Network reconstruction based on quasi-steady state data

Eduardo D. Sontag
Department of Mathematics, Rutgers University

February 2, 2008

Abstract

This note discusses a theoretical issue regarding the application of the “Modular Response Analysis” method to quasi-steady state (rather than steady-state) data.

1 Introduction

The reverse engineering problem in systems biology is, loosely speaking, that of unraveling the web of interactions among the components of protein and gene regulatory networks. A major goal is to map out the direct functional interactions among components, a problem that is difficult to approach by means of standard statistical and machine learning approaches such as clustering into co-expression patterns. Information on direct functional interactions throws light upon the possible mechanisms and architecture underlying the observed behavior of complex molecular networks.

An intrinsic difficulty in capturing direct interactions in intact cells by traditional genetic experiments, RNA interference, hormones, growth factors, or pharmacological interventions, is that any perturbation to a particular gene or signaling component may rapidly propagate throughout the network, thus causing global changes which cannot be easily distinguished from direct (local) effects. Thus, one central goal in reverse engineering is to use the observed global responses (such as steady-state changes in concentrations of active proteins, mRNA levels, or transcription rates) in order to infer the local interactions between individual nodes. One potentially very powerful approach to solve the global to local problem is the Modular Response Analysis (MRA) or “unraveling” method proposed in [5] and further elaborated upon in [9, 1, 2, 3] (see [10, 4] for reviews).

The MRA experimental design compares the steady states that result after performing independent perturbations to each “modular component” of a network. These perturbations might be genetic or biochemical, or (in eukaryotes) they might be achieved through the down-regulation of protein levels by means of RNAi. This latter experimental approach to MRA was the one taken in [7], which quantified positive and negative feedback effects in the Raf/Mek/Erk MAPK network in rat adrenal pheochromocytoma (PC-12) cells. Using the formulas given in [9] and [1], the authors of [7] uncovered connectivity differences depending on whether the cells are stimulated with epidermal growth factor (EGF) or instead with neuronal growth factor (NGF). There are a couple of subtle theoretical gaps, however, when applying MRA algorithms to data like that employed in [7]. The main gap is due to the fact that the data fed into the MRA algorithm included non-steady state measurements. Specifically, for EGF stimulation, network responses were measured at the peak of Erk activity (at 5 minutes) and not at steady state. This note fills that gap, providing a theoretical justification for the use of quasi-steady state information.

1.1 Mathematical formulation

We assume that there are $n$ quantities $x_i(t)$ that can be in principle measured, such as the levels of activity of selected proteins, or the transcription rates of certain genes. These quantities are thought of as state
variables in a dynamical system and are collected into a time-dependent vector \( x(t) = (x_1(t), \ldots, x_n(t)) \). The dynamical system is described by a system of differential equations:

\[
\begin{align*}
\dot{x}_1 &= f_1(x_1, \ldots, x_n, p_1, \ldots, p_m) \\
\dot{x}_2 &= f_2(x_1, \ldots, x_n, p_1, \ldots, p_m) \\
& \quad \vdots \\
\dot{x}_n &= f_n(x_1, \ldots, x_n, p_1, \ldots, p_m)
\end{align*}
\]

or, in more convenient vector form,

\[
\dot{x} = f(x, p)
\]

(dot indicates time derivative, and arguments \( t \) are omitted when clear). The \( p_i \)'s are parameters, collected into a vector \( p = (p_1, \ldots, p_m) \). These parameters can be manipulated, but, once changed, they remain constant for the duration of the experiment. An example would be that in which the variables \( x_i \) correspond to the levels of protein products corresponding to equations, such as phosphorylation modifications of these proteins in a signaling pathway.

The ultimate goal is to obtain, for each pair of variables \( x_i \) and \( x_j \), the relative signs and magnitudes of the partial derivatives

\[
\frac{\partial f_i}{\partial x_j},
\]

which quantify the direct effects of each variable \( x_j \) on each variable \( x_i \).

The critical assumption for MRA, and indeed the main point of [5, 6, 9], is that, while one may not know the detailed form of the vector field \( f \), often one does know which parameters \( p_j \) directly affect which variables \( x_i \). For example, \( x_i \) may be the level of activity of a particular protein, and \( p_i \) might be the total amount (active plus inactive) of that particular protein; in that case, we know that \( p_i \) only directly affects \( x_i \).

