# AUTOMATIC MODEL GENERATION FOR SIGNAL TRANSDUCTION WITH APPLICATIONS TO MAP-KINASE PATHWAYS

**Category: Methodologies for system-level understanding of life**

**Bruce E Shapiro*[1], Andre Levchenko**[2], Eric Mjolsness*[3]**
* Jet Propulsion Laboratory, California Institute of Technology
** Division of Biology and Division of Engineering and Applied Science, California Institute of Technology

## ABSTRACT

We describe a general approach to automatic model generation in the description of dynamic regulatory networks. Several potential areas of application of this technique are outlined. We then describe how a particular implementation of this approach, Cellerator©, has been used to study the mitogen-activated protein kinase (MAPK) cascade. These signal transduction modules occur both in solution and when bound to a scaffold protein, and we have generalized the technique to include both types of module. We show that the results of simulations with the Cellerator©–created model are consistent with our previously published report, where an independently written model was developed. New results made possible by the use of Cellerator© are also presented. An important aspect of Cellerator© operation – explicit output description at several steps during model generation – is emphasized. This design allows intervention and modification of the model "on the go" leading to both a more flexible of model description and a straightforward error correction mechanism. We also outline our future plans in Cellerator© development.

## INTRODUCTION

In the past few decades the rapid gain of information about intracellular signal transduction and genetic networks has led to the view of regulatory biomolecular circuits as highly structured multi-component systems that have evolved to perform optimally in very uncertain environments. This emergent complexity of biochemical regulation necessitates the development of new tools for analysis, most notably computer assisted mathematical models. Computer modeling has proved to be of crucial importance in the analysis of genomic DNA sequences and molecular dynamics simulations and is likely to become an indispensable tool in biochemical and genetic research. Several platforms have been (or are being) developed that enable biologists to do complex computational simulations of various aspects of cellular signaling and gene regulatory networks.

In spite of their promise, these new modeling environments have not been widely utilized in the biological research community. Arguably, among the reasons for this is a relative inaccessibility of the modeling interface for the typical classically trained geneticist or biochemist. Instead of cartoon representations of signaling pathways, in which activation can be represented simply by an arrow connecting two molecular species, users are often asked to write specific differential equations or chose among different modeling approximations. Even for fairly modest biomolecular circuits such a technique would involve explicitly writing dozens (or even hundreds) of differential equations, a job that can be tedious, difficult, and highly error prone, even for an experienced modeler. Thus it would be extremely helpful to have a modeling interface that would automatically convert a cartoon- or reaction-based biochemical pathway description into a mathematical representation suitable for the solvers built into various currently existing software packages.

[1] Bruce E. Shapiro, Machine Learning Systems Group, Jet Propulsion Laboratory, California Institute of Technology, M/S 264-355, 4800 Oak Grove Drive, Pasadena CA, USA 91109; tel. (818) 393 0980 bshapiro@jpl.nasa.gov.
[2] Andre Levchenko, Division of Biology, California Institute of Technology, Mail Stop 156-29, Pasadena, CA USA 91125; tel. (626) 395 8542 andre@vigeland.paradise.caltech.edu.
[3] Eric Mjolsness, Machine Learning Systems Group, Jet Propulsion Laboratory, California Institute of Technology, M/S 126-347, 4800 Oak Grove Drive, Pasadena, CA USA 91109; tel. (818) 393 5311mjolsness@jpl.nasa.gov.

In addition to being more accessible to a wider research community, a tool allowing the automatic generation of mathematical models would facilitate the modeling of complex networks and interactions. For example, in intracellular signal transduction it is not uncommon to find multi-molecular complexes of modifiable proteins. The number of different states that a multi-molecular complex, along with the number of equations required to fully describe the dynamics of such a system, increases exponentially with the number of participating molecules or classes of molecules. One typical complex – scaffolds in MAPK cascades – will be studied in detail later in this report. It is often the case that the dynamics of each state is of interest. A modeler then faces the unpleasant, and potentially error prone task, of writing dozens, if not hundreds, of equations. Automatic equation generation can significantly ease this task.

In this report we consider a general approach to automatic model generation for the description of dynamic regulatory networks. Several potential areas of application of this technique will be outlined. We then will describe how a particular implementation of this approach, Cellerator©, has been used to study the mitogen-activated protein kinase (MAPK) cascade signal transduction modules operating in solution or when bound to a scaffold protein. An important aspect of Cellerator© operation – explicit output description and flexible user intervention at several steps through the model generation -- will be emphasized. This design, which allows intervention and modification of the model "on the fly" leads to increased model design flexibility and provides an immediate error correction mechanism.

## AUTOMATIC MODEL GENERATION

### Canonical Forms for Cell Simulation

We can loosely classify the components needed to perform cell simulation in order of their biological complexity: simple chemical reactions including degradation, enzymatic reactions in solution, multi-molecular complexes with a non-trivial number of states (e.g., scaffold proteins), multiple interacting and non-overlapping pathways, transcription, translation, intracellular components, transport processes and morphogenesis. We will examine these processes and attempt to derive general *canonical forms* that can be used to describe these processes in the following paragraphs. These canonical forms can be either *input forms*, such as chemical reactions, or *output forms*, such as differential equations that are automatically generated by the program. It is crucial to identify these canonical forms so that an efficient mapping from the input forms to the output forms can be implemented. Specific examples of how these forms may be implemented in a computer program are given in the following section.

Biochemistry is frequently referred to as the language of biology, in much the same way that mathematics has been called the language of physics. Cellular activity is generally expressed in terms of the biochemical cascades that occur. These chemical reactions constitute the core of our input forms; the corresponding differential equations constitute the core of our output forms. (Differential equations can be thought of as output because they are passed on to solver and/or optimizer modules to handle). A fundamental library of simple chemical reactions can be quickly developed; such reactions take the form

$$\sum_{X_i \in S' \subset S} X_i \xrightarrow{k} \sum_{Y_i \in S'' \subset S} Y_i \tag{1}$$

where $S$ is a set of reactants and $S'$ and $S''$ are (possible empty and possibly non-distinct) subsets of $S$ and $k$ is a representation of the rate at which the reaction proceeds. In general there are rarely more than two elements in either $S'$ or $S''$ but it is possible for there to be more. Fore example, all of the following chemical reactions fall into this form:

| | |
|---|---|
| $A + B \rightarrow C = AB$ | complex formation |
| $C = AB \rightarrow A + B$ | dissociation |
| $A \rightarrow B$ | conversion |
| $A \rightarrow \phi$ | degradation |
| $\phi \rightarrow A$ | creation (*e.g.*, through transcription) |

(2)

Enzyme kinetic reactions, which are usually written as

$$S + E \rightarrow P + E \tag{3}$$

where $E$ is an enzyme that facilitates the conversion of the substrate $S$ into the product $P$, would also fall into this class. More generally, equation 3 is a simplification of the cascade

$$S + E \leftrightarrow SE \rightarrow S + P \tag{4}$$

where the bi-directional arrow indicates that the first reaction is reversible. Thus (4) is equivalent to the triplet of reactions

$$\{S + E \rightarrow SE,\ SE \rightarrow S + E,\ SE \rightarrow S + P\} \tag{5}$$

The reactions (4) or (5) can be written compactly with the following double-arrow notation

$$S \overset{E}{\Rightarrow} P \tag{6}$$

which should be read as " the conversion of $S$ to $P$ is catalyzed by an enzyme $E$." If there is also an second enzyme, $G$, that can catalyze the reverse reaction

$$P \overset{G}{\Rightarrow} S = \{P + G \rightarrow GP,\ GP \rightarrow G + P,\ GP \rightarrow G + S\} \tag{7}$$

we further use the double-double arrow notation

$$S \underset{G}{\overset{E}{\Leftrightarrow}} P \tag{8}$$

to compactly indicate the pair of enzymatic reactions given by (6) and (7). The enzyme above the arrow always facilitates the forward reaction, and the enzyme beneath the reaction always facilitates the reverse reaction. For example, $E$ might be a kinase and $G$ might be a phosphatase molecule. Since each of equations (6) and (7) represent a triplet of simpler reactions, we observe that the notation of equation (8) compactly represents a total of six elementary reactions, each of which is in the form given by equation (1). We therefore take equation (1) as our input canonical form for chemical reactions. The corresponding output canonical form is given by the set of differential equations

$$\tau_i \dot{X}_i = \sum_{\alpha} c_{i\alpha} \prod_j X_j^{n_{i\alpha j}} \tag{9}$$

where the $\tau_i$ and $c_{i\alpha}$ are constants that are related to the rate constants , the signs of the $c_{i\alpha}$ are determined from which side of equation (1) the terms in equation (9) correspond to, and the $n_{i\alpha j}$ represent the cooperativity of the reaction. The summation is taken over all equations in which $X_i$ appears. Multi-molecular reactions (e.g., binding to a scaffold protein) and multiple interacting and overlapping pathways are described in much the same way - there are just more reactions that must be included in our model. The canonical forms (1) and (9) can still describe each one of these reactions.

Genetic transcription and translation into proteins can be described by an extension of equation (9) to include terms of the form

$$\tau_i \dot{X}_i = \prod_{\beta} \frac{c_{i\beta} X_\beta^{n\beta}}{K_{i\beta}^{n\beta} + X_\beta^{n\beta}} \tag{10}$$

where the product runs over the various transcription factors $\{X_\beta\}$ that influence production of $X_i$. If there are any reactions of the form (1) for $X_i$ then the expression on the right side of equation (10) would be added to the right hand side of (9). In a more realistic system, a gene would b e influenced by a (possibly large) set of promoter and enhancer elements $X_i$ that bind to different sites. A hierarchical model could describe this set of interactions

$$\tau_i \dot{X}_i = \frac{J u_i}{1 + J u_i} - \lambda_i X_i \tag{11}$$

$$u_i = \prod_{\alpha \in i} \frac{1 + J_\alpha \tilde{v}_\alpha}{1 + \hat{J}_\alpha \tilde{v}_\alpha} \tag{12}$$

$$\tilde{v}_\alpha = \frac{\tilde{K}_\alpha \tilde{u}_\alpha}{1 + \tilde{K}_\alpha \tilde{u}_\alpha} \tag{13}$$

$$\tilde{u}_\alpha = \prod_{b \in \alpha} \frac{1 + K_b v_{j(b)}^{n(b)}}{1 + \hat{K}_b v_{j(b)}^{n(b)}} \tag{14}$$

where $i$ and $j$ index transcription factors, $\alpha$ indexes promoter modules, $b$ indexes binding sites, the function $j(b)$ determines which transcription factor $j$ binds at site $j$, the $J$ and $K$ are constants, and $\lambda$ is a degradation rate.

Sub-cellular components represent a higher order of biological complexity. If we assume perfect mixing each component can be treated as a separate pool of reactants which we can describe by the reaction

$$X_A \rightarrow X_B \tag{15}$$

This is taken to mean that $X$ in pool $A$ is transported into pool $B$ at some rate. When the concentration changes and distances involved are small such processes can be described by the canonical forms in equation (1). In large or elongated cells with long processes (such as neurons) or when the molecules have a net charge the transport process defined in equation (15) can not be described by the output canonical form (9). Instead we must modify this ordinary differential equation into a partial differential equation to allow for diffusion,

$$\tau_i \frac{\partial X_i}{\partial t} = \nabla \bullet \left( D_i \nabla X_i + C_i D_i \nabla V \right) + \sum_\alpha c_{i\alpha} \prod_j X_j^{n_{i\alpha j}} \tag{16}$$

where the $D_i$ are (possibly spatially dependent) diffusion constants for species $X_i$, $C_i$ are charge and temperature dependent constants, and $V$ is the voltage. Other voltage and pressure dependent movement between compartments (especially those with membranes) that are controlled by channels and transport proteins could be described by including additional terms on the right hand side of equation (16) (e.g., Hodgkin-Huxley type expressions).

**Implementation**

In standard biochemical notation, protein cascades are represented by a arrow-sequence of the form

$$A \Rightarrow B \Rightarrow \cdots \tag{17}$$

where each step (the $A$, $B$, ...) would represent, for example, the activation of a particular molecular species. Our goal is to translate the cascade (17) into a computable form while retaining the biological notation in the user interface. Mathematically, we can specify such as cascade as a multiset

$$C = \{P, R, IC, I, F\} \tag{18}$$

where $P$ is a set of proteins, $R$ is a set of reactions, $IC$ is a set of initial conditions, $I$ is a set of input functions, and $F$ is a set of output functions.

To illustrate this transformation process (from the biochemical notation, such as in equation (17), to the mathematical notation, as in equation (18)), we consider the example where equation (17) represents a simple linear phosphorylation cascade. In this case equation (17) would mean that $A$ facilitates the phosphorylation of $B$, which in turn facilitates the phosphorylation of $C$, and so forth. In general, a cascade can have any length, so we define the elements of a cascade with a simple indexed notation, *e.g.*,

$$K_4 \Rightarrow K_3 \Rightarrow K_2 \Rightarrow K_1 \tag{19}$$

where $K$ is used to indicate that all the members of the cascade induce phosphorylation of their substrates, that is they are kinases. In general, activation can proceed by any specified means.

This indexed notation is always used internally by the program. The user, however, has the option of using either common names or the indexed variables. There is still a great deal of information hidden in this expression, such as how many phosphate groups must be added to make each successive protein active. In the MAPK cascade for example (as explained below), the input signal that starts this cascade is $K_4$. The output, however, is not $K_1$, as this notation would suggest, but a doubly phosphorylated version of $K_1$. Hence for MAPK cascade we introduce a modified notation:

$$\begin{aligned} & K_3 \overset{K_4}{\Rightarrow} K_3^* \\ & K_2 \overset{K_3^*}{\Rightarrow} K_2^* \overset{K_3^*}{\Rightarrow} K_2^{**} \\ & K_1 \overset{K_2^{**}}{\Rightarrow} K_1^* \overset{K_2^{**}}{\Rightarrow} K_1^{**} \end{aligned} \tag{20}$$

where each phosphate group that has been added is indicated with an asterisk. From this notation it is clear that the input is $K_4$ and the output is $K_1$**.

In general, suppose we have a cascade formed by n proteins $K_1$, $K_2$, ..., $K_n$, and that the $i^{th}$ protein $K_i$ can be phosphorylated $a_i$ times. Denote by $K_i^j$ the fact that kinase $K_i$ has be phosphorylated $j$ (possibly zero) times. The set $P$ of all kinases $K_i^j$ in an $n$-component cascade is then

$$P = \{K_i^j \mid i = 1,2,...,n,\ j = 0,1,\ldots,a_i\} \tag{21}$$

The *reactions* in the cascade are of the form

$$R = \left\{ K_i^j \overset{K_{i+1}^{a_{i+1}}}{\Rightarrow} K_i^{j+1} \,\middle|\, i = 1,\ldots,n-1,\ j = 0,\ldots,a_j - 1 \right\} \tag{22}$$

We note at this point that this notation describes a linear cascade, in which each element $K_i$ is only phosphorylated by the active form of $K_{i-1}$. It does not include other reactions, when, for example, $K_3$ might, under special circumstances, phosphorylate $K_1$ directly without the intermediate step of first phosphorylating $K_2$. Such additional reactions could be added, but they have been omitted from this presentation to simplify the discussion. We can also add the dephosphorylation enzymes, or phosphatases, with a double-arrow notation:

$$R = \left\{ K_i^j \underset{Ph_i}{\overset{K_{i+1}^{a_{i+1}}}{\Leftrightarrow}} K_i^{j+1} \,\middle|\, i = 1,\ldots,n-1,\ j = 0,\ldots,a_j - 1 \right\} \tag{23}$$

In general, it is not necessary to specify explicit conservation laws with this notation because they are built directly into the equations. For example, we do not have to separately specify that the quantities

$$K_i^{Total} = \sum_{j=0}^{a_i} K_i^j \tag{24}$$

because this is implicit in the differential equations that are built using this notations. We do, however, have to specify the initial conditions,

$$IC = \left\{ K_i^j(0) \,\middle|\, i = 1,2,...,n,\ j = 0,1,\ldots,a_i \right\} \tag{25}$$

Next, we need to specify how the cascade is initiated. For example if $K_4$ is not present until some time $t_{on}$ and then is fixed at a level $c$, would write the set of *input functions* as

$$I = \{K_4(t) = cH(t - t_{on})\} \tag{26}$$

where $H(t) = \begin{cases} 0, t < 0 \\ 1, t \geq 0 \end{cases}$ is the Heaviside step function. In some cases, we are only interested in the total quantity of each substance produced as a function of time, e.g., $K_i^j(t)$ . More generally, we would also specify a set of *output functions* $F$. For example we might have $F = \{f, g\}$ where $f(T)$ is the total accumulated protein concentration after some time $T$,

$$f(T) = \int_{t_{on}}^{T} K_1^{a_1}(t)dt \tag{27}$$

and $g(c)$ is the steady state concentration of activated kinase,

$$g(c) = \lim_{t_{on} \to \infty} \left[ \lim_{t \to \infty} K_1^{a_1}(t) \right] \tag{28}$$

where $c$ is the input signal specified $I$. Then the cascade is then completely specified by the multiset $C = \{P, R, IC, I, F\}$.

If we have an additional regulatory protein, such as a scaffold that holds the various proteins in equation (20) together there are additional reactions. These describe binding of the enzymes to the scaffold and phosphorylation within the scaffold. We describe the scaffold itself by defining an object $S_{p_1,p_2,\cdots,p_n}$ where $n$ is as before (the number of kinases that may bind to the scaffold, or alternatively, the number of "slots" in the scaffold) and $p_i \in \{\varepsilon, 0, 1, \ldots, a_i\}$ indicates the state of phosphorylation of the proteins in each slot. Thus if $p_i = \varepsilon$ (or, alternatively, -1) the slot for $K_i$ is empty, if $p_i = 0$, $K_i^0$ is in the slot, etc. For a three-slot scaffold, for example, we would add to the set $P$ the following set

$$P' = \left\{ S_{ijk} \;\middle|\; i = \varepsilon, 0, 1, \ldots, a_1,\ j = \varepsilon, 0, 1, \ldots, a_2,\ k = \varepsilon, 0, 1, \ldots, a_3 \right\} \tag{29}$$

To describe binding to the scaffold, we would also add to the set $R$ the following reactions

$$R' = \left\{ S_{p_1,\cdots,p_i=\varepsilon,\cdots,p_n} + K_i^j \leftrightarrow S_{p_1,\cdots,p_i=j,\cdots,p_n} \right\} \tag{30}$$

where the indices run over all values in the range

$$p_i = \begin{cases} \varepsilon, 0, 1, \ldots, a_i, & i \neq j \\ 0, 1, \ldots, a_i, & i = j \end{cases} \tag{31}$$

For the three-member scaffold this would be

$$\begin{aligned} R' = &\left\{ S_{\varepsilon jk} + K_1^i \leftrightarrow S_{ijk}, i = 0, \ldots, a_1,\ j = \varepsilon, 0, \ldots, a_2,\ k = \varepsilon, 0, \ldots, a_3 \right\} \\ &\cup \left\{ S_{i\varepsilon k} + K_2^j \leftrightarrow S_{ijk}, i = \varepsilon, 0, \ldots, a_1,\ j = 0, \ldots, a_2,\ k = \varepsilon, 0, \ldots, a_3 \right\} \\ &\cup \left\{ S_{ij\varepsilon} + K_3^k \leftrightarrow S_{ijk}, i = \varepsilon, 0, \ldots, a_1,\ j = \varepsilon, 0, \ldots, a_2,\ k = 0, \ldots, a_3 \right\} \end{aligned} \tag{32}$$

Finally, we have phosphorylation in the scaffold. This can be done either by a protein that is not bound to the scaffold, *e.g.*, for the input signal,

$$R'' = \left\{ S_{p_1,\cdots,p_{i-1}=j<a_{i-1},p_i=a_i,\cdots,p_n} + K \Leftrightarrow S_{p_1,\cdots,p_{i-1}=j+1,p_i=a_i,\cdots,p_n} \right\} \tag{33}$$

where the two-sided double arrow ($\Leftrightarrow$) is used as shorthand for the (possibly bi-directional) enzymatic reaction, or by one that is bound to the scaffold,

$$R''' = \left\{ S_{p_1,\cdots,p_{i-1}=j<a_{i-1},p_i=a_i,\cdots,p_n} \rightarrow S_{p_1,\cdots,p_{i-1}=j+1,p_i=a_i,\cdots,p_n} \right\} \tag{34}$$

or by some combination of the two, all of which must be added to the reaction list $R$. For the three-slot scaffold with external signal $K_4$ that activates $K_3$, we have

$$\begin{aligned} R'' = &\left\{ S_{i,a_2,k} \rightarrow S_{i+1,a_2,k}, i = 0,\ldots,a_1 - 1, k = \varepsilon, 0,\ldots,a_3 \right\} \\ &\cup \left\{ S_{i,j,a_3} \rightarrow S_{i,j+1,a_3}, i = \varepsilon, 0,\ldots,a_1, j = \varepsilon, 0,\ldots,a_2 - 1 \right\} \end{aligned} \tag{35}$$

and

$$R''' = \left\{ S_{ijk} \underset{Ph_3}{\overset{K_4}{\Leftrightarrow}} S_{i,j,k+1},\ i = \varepsilon, 0,\ldots,a_1,\ j = \varepsilon, 0,\ldots,a_2,\ k = 0,\ldots,a_3 - 1 \right\} \tag{36}$$

Typical $a_i$ values for this type of cascade are $a_1$=$a_2$=2 and $a_3$=1.

As an example, let us continue with the above-mentioned three-member cascade that is initiated with $K_4$. In what follows, we refer to Cellerator©, a Mathematica® package that implements the above algorithms. In Cellerator© we have defined the function

**genReacts**[*kinase-name*, *n*, {$a_i$}, *phosphatase-name*],

where *kinase-name* and *phosphatase-name* are the names we want to give to the sequences of kinases and phosphatases, respectively, and $n$ and $a_i$ are as before. The following Cellerator© command then generates the above set of reactions (20),

```
genReacts[K, 3, {2, 2, 1, 1}, kpase]
```

```
                K[2,2]          K[2,2]          K[3,1]          K[3,1]          K[4,1]
{K[1, 0] <=> K[1, 1], K[1, 1] <=> K[1, 2], K[2, 0] <=> K[2, 1], K[2, 1] <=> K[2, 2], K[3, 0] <=> K[3, 1]}
     kpase[1]        kpase[1]        kpase[2]        kpase[2]        kpase[3]
```

The input is in the first line while the output is the second line. Alternatively, the user could specify the set of reactions explicitly, or copy the output to a later cell to manually add additional reactions. If RAF has been set up as an alias for $K_3$ then the rate constants are specified by a content-addressable syntax, e.g., as

```
                                  RAFK
storeRateConstant[db, RAF <=> RAF*, a1, d1, k1, a2, d2, k2];
                                  RAFP
```

corresponding to

$$RAF + RAFK \underset{d_1}{\overset{a_1}{\Leftrightarrow}} RAF - RAFK \overset{k_1}{\Rightarrow} RAF^* + RAFK \tag{37}$$

and

$$RAF^* + RAFP \underset{d_2}{\overset{a_2}{\Leftrightarrow}} RAF^* - RAFP \overset{k_2}{\Rightarrow} RAF + RAFP \tag{38}$$

and so forth, where the numbers over the arrows indicate the rate constants (and not enzymes, as with the double arrow notation). Cellerator first translates the five high-order reactions (equation 20) into the corresponding set of 30 low-level reactions. Each low-level reaction (such as intermediate compound formation) is determined by applying the appropriate enzyme-kinetics description, and has a unique rate constant. The low-level reactions are subsequently translated into the appropriate set of 21 differential equations for the eight kinases, three phosphatases and ten intermediate compounds. When scaffold proteins are included (discussed below) these numbers increase to 139 high level reactions, 348 low-level reactions (300 without kinases), and 101 differential equations (85 without kinases).

## MAPK PATHWAY WITH SCAFFOLDS: EXPERIMENTAL BACKGROUND

The mitogen-activated protein kinase (MAPK) cascades (Fig. 1) are a conserved feature of a variety of receptor mediated signal transduction pathways ((Garrington and Johnson, 1999; Widmann et al., 1999; Gustin et al., 1998)). In humans they have been implicated in transduction of signals from growth factor, insulin and cytokine receptors, T cell receptor, heterotrimeric G proteins and in response to various kinds of stress ((Garrington and Johnson, 1999; Putz et al., 1999; Sternberg and Alberola-Ila, 1998; Crabtree and Clipstone, 1994; Kyriakis, 1999)). A MAPK cascade consists of three sequentially acting kinases. The last member of the cascade, MAPK is activated by dual phosphorylation at tyrosine and threonine residues by the second member of the cascade: MAPKK. MAPKK is activated by phosphorylation at threonine and serine by the first member of the cascade: MAPKKK. Activation of MAPKKK apparently proceeds through different mechanisms in different systems. For instance, MAPKKK Raf-1 is thought to be activated by translocation to the cell membrane, where it is phosphorylated by an unknown kinase. All the reactions in the cascade occur in the cytosol with the activated MAPK translocating to the nucleus, where it may activate a battery of transcription factors by phosphorylation.

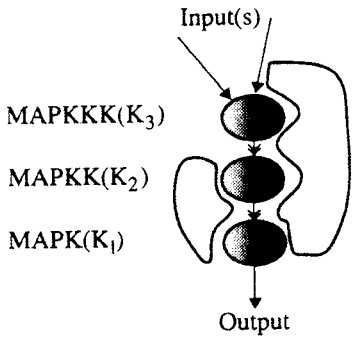


Figure 1. The topology of MAPK signaling cascade. Each double arrow represents activation through dual phosphorylation. Two and three-member scaffolds have been identified experimentally and are depicted here.

MAPK cascades have been implicated in a variety of intercellular processes including regulation of the cell cycle, apoptosis, cell growth and responses to stress. These molecules are of crucial importance in the development of memory and wound healing. Abnormal changes in MAPK pathway regulation often mediate various pathologies, most notably cancer. This central role of MAPK mediated signal transduction in most regulatory processes makes it an especially attractive research and modeling object.

Signal transduction through a MAPK cascade can be very inefficient unless additional regulatory proteins, called scaffolds, are present in the cytosol. Scaffold proteins nucleate signaling by binding two or more MAP kinases into a single multi-molecular complex. It has been reported previously that scaffolds can both increase and decrease the efficiency of signaling in a concentration dependent manner (Levchenko *et. al.*, 2000). In addition they can reduce the non-linear activation characteristics of the cascade. These properties may be crucial for global and local activation of MAPK as scaffold proteins may selectively translocate to small subcellular compartments, thus locally facilitating or inhibiting MAPK activation. In this report we show how the use of Cellerator© software package has allowed us to substantially improve our earlier model and study its parametric dependence in a manner not investigated in the preceding report.

## MAPK PATHWAY WITH SCAFFOLDS: RESULTS

As described above, addition of scaffold proteins into the MAPK reaction system results in markedly increased number of states and equations describing transitions between them. Here the benefits provided by Cellerator© can really be appreciated, as a simple sequence of commands can lead to automatic description of all reactions involving scaffold-kinase complexes (see Fig. 2).

In our simulations the first goal was to verify the automatic model generation for scaffold-medicated MAPK cascade as implemented in Cellerator©. As a basis for the comparison we referred to our previous report describing a quantitative model of the effect scaffold proteins can play in MAPK mediated signal transduction. When all the assumptions of that model were made again exactly the same solution for the three-member scaffold case was obtained. This convergence of results verified the model generated by Cellerator©. In addition, the difficulty of manual generation of all the necessary equations, a limiting factor of the previous study, has now been removed. We thus attempted to study a more detailed model, in which some of the previous assumptions were relaxed.

```
phosphorylationReactions = genScafPhosReacts[S, {2, 2, 1, 1}, K]

{S[0, 2, -1] → S[1, 2, -1], S[0, 2, 0] → S[1, 2, 0], S[0, 2, 1] → S[1, 2, 1], S[1, 2, -1] → S[2, 2, -1],
 S[1, 2, 0] → S[2, 2, 0], S[1, 2, 1] → S[2, 2, 1], S[-1, 0, 1] → S[-1, 1, 1], S[-1, 1, 1] → S[-1, 2, 1],
 S[0, 0, 1] → S[0, 1, 1], S[0, 1, 1] → S[0, 2, 1], S[1, 0, 1] → S[1, 1, 1], S[1, 1, 1] → S[1, 2, 1],
                                                                    K[4,1]
 S[2, 0, 1] → S[2, 1, 1], S[2, 1, 1] → S[2, 2, 1], S[-1, -1, 0] ⇌ S[-1, -1, 1],
          K[4,1]                    K[4,1]                    K[4,1]
 S[-1, 0, 0] ⇌ S[-1, 0, 1], S[-1, 1, 0] ⇌ S[-1, 1, 1], S[-1, 2, 0] ⇌ S[-1, 2, 1],
          K[4,1]                    K[4,1]                    K[4,1]                    K[4,1]
 S[0, -1, 0] ⇌ S[0, -1, 1], S[0, 0, 0] ⇌ S[0, 0, 1], S[0, 1, 0] ⇌ S[0, 1, 1], S[0, 2, 0] ⇌ S[0, 2, 1],
          K[4,1]                    K[4,1]                    K[4,1]                    K[4,1]
 S[1, -1, 0] ⇌ S[1, -1, 1], S[1, 0, 0] ⇌ S[1, 0, 1], S[1, 1, 0] ⇌ S[1, 1, 1], S[1, 2, 0] ⇌ S[1, 2, 1],
          K[4,1]                    K[4,1]                    K[4,1]                    K[4,1]
 S[2, -1, 0] ⇌ S[2, -1, 1], S[2, 0, 0] ⇌ S[2, 0, 1], S[2, 1, 0] ⇌ S[2, 1, 1], S[2, 2, 0] ⇌ S[2, 2, 1]}
```

Figure 2. The implementation of automatic generation of the MAP kinases activation reactions (through phosphorylation) in the scaffold in the Cellerator© environment. All the possible scaffold states (species) are generated as are the transition reactions between them. The indexes in the parentheses indicate the phosphorylation status of the kinase in the corresponding position, with -1 corresponding to the absence of the kinase from the scaffold complex. K[4,1] represents the external kinase activating the first MAP kinase (MAPKKK) in the cascade.

The use of Cellerator© has allowed us to perform systematic sensitivity analyses of the assumptions made in our description of the role of scaffold proteins in MAPK cascade regulation (Levchenko et.al., 2000). We previously described dual MAPKK and MAPK phosphorylation within the scaffold to proceed as a single step (processive activation). This is substantially different from a two-step dual phosphorylation sequence occurring in solution. In this distributive activation, the first phosphorylation event is first followed by complete dissociation from the activating kinase and subsequently the second phosphorylation reaction occurs. The assumption of processive phosphorylation in the scaffold has some experimental basis. Mathematically, it is equivalent to assuming that the rate of the second phosphorylation reaction is fast compared to the first reaction. Although this assumption was partially relaxed in our previous report, no systematic study of relaxation of this assumption has been performed. Using Cellerator© we performed a systematic investigation of the role of increasing or decreasing the rate of the second phosphorylation within the scaffold compared to reactions in solution. The results for the case when the two rates are equal are presented in Fig. 3. It is clear that relaxation of this assumption results in a substantial decrease of efficiency of signal propagation.

Similar simulations were performed to investigate the effect of allowing formation of a complex between MAPKKK in the scaffold and MAPKKK-activating kinase, as well as the effect of allowing phosphatases to dephosphorylate scaffold-bound kinases. In all cases the parameter values used in simulation are equal

to those used for corresponding reactions in solution (for the full list of parameters see Levchenko et al., 2000). The results are presented in Fig. 3. Again, new assumptions resulted in substantial down-regulation of efficiency of signal propagation. It is of interest that the position of the optimum scaffold concentration (at which the maximum signaling is achieved) is insensitive to making these new assumptions. This agrees with the analysis in (Levchenko *et. al.*, 2000), which suggested that the position of the optimum is determined only by the total concentrations of the kinases and their mutual interaction with the scaffold.

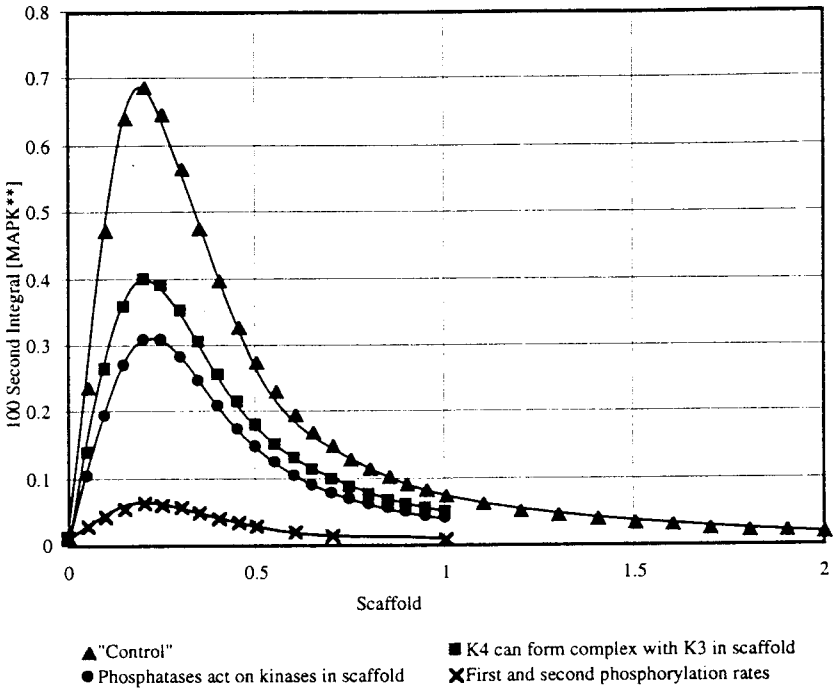


Figure 3. The effect of relaxing several assumptions made in the previous report. The time integral of free dually phosphorylated MAPK over first 100 sec. is plotted vs. scaffold concentration. The "control" curve reproduces the data with all the assumptions made previously, whereas the other curves represent the results of relaxation of these assumptions as described in the legend. All data are obtained using the **Cellerator**[c] package and are plotted in Microsoft Excel.

## DISCUSSION AND FUTURE DIRECTIONS

We have shown that automatic model generation can simplify the transition from an informal, cartoon-based description of a reaction pathway (or a network of pathways) to a system of differential equations. This transition is obtained via a rigorous description of enzymatic kinetics and other biochemical processes and is implemented utilizing symbolic translation. In addition to facilitating the potentially burdensome task of correctly writing out all of the necessary equations, this methodology provides an explicit and flexible way of controlling successive stages of model creation. Furthermore, user intervention is possible both at the stage of conversion of an informal pathway description into a set of chemical reactions and at the later stage of mapping these reactions to the corresponding mathematical forms. This flexibility is

likely to increase the ability of the user to participate in building and modifying the model at a level limited only by his or her expertise.

We have demonstrated the automatic generation of symbolic differential equations using a generic three-member scaffold, the MAPK cascade mediated signaling system. The implementation that we have presented -- Cellerator© -- is capable of generating and solving these 101 differential equations, a task not achieved in the previous detailed study of the effect of scaffolds. Such automated model generation will prove especially useful in describing even more complex biochemical reactions that involve the formation of multi-molecular complexes. Such complexes may exist in numerous states, each requiring a corresponding equation for its dynamical description. Because of the combinatoric expansion of reaction possibilities, correctly writing out all of these equations by hand rapidly becomes impossible.

We intend to pursue the research into role of scaffolds in signal transduction regulation using this new tool. In particular we intend to use extended indexing to specify reactions occurring in various sub-cellular compartments. This will facilitate the study of the effect of scaffold translocation to the cell membrane observed in gradient sensing and other important regulatory processed. In addition we will attempt to develop our algorithm to allow for scaffold dimerization, an experimentally observed phenomenon.

Currently, Cellerator© is "tailor-made" for modeling events in a linear pathway mediated by sequential covalent modification. It is within our immediate plans to make the code more universal to include other canonical forms and variable structure systems. In particular, we are in the process of adapting Cellerator© to two test cases: NF-κB and PKA pathways. Consideration of these pathways will necessitate implementation of the elementary reactions describing transcription, translation and protein degradation. In addition, complex formation will be considered as a high level reaction leading to an activation step within the pathway.

## ACKNOWLEDGEMENTS

We have benefited from discussions with B. Wold and H. Bolouri. This work was supported in part by the Whittier Foundation, the Office of Naval Research under contract N00014-97-1-0422, the NASA Advanced Concepts program and by a Borroughs-Wellcome Fund Computational Molecular Biology Postdoctoral Fellowship to A.L

## BIBLIOGRAPHY

Crabtree, G.R. and Clipstone, N.A. (1994). Signal transmission between the plasma membrane and nucleus of T lymphocytes. Annu.Rev.Biochem. *63*, 1045-1083.

Garrington, T.P. and Johnson, G.L. (1999). Organization and regulation of mitogen-activated protein kinase signaling pathways. Curr.Opin.Cell.Biol. *11*, 211-218.

Gustin, M.C., Albertyn, J., Alexander, M., and Davenport, K. (1998). MAP Kinase Pathways in the Yeast Saccharomyces cerevisiae. Microbiol.Mol.Biol.Rev. *62*, 1264-1300.

Kyriakis, J.M. (1999). Making the connection: coupling of stress-activated ERK/MAPK (extracellular-signal-regulated kinase/mitogen-activated protein kinase) core signaling modules to extracellular stimuli and biological responses. Biochem.Soc.Symp. *64*, 29-48.

Levchenko, A., Bruck, J., Sternberg, P.W. (2000) Scaffold proteins may biphasically affect the levels of mitogen-activated protein kinase signaling and reduce its threshold properties. Proc.Natl.Acad.Sci.USA. 97(11):5818-23.

Putz, T., Culig, Z., Eder, I.E., Nessler-Menardi, C., Bartsch, G., Grunicke, H., Uberall, F., and Klocker, H. (1999). Epidermal growth factor (EGF) receptor blockade inhibits the action of EGF, insulin-like growth factor I, and a protein kinase A activator on the mitogen-activated protein kinase pathway in prostate cancer cell lines. Cancer Res. *59*, 227-233.

Sternberg, P.W. and Alberola-Ila, J. (1998). Conspiracy theory: RAS and RAF do not act alone. Cell *95*, 447-450.

Widmann, C., Gibson, S., Jarpe, M.B., and Johnson, G.L. (1999). Mitogen-activated protein kinase: conservation of a three-kinase module from yeast to human. Physiol.Rev. *79*, 143-180