

Innovations of the rule-based modeling approach

Lily A. Chylek, Edward C. Stites, Richard G. Posner, and William S. Hlavacek

Abstract

New modeling approaches are needed to tackle the complexity of cell signaling systems. An emerging approach is rule-based modeling, in which protein-protein interactions are represented at the level of functional components. By using rules to represent interactions, a modeler can avoid enumerating the reachable chemical species in a system, which is a necessity in traditional modeling approaches. A set of rules can be used to generate a reaction network, or to perform simulations with or without network generation. Although the rule-based approach is a relatively recent development in biology, it is based on concepts that have proven useful in other fields. In this chapter, we discuss innovations of the rule-based modeling approach, relative to traditional approaches for modeling chemical kinetics. These innovations include the use of rules to concisely capture the dynamics of molecular interactions, the view of models as programs, and agent-based computational approaches that can be applied to simulate the chemical kinetics of a system characterized by a large traditional model. These innovations should enable the development of models that can relate the molecular state of a cell to its phenotype, even though vast and complex networks bridge perturbations at the molecular level to fates and activities at the cellular level. In the future, we expect that validated rule-based models will be useful for model-guided studies of cell signaling mechanisms, interpretation of temporal phosphoproteomic data, and cell engineering applications. **Keywords:** Computational modeling, Combinatorial complexity, Protein-protein interactions, Cell signaling, Rule-based modeling, Formal languages, Simulation algorithms, Chemical kinetics

Lily A. Chylek

Department of Chemistry and Chemical Biology, Cornell University, Ithaca, NY 14853, USA

Edward C. Stites

Clinical Translational Research Division, Translational Genomics Research Institute, Phoenix, AZ 85004, USA

Richard G. Posner

Clinical Translational Research Division, Translational Genomics Research Institute, Phoenix, AZ 85004, USA and Department of Biological Sciences, Northern Arizona University, Flagstaff, AZ 86011, USA

William S. Hlavacek

Theoretical Biology and Biophysics Group, Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA and Department of Biology, University of New Mexico, Albuquerque, NM 87131, USA

Abbreviations: BEM (bond electron matrix), BNGL (BioNetGen Language), ODE (ordinary differential equation), SBGN (Systems Biology Graphical Notation), SBML (Systems Biology Markup Language)

Introduction

An important aim of systems biology is to understand phenomena that arise from the interactions of the component parts of cellular regulatory systems [1], such as genes, proteins, and metabolites. Key components of many regulatory systems have been studied extensively in isolation, which remains a common approach for investigating cellular regulation. Synthesis of the knowledge gained from reductionist studies, and accompanying development of systems-level understanding, necessitates the use of computational models that can account for the complexity of cellular regulatory networks [2, 3, 4, 5, 6]. Models are useful because they can make testable predictions and elucidate the logical consequences of the assumptions upon which a model is based. Models can advance understanding in other ways [7], for example, by consolidating available knowledge, visualizing this knowledge to make it more accessible, and revealing knowledge gaps. For a model to be useful, it need not capture all known mechanistic details, but the level of detail included in a model should be suitable for the system of interest and the questions that a modeler intends to ask.

Here, we focus on cell signaling systems. These systems consist of interacting molecules that coordinate responses to changes in the environment (signals). Aspects of these responses may not always be possible to predict using intuition alone. Indeed, molecularly targeted therapies, such as RAF inhibitors for cancer treatment [8], may lead to unexpected and even harmful outcomes due to complex repercussions emanating from perturbed molecular states. To better understand how cell signaling systems process information and respond to stimuli, we need mathematical/computational models that capture the chemical kinetics of molecular interactions in these systems. These physical interactions have been found to be dynamic [9, 10], regulated (*viz.*, protein-protein interactions that are affected by post-translational modifications [11]), and mediated by modular components (*e.g.*, domains and linear motifs [12]). Thus, it seems worthwhile to develop models that can account for these mechanistic details.

However, mechanistic details of protein-protein interactions in cell signaling systems give rise to at least two significant challenges for modelers. The first challenge is size: a signaling system typically contains numerous proteins [13]. The second challenge is combinatorial complexity [14, 15]: a protein may participate in multiple interactions and undergo post-translational modifications at multiple sites. As a result, a large number of chemical species can potentially be populated. Traditional modeling approaches, such as those indicated in Fig. 1, are poorly suited to cope with combinatorial complexity because they require enumeration of every reachable species. An alternative approach more suited for modeling of cell signaling systems, and other types of biochemical systems, is that of rule-based modeling, which is distinguished from traditional modeling approaches in several ways. Here, we review key innovative features of the rule-based modeling approach. It is a method of systems biology that is likely to grow in importance in the future, in part because of the number of sophisticated software tools now available to support it. For example, see [16, 17, 18, 19, 20, 21, 22, 23, 24]. There is also a large body of knowledge available about a number of cell signaling systems and a need to formalize this knowledge.

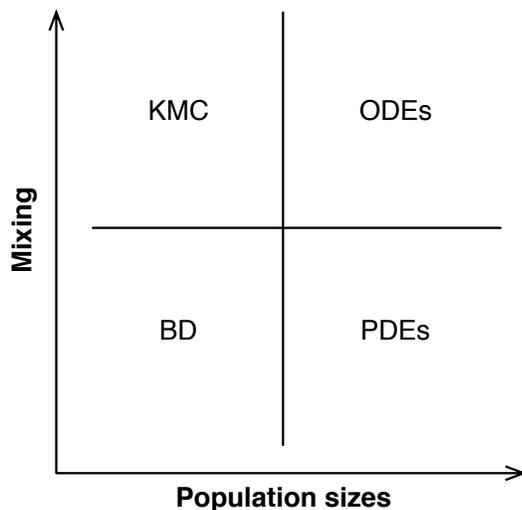


Fig. 1 Traditional modeling approaches. For well-mixed systems with large population sizes (i.e., populations that are large enough for concentrations to be continuous), ordinary differential equations (ODEs) can be used. For well mixed-systems with small population sizes, kinetic Monte Carlo (KMC) methods (e.g., Gillespie’s method) may be more appropriate. If the rate of mixing in a system is slower than the rate of chemical reactions, spatial effects can be expected to be important. In these cases, partial differential equations (PDEs) and Brownian dynamics (BD) can be used for systems with large and small population sizes, respectively. For each of these traditional modeling approaches, there is a corresponding rule-based approach. BioNetGen [16] and Smoldyn [17] can perform ODE-based and KMC-based simulations. Other simulators, including DYNSTOC [18], KaSim [19], NFsim [20], and RuleMonkey [21] can also perform KMC-based simulations. BNG@VCell [22] and Simmune [23, 24] can perform PDE-based simulations. Smoldyn [17] can also use BD to model diffusion of molecules.

To illustrate the size and combinatorial complexity of a well-studied cell signaling system, let us consider a subset of the proteins involved in signaling via the epidermal growth factor receptor (EGFR). Specifically, let us focus on 21 proteins included in the model of Chen et al. [25]; it is worth noting that the NetPath database lists over 400 proteins involved in EGFR signaling [13]. Based on information available in public databases, on average each of the 21 proteins contains 8.2 sites of phosphorylation [26], 2.6 domains [27], and 0.6 motifs [28], and has 6.5 interaction partners among the other 20 proteins [29]. These statistics are summarized in Fig. 2. Enumeration of every possible species of interest that could arise in this subsystem would be impractical, if not impossible, without the use of simplifying assumptions to reduce combinatorial complexity. For example, consider Gab1, Raf-1, and EGFR. According to Phospho.ELM, these proteins have 14, 21, and 35 sites of phosphorylation, respectively [26]. As a result, Gab1 has $2^{14} = 16,384$ possible phosphorylation states, Raf-1 has $2^{21} \approx 2 \times 10^6$ possible phosphorylation states, and EGFR has $2^{35} \approx 3.4 \times 10^{10}$ possible phosphorylation states.

The challenge of combinatorial complexity can be addressed using the rule-based modeling approach [30, 31, 32]. In this approach, proteins are represented as structured objects whose components can interact independently of one another unless otherwise specified. Contextual constraints on protein-protein interactions can be captured in rules, which include necessary and sufficient conditions for firing of reaction events. One can view reactants as satisfying conditions required at specific sites, as specified in rules. Fewer simplifying assumptions are typically required and a more

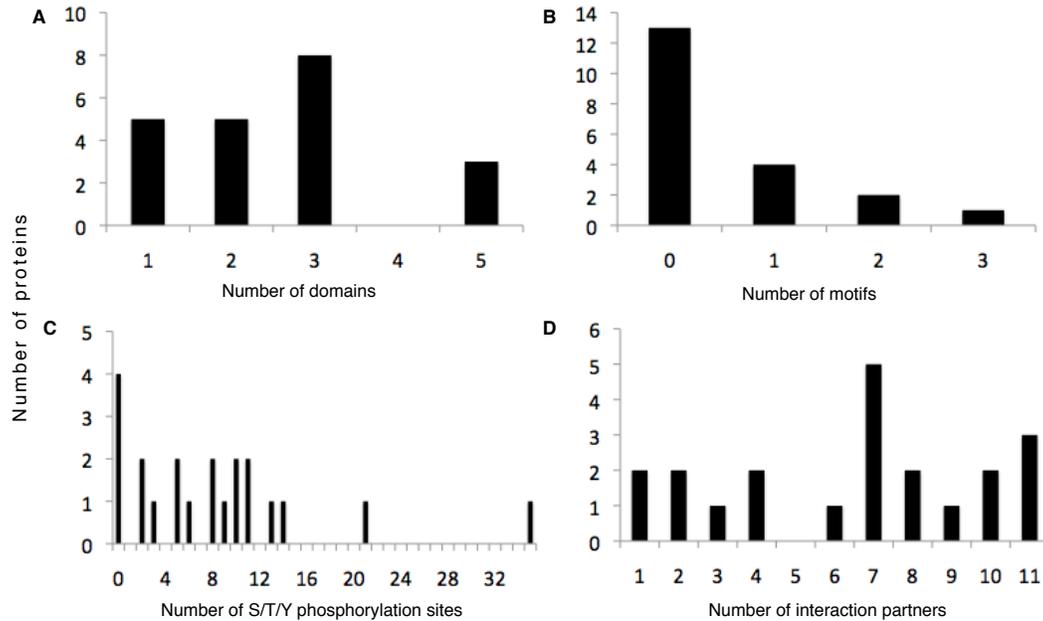


Fig. 2 Sites, modifications and interactions of proteins involved in EGFR signaling. Gene names of the proteins considered here are EGF, NRG1, EGFR, ERBB2, ERBB3, ERBB4, SHC1, GRB2, SOS1, GAB1, PIK3RI, PIK3CA, PDPK1, AKT1, KRAS, RASA1, RAF1, MAP2K1, MAPK1, PTEN, and PTPN11. **A.** Domains considered are those documented in the Pfam database [27]. **B.** Similarly, motifs were obtained from ELM [28]. **C** Phosphorylation sites were obtained from Phospho.ELM [26]. **D** Interaction partners were obtained from HPRD [29].

comprehensive picture of a signaling system can be developed that is more aligned with mechanistic understanding.