In the standard version of MRA, one first measures a steady state \( \bar{x} \) corresponding to a “wild type” vector of parameters \( \bar{p} \), that is \( f(\bar{x}, \bar{p}) = 0 \). Subsequent perturbations are separately performed to each entry of \( \bar{p} \), and a new steady state is measured, one for each such perturbation. Using these data (and assuming a that certain independence condition which we review later is satisfied), it is possible to calculate, at least in the ideal noise-free case, the Jacobian of \( f \), evaluated at \( (\bar{x}, \bar{p}) \), up to a scalar multiplicative factor uncertainty on each row. (Such uncertainty is unavoidable when using only steady state measurements, since multiplying a row of the vector field \( f \) by a nonzero constant does not affect the location of steady states.) A variation of MRA is possible, which allows for the use of non-steady state, time-series data. However, this alternative method, developed in [9], requires one to compute time derivatives, and hence is hard to apply when time measurements are spaced far apart and/or are noisy. An intermediate possibility is to use quasi-steady state data, meaning, just as in the experimental setup of [7], that one employs data collected at times when a variable has been observed to attain a local maximum or local minimum. That is the case addressed in this note.

More precisely, we will consider the following scenario. For any fixed variable, let us say the \( i \)th component \( x_i \) of \( x \), we consider some time instant \( \bar{t}_i \) at which \( \dot{x}_i(\bar{t}_i) \) is zero. Under the same independence hypothesis as in the classical MRA case, plus the nondegeneracy assumption that the second time derivative \( \ddot{x}_i(\bar{t}_i) \) is not zero (so that we have a true local minimum or local maximum, but not an inflection point), we show here that the MRA approach applies in exactly the same manner as in the steady-state case. Specifically, the \( i \)th row of the Jacobian of \( f \), evaluated at the vector \( (\bar{x}, \bar{p}) \), is recovered up to a constant multiple, where \( \bar{x} = x(\bar{t}_i) \) is the full state \( x \) at time \( \bar{t}_i \). The main difference with the steady-state case is that different rows of \( f \) are estimated at different pairs \( (\bar{x}, \bar{p}) \), since the considered times \( \bar{t}_i \) at which each individual \( \dot{x}_i(\bar{t}_i) \) vanishes are in general different for different indices \( i \), and so the state \( \bar{x} \) is different for different \( i \)'s.

2
2 Using quasi-steady state data

We fix an index \( i \in \{1, \ldots, n\} \), and an initial condition \( x(0) \), and assume that the solution \( x(t) \) with this initial condition and a given parameter vector \( \bar{p} \) has the property that, for some time \( t = \bar{t} \), we have that both \( \dot{x}_i(\bar{t}) = 0 \) and \( \ddot{x}_i(\bar{t}) \neq 0 \). At the instant \( t = \bar{t} \), \( x_i \) achieves a local minimum or a local maximum as a function of \( t \). We describe the reconstruction of the \( i \)th row of the Jacobian of \( f \), which is the same as the gradient \( \nabla f_i \), where \( f_i \) is the \( i \)th coordinate of \( f \), evaluated at \( x = \bar{x} \) and \( p = \bar{p} \), where \( \bar{x} = x(\bar{t}) \).

To emphasize the dependence of the solution on the parameters (the initial condition \( x(0) \) will remain fixed), we will denote the solution of the differential equation \( \dot{x} = f(x,p) \) by \( x(t,p) \). The function \( x(t,p) \) is jointly continuously differentiable in \( x \) and \( p \), if the vector field \( f \) is continuously differentiable (see e.g. [5], Appendix C). Note that, with this notation, the left-hand side of the differential equation can also be written as \( \partial x/\partial t \), and that \( x(t,\bar{p}) = \bar{x} \).

We introduce the following function:

\[
\alpha(t,p) = \frac{\partial x_i}{\partial t}(t,p) = f_i(x(t,p),p).
\]

Note that \( \alpha(\bar{t},\bar{p}) = 0 \). Also,

\[
\frac{\partial \alpha}{\partial t}(t,p) = \frac{\partial^2 x_i}{\partial t^2}(t,p) = \nabla f_i(x(t,p),p)f(x(t,p),p).
\]

The assumption that \( \dot{x}_i(\bar{t}) \neq 0 \) when \( p = \bar{p} \) means that \( \frac{\partial \alpha}{\partial t}(\bar{t},\bar{p}) \neq 0 \). Therefore, we may apply the implicit function theorem and conclude the existence of a mapping \( \tau \), defined on a neighborhood of \( \bar{p} \), with the property that

\[
\alpha(\tau(p),p) = 0 \quad \text{for all } p \approx \bar{p}
\]

and \( \tau(\bar{p}) = \bar{t} \) (and, in fact, \( t = \tau(p) \) is the unique value of \( t \) near \( \bar{t} \) such that \( (\partial x_i/\partial t)(t,p) = \alpha(t,p) = 0 \)).

We define, also in a neighborhood of \( \bar{p} \), the differentiable function

\[
\varphi(p) = x(\tau(p),p)
\]

and note that \( \varphi(\bar{p}) = \bar{x} \). Observe that, from the definition of \( \alpha \), we have:

\[
f_i(\varphi(p),p) = 0 \quad \text{for all } p \approx \bar{p}. \tag{1}
\]

We next discuss how to reconstruct \( \nabla f_i(\bar{x},\bar{p}) \), up to a constant multiple, under the assumption (as in [5]) that it is possible to apply \( n-1 \) independent parameter perturbations to all species different from the \( i \)th one. This discussion is basically identical to that for the steady state case, given in [5][1][2].

Mathematically, we assume that there are \( n-1 \) indices \( j_1, j_2, \ldots, j_{n-1} \) with the properties that (a) \( f_i \) does not depend directly on any \( p_j \): \( \partial f_i/\partial p_j \equiv 0 \), for \( j \in \{ j_1, j_2, \ldots, j_{n-1} \} \), and (b) the vectors \( v_j = (\partial \varphi/\partial p_j)(\bar{p}) \), for these \( j \)’s, are linearly independent. Assumption (a) is structural, and is key to the method and nontrivial, but assumption (b) is a weak genericity assumption.

We then have, taking total derivatives in [1]:

\[
\nabla f_i(\bar{x},\bar{p}) v_j = 0, \quad j \in \{p_{j_1}, p_{j_2}, \ldots, p_{j_{n-1}}\}.
\]

Thus, the vector \( \nabla f_i(\bar{x},\bar{p}) \) is orthogonal to the \( n-1 \) dimensional subspace spanned by \( \{v_1, \ldots, v_{n-1}\} \), and hence is uniquely determined up to multiplication by a positive scalar. Another way to phrase this is to say that \( \nabla f_i(\bar{x},\bar{p}) \) is in the (one-dimensional) left nullspace of the matrix \( A \) whose rows are the \( v_i \)’s, or that (if nonzero) the transpose of this gradient can be found as an (any) eigenvector associated to the zero eigenvalue of the transpose of \( A \).
Numerical approximation by finite differences

Approximating the vectors $v_j$ by finite differences, one has that $\nabla f_i(\bar{x}, \bar{p})$ is approximately orthogonal to these differences as well. Explicitly, suppose that we approximate $v_j = (\partial \varphi / \partial p_j)(\bar{p})$ by:

$$\frac{1}{h} (\varphi(\bar{p} + he_j) - \bar{x}),$$

where $h$ is small and where $e_j$ is the vector having a one in the $j$th position and zeros elsewhere. Then, $\nabla f_i(\bar{x}, \bar{p})$ is (approximately) orthogonal to the differences

$$\varphi(\bar{p} + he_j) - \bar{x},$$

which form a set of $n - 1$ linearly independent vectors (if $h$ is small). A simple matrix inversion (after fixing an arbitrary value for one of its entries) allows the computation of $\nabla f_i(\bar{x}, \bar{p})$. Observe that division by the potentially small number $h$ is not required in performing these operations. In fact, no knowledge whatsoever about parameter values is needed by the algorithm.

Note that $\varphi(\bar{p} + he_j)$ is the state $x(t)$ at the time $t$ at which the particular coordinate $x_i$ achieves a local extremum value, if the parameters have been perturbed to $p = \bar{p} + he_j$. To be more precise, $t$ is the unique time close to $\bar{t}$ such that $\dot{x}_i(t) = 0$ when parameter vector $p$ is being used. Theoretically, we must have $p \approx \bar{p}$, so $h$ must be very small, but, in practice, quite large perturbations of $p$ also work fine.

3 A simple numerical example

We illustrate the calculations with a very simple example, the following system (writing $x$ instead of $x_1$ and $y$ instead of $x_2$):

$$\begin{align*}
\dot{x} &= -3x + \frac{10}{1+y} \\
\dot{y} &= px + 1 - 3y
\end{align*}$$

with initial state $(0, 0)$ and reference parameter $\bar{p} = 2$. This might represent the simplified dynamics of a two-gene network, in which the first gene enhances the expression of the second gene, which in turn represses the rate of expression of the first one, there is a constitutive rate of production of the second gene, and both protein products decay at rate 3 sec$^{-1}$. The single parameter $p$ may represent a promoter strength, and we assume that there is a way to perturb it (perhaps by duplication or sequence change). The solid lines in Figure [1] (and also in Figures [2] and [3]) show plots of the solution coordinates $x(t)$ and $y(t)$.

Let us pose the following problem: not knowing the above equations, estimate the relative strength of the second gene’s effect on the rate of expression of the first one. The only data to be used are the levels of both gene products ($x(t)$ and $y(t)$) at the time when $x(t)$ achieves its local maximum. We do assume known the fact that the parameter $p$ affects directly only the rate of expression of the second gene, not the first. Observe that the maximum of $x$ is attained at $t \approx 0.5275$, and the values there are (approximately) $x(t) = 1.6553$ and $y(t) = 1.0138$. The gradient $\nabla f_1$ of $-3x + \frac{10}{1+y}$, evaluated at $(1.407, 1.3695)$, has the true (but unknown to the algorithm) value:

$$\left(-3, -\frac{10}{(1+y)^2}\right)_{y=1.0138} \approx (-3, -2.4659).$$

Next, we perform the “experiment” in which $p$ is up-perturbed by 25%. With the new parameter $p = 2.5$, we obtain plots as shown by the dotted lines in Figure [1]. Now the maximum of $x$ is attained at $t \approx 0.4268$, and the values there are $x(t) = 1.407$ and $y(t) = 1.3695$. Letting $\delta = (1.407, 1.3695) - (1.6553, 1.0138)$, the unknown (to the algorithm) gradient $\nabla f_1$ is known to be (approximately) orthogonal
Figure 1: Trajectories; dashed is perturbed motion with 25% change in parameter to $\delta$. Any vector perpendicular to $\delta$ must be a multiple of $(-3, -2.3455)$. (We normalized the first entry to -3 merely in order to compare our result to the true gradient; the algorithm does not know the value "-3". In practice, however, one may assume that the first entry of the vector is negative, reflecting degradation or dilution effects, so the algorithm will give the correct sign for the second term, as well as its magnitude relative to the rate of degradation or dilution.) The relative error in our estimate is less than 5%.

Even larger perturbations may be performed. For example, a 50% perturbation from $\bar{p} = 2$ to $p = 3$, provides the dashed lines in Figure 2. Now the maximum for $x$ is attained at $t \approx 0.4658$, and there

Figure 2: Trajectories; dashed is perturbed motion with 50% change in parameter

$x(t) = 1.5103$ and $y(t) = 1.2073$. The estimated gradient is now $(-3, -2.2476)$, which gives a relative error of less than 9%. Finally, a 100% perturbation to $p = 4$ provides the dashed lines in Figure 3. Now the maximum for $x$ is attained at $t \approx 0.4268$, and there $x(t) = 1.4071$ and $y(t) = 1.3695$. The estimated gradient is now $(-3, -2.0936)$, which gives a relative error of about 15%.

Remarks

As its name implies, one of the main advantages of the MRA method in the steady-state case is that only "communicating intermediates" in-between "modules" need to be measured (for example, just the active forms of Erk1/2, Mek1/2 and Raf-1, in [7]). Here, we only carried out the analysis in the case in
which all the variables $x_i$ can be measured. In the general case, if one assumes that hidden (internal) variables are at quasi-steady state at the same times as the communicating variables, then an implicit function argument as in [6] allows one to reduce to the present situation, by writing the hidden variables in terms of the communicating quantities. However, there is no reason for the method to work when the hidden variables do not have this property.

Also, we assume perfect “noise free” data. The analysis of noise performed in [1] carries over with no changes to the quasi-steady state case.
References

[1] M. Andrec, B.N. Kholodenko, R.M. Levy, and E.D. Sontag. Inference of signaling and gene regulatory networks by steady-state perturbation experiments: structure and accuracy. *J. Theoret. Biol.*, 232(3):427–441, 2005.

[2] P. Berman, B. Dasgupta, and E.D. Sontag. Algorithmic issues in reverse engineering of protein and gene networks via the modular response analysis method. *Annals of the NY Academy of Sciences*, 1115:132–141, 2007.

[3] P. Berman, B. Dasgupta, and E.D. Sontag. Randomized approximation algorithms for set multico ver problems with applications to reverse engineering of protein and gene networks. *Discrete Applied Mathematics Special Series on Computational Molecular Biology*, 155:733–749, 2007.

[4] E. J. Crampin, S. Schnell, and P. E. McSharry. Mathematical and computational techniques to deduce complex biochemical reaction mechanisms. *Prog Biophys Mol Biol*, 86(1):77–112, September 2004.

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

[6] B.N. Kholodenko and E.D. Sontag. Determination of functional network structure from local parameter dependence data. Technical report, 2002. arXiv [physics/0205003], May 2002.

[7] S.D.M. Santos, P.J. Verveer, and P.I.H. Bastiaens. Growth factor induced MAPK network topology shapes Erk response determining PC-12 cell fate. *Nature Cell Biology*, 9:324–330, 2007.

[8] E.D. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems*. Springer-Verlag, New York, second edition, 1998.

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

[10] J. Stark, R. Callard, and M. Hubank. From the top down: towards a predictive biology of signalling networks. *Trends Biotechnol*, 21(7):290–293, Jul 2003.