signs =
    -1     1     1     1    -1     1    -1    -1    -1     1    -1    -1    -1
     e    m0     a    m1     g     b    n0     c    n1     f     d     x     y
```

Kinase/substrate example, keep $E_{\mathrm{T}}, G_{\mathrm{T}}, M_{\mathrm{T}}, N_{\mathrm{T}}$ constant

```
how many possible signs: 88573
after using constraint     1, possible signs left: 42646
constraint used is:
    -1    -1     2     0     0     0     0     0     0     0     0
corresponding to this combination of species:
     1     0     0     0     0     0     0     0     0     0     0
after using constraint     2, possible signs left: 40459
constraint used is:
    -1    -1     1     0     0     1     0     0     0     0     0
corresponding to this combination of species:
```

```
     0     1     0     0     0     0     0     0     0     0     0
after using constraint     4, possible signs left: 34060
constraint used is:
     0     0     1    -2    -1     1    -1     2     0     0     0
corresponding to this combination of species:
     0     0     0     1     0     0     0     0     0     0     0
after using constraint     5, possible signs left: 18535
constraint used is:
     0     0     0    -1    -1     2     0     0     0     0     0
corresponding to this combination of species:
     0     0     0     0     1     0     0     0     0     0     0
after using constraint     7, possible signs left: 12334
constraint used is:
     0     0     0    -1     0     0    -1     1     0     0     1
corresponding to this combination of species:
     0     0     0     0     0     0     1     0     0     0     0
after using constraint     8, possible signs left:  9355
constraint used is:
     0     0     0     1     0     0     1    -2     0     0     0
corresponding to this combination of species:
     0     0     0     0     0     0     0     1     0     0     0
after using constraint     9, possible signs left:  6172
constraint used is:
     0     0     0     0     0     0     0     1    -1    -1     1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1     0     0
after using constraint    10, possible signs left:  4205
constraint used is:
     0     0     0     0     0     0     0     0    -1    -1     2
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     0     1     0
after using constraint    12, possible signs left:  1578
constraint used is:
     0     0     1     0     0    -1     0     0     0     0     0
corresponding to this combination of species:
     1    -1     0     0     0     0     0     0     0     0     0
after using constraint    40, possible signs left:  1557
constraint used is:
     0     0     1    -1    -1     1     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     1     0     0    -1     0     0     0     0
after using constraint    60, possible signs left:   613
constraint used is:
     0     0     0     0     0     0     0     1     0     0    -1
corresponding to this combination of species:
     0     0     0     0     0     0     0     0     1    -1     0
after using constraint   305, possible signs left:   134
constraint used is:
```

```
     1     0     1     0     0     0     0     0     0     0     0
corresponding to this stoichiometry constraint:
     1     0     1     0     0     0     0     0     0     0     0
after using constraint   306, possible signs left:    29
constraint used is:
     0     0     0     0     1     1     0     0     0     0     0
corresponding to this stoichiometry constraint:
     0     0     0     0     1     1     0     0     0     0     0
after using constraint   307, possible signs left:     6
constraint used is:
     0     1     1     1     0     1     0     1     0     0     0
corresponding to this stoichiometry constraint:
     0     1     1     1     0     1     0     1     0     0     0
after using constraint   308, possible signs left:     1
constraint used is:
     0     0     0     0     0     0     1     1     1     0     1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     1     1     1     0     1
ended search

possible_signs =
    -1     1     1     1    -1     1    -1    -1     1    -1    -1
     e    m0     a    m1     g     b    n0     c    n1     f     d
```

Additional computations for "artificial variables" x and y:

```
how many possible signs:     9
after using constraint    17, possible signs left:     3
constraint used is:
     0     1     1     0     0     0     0     0     0     0     0     1     0
corresponding to this stoichiometry constraint:
     0     1     1     0     0     0     0     0     0     0     0     1     0
after using constraint    20, possible signs left:     1
constraint used is:
     0     0     0     0     0     0     1     1     0     0     0     0     1
corresponding to this stoichiometry constraint:
     0     0     0     0     0     0     1     1     0     0     0     0     1
ended search

possible_signs =
    -1     1     1     1    -1     1    -1    -1     1    -1    -1    -1     1
     e    m0     a    m1     g     b    n0     c    n1     f     d     x     y
```

S2. Phosphotransfer example

Since this is a smaller example, the complete output is shown.

Phosphotransfer example, keep X_{T} and P_{T} constant

```
>> find_signs
G =
    -1     0     1     0
     1    -1    -1     1
     0     1     0    -1
     1     1    -1    -1
    -1    -1     1     1
generating first single rows of n*G*A^T that pass test
next, generating combinations 1-1, 1+1, 1+2, 1-2, 2-1, 2+1 of rows
testing the new vectors and adding to list the ones that pass
total number of constraints so far =   13, listed below
constraints =
    -1     1     0     1    -1
     0     1    -1    -1     1
     0     1    -1    -1     1
     0    -2     2     2    -2
     2    -2     0    -2     2
    -1     1     0     1    -1
     0    -1     1     1    -1
     0     1    -1    -1     1
     1    -1     0    -1     1
     2    -2     0    -2     2
     0    -2     2     2    -2
    -1     1     0     1    -1
     0     0     0     0     0
making initial stoichiometric constraints
total number of initial stoichiometry constraints =    2, listed below
stoichiometry_constraints =
     1     1     1     0     0
     0     1     2     0     1
next, generating combinations 1-1, 1+1, 1+2, 1-2, 2-1, 2+1 of stoichiometry
total number of added stoichiometry constraints =    6, listed below
additional_stoichiometry =
     1     0    -1     0    -1
     1     2     3     0     1
     1     3     5     0     2
     1    -1    -3     0    -2
     2     1     0     0    -1
     2     3     4     0     1
adding virtual constraint(s), if any
total number of virtual constraints =    1, listed below
virtual =
```

```
    -1     1    -1     0     0
total number of constraints so far =   22, listed below
constraints =
    -1     1     0     1    -1
     0     1    -1    -1     1
     0     1    -1    -1     1
     0    -2     2     2    -2
     2    -2     0    -2     2
    -1     1     0     1    -1
     0    -1     1     1    -1
     0     1    -1    -1     1
     1    -1     0    -1     1
     2    -2     0    -2     2
     0    -2     2     2    -2
    -1     1     0     1    -1
     0     0     0     0     0
     1     1     1     0     0
     0     1     2     0     1
     1     0    -1     0    -1
     1     2     3     0     1
     1     3     5     0     2
     1    -1    -3     0    -2
     2     1     0     0    -1
     2     3     4     0     1
    -1     1    -1     0     0
how many possible signs:   121
after using constraint     1, possible signs left:    76
constraint nu*gamma*A^T used is:
    -1     1     0     1    -1
corresponding to this combination of species:
     1     0     0     0     0
after using constraint     2, possible signs left:    49
constraint nu*gamma*A^T used is:
     0     1    -1    -1     1
corresponding to this combination of species:
     0     0     1     0     0
after using constraint    14, possible signs left:    22
constraint nu*gamma*A^T used is:
     1     1     1     0     0
corresponding to this stoichiometry constraint:
     1     1     1     0     0
after using constraint    15, possible signs left:    12
constraint nu*gamma*A^T used is:
     0     1     2     0     1
corresponding to this stoichiometry constraint:
     0     1     2     0     1
after using constraint    16, possible signs left:     7
constraint nu*gamma*A^T used is:
```

```
     1     0    -1     0    -1
corresponding to this stoichiometry constraint:
     1     0    -1     0    -1
after using constraint    22, possible signs left:     3
constraint nu*gamma*A^T used is:
    -1     1    -1     0     0
corresponding to this virtual constraint:
    -1     1    -1     0     0
ended search

possible_signs =
    -1    -1     1    -1    -1
    -1     0     1    -1    -1
    -1     1     1    -1    -1
    x0    x1    x2    y0    y1
```

Phosphotransfer example, keep X_{T} and Y_{T} constant

```
>> find_signs
G =
    -1     0     1     0
     1    -1    -1     1
     0     1     0    -1
     1     1    -1    -1
    -1    -1     1     1
generating first single rows of n*G*A^T that pass test
next, generating combinations 1-1, 1+1, 1+2, 1-2, 2-1, 2+1 of rows
testing the new vectors and adding to list the ones that pass
total number of constraints so far =   13, listed below
constraints =
    -1     1     0     1    -1
     0     1    -1    -1     1
     0     1    -1    -1     1
     0    -2     2     2    -2
     2    -2     0    -2     2
    -1     1     0     1    -1
     0    -1     1     1    -1
     0     1    -1    -1     1
     1    -1     0    -1     1
     2    -2     0    -2     2
     0    -2     2     2    -2
    -1     1     0     1    -1
     0     0     0     0     0
stoichiometry_constraints =
     1     1     1     0     0
     0     0     0     1     1
total number of initial stoichiometry constraints =    2, listed below
```

```
stoichiometry_constraints =
     1     1     1     0     0
     0     0     0     1     1
next, generating combinations 1-1, 1+1, 1+2, 1-2, 2-1, 2+1 of stoichiometry
total number of added stoichiometry constraints =    6, listed below
additional_stoichiometry =
     1     1     1    -1    -1
     1     1     1     1     1
     1     1     1     2     2
     1     1     1    -2    -2
     2     2     2    -1    -1
     2     2     2     1     1
adding virtual constraint(s), if any
total number of virtual constraints =    1, listed below
virtual =
    -1     1    -1     0     0
total number of constraints so far =   22, listed below
constraints =
    -1     1     0     1    -1
     0     1    -1    -1     1
     0     1    -1    -1     1
     0    -2     2     2    -2
     2    -2     0    -2     2
    -1     1     0     1    -1
     0    -1     1     1    -1
     0     1    -1    -1     1
     1    -1     0    -1     1
     2    -2     0    -2     2
     0    -2     2     2    -2
    -1     1     0     1    -1
     0     0     0     0     0
     1     1     1     0     0
     0     0     0     1     1
     1     1     1    -1    -1
     1     1     1     1     1
     1     1     1     2     2
     1     1     1    -2    -2
     2     2     2    -1    -1
     2     2     2     1     1
    -1     1    -1     0     0
how many possible signs:   121
after using constraint     1, possible signs left:    76
constraint used is:
    -1     1     0     1    -1
corresponding to this combination of species:
     1     0     0     0     0
after using constraint     2, possible signs left:    49
constraint used is:
```

```
     0     1    -1    -1     1
corresponding to this combination of species:
     0     0     1     0     0
after using constraint    14, possible signs left:    22
constraint used is:
     1     1     1     0     0
corresponding to this stoichiometry constraint:
     1     1     1     0     0
after using constraint    15, possible signs left:     3
constraint used is:
     0     0     0     1     1
corresponding to this stoichiometry constraint:
     0     0     0     1     1
ended search

possible_signs =
    -1    -1     1    -1     1
    -1     0     1    -1     1
    -1     1     1    -1     1
    x0    x1    x2    y0    y1
```

Phosphotransfer example, keep P_{T} and Y_{T} constant

```
>> find_signs
G =
    -1     0     1     0
     1    -1    -1     1
     0     1     0    -1
     1     1    -1    -1
    -1    -1     1     1
generating first single rows of n*G*A^T that pass test
next, generating combinations 1-1, 1+1, 1+2, 1-2, 2-1, 2+1 of rows
testing the new vectors and adding to list the ones that pass
total number of constraints so far =   13, listed below
constraints =
    -1     1     0     1    -1
     0     1    -1    -1     1
     0     1    -1    -1     1
     0    -2     2     2    -2
     2    -2     0    -2     2
    -1     1     0     1    -1
     0    -1     1     1    -1
     0     1    -1    -1     1
     1    -1     0    -1     1
     2    -2     0    -2     2
     0    -2     2     2    -2
    -1     1     0     1    -1
```

```
     0     0     0     0     0
stoichiometry_constraints =
     0     0     0     1     1
     0     1     2     0     1
total number of initial stoichiometry constraints =    2, listed below
stoichiometry_constraints =
     0     0     0     1     1
     0     1     2     0     1
next, generating combinations 1-1, 1+1, 1+2, 1-2, 2-1, 2+1 of stoichiometry
total number of added stoichiometry constraints =    6, listed below
additional_stoichiometry =
     0    -1    -2     1     0
     0     1     2     1     2
     0     2     4     1     3
     0    -2    -4     1    -1
     0    -1    -2     2     1
     0     1     2     2     3
adding virtual constraint(s), if any
total number of virtual constraints =    1, listed below
virtual =
    -1     1    -1     0     0
total number of constraints so far =   22, listed below
constraints =
    -1     1     0     1    -1
     0     1    -1    -1     1
     0     1    -1    -1     1
     0    -2     2     2    -2
     2    -2     0    -2     2
    -1     1     0     1    -1
     0    -1     1     1    -1
     0     1    -1    -1     1
     1    -1     0    -1     1
     2    -2     0    -2     2
     0    -2     2     2    -2
    -1     1     0     1    -1
     0     0     0     0     0
     0     0     0     1     1
     0     1     2     0     1
     0    -1    -2     1     0
     0     1     2     1     2
     0     2     4     1     3
     0    -2    -4     1    -1
     0    -1    -2     2     1
     0     1     2     2     3
    -1     1    -1     0     0
how many possible signs:   121
after using constraint     1, possible signs left:    76
constraint used is:
```

```
    -1     1     0     1    -1
corresponding to this combination of species:
     1     0     0     0     0
after using constraint     2, possible signs left:    49
constraint used is:
     0     1    -1    -1     1
corresponding to this combination of species:
     0     0     1     0     0
after using constraint    14, possible signs left:     8
constraint used is:
     0     0     0     1     1
corresponding to this stoichiometry constraint:
     0     0     0     1     1
after using constraint    15, possible signs left:     3
constraint used is:
     0     1     2     0     1
corresponding to this stoichiometry constraint:
     0     1     2     0     1
ended search

possible_signs =
    -1    -1    -1    -1     1
    -1    -1     0    -1     1
    -1    -1     1    -1     1
    x0    x1    x2    y0    y1
```

S3. Ligand/receptor/antagonist/trap example

```
>> find_signs

G =

    -1     0     0    -1     1     0     0     1
    -1    -1     0     0     1     1     0     0
     0    -1    -1     0     0     1     1     0
     0     0    -1    -1     0     0     1     1
     1     0     0     0    -1     0     0     0
     0     1     0     0     0    -1     0     0
     0     0     1     0     0     0    -1     0
     0     0     0     1     0     0     0    -1

generating first single rows of n*G*A^T that pass test
next, generating combinations 1-1, 1+1, 1+2, 1-2, 2-1, 2+1 of rows
testing the new vectors and adding to list the ones that pass
total number of constraints so far =  118, listed below

constraints =

    -2    -1     0    -1     1     0     0     1
    -1    -2    -1     0     1     1     0     0
     0    -1    -2    -1     0     1     1     0
    -1     0    -1    -2     0     0     1     1
     1     1     0     0    -1     0     0     0
     0     1     1     0     0    -1     0     0
     0     0     1     1     0     0    -1     0
     1     0     0     1     0     0     0    -1
    -1     1     1    -1     0    -1     0     1
    -1    -1     1     1     1     0    -1     0
    -3    -2     0    -1     2     0     0     1
    -2    -2    -1    -1     1     1     0     1
    -2    -1    -1    -2     1     0     1     1
    -3    -1     0    -2     1     0     0     2
    -1    -1     1     1     1     0    -1     0
    -2    -3    -1     0     2     1     0     0
    -1    -3    -2     0     1     2     0     0
    -1    -2    -2    -1     1     1     1     0
    -2    -2    -1    -1     1     1     0     1
     1    -1    -1     1     0     1     0    -1
    -1    -2    -2    -1     1     1     1     0
     0    -2    -3    -1     0     2     1     0
     0    -1    -3    -2     0     1     2     0
    -1    -1    -2    -2     0     1     1     1
    -2    -1    -1    -2     1     0     1     1
```

-1	-1	-2	-2	0	1	1	1
-1	0	-2	-3	0	0	2	1
-2	0	-1	-3	0	0	1	2
1	1	-1	-1	-1	0	1	0
-1	1	1	-1	0	-1	0	1
-3	-3	-1	-1	2	1	0	1
-2	-2	-2	-2	1	1	1	1
-3	-1	-1	-3	1	0	1	2
-1	0	0	-1	0	0	0	1
-1	-1	0	0	1	0	0	0
-1	-3	-3	-1	1	2	1	0
-2	-2	-2	-2	1	1	1	1
0	-1	-1	0	0	1	0	0
-1	-1	0	0	1	0	0	0
-1	-1	-3	-3	0	1	2	1
0	0	-1	-1	0	0	1	0
0	-1	-1	0	0	1	0	0
-1	0	0	-1	0	0	0	1
0	0	-1	-1	0	0	1	0
1	2	1	0	-1	-1	0	0
1	1	1	1	-1	0	-1	0
2	1	0	1	-1	0	0	-1
0	1	2	1	0	-1	-1	0
1	1	1	1	0	-1	0	-1
1	0	1	2	0	0	-1	-1
-4	-5	-2	-1	3	2	0	1
-2	-3	-4	-3	1	2	2	1
-4	-1	-2	-5	1	0	2	3
-1	-4	-5	-2	1	3	2	0
-3	-2	-3	-4	1	1	2	2
-2	-1	-4	-5	0	1	3	2
1	3	2	0	-1	-2	0	0
1	1	2	2	-1	0	-2	0
3	1	0	2	-1	0	0	-2
0	1	3	2	0	-1	-2	0
2	1	1	2	0	-1	0	-2
2	0	1	3	0	0	-1	-2
-4	-3	0	-1	3	0	0	1
-2	-3	-2	-1	1	2	0	1
-2	-1	-2	-3	1	0	2	1
-4	-1	0	-3	1	0	0	3
-3	-4	-1	0	3	1	0	0
-1	-4	-3	0	1	3	0	0
-1	-2	-3	-2	1	1	2	0
-3	-2	-1	-2	1	1	0	2
-2	-3	-2	-1	2	1	1	0
0	-3	-4	-1	0	3	1	0
0	-1	-4	-3	0	1	3	0

```
-2    -1    -2    -3     0     1     1     2
-3    -2    -1    -2     2     0     1     1
-1    -2    -3    -2     0     2     1     1
-1     0    -3    -4     0     0     3     1
-3     0    -1    -4     0     0     1     3
 1     1    -2    -2    -1     0     2     0
-2     1     1    -2     0    -1     0     2
-5    -3     0    -2     3     0     0     2
-4    -3    -1    -2     2     1     0     2
-4    -2    -1    -3     2     0     1     2
-5    -2     0    -3     2     0     0     3
-3    -5    -2     0     3     2     0     0
-2    -5    -3     0     2     3     0     0
-2    -4    -3    -1     2     2     1     0
-3    -4    -2    -1     2     2     0     1
-1    -3    -4    -2     1     2     2     0
 0    -3    -5    -2     0     3     2     0
 0    -2    -5    -3     0     2     3     0
-1    -2    -4    -3     0     2     2     1
-3    -1    -2    -4     1     0     2     2
-2    -1    -3    -4     0     1     2     2
-2     0    -3    -5     0     0     3     2
-3     0    -2    -5     0     0     2     3
 2     2    -1    -1    -2     0     1     0
-1     2     2    -1     0    -2     0     1
-5    -4    -1    -2     3     1     0     2
-4    -3    -2    -3     2     1     1     2
-5    -2    -1    -4     2     0     1     3
-3    -1     0    -2     1     0     0     2
-3    -2     0    -1     2     0     0     1
-2    -5    -4    -1     2     3     1     0
-3    -4    -3    -2     2     2     1     1
-1    -3    -2     0     1     2     0     0
-2    -3    -1     0     2     1     0     0
-1    -2    -5    -4     0     2     3     1
 0    -1    -3    -2     0     1     2     0
 0    -2    -3    -1     0     2     1     0
-2     0    -1    -3     0     0     1     2
-1     0    -2    -3     0     0     2     1
 2     3     1     0    -2    -1     0     0
 2     2     1     1    -2     0    -1     0
 3     2     0     1    -2     0     0    -1
 0     2     3     1     0    -2    -1     0
 1     2     2     1     0    -2     0    -1
 1     0     2     3     0     0    -2    -1

total number of initial stoichiometry constraints =    3, listed below
```

```
stoichiometry_constraints =

     0     1     0     0     1     1     0     0
     0     0     1     0     0     1     1     0
     0     0     0     1     0     0     1     1


how_many_stoichiometry =

     3

next, generating combinations 1-1, 1+1, 1+2, 1-2, 2-1, 2+1 of stoichiometry
total number of added stoichiometry constraints =   18, listed below
adding virtual constraint(s), if any
total number of virtual constraints =    1, listed below
total number of constraints so far =  140, listed below

constraints =

    -2    -1     0    -1     1     0     0     1
    -1    -2    -1     0     1     1     0     0
     0    -1    -2    -1     0     1     1     0
    -1     0    -1    -2     0     0     1     1
     1     1     0     0    -1     0     0     0
     0     1     1     0     0    -1     0     0
     0     0     1     1     0     0    -1     0
     1     0     0     1     0     0     0    -1
    -1     1     1    -1     0    -1     0     1
    -1    -1     1     1     1     0    -1     0
    -3    -2     0    -1     2     0     0     1
    -2    -2    -1    -1     1     1     0     1
    -2    -1    -1    -2     1     0     1     1
    -3    -1     0    -2     1     0     0     2
    -1    -1     1     1     1     0    -1     0
    -2    -3    -1     0     2     1     0     0
    -1    -3    -2     0     1     2     0     0
    -1    -2    -2    -1     1     1     1     0
    -2    -2    -1    -1     1     1     0     1
     1    -1    -1     1     0     1     0    -1
    -1    -2    -2    -1     1     1     1     0
     0    -2    -3    -1     0     2     1     0
     0    -1    -3    -2     0     1     2     0
    -1    -1    -2    -2     0     1     1     1
    -2    -1    -1    -2     1     0     1     1
    -1    -1    -2    -2     0     1     1     1
    -1     0    -2    -3     0     0     2     1
    -2     0    -1    -3     0     0     1     2
     1     1    -1    -1    -1     0     1     0
```

```
-1   1   1  -1   0  -1   0   1
-3  -3  -1  -1   2   1   0   1
-2  -2  -2  -2   1   1   1   1
-3  -1  -1  -3   1   0   1   2
-1   0   0  -1   0   0   0   1
-1  -1   0   0   1   0   0   0
-1  -3  -3  -1   1   2   1   0
-2  -2  -2  -2   1   1   1   1
 0  -1  -1   0   0   1   0   0
-1  -1   0   0   1   0   0   0
-1  -1  -3  -3   0   1   2   1
 0   0  -1  -1   0   0   1   0
 0  -1  -1   0   0   1   0   0
-1   0   0  -1   0   0   0   1
 0   0  -1  -1   0   0   1   0
 1   2   1   0  -1  -1   0   0
 1   1   1   1  -1   0  -1   0
 2   1   0   1  -1   0   0  -1
 0   1   2   1   0  -1  -1   0
 1   1   1   1   0  -1   0  -1
 1   0   1   2   0   0  -1  -1
-4  -5  -2  -1   3   2   0   1
-2  -3  -4  -3   1   2   2   1
-4  -1  -2  -5   1   0   2   3
-1  -4  -5  -2   1   3   2   0
-3  -2  -3  -4   1   1   2   2
-2  -1  -4  -5   0   1   3   2
 1   3   2   0  -1  -2   0   0
 1   1   2   2  -1   0  -2   0
 3   1   0   2  -1   0   0  -2
 0   1   3   2   0  -1  -2   0
 2   1   1   2   0  -1   0  -2
 2   0   1   3   0   0  -1  -2
-4  -3   0  -1   3   0   0   1
-2  -3  -2  -1   1   2   0   1
-2  -1  -2  -3   1   0   2   1
-4  -1   0  -3   1   0   0   3
-3  -4  -1   0   3   1   0   0
-1  -4  -3   0   1   3   0   0
-1  -2  -3  -2   1   1   2   0
-3  -2  -1  -2   1   1   0   2
-2  -3  -2  -1   2   1   1   0
 0  -3  -4  -1   0   3   1   0
 0  -1  -4  -3   0   1   3   0
-2  -1  -2  -3   0   1   1   2
-3  -2  -1  -2   2   0   1   1
-1  -2  -3  -2   0   2   1   1
-1   0  -3  -4   0   0   3   1
```

-3	0	-1	-4	0	0	1	3
1	1	-2	-2	-1	0	2	0
-2	1	1	-2	0	-1	0	2
-5	-3	0	-2	3	0	0	2
-4	-3	-1	-2	2	1	0	2
-4	-2	-1	-3	2	0	1	2
-5	-2	0	-3	2	0	0	3
-3	-5	-2	0	3	2	0	0
-2	-5	-3	0	2	3	0	0
-2	-4	-3	-1	2	2	1	0
-3	-4	-2	-1	2	2	0	1
-1	-3	-4	-2	1	2	2	0
0	-3	-5	-2	0	3	2	0
0	-2	-5	-3	0	2	3	0
-1	-2	-4	-3	0	2	2	1
-3	-1	-2	-4	1	0	2	2
-2	-1	-3	-4	0	1	2	2
-2	0	-3	-5	0	0	3	2
-3	0	-2	-5	0	0	2	3
2	2	-1	-1	-2	0	1	0
-1	2	2	-1	0	-2	0	1
-5	-4	-1	-2	3	1	0	2
-4	-3	-2	-3	2	1	1	2
-5	-2	-1	-4	2	0	1	3
-3	-1	0	-2	1	0	0	2
-3	-2	0	-1	2	0	0	1
-2	-5	-4	-1	2	3	1	0
-3	-4	-3	-2	2	2	1	1
-1	-3	-2	0	1	2	0	0
-2	-3	-1	0	2	1	0	0
-1	-2	-5	-4	0	2	3	1
0	-1	-3	-2	0	1	2	0
0	-2	-3	-1	0	2	1	0
-2	0	-1	-3	0	0	1	2
-1	0	-2	-3	0	0	2	1
2	3	1	0	-2	-1	0	0
2	2	1	1	-2	0	-1	0
3	2	0	1	-2	0	0	-1
0	2	3	1	0	-2	-1	0
1	2	2	1	0	-2	0	-1
1	0	2	3	0	0	-2	-1
0	1	0	0	1	1	0	0
0	0	1	0	0	1	1	0
0	0	0	1	0	0	1	1
0	1	-1	0	1	0	-1	0
0	1	0	-1	1	1	-1	-1
0	0	1	-1	0	1	0	-1
0	1	1	0	1	2	1	0

```
     0     1     0     1     1     1     1     1
     0     0     1     1     0     1     2     1
     0     1     2     0     1     3     2     0
     0     1     0     2     1     1     2     2
     0     0     1     2     0     1     3     2
     0     1    -2     0     1    -1    -2     0
     0     1     0    -2     1     1    -2    -2
     0     0     1    -2     0     1    -1    -2
     0     2    -1     0     2     1    -1     0
     0     2     0    -1     2     2    -1    -1
     0     0     2    -1     0     2     1    -1
     0     2     1     0     2     3     1     0
     0     2     0     1     2     2     1     1
     0     0     2     1     0     2     3     1
     0     0     0     0     1    -1     1    -1

how many possible signs:  3280
after using constraint     1, possible signs left:  2443
constraint used is:
    -2    -1     0    -1     1     0     0     1

corresponding to this combination of species:
     1     0     0     0     0     0     0     0

after using constraint     2, possible signs left:  1996
constraint used is:
    -1    -2    -1     0     1     1     0     0

corresponding to this combination of species:
     0     1     0     0     0     0     0     0

after using constraint     3, possible signs left:  1591
constraint used is:
     0    -1    -2    -1     0     1     1     0

corresponding to this combination of species:
     0     0     1     0     0     0     0     0

after using constraint     4, possible signs left:  1313
constraint used is:
    -1     0    -1    -2     0     0     1     1

corresponding to this combination of species:
     0     0     0     1     0     0     0     0

after using constraint     5, possible signs left:   914
constraint used is:
     1     1     0     0    -1     0     0     0
```

```
corresponding to this combination of species:
     0     0     0     0     1     0     0     0

after using constraint     6, possible signs left:   578
constraint used is:
     0     1     1     0     0    -1     0     0

corresponding to this combination of species:
     0     0     0     0     0     1     0     0

after using constraint     7, possible signs left:   378
constraint used is:
     0     0     1     1     0     0    -1     0

corresponding to this combination of species:
     0     0     0     0     0     0     1     0

after using constraint     8, possible signs left:   224
constraint used is:
     1     0     0     1     0     0     0    -1

corresponding to this combination of species:
     0     0     0     0     0     0     0     1

after using constraint   119, possible signs left:    87
constraint used is:
     0     1     0     0     1     1     0     0

corresponding to this stoichiometry constraint:
     0     1     0     0     1     1     0     0

after using constraint   120, possible signs left:    28
constraint used is:
     0     0     1     0     0     1     1     0

corresponding to this stoichiometry constraint:
     0     0     1     0     0     1     1     0

after using constraint   121, possible signs left:     9
constraint used is:
     0     0     0     1     0     0     1     1

corresponding to this stoichiometry constraint:
     0     0     0     1     0     0     1     1

after using constraint   140, possible signs left:     5
constraint used is:
```

```
     0     0     0     0     1    -1     1    -1

corresponding to this virtual constraint:
     0     0     0     0     1    -1     1    -1

ended search


possible_signs =

    -1     1    -1     1    -1    -1     1    -1
    -1     1    -1     1    -1     0     1    -1
    -1     1    -1     1    -1     1    -1    -1
    -1     1    -1     1    -1     1     0    -1
    -1     1    -1     1    -1     1     1    -1

    x1    x2    x3    x4    y1    y2    y3    y4
```