In this chapter, we discuss innovations of the rule-based modeling approach. The first innovation that we discuss is the use of rules, which builds on concepts that have proven useful in other fields. A second innovation is the use of formal languages to specify models, allowing models to be viewed as programs. A third innovation is network-free algorithms for stochastic simulation of agent-based models consistent with the law of mass action. These algorithms are needed for mechanistic modeling of cell signaling on a large scale.

1 Use of rules to represent molecular interactions in cellular and molecular biology

The network motifs (e.g., the writer, reader, eraser motif, which consists of tyrosine phosphorylation, SH2 domain binding, and dephosphorylation [33]) and subsystems that constitute a signaling system may each involve only a few different proteins. However, interactions among these proteins may give rise to far larger numbers of distinct chemical species through combinations of the different possible interactions and modifications [34, 14, 15, 35]. To capture these effects, a number of tools and

modeling frameworks have been developed that use rules to represent molecular interactions at the level of molecular components, or sites.

Among the first software tools developed for rule-based modeling of biological systems were OLIGO [37] and STOCHSIM [38, 39]. OLIGO is capable of generating reaction networks for assembly of oligomeric complexes, but does not capture regulation of interactions through post-translational modifications. This capability is provided in STOCHSIM, where proteins are represented as multi-state entities. A protein is encoded as a set of “flags” that represent binding or modification states. During a simulation, molecules are selected randomly and a list of rules is used to determine whether a reaction can occur between them (i.e., whether states can change). Although STOCHSIM can be used to effectively capture changes in state, it is poorly suited for explicitly tracking the connectivity of molecular complexes.

Another early approach, developed by Regev et al. [40], uses π -calculus to model a cell signaling system as a concurrent computational system. In this approach, molecules and sites are treated as parallel processes that can behave independently of one another, in accordance with a set of rules. Stochastic π -calculus [41] and tools implementing this method, such as BioSPI [42], BlenX [43], and SPiM [44, 45], enable simulation of biochemical kinetics. However, the use of π -calculus introduces artifacts from the study of concurrency, such as directionality of communication.

An early example of a non-trivial rule-based model is that of Goldstein et al. [46] and Faeder et al. [47]. This model is equivalent to 354 ODEs with 3,680 distinct right-hand-side terms, making it tedious to specify using traditional approaches. The model was used to investigate early events in signaling via the high-affinity receptor for IgE. The rule-based approach has since been applied

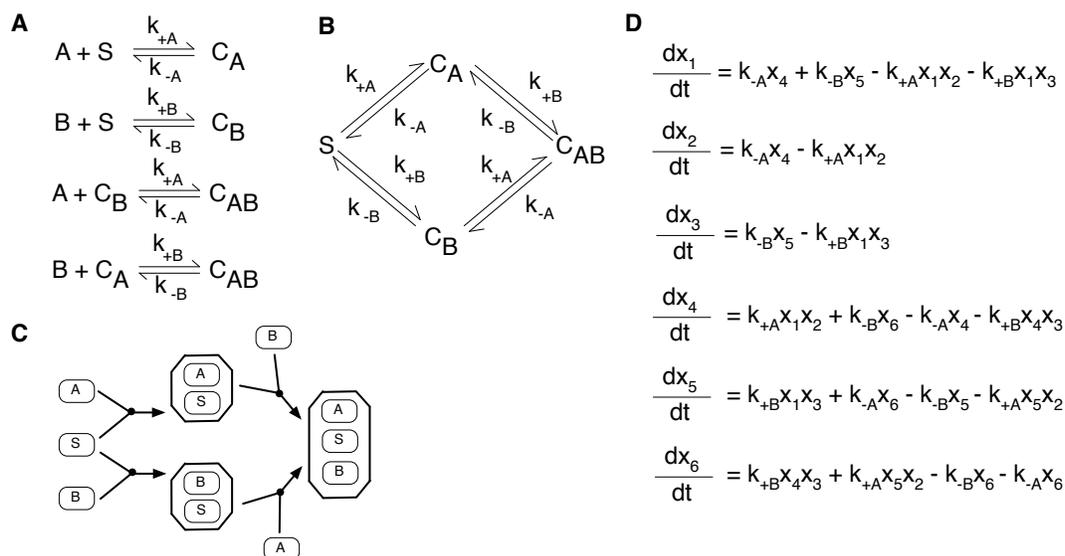


Fig. 3 A model of scaffold-ligand interactions traditionally formulated. Ligands A and B bind non-cooperatively to scaffold S . **A.** A list of reactions with associated rate constants for forward and reverse reactions. A , B , and S represent unbound proteins. C_A and C_B represent S bound to A and B , respectively. C_{AB} represents the ternary complex of S , A , and B . **B.** A reaction scheme, i.e., an organized layout of Panel A. **C.** An SBGN (Process Diagram) [36] representation of the model of Panels A and B. **D.** The model in the form of a system of ODEs: the variables x_1, \dots, x_6 represent the concentrations of S, A, B, C_A, C_B , and C_{AB} , respectively.

extensively to study immunoreceptor signaling [48, 49, 50, 51, 52, 53, 54, 55]. However, these and other applications are not the main subject of this chapter; instead, we focus on methodology.

To demonstrate the use of rules, let us consider a system in which a scaffold, S , may bind ligand A with a forward rate constant of k_{+A} and a reverse rate constant of k_{-A} . The scaffold S may also independently bind ligand B with a forward rate constant of k_{+B} and a reverse rate constant of k_{-B} . Thus, the system has six species: S , A , B , a complex of S and A , a complex of S and B , and a ternary complex of S , A , and B . Fig. 3 illustrates traditional formulations of a model of this system. Panels A and B show the reactions of the model as a list and as a reaction scheme, respectively. Panel C is a visualization of the model using the conventions of Systems Biology Graphical Notation (SBGN) [36]. Panel D shows the six ODEs of the model that follow from mass-action kinetics. The ODEs characterize the change with time of each of the six concentrations for a well-mixed reaction compartment and continuous population levels (i.e., large numbers of molecules).

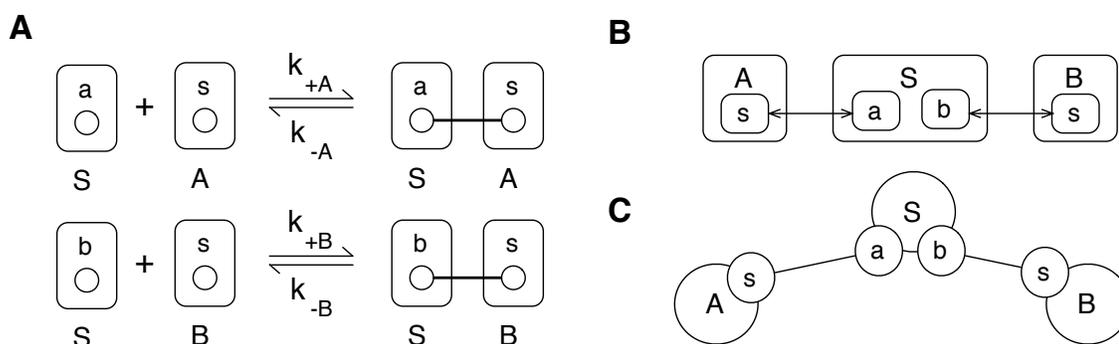


Fig. 4 A rule-based model of scaffold-ligand interactions. **A**. Two rules visualized using the graphical conventions of Faeder et al. [56]. Components not affecting an interaction are omitted from a rule. Proteins are represented as simple colored graphs. The “color” of a graph is the name of the protein that the graph represents. By convention, boxes enclose vertices of the same color. Bonds are represented as edges, which connect vertices that represent cognate binding sites. A BioNetGen Language (BNGL) encoding of the first rule is $S(a) + A(s) \leftrightarrow S(a!1).A(s!1) \text{ kpa, kma}$. A BNGL encoding of the second rule is $S(b) + B(s) \leftrightarrow S(b!1).B(s!1) \text{ kpb, kmb}$. **B**. The model visualized as an extended contact map [57]. Boxes represent proteins and components. A double-headed arrow represents a noncovalent bond. Contact maps can be generated using RuleBender [58]. **C**. An alternative rendering of a contact map, consistent with conventions of Danos et al. [31].

A rule-based formulation of the same model is illustrated graphically in Fig. 4. This model can be encoded in a number of rule-based modeling languages. As we will discuss, the most commonly used languages for rule-based modeling are BioNetGen Language (BNGL) [59, 16] and Kappa [60, 31, 61], and we will use shared conventions of these languages in our description of the model. The scaffold is represented as a structured object, S , with two components, a and b . These components are binding sites that recognize ligands A and B , respectively. Ligand A contains a component s that binds a in S . Similarly, ligand B contains a component s that binds b in S . Fig. 4A illustrates two rules that capture the interactions among these molecules. The first rule specifies the conditions necessary for S to bind A : S must have an unbound component a and A must have an unbound component s . We assume that the state of site b does not affect the interaction between S and A , so it is omitted from the rule. If b could affect the interaction between S and A (e.g., through an allosteric mechanism), it would be possible to express such an effect by appropriate modification of

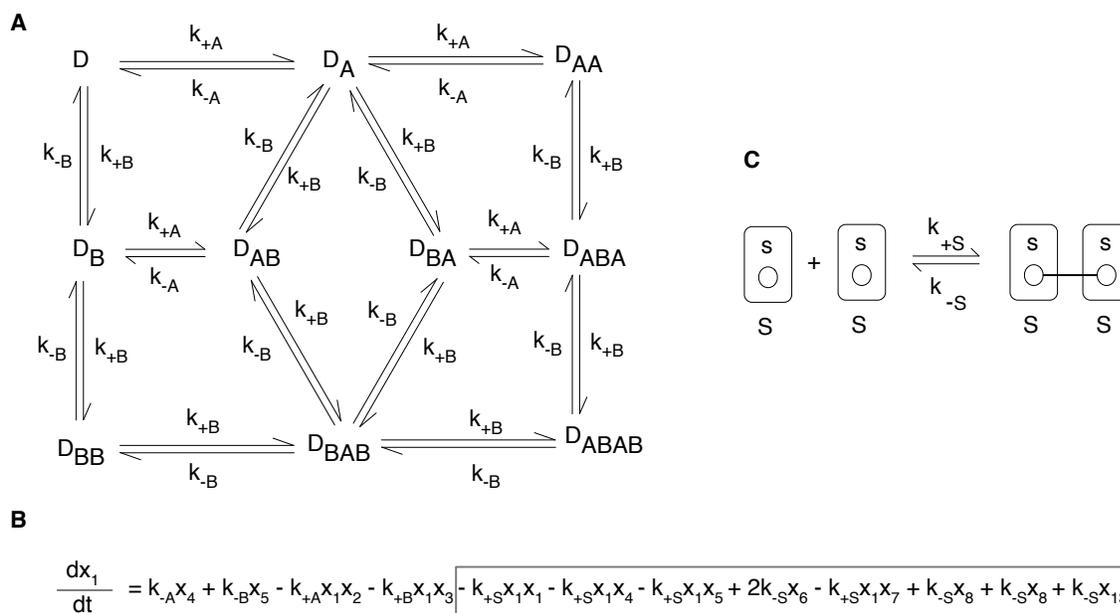


Fig. 5 Illustration of an extension of the scaffold model of Figs. 3 and 4 to allow the scaffold to dimerize. **A.** A partial reaction scheme showing the ten additional species that arise when scaffold dimerization is allowed. **B.** An example of an equation from the scaffold model of Fig. 1D that must be modified if scaffold dimerization is allowed. The terms that must be added to account for scaffold dimerization are highlighted. **C.** The rule that is added to the rule-based model of Fig. 2A to account for scaffold dimerization. A BNGL encoding of this rule is $S(s) + S(s) \xrightarrow{k_{+S}} S(s) - S(s)$. $S(s) - S(s)$ kps,kms

the rules that comprise the model. The second rule specifies the conditions necessary for S to bind B, which are similar to those of the first rule. These two rules represent the same set of interactions as the eight unidirectional reactions and the six ODEs shown in Fig. 3. Fig. 4B shows the model visualized as a contact map, which in general provides an illustration of all molecules, components, modifications (none in this model) and interactions that are included in a model. Fig. 4C is an alternative rendering of the contact map.

From the simple example given above, the benefits of the rule-based approach may not be evident. However, consider addition of one more interaction to the model: dimerization of the scaffold protein. This added interaction gives rise to ten additional species (Fig 5A). Thus, ten additional ODEs must be added to capture this one additional interaction. Furthermore, the original six equations must be modified to account for additional reactions that each of the original six species can now undergo. Fig. 5B shows a modified equation from the original model; the highlighted terms are added to account for interactions that become possible if the scaffold dimerizes. In contrast, scaffold dimerization can be incorporated into a rule-based model through single addition of the rule of Fig. 5C without modification of the rules of Fig. 4A. Thus, rule-based modeling is more extensible than traditional modeling for chemical kinetics. It is worth noting that the rule-based approach has been used to study scaffold effects in cell signaling [62, 63]. A complete specification of the rule-based model in which scaffold molecules may dimerize is provided in Fig. 6, wherein a BNGL [59, 16] encoding (Listing 1) and an equivalent Kappa [60, 31, 61] encoding (Listing 2) are given. Listing 1

can be used to generate a MATLAB (The MathWorks, Natick, MA) definition of a system of ODEs (i.e. a traditional model specification) by adding the command “`writeMfile()`” to the end of the listing (i.e., in the actions block of the code, which is not shown). Listing 1 can be processed by BioNetGen [59, 16] to generate an M-file consisting of 113 lines of code. The M-file defines a system of 16 ODEs with 60 different right-hand-side terms. In contrast, the model specification of Listing 1 in Fig. 6 consists of only three rules and three molecule type definitions (molecule type definitions are not shown in Fig. 6), as well as specifications of parameter values and initial conditions.

Listing 1

```
begin molecule types
S(s,a,b)
A(s)
B(s)
end molecule types
begin seed species
S(s,a,b) S_init
A(s) A_init
B(s) B_init
end seed species
begin reaction rules
S(a) + A(s) <-> S(a!1).A(s!1) kpa,kma
S(b) + B(s) <-> S(b!1).B(s!1) kpb,kmb
S(s) + S(s) <-> S(s!1).S(s!1) kps,kms
end reaction rule
```

Listing 2

```
%agent: S(s,a,b)
%agent: A(s)
%agent: B(s)
%init: 1e5 * S(s,a,b)
%init: 1e5 * A(s)
%init: 1e5 * B(s)
S(a),A(s) ->S(a!1),A(s!1) @0.1
S(a!1),A(s!1) ->S(a),A(s) @0.1
S(b),B(s) ->S(B!1),B(s!1) @0.1
S(B!1),B(s!1) ->S(b),B(s) @0.1
S(S),S(s) ->S(s!1),S(s!1) @0.1
S(s!1),S(s!1) ->S(S),S(s) @0.1
```

Fig. 6 The model composed of the rules illustrated in Fig. 4A and 5C encoded in two formal languages, the BioNetGen Language (BNGL) and Kappa. **Listing 1.** BNGL-encoded model specification. This excerpt from a BioNetGen input file illustrates the definition of molecule types, seed species (initial conditions for a simulation), and rules. Lines of code for setting parameter values and actions are not shown. The model specification can be simulated using different methods available in BioNetGen [59, 16] or other BNGL-compliant software tools [22, 18, 20, 21]. Note that the first rule corresponds to the top rule of Fig. 4A, the second rule corresponds to the bottom rule of Fig. 4A, and the third rule corresponds to the rule of Fig. 5C. **Listing 2.** Kappa-encoded model specification. This model specification, which is equivalent to Listing 1, can be processed by KaSim [61].

Although the rule-based modeling approach is a relatively recent development in biology, similar concepts have long been used in other fields. Below, we briefly discuss related approaches that have been developed for a variety of problems in physics, chemistry, and computer science. The success of this approach in other fields suggests that it will also be useful for studying the systems of cellular and molecular biology.

1.1 Precedents in physics

The Ising model was originally developed to study ferromagnetism: the emergence of a magnetic moment through alignment of atomic spin states. The model, which has a number of other applications, involves a lattice of sites, each of which has one of two states, e.g., spin-up or spin-down. The state or spin of a site can be reversed. The probability of a site’s spin reversing depends on the spin states of its neighbors. The Ising model can be simulated using a number of methods. In the classic

Metropolis method [64], a site (if spin is flipped) or pair of sites (if spins are exchanged) is first selected at random. For the purposes of this discussion, we assume that a single site is chosen. The probability of spin reversal is then computed based on temperature and the configuration of a sites neighbors. This probability is then compared to a random number. If the random number is less than or equal to the probability of flipping, the spin of the site is reversed and time is incremented. If the random number is greater than the probability of flipping, spin is reversed with a probability equal to the ratio of the probabilities of the initial and final states. If spin is not reversed, a null event occurs and time is incremented. A drawback of this method is that a high frequency of null events causes simulation to slow significantly.

A			B				
Class	Spin	Number of spin-up nearest neighbors	5	7	3	9	8
1	Up	4	6	3	3	8	4
2	Up	3	3	2	8	8	2
3	Up	2	7	4	7	4	3
4	Up	1	9	8	4	8	9
5	Up	0					
6	Down	4					
7	Down	3					
8	Down	2					
9	Down	1					
10	Down	0					

Fig. 7 Classes used in KMC simulation of the Ising model. **A.** Scheme used for classification of lattice sites. A site is classified based on its spin and the number of its nearest neighbors that are spin-up. **B.** In this example, white squares are used for spin-up sites and dark squares are used for spin-down sites. Class numbers are shown on squares in accordance with the scheme of Panel A. The lattice is assumed to have periodic boundary conditions, i.e., the lower boundary is replicated above the upper boundary and the left boundary is replicated after the right boundary.

An alternative approach is the n -fold way [65], a kinetic Monte Carlo (KMC) method [66], in which null events are avoided. In this algorithm, a site is classified based on its spin state and the spin states of its nearest neighbors. A classification scheme for a square lattice is shown in Fig. 7A. Use of this scheme is illustrated in Fig. 7B; white squares represent spin-up sites, gray squares represent spin-down sites, and the number of a square indicates its class. Rather than selecting a site randomly, the probability of a site being selected is related to the probability of its spin flipping. Once a site is selected, its spin is flipped immediately. Thus, null events do not arise. The n -fold way for the example of Fig. 7 consists of the following steps:

1. Assign each site to one of ten possible classes.
2. Choose a class $r \in [1, 2, \dots, 10]$. A class is chosen by first calculating cumulative rates Q_1, Q_2, \dots, Q_{10} , where

$$Q_r = \sum_{j=1}^r n_j P_j \quad (1)$$

In the above expression, n_j is the number of sites in class j and P_j is the probability of spin reversal for a site in class j . Then, a random number R_1 , uniformly distributed between 0 and Q_{10} , is chosen, and a class r is chosen such that $Q_{r-1} \leq R_1 < Q_r$.

3. Randomly choose a site i within r .
4. Flip the spin of site i .
5. Update classes and the rates Q_1, Q_2, \dots, Q_{10} based on the new configuration of the lattice.
6. Increment time. The time step is calculated as

$$\Delta t = \frac{\tau \log_2 R_2}{Q_{10}} \quad (2)$$

where R_2 is a random number and τ is the expectation value (i.e., the average time per spin flip). Recall that Q_{10} is the overall rate of spin flipping.

This procedure applied to stochastic simulation of chemical reaction systems is known as Gillespie’s method [67, 68, 69], which is discussed below. The similarity between the n -fold way and the rule-based modeling approach lies in the use of classes. In the n -fold way, a class defines a set of lattice sites that have a particular spin state and configuration of neighbors. Sites within a class all have the same probability of undergoing a transition. Similarly, a rule defines a class of reactions whose reactants share certain local component properties and reactions that are defined by a rule are taken to have the same kinetic parameters.

1.2 Precedents in chemistry

In modeling chemical reactions, matrices and matrix operations can serve as useful abstractions for representing molecular structures and functional group transformations. Ugi and co-workers developed a formalism in which of a bond electron matrix (BEM) is used to represent the atoms present in a molecule (or set of molecules) and the sharing of electrons between them. In this formalism, a chemical reaction is viewed as converting a BEM into an isomeric BEM by redistributing valence electrons. A BEM for n atoms contains n rows and columns. The i th row and column correspond to the i th atom of the molecule or set of molecules. The matrix entry b_{ij} is the number of bonds between atoms i and j and the diagonal matrix entry b_{ii} is the number of free valence electrons of atom i [70]. (When applied to reactions on surfaces, b_{ii} can also represent the number of electrons backdonated to the adsorbate.) Electrons are redistributed (i.e., a chemical reaction is executed) by addition of a reaction matrix R to a reactant matrix B . An entry in a reaction matrix corresponds to the number of bonds formed (positive numbers) or broken (negative numbers) between atoms as a result of a reaction. The matrix $E \equiv B + R$ represents the product molecule(s) of a reaction. The BEM formalism can be used to generate reaction networks and elucidate possible synthetic routes between reactant and target molecules [71, 72, 73, 74]. This method has also been used for time-scale analysis of rule-based models in which reactions within the same class have different kinetic parameters [75].

BEMs have been applied by Broadbelt and co-workers to investigate reaction mechanisms for heterogeneous catalytic chemistry [76] and novel metabolites and pathways in metabolic networks [77, 78]. The assumption underlying the latter is that the large number of reactions found in a metabolic network can be represented by a smaller number of rules for common functional group

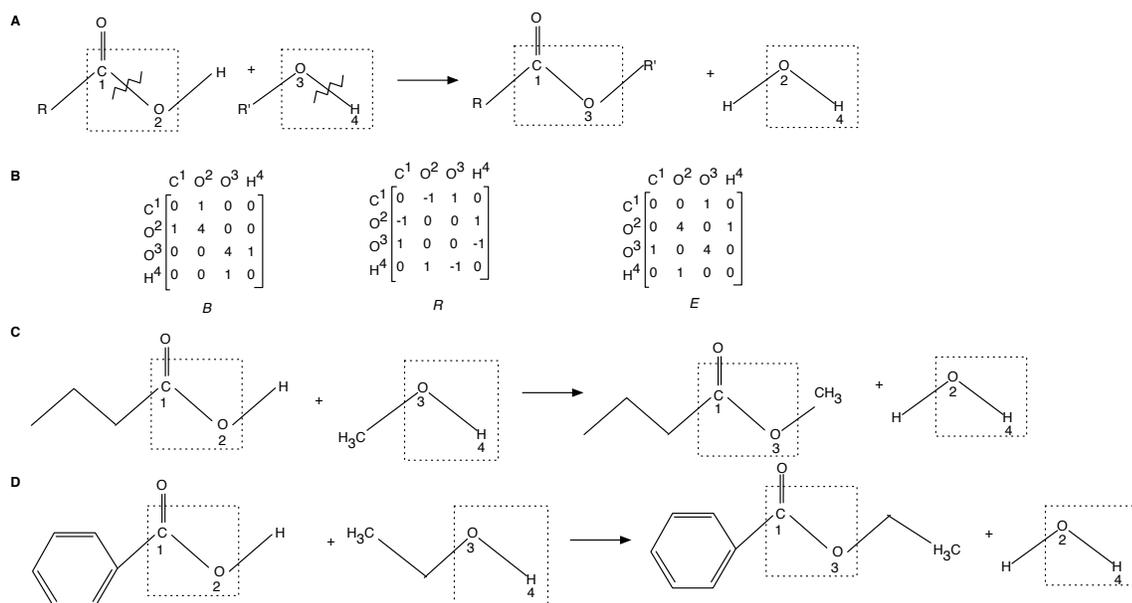


Fig. 8 Bond electron matrices and matrix operations can be used to model functional group transformations in organic chemistry. **A.** A general Fischer esterification reaction between a carboxylic acid and an alcohol. A box is placed around functional groups that participate in the reaction, and a jagged line is used to mark the bonds that are broken. **B.** *B* is a bond electron matrix for the reactants. Rows and columns are labeled to correspond to labeling of atoms in Panel A. *R* is a reaction matrix showing the bonds that are broken and formed as a result of esterification. *E* is a bond electron matrix for the products. **C, D.** Two instances of rule application. Functional groups are enclosed within a box and atoms are numbered to correspond to panels A and B.

transformations in metabolism [77, 79]. Functional groups can be encoded as BEMs and associated with reaction matrices. An input molecule can also be encoded as a BEM and compared to the BEM of a functional group to determine whether the molecule contains the functional group necessary to undergo a reaction. If so, a reaction matrix is added to the appropriate part of the reactant matrix to yield a matrix for a product molecule or set of molecules. If the product is a chemical species that has not yet been generated, it is evaluated to determine whether it can undergo further reactions. A maximum number of generations can be specified as a stopping criterion. In this way, a set of rules can be identified that can generate potentially novel reaction paths from reactants to products. This approach has been complemented by thermodynamic studies to evaluate the feasibility of possible reaction paths [78].

An example of the use of BEMs to model a chemical reaction is shown in Fig. 8. Panel A shows a rule that specifies the functional groups involved in an esterification reaction. Panel B shows the same functional groups in the form of BEMs, with atoms numbered to correspond to Panel A. Panels C and D show two instances of the rule acting on specific molecules.

1.3 Precedents in computer science

A concurrent computational system is one in which multiple processes are executed in parallel and can potentially influence each other. Interaction among processes can lead to many possible outcomes. The complexity of concurrent systems necessitates a language that can be used to analyze and reason about a system's behavior. This need is addressed by process algebras [80, 81]. Here, we focus on π -calculus, a process algebra that has been intensely studied in computer science and that has also been applied to model biological systems, as noted earlier. A notable feature of π -calculus is that it allows explicit representation of communication channels and allows system components to be modeled independently.

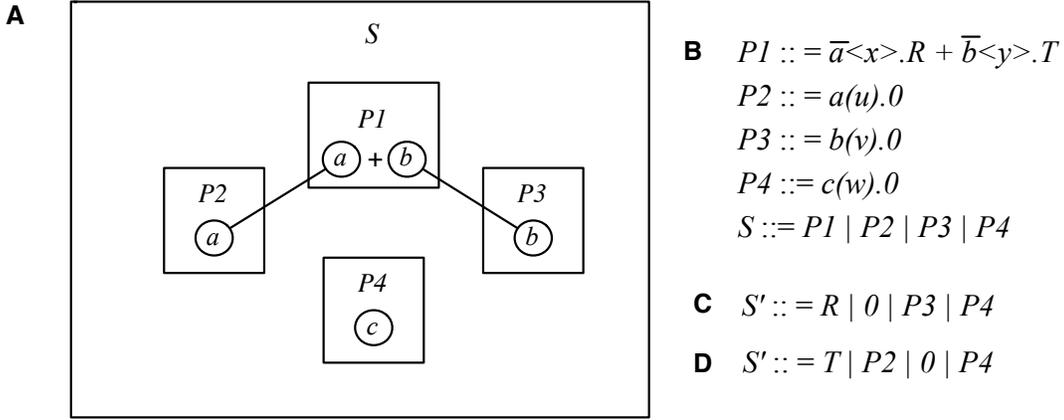


Fig. 9 The process algebra π -calculus is used to model concurrent computational systems. **A.** System S contains processes $P1$, $P2$, $P3$, and $P4$. $P1$ can communicate with $P2$ using channel a . $P1$ can communicate with $P3$ using channel b . $P4$ can receive information along channel c ; however, there is no complementary channel in S . **B.** Representation of the system in π -calculus. **C.** The system that S reduces to if $P1$ sends a message on channel a . **D.** The system that S reduces to if $P1$ sends a message on channel b .

An example of the use of π -calculus is shown in Fig. 9. In π -calculus, “+” designates choice, “.” designates sequence, and “|” designates processes executed in parallel. The symbol 0 designates an inert process (i.e., a process that does nothing further). A process can contain one or more channels, which can be used to communicate with other processes. Channels that can communicate with one another are referred to as complementary channels. Complementary channels share the same name, and prefixing conventions are used to distinguish an input channel from an output channel. For example, $\bar{a}\langle x \rangle$ is an output channel named a that sends a piece of information named x . A complementary input channel can be designated $a(u)$. When $a(u)$ receives information, the name received (e.g., x) becomes bound to u .

In the example of Fig. 9, $P1$ is a process that can send x on channel a . $P1$ then behaves as R . Alternatively, $P1$ can send y on channel b and then behave as T . In the first scenario, $P1$ becomes R , and $P2$ uses channel a to receive x from $P1$. The message x becomes bound to u . Then, $P2$ becomes 0 . The processes $P3$ and $P4$ are unaffected by this communication event. As a result, $P1|P2|P3|P4$ becomes $R|0|P3|P4$. Similarly, if $P1$ chooses the second option, the system becomes $T|P2|0|P4$.

The similarity between π -calculus in computer science and rule-based modeling approaches in systems biology lies in modularity. In a rule-based model, one may specify an interaction using only the sites that participate in the interaction. In π -calculus, one may likewise specify the effect of communication using expressions that only include the relevant (sub)processes and channels. For example, in the system of Fig. 9, communication between $P1$ and $P2$ or $P3$ is expressed without the inclusion of $P4$, which does not communicate with the other processes. Rule-based modeling approaches and process algebras share context-free properties, meaning that context can be omitted from a rule. Omitted contexts have no effect on the transformation specified in a rule, so that the rule can be applied in multiple contexts that need not be specified by the modeler. However, in some cases it is necessary for rule application to be restricted by context (e.g., when a reaction can only occur intramolecularly). In these cases, features of rule-based modeling languages, such as the dot-plus notation of BNGL (see Section 2), can be used to impose contextual constraints. Different but functionally equivalent notation is available in Kappa.

2 Models as programs

A model can be formalized using mathematical expressions. A different approach is to formalize a model as an executable program, which can potentially facilitate analysis [61], extensibility [82], and high-level abstractions [83, 84, 85]. A number of languages for modeling biological systems have been developed, including languages designed for specification of rule-based models. As we will discuss below, BNGL [59, 16] and Kappa [60, 61] are the most widely used rule-based modeling languages. A BNGL-encoded model and an equivalent Kappa-encoded model can be found in Listings 1 and 2 of Fig. 6, respectively.

A model, once specified using rules, may be simulated in a number of different ways without modification of the model specification *per se*. The model of Listing 1 can be simulated deterministically with the command `simulate_ode`, or stochastically with `simulate_ssa`. (For a complete description of BNGL syntax, see Faeder et al. [16].) Thus, model specification is separated from simulation. For an example of a rule-based model simulated in multiple ways, see Lipniacki et al. [86].

Methods for simulating rule-based models include generate-first, on-the-fly, and network-free methods. In generate-first methods, rules are iteratively applied to a set of species to generate new reactions and new species. This process continues until the full network is generated (i.e., no new species are generated) or until a stopping criterion is satisfied [87]. The dynamics of the network can then be simulated through numerical integration of a corresponding system of ODEs or a stochastic simulation algorithm. In on-the-fly methods, a network is generated as simulation progresses rather than before simulation begins [87, 88]. When a species is first populated, rules are applied to it, and new reactions and species may be generated. This approach can be useful if a set of rules implies a large number of possible species, some of which might never become populated. However, this method still relies on a computer's memory to store the network, which can be a limitation. The step of network generation is avoided entirely with network-free methods, which are discussed in more detail in the next section. In short, in a network-free method, all components of a system are tracked individually and rules are used directly to advance the state of a system by modifying states of components. Thus, network-free methods are particle- or agent-based.

A number of software tools compatible with BNGL and/or Kappa implement the simulation methods described above, in addition to providing other capabilities. These tools are listed in Table

1. Other languages that may be used to specify rule-based models include cBNGL, a form of BNGL that allows for explicit representation of compartments [89]; ML-rules, designed for multi-level rule-based modeling [90]; and SBML-multi, which is in development. See the Systems Biology Markup Language (SBML) website (<http://sbml.org>).

Tool	Language	Reference
BioNetGen	BNGL	[59, 87, 16]
BNG@VCell	BNGL	[22]
little b	little b	[91]
Smoldyn/libMolecularizer	BNGL/Kappa-like	[88, 17]
SSC	SSC	[92]
DYNSTOC	BNGL	[18]
NFsim	BNGL	[93, 20]
RuleMonkey	BNGL	[21]
KaSim	Kappa	[61, 19]
SRSim	BNGL	[94]
RuleBender	BNGL	[58]
RuleStudio	Kappa	[19]
RuleBase	BNGL and Kappa	[95]
GetBonNie	BNGL	[96]
BioLab	BNGL	[97]
complx	Kappa	[19]
PySB	BNGL and Kappa	[85]

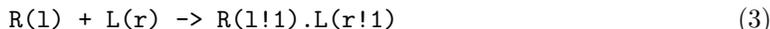
Table 1 Software tools that use BNGL and/or Kappa. Capabilities: BioNetGen is capable of network generation, ODE-based simulation, and generate-first and on-the-fly stochastic simulation. BNG@VCell has the additional capability of PDE-based simulation. The little b environment uses BioNetGen to perform network generation. Smoldyn/libMolecularizer and SSC can perform particle-based reaction diffusion calculations. BioNetGen can convert BNGL-specified rules into SSC format. DYNSTOC, NFsim, and RuleMonkey perform network-free simulation. KaSim performs network-free and ODE-based simulations. SRSim combines rule-based modeling with atomistic modeling (i.e., molecular dynamics simulation). RuleBender and RuleStudio are modeling interfaces, and RuleBender provides visualization capabilities. RuleBase and GetBonnie are model databases. BioLab is a model-checking tool, complex is a tool for static analysis, and PySB is tool for model building and analysis. Other software tools for rule-based modeling that do not use BNGL or Kappa include ALC [98], ANC [99], BIOCHAM [100], BioSPI [40], BlenX4Bio [101], CplexA [102], Meredys [103], ML-Rules [90], Molecularizer [88], Pathway Logic Assistant [104] PottersWheel [105], Simmune [23, 24], and StochSim [39].

BNGL and Kappa are closely related but differ in several details. One difference is the treatment of indistinguishable sites. In BNGL, a molecule is allowed to have two or more sites that have the same name. Such sites are taken to be indistinguishable. This capability is useful for molecules such as an IgG or IgE antibody, which contains two antigen-combining sites that are essentially identical. A bivalent antibody can be captured in BNGL with a molecule type definition such as $\text{IgE}(\text{Ag}, \text{Ag})$. In contrast, Kappa requires that every site have a unique name. Thus, the same molecule would necessarily require a definition of the form $\text{IgE}(\text{Ag1}, \text{Ag2})$.

Reaction rules in BNGL constrain the molecularity of reactions using “dot-plus” notation. This notation does not exist in Kappa; however, equivalent distinctions can now be made through other conventions [106].

The dot-plus notation is used to distinguish molecules that are part of the same chemical species (i.e., molecules that are directly or indirectly connected) from molecules that are part of separate species (i.e., not connected). For example, the following rule states that a bond forms between

molecules L and R.



The “+” sign specifies that the two reactant sites must be part of separate species for the rule to be applied. Thus, the rule defines only bimolecular association reactions. In general, the molecularity of a reaction is $1 + p$, where p is the number of “+” signs on the left-hand-side of a rule. In contrast, the following rule states that a bond forms between molecules L and R only when they are part of the same species.



The absence of a “+” sign is an application condition of the rule, which indicates that the rule generates only unimolecular reactions. This rule defines reactions that form intramolecular bonds.

The dot-plus notation of BNGL allows a modeler to not only impose molecularity constraints but also specify that a pair of molecules are connected without explicitly specifying connectivity. For example, to obtain the number of complexes that contain two receptors, one may specify an observable $R() \cdot R()$, which encompasses all complexes that contain at least two receptors.

3 Agent-based modeling consistent with the law of mass action

Traditional models are usually simulated via population-based methods, which require explicit tracking of all potentially populated chemical species. A rule-based model can also be simulated with population-based methods; however, combinatorial complexity can give rise to a large number of species, which makes the approach impractical or, in some cases, impossible. An alternative method is network-free simulation. Algorithms for network-free simulation are agent-based simulation protocols consistent with the law of mass action. Agent-based models are used in a variety of fields [107], and most algorithms for agent-based simulation are not guided by physicochemical principles. Thus, the innovation of network-free methods is that agents behave according to rules that recapitulate chemical kinetics.

To illustrate agent-based simulation of a rule-based model, let us consider a model of a bivalent ligand and a bivalent cell-surface receptor (Fig. 10). The ligand contains two identical, independent sites that can bind receptors. The receptor contains two identical, independent sites that can bind ligands. Interactions between ligands and receptors can give rise to chains (i.e., linear aggregates) and rings (i.e., cyclic aggregates). The two molecule type definitions and three rules that form this model are shown in Figs. 10A and 10B, respectively.

The rate for a free ligand binding a free receptor site (Fig. 10B, Rule 1) is given by the following equation [108]:

$$\eta_1 = 4k_{+1}F_L(N_R - N_B) \quad (5)$$

where k_{+1} is the forward rate constant associated with the rule, F_L is the number of free ligands, N_R is the number of receptors, and N_B is the number of bonds. The statistical factor of four arises from the two identical binding sites per receptor and two identical binding sites per ligand.

The rate of dissociation of ligand from receptor (Fig. 10B, Rules 1, 2, and 3), including breaking of a cyclic aggregate, is proportional to the number of ligand-receptor bonds [108, 109]:

$$\eta_{1r} = k_{\text{off}}N_B \quad (6)$$

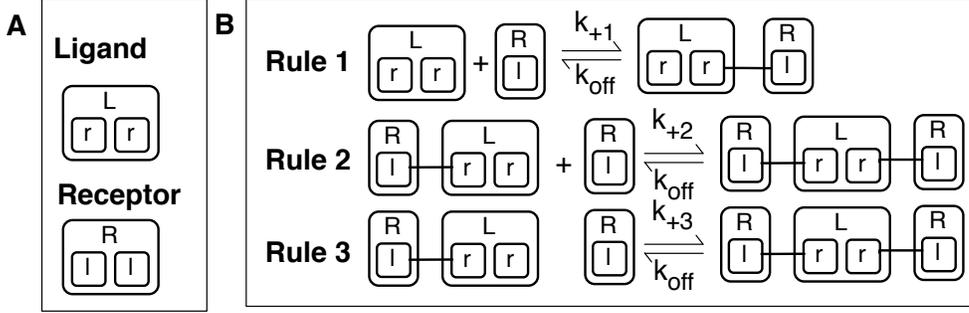


Fig. 10 A rule-based model for multivalent ligand-receptor binding. **A.** Molecule type definitions for a bivalent ligand and bivalent receptor. **B.** Rules for interactions between a bivalent ligand and a bivalent receptor. Note that the receptor **R** contains two identical sites (Panel A), but only one site is shown in Rules 1–3 (in accordance with the conventions of Faeder et al. [56]) because we assume that the bound state of the second site does not affect interactions represented by these rules. Rule 1 characterizes binding of a free ligand to a receptor. Rule 2 characterizes binding of a tethered ligand to a second receptor, thereby cross-linking a pair of receptors. Rule 3 characterizes ring closure and opening. The reactant sites of Rule 3 are required to be (indirectly) connected (i.e., they must exist within the same complex). Rules 2 and 3 differ on their left-hand sides. The plus sign in Rule 2 indicates that the rule defines reactions with molecularity of 2, whereas the absence of a plus sign in Rule 3 indicates that the rule defines reactions with molecularity of 1.

We assume that a single dissociation rate constant, k_{off} , applies for all dissociation reactions.

The rate for a tethered ligand binding a receptor site that is not part of the same complex as the ligand (Fig. 10B, Rule 2) is given by the following equation [108]:

$$\eta_2 = k_{+2} \sum_{i=1}^{N_A} l_i (v_r N_R - N_B - r_i) \quad (7)$$

where N_A is the number of aggregates, l_i is the number of free ligand sites in the i th aggregate, and r_i is the number of free receptor sites in the i th aggregate.

The rate for ring closure (Fig. 10B, Rule 3) is given by the following equation:

$$\eta_3 = k_{+3} \sum_{i=1}^{N_A} \frac{l_i r_i}{L_i} \quad (8)$$

The rate constant for ring closure can be taken to be inversely proportional to the length of a chain [109]. Here, we assume that rings of size one (i.e., containing one ligand and one receptor) are prohibited, and that k_{+3} is the rate constant for closure of a chain that yields a ring of size two (i.e., containing two ligands and two receptors). L_i is proportional to the length of a chain. For a ring of size two, we take $L_i = 1$.

Information about rates is used by network-free simulation algorithms to select rules to apply. Sequential rule application produces a system trajectory. Fig. 11 shows an example of a trajectory in a network-free simulation of the model of Fig. 10. It is worth noting that Rule 1 is executed twice, in Panel B and Panel F. The two instances represent different reactions, but both reactions are captured by the same rule. A rule can be viewed as a generalized reaction, and algorithms for

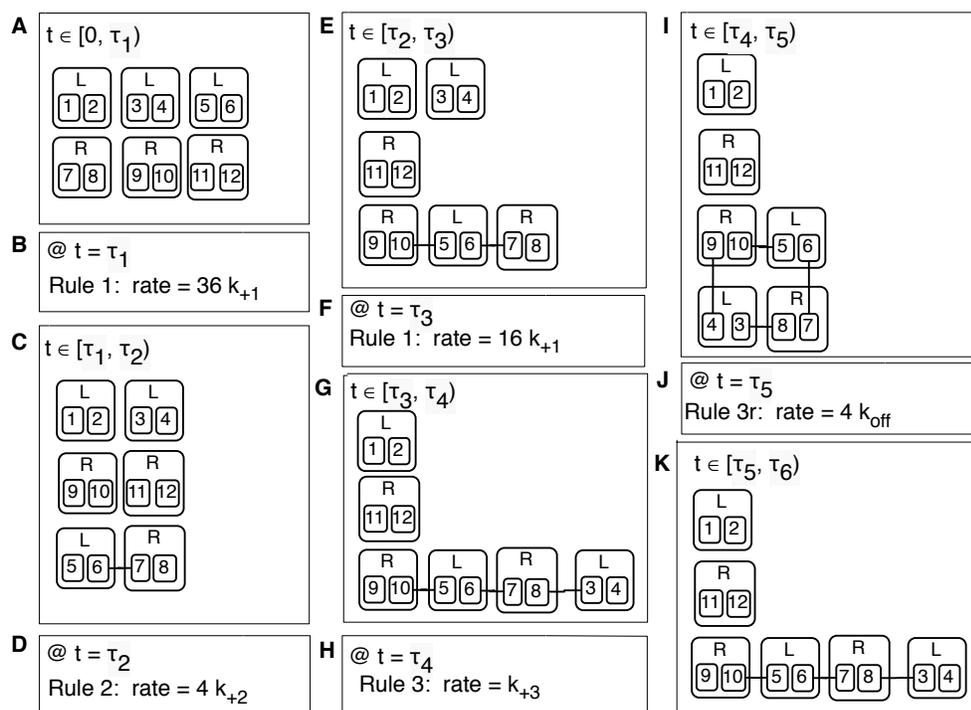


Fig. 11 An example trajectory in agent-based simulation of the model of Fig. 10. Each site is labeled with a unique number because sites are tracked individually. Reaction rates are calculated using Eqns. 5-8. **A.** The initial state of the system contains three free ligands and three free receptors. **B.** At time τ_1 , Rule 1 is fired and a ligand binds a receptor. **C.** Between times τ_1 and τ_2 , the system contains two free receptors, two free ligands, and a ligand-receptor complex. **D.** At time τ_2 , Rule 2 is fired and a pair of receptors are cross-linked. **E.** Between times τ_2 and τ_3 , the system contains two free ligands, one free receptor, and a complex of two receptors and one ligand. **F.** At time τ_3 , Rule 1 is fired and a ligand binds a receptor. This reaction differs from the instance of Rule 1 in Panel B because the receptor that undergoes reaction is already part of a complex. **G.** Between times τ_3 and τ_4 , the system contains one free ligand, one free receptor, and a chain of two ligands and two receptors. **H.** At time τ_4 , Rule 3 is fired and a ring or cyclic aggregate forms. **I.** Between times τ_4 and τ_5 , the system contains one free ligand, one free receptor, and a cyclic aggregate of two ligands and two receptors. **J.** At time τ_5 , the reverse of Rule 3 is fired and the cyclic aggregate is transformed into a chain or linear aggregate. **K.** Between times τ_5 and τ_6 , the system contains one free ligand, one free receptor, and a chain of two ligands and two receptors. This state is identical to the state of Panel G.

network-free simulation can be viewed as generalizations of Gillespie's method, which we briefly present below before reviewing different network-free algorithms reported in the literature.

3.1 Gillespie's method

Gillespie's method [67, 68, 69], a method for stochastic simulation of chemical reaction systems, is useful because it takes into account two facts that a deterministic method is not designed to capture: a system contains a whole number of molecules, and reactions among molecules are subject

to randomness. These qualities are likely to be important in systems where population sizes are small.

Gillespie’s method consists of essentially the same steps as the n -fold way, described above. Both methods belong to the class of kinetic Monte Carlo methods [66]. An implicit assumption of Gillespie’s method is the assumption that an explicit list of the reactions that can occur in a system is available. A simulation proceeds as follows. First, initial population sizes and reaction rates are calculated. Reaction rates are calculated based on rate constants and numbers of reactant species. Rates are used to select the next event time and the next reaction. A reaction is then fired. Populations and rates are updated for the new state of the system, and simulation continues until a stopping criterion is satisfied. Variations of this method have been developed to increase its speed. For example, efficiency of simulation can be improved through use of a reaction classification scheme, as demonstrated in the method of Blue et al. [110] or Gibson and Bruck [111]. More recently, the method of Slepoy et al. [112] groups together reactions that share similar rates.

Reaction classification is an inherent feature of the rule-based approach: as a coarse-graining assumption, reactions implied by the same rule are assigned the same rate law. Thus, Gillespie’s method is well-suited for simulation of rule-based models, if rates of all reactions implied by a rule can be calculated without explicitly deriving the reactions. These calculations are performed in network-free methods, of which there are multiple variants.

3.2 Algorithms for network-free simulation

Gillespie’s method has been generalized for simulation of rule-based models. These simulation methods are termed “network-free” because rules are used directly to advance the state of a system, thereby avoiding network generation. Currently, four related algorithms have been described for network-free simulation. These algorithms are summarized in Fig. 12. A main point of difference between them lies in the handling of non-local site properties. An example of a non-local site property is connectivity. The non-local environment of two sites must be examined if they are connected indirectly. Determining if two sites are indirectly connected is important for enforcing rule application conditions that place constraints on molecularity of rule-defined reactions. In general, non-local properties are more difficult to evaluate than local properties (e.g., whether a site is bound or free).

In the method of Danos et al. [31], rates are assumed to depend on local properties only. A waiting time is determined, a rule is selected, and sites are selected for rule application. The system is updated, rates and populations are recalculated, time is incremented, and simulation continues. The method of Yang et al. [93] performs the same calculation of rates as the method of Danos et al. [31]. However, after sites are selected based on local properties, non-local properties are checked. If a site is found to lack permissive non-local properties, it is rejected and a null event occurs. The method of Colvin et al. [21] avoids the rejection step by calculating rates exactly (i.e., with consideration of both local and non-local properties) before selecting rules and sites. Lastly, the method of Colvin et al. [18] differs from the other algorithms in a number of ways. The time step is fixed, and sites are selected before rules are selected. This method yields results consistent with the other methods as long as the time step is below a certain threshold, which is checked during simulation. The performance of tools implementing these methods have been compared, to a limited extent [21, 20, 108].

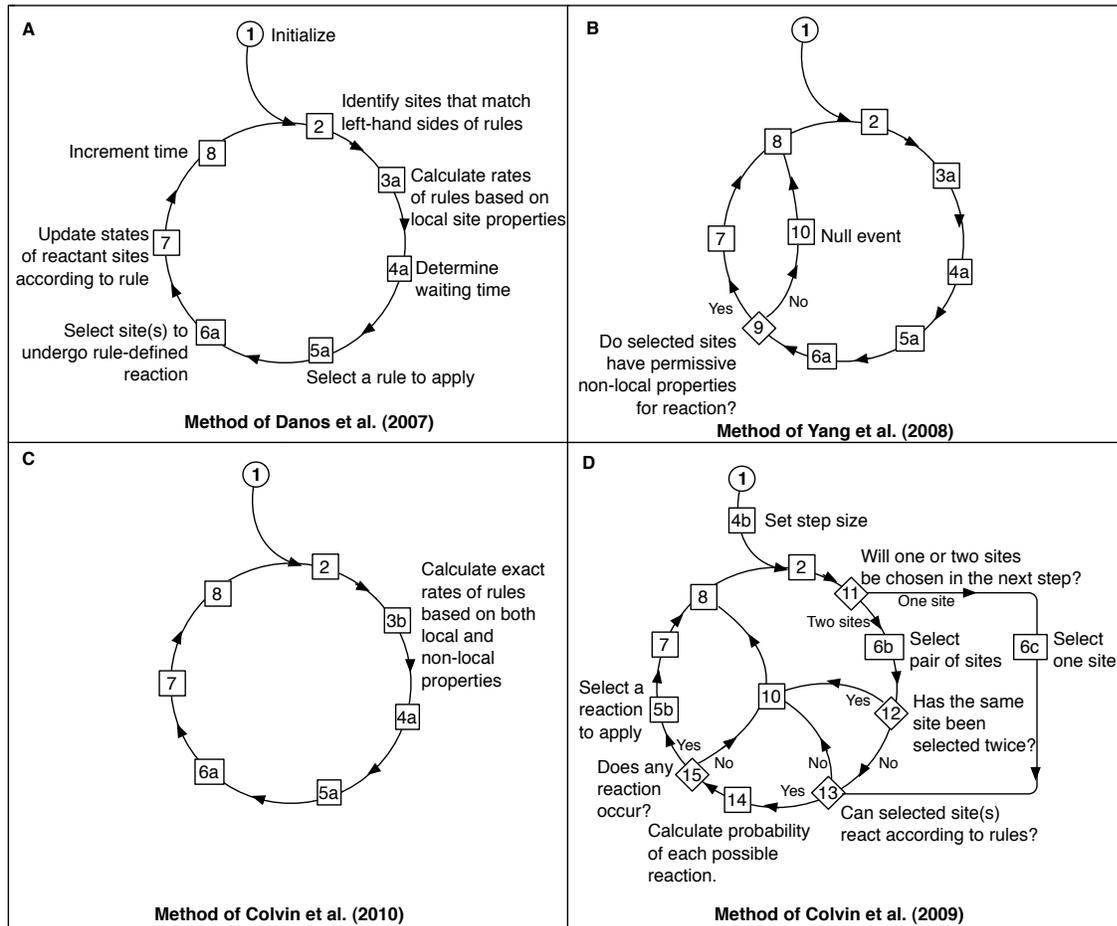


Fig. 12 A comparison of algorithms for network-free simulation. **A.** The method of Danos et al. [113] can be applied if rule application depends on local context only. **B.** The method of Yang et al. [93] introduces a rejection step to account for non-local site properties. **C.** The method of Colvin et al. [21] calculates exact rule rates considering both local and non-local site properties. Thus, it has no rejection step. **D.** The method of Colvin et al. [18] is a generalization of the STOCHSIM method [38, 114], which has a number of distinguishing features, including a fixed time step and reversal of the site and rule selection steps. However, the method yields results consistent with other methods, as long as step size is below a threshold.

4 Outlook and closing remarks

Our accumulated knowledge about the mechanisms of cell signaling motivates the development of models that can capture these details. Current experimental capabilities that allow us to characterize the functional roles of specific protein sites and to monitor the dynamics of protein-protein interactions [115, 116, 117, 118, 119, 120, 121] makes the development of complementary modeling methods especially timely. A method that addresses this need is the rule-based modeling approach. By using rules to represent interactions, a modeler can avoid enumerating the reachable chemical

species in a system, which is required in traditional approaches and which is a severe limitation given the typical size and combinatorial complexity of cell signaling (sub)systems. The rule-based approach allows models to be specified compactly, with simplifying assumptions that are more transparent and arguably less *ad hoc* than traditional modeling approaches [122]. With recent advances in simulation methodology, simulation of models that imply large reaction networks has become feasible. As a result, it is now possible to develop models that capture site-specific details of a large number of protein-protein interactions.

These capabilities are relevant for the study and, potentially, manipulation of cell signaling mechanisms. For example, different residues in the same protein may have different kinetics of phosphorylation, and each phosphorylated residue may regulate a distinct set of interactions (for example, see Houtman et al. [123]). As a result, perturbations that affect phosphorylation kinetics of specific sites (e.g., therapeutic kinase inhibitors, such as imatinib [124]) may be difficult to analyze without a model in which individual sites of phosphorylation are distinguished. However, traditional modeling approaches often necessitate a “virtual phosphorylation site” assumption [125], meaning that multiple sites are lumped together as a single, virtual site of phosphorylation. Roles for individual sites are not distinguished. This assumption can be lifted in a rule-based model more easily than in a traditional model.

Rule-based models can be specified using formal domain-specific languages (i.e., programming languages specialized for modeling). In contrast, traditional models for chemical kinetics formulated in terms of equations are more suitable for analysis (e.g., integration or differentiation) than for computation. Traditional modeling forms are used by many software tools, including tools that bridge equations to numerical methods of analysis (e.g., numerical integration), such as MATLAB (The MathWorks, Natick, MA). However, departure from traditional forms can be advantageous [82] and for mechanistic modeling of cell signaling systems, it is necessary. This need arises from the size and combinatorial complexity of signaling systems, which can be better captured if a model is viewed as a program. The reason is that a programming language can be tailored for the problem at hand. A model specified as a program has a number of other advantages over a set of equations. One advantage that has perhaps not yet been fully appreciated is greater extensibility and a potential for clearer annotation. As demonstrated by Thomson et al. [126], the formal elements of a rule-based model can be specified incrementally (i.e., one at a time), annotated independently, and then later assembled to address specific questions about system properties, which can also be formalized [97, 127, 128]. Guidelines for annotating rule-based models have been proposed [57], which if adopted, could make models more understandable and reusable. Rule-based modeling provides a general paradigm for modeling interactions of structured objects, with proven applications in physics, chemistry, and computer science. The approach is being used increasingly often in systems biology. In the future, we expect it to be a foundational method of the field because its extensibility addresses large network size, and the use of rules addresses combinatorial complexity, which are two inherent features of cell signaling systems.

5 Acknowledgements

We thank Michael L. Blinov, James R. Faeder, David J. Klink II, Jean Krivine, and Carlos F. Lopez for helpful discussions.

References

1. Kitano H: **Systems biology: a brief overview**. *Science* 2002, **295**:1662–1664.
2. Kitano H: **Computational systems biology**. *Nature* 2002, **420**:206–210.
3. Lazebnik Y: **Can a biologist fix a radio?—Or, what I learned while studying apoptosis**. *Cancer Cell* 2002, **2**:179–182.
4. Kreeger PK, Lauffenburger DA: **Cancer systems biology: a network modeling perspective**. *Carcinogenesis* 2010, **31**:2–8.
5. Chakraborty AK, Das J: **Pairing computation with experimentation: a powerful coupling for understanding T cell signalling**. *Nat Rev Immunol* 2010, **10**:59–71.
6. Germain RN, Meier-Schellersheim M, Nita-Lazar A, Fraser IDC: **Systems biology in immunology—A computational modeling perspective**. *Annu Rev Immunol* 2011, **29**:527–585.
7. Lander AD: **The edges of understanding**. *BMC Biol* 2010, **8**:40.
8. Downward J: **Targeting RAF: trials and tribulations**. *Nat Med* 2011, **17**:286–288.
9. Kholodenko BN: **Cell-signalling dynamics in time and space**. *Nat Rev Mol Cell Biol* 2006, **7**:165–176.
10. Kholodenko BN, Hancock JF, Kolch W: **Signalling ballet in space and time**. *Nat Rev Mol Cell Biol* 2010, **11**:414–426.
11. Hunter T: **Signaling—2000 and beyond**. *Cell* 2000, **100**:113–127.
12. Pawson T, Nash P: **Assembly of cell regulatory systems through protein interaction domains**. *Science* 2003, **300**:445–52.
13. Kandasamy K, Mohan SS, Raju R, Keerthikumar S, Kumar G, Venugopal A, Telikicherla D, Navarro JD, Mathivanan S, Pecquet C, Gollapudi S, Tattikota S, Mohan S, Padhukasahasram H, Subbannayya Y, Goel R, Jacob H, Zhong J, Sekhar R, Nanjappa V, Balakrishnan L, Subbaiah R, Ramachandra Y, Rahiman BA, Prasad TK, Lin JX, Houtman J, Desiderio S, Renauld JC, Constantinescu S: **NetPath: a public resource of curated signal transduction pathways**. *Genome Biol* 2010, **11**:R3.
14. Hlavacek WS, Faeder JR, Blinov ML, Perelson AS, Goldstein B: **The complexity of complexes in signal transduction**. *Biotechnol Bioeng* 2003, **84**:783–794.
15. Mayer BJ, Blinov ML, Loew LM: **Molecular machines or pleiomorphic ensembles: signaling complexes revisited**. *J Biol* 2009, **8**:81.
16. Faeder JR, Blinov ML, Hlavacek WS: **Rule-based modeling of biochemical systems with BioNetGen**. *Methods Mol Biol* 2009, **500**:113–167.
17. Andrews SS, Addy NJ, Brent R, Arkin AP: **Detailed simulations of cell biology with Smoldyn 2.1**. *PLoS Comput Biol* 2010, **6**:e1000705.
18. Colvin J, Monine MI, Faeder JR, Hlavacek WS, Von Hoff DD, Posner RG: **Simulation of large-scale rule-based models**. *Bioinformatics* 2009, **25**:910–917.
19. **Website about Kappa and Kappa-based software tools** [<http://kappalanguage.org/>].
20. Sneddon MW, Faeder JR, Emonet T: **Efficient modeling, simulation, and coarse-graining of biological complexity with NFsim**. *Nat Methods* 2011, **8**:177–183.
21. Colvin J, Monine MI, Gutenkunst R, Hlavacek WS, Von Hoff DD, Posner RG: **RuleMonkey: software for stochastic simulation of rule-based models**. *BMC Bioinformatics* 2010, **11**:404.
22. Moraru II, Schaff JC, Slepchenko BM, Blinov ML, Morgan F, Lakshminarayana A, Gao F, Li Y, Loew LM: **Virtual Cell modelling and simulation software environment**. *IET Systems Biology* 2008, **2**:352–362.
23. Meier-Schellersheim M, Xu X, Angermann B, Kunkel E, Jin T, Germain RN: **Key role of local regulation chemosensing revealed by a new molecular interaction-based modeling method**. *PLoS Comput Biol* 2006, **2**:e82.
24. Angermann BR, Klauschen F, Garcia AD, Prustel T, Zhang F, Germain RN, Meier-Schellersheim M: **Computational modeling of cellular signaling processes embedded into dynamic spatial contexts**. *Nat Methods* 2012.
25. Chen WW, Schoeberl B, Jasper PJ, Niepel M, Nielse UB, Lauffenburger D, Sorger PK: **Input–output behavior of ErbB signaling pathways as revealed by a mass action model trained against dynamic data**. *Mol Syst Biol* 2009, **5**:239.
26. Dinkel H, Chica C, Via A, Gould CM, Jensen LJ, Gibson TJ, Diella F: **Phospho.ELM: a database of phosphorylation sites—update 2011**. *Nucleic Acids Res* 2011, **39**:D261.
27. Bateman A, Coin L, Durbin R, Finn RD, Hollich V, GriffithsJones S, Khanna A, Marshall M, Moxon S, Sonnhammer ELL, Studholme DJ, Yeats C, Eddy SR: **The Pfam protein families database**. *Nucleic Acids Res* 2004, **32**:D138–D141.

28. Dinkel H, Michael S, Weatheritt RJ, Davey NE, Van Roey K, Altenberg B, Toedt G, Uyar B, Seiler M, Budd A, Jdicke L, Dammert MA, Schroeter C, Hammer M, Schmidt T, Jehl P, McGuigan C, Dymecka M, Chica C, Luck K, Via A, Chatr-aryamontri A, Haslam N, Grebnev G, Edwards RJ, Steinmetz MO, Meiselbach H, Diella F, Gibson TJ: **ELM—the database of eukaryotic linear motifs**. *Nucleic Acids Res* 2012, **40**:D242–D251.
29. Keshava Prasad TS, Goel R, Kandasamy K, Keerthikumar S, Kumar S, Mathivanan S, Telikicherla D, Raju R, Shafreen B, Venugopal A, Balakrishnan L, Marimuthu A, Banerjee S, Somanathan DS, Sebastian A, Rani S, Ray S, Harrys Kishore CJ, Kanth S, Ahmed M, Kashyap MK, Mohmood R, Ramachandra YL, Krishna V, Rahiman BA, Mohan S, Ranganathan P, Ramabadran S, Chaerkady R, Pandey A: **Human protein reference database—2009 update**. *Nucleic acids research* 2009, **37**:D767–D772.
30. Hlavacek WS, Faeder JR, Blinov ML, Posner RG, Hucka M, Fontana W: **Rules for modeling signal-transduction systems**. *Sci STKE* 2006, **2006**:re6.
31. Danos V, Feret J, Fontana W, Harmer R, Krivine J: **Rule-based modelling of cellular signalling**. *Lect Notes Comput Sci* 2007, **4703**:17–41.
32. Fisher J, Harel D, Henzinger TA: **Biology as reactivity**. *Commun ACM* 2011, **54**:72–82.
33. Lim WA, Pawson T: **Phosphotyrosine signaling: evolving a new cellular communication system**. *Cell* 2010, **142**:661–667.
34. Endy D, Brent R: **Modeling cellular behavior**. *Nature* 2001, **409**:391–395.
35. Bray D: **Molecular prodigality**. *Science* 2003, **299**:1189–1190.
36. Le Novère N, Hucka M, Mi H, Moodie S, Schreiber F, Sorokin A, Demir E, Wegner K, Aladjem MI, Wimalaratne SM, Bergman FT, Gauges R, Ghazal P, Kawaji H, Li L, Matsuoka Y, Villegier A, Boyd SE, Calzone L, Courtot M, Dogrusoz U, Freeman TC, Funahashi A, Ghosh S, Jouraku A, Kim S, Kolpakov F, Luna A, Sahle S, Schmidt E, Watterson S, Wu G, Goryanin I, Kell DB, Sander C, Sauro H, Snoep JL, Kohn K, Kitano H: **The systems biology graphical notation**. *Nat Biotechnol* 2009, **27**:735–741.
37. Bray D, Lay S: **Computer-based analysis of the binding steps in protein complex formation**. *Proc Natl Acad Sci* 1997, **94**:13493–13498.
38. Morton-Firth CJ, Bray D: **Predicting temporal fluctuations in an intracellular signalling pathway**. *J Theor Biol* 1998, **192**:117–128.
39. Le Novère N, Shimizu TS: **STOCHSIM: modelling of stochastic biomolecular processes**. *Bioinformatics* 2001, **17**:575–576.
40. Regev A, Silverman W, Shapiro E: **Representation and simulation of biochemical processes using the π -calculus process algebra**. *Pac Symp Biocomput* 2001, :459–470.
41. Priami C, Regev A, Shapiro E, Silverman W: **Application of a stochastic name-passing calculus to representation and simulation of molecular processes**. *Information Processing Letters* 2001, **80**:25–31.
42. **The BioSPI project homepage** [<http://www.wisdom.weizmann.ac.il/~biospi/>].
43. Dematté L, Priami C, Romanel A: **The BlenX language: a tutorial**. *Lect Notes Comput Sci* 2008, **5016**:313–365.
44. Kahramanogullari O, Cardelli L, Caron E: **An Intuitive Automated Modelling Interface for Systems Biology**. In *DCM* 2009:73–86.
45. Phillips A, Cardelli L: **Efficient, correct simulation of biological processes in the stochastic pi-calculus**. *Lect Notes Comput Sci* 2007, **4695**:184–199.
46. Goldstein B, Faeder JR, Hlavacek WS, Blinov ML, Redondo A, Wofsy C: **Modeling the early signaling events mediated by Fc ϵ RI**. *Mol Immunol* 2002, **38**:1213–1219.
47. Faeder JR, Hlavacek WS, Reischl I, Blinov ML, Metzger H, Redondo A, Wofsy C, Goldstein B: **Investigation of early events in Fc ϵ RI-mediated signaling using a detailed mathematical model**. *J Immunol* 2003, **170**:3769–3781.
48. Nag A, Monine MI, Faeder JR, Goldstein B: **Aggregation of membrane proteins by cytosolic cross-linkers: theory and simulation of the LAT-Grb2-SOS1 system**. *Biophys J* 2009, **96**:2604–2623.
49. Nag A, Monine MI, Blinov ML, Goldstein B: **A detailed mathematical model predicts that serial engagement of IgE-Fc ϵ RI complexes can enhance Syk activation in mast cells**. *J Immunol* 2010, **185**:3268–3276.
50. Nag A, Blinov ML, Goldstein B: **Shaping the response: the role of Fc ϵ RI and Syk expression levels in mast cell signaling**. *IET Syst Biol* 2010, **4**:334–347.
51. Monine MI, Posner RG, Savage PB, Faeder JR, Hlavacek WS: **Modeling multivalent ligand-receptor interactions with steric constraints on configurations of cell-surface receptor aggregates**. *Biophys J* 2010, **98**:48–56.

52. Lee KH, Dinner AR, Tu C, Campi G, Raychaudhuri S, Varma R, Sims TN, Burack WR, Wu H, Wang J, Kanagawa O, Markiewicz M, Allen PM, Dustin ML, Chakraborty AK, Shaw AS: **The immunological synapse balances T cell receptor signaling and degradation.** *Science* 2003, **302**:1218–1222.
53. Li QJ, Dinner AR, Qi S, Irvine DJ, Huppa JB, Davis MM, Chakraborty AK: **CD4 enhances T cell sensitivity to antigen by coordinating Lck accumulation at the immunological synapse.** *Nat Immunol* 2004, **5**:791–799.
54. Altan-Bonnet G, Germain RN: **Modeling T cell antigen discrimination based on feedback control of digital ERK responses.** *PLoS Biol* 2005, **3**:e356.
55. Nag A, Monine M, Perelson AS, Goldstein B: **Modeling and simulation of aggregation of membrane protein LAT with molecular variability in the number of binding sites for cytosolic Grb2-SOS1-Grb2.** *PLoS ONE* 2012, **7**:e28758.
56. Faeder JR, Blinov ML, Hlavacek WS: **Graphical rule-based representation of signal transduction networks.** In *Proc. 2005 ACM Symp. Appl. Computing*. Edited by Liebrock L, ACM Press, New York, NY 2005:133–140.
57. Chylek LA, Hu B, Blinov ML, Emonet T, Faeder JR, Goldstein B, Gutenkunst RN, Haugh JM, Lipniacki T, Posner RG, Yang J, Hlavacek WS: **Guidelines for visualizing and annotating rule-based models.** *Mol Biosyst* 2011, **7**:2779–2795.
58. Xu W, Smith AM, Faeder JR, Marai GE: **RuleBender: a visual interface for rule-based modeling.** *Bioinformatics* 2011, **27**:1721–1722.
59. Blinov ML, Faeder JR, Goldstein B, Hlavacek WS: **BioNetGen: software for rule-based modeling of signal transduction based on the interactions of molecular domains.** *Bioinformatics* 2004, **20**:3289–3291.
60. Danos V, Laneve C: **Formal molecular biology.** *Theoretical Computer Science* 2004, **325**:69–110.
61. Feret J, Danos V, Krivine J, Harmer R, Fontana W: **Internal coarse-graining of molecular systems.** *Proc Natl Acad Sci USA* 2009, **106**:6453–6458.
62. Barua D, Faeder JR, Haugh JM: **A bipolar clamp mechanism for activation of Jak-family protein tyrosine kinases.** *PLoS Comput Biol* 2009, **5**:e1000364.
63. Dushek O, Das R, Coombs D: **A role for rebinding in rapid and reliable T cell responses to antigen.** *PLoS Comput Biol* 2009, **5**:e1000578.
64. Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E: **Equation of state calculations by fast computing machines.** *J Chem Phys* 1953, **21**:1087–1092.
65. Bortz AB, Kalos M, Lebowitz J: **A new algorithm for Monte Carlo simulations of Ising spin systems.** *J Comput Phys* 1975, **17**:10–18.
66. Voter AF: **Introduction to the kinetic Monte Carlo method.** In *Radiation Effects in Solids*. Edited by Sickafus KE, Kotomin EA, Springer 2007:1–21.
67. Gillespie DT: **A general method for numerically simulating the stochastic time evolution of coupled chemical reactions.** *J Comput Phys* 1976, **22**:403–434.
68. Gillespie DT: **Exact stochastic simulation of coupled chemical reactions.** *J Phys Chem* 1977, **81**:2340–2361.
69. Gillespie DT: **Stochastic simulation of chemical kinetics.** *Annu Rev Phys Chem* 2007, **58**:35–55.
70. Dugundji J, Ugi I: **An algebraic model of constitutional chemistry as a basis for chemical computer programs.** *Top Curr Chem* 1973, **39**:19–64.
71. Ugi I, Bauer J, Bley K, Dengler A, Dietz A, Fontain E, Gruber B, Herges R, Knauer M, Reitsman K, Stein N: **Computer-assisted solution of chemical problems—the historic development and the present state of the art of a new discipline of chemistry.** *Angew Chem Int Ed Engl* 1993, **32**:201–227.
72. Green WH Jr: **Predictive kinetics: a new approach for the 21st century.** *Adv Chem Eng* 2007, **32**:1–50.
73. Faulon JL, Carbonell P: **Reaction network generation.** In *Handbook of Chemoinformatics Algorithms*. Edited by Faulon JL, Bender A, Chapman & Hall/CRC Press, Boca Raton, FL 2010:317–341.
74. Rangarajan S, Bahn A, Daoutidis P: **Rule-based generation of thermochemical routes to biomass conversion.** *Ind Eng Chem Res* 2010, **49**:10459–10470.
75. Klinke DJ II, Finley SD: **Timescale analysis of rule-based biochemical reaction networks.** *Biotechnol Progr* 2012.
76. Klinke DJ II, Broadbelt LJ: **Construction of a mechanistic model of Fischer-Tropsch synthesis on Ni (1 1 1) and Co (0 0 1) surfaces.** *Chem Eng Sci* 1999, **54**:3379–3389.
77. Broadbelt LJ, Pfaendtner J: **Lexicography of kinetic modeling of complex reaction networks.** *AIChE J* 2005, **51**:2112–2121.

78. Hatzimanikatis V, Li C, Ionita JA, Henry CS, Jankowski MD, Broadbelt LJ: **Exploring the diversity of complex metabolic networks**. *Bioinformatics* 2005, **21**:1603–1609.
79. Mu F, Unkefer CJ, Unkefer PJ, Hlavacek WS: **Prediction of metabolic reactions based on atomic and molecular properties of small-molecule compounds**. *Bioinformatics* 2011, **27**:1537–1545.
80. Milner R, Parrow J, Walker D: **A calculus of mobile processes, I**. *Inform Comput* 1992, **100**:1–40.
81. Fokkink W: *Introduction to process algebra*. Springer 2000.
82. Fisher J, Henzinger TA: **Executable cell biology**. *Nat Biotechnol* 2007, **25**:1239–1249.
83. Harmer R: **Rule-based modeling and tunable resolution**. *EPTCS* 2009, **9**:65–72.
84. Harmer R, Danos V, Feret J, Krivine J, Fontana W: **Intrinsic information carriers in combinatorial dynamical systems**. *Chaos* 2010, **20**:037108.
85. **PySB: Python framework for Systems Biology modeling** [<https://github.com/sorgerlab/pysb>].
86. Lipniacki T, Hat B, Faeder JR, Hlavacek WS: **Stochastic effects and bistability in T cell receptor signaling**. *J Theor Biol* 2008, **254**:110–122.
87. Faeder JR, Blinov ML, Goldstein B, Hlavacek WS: **Rule-based modeling of biochemical networks**. *Complexity* 2005, **10**:22–41.
88. Lok L, Brent R: **Automatic generation of cellular reaction networks with Molecuizer 1.0**. *Nat Biotechnol* 2005, **23**:131–136.
89. Harris LA, Hogg JS, Faeder JR: **Compartmental rule-based modeling of biochemical systems**. In *Proceedings of the 2009 Winter Simulation Conference*. Edited by Rossetti M, Hill R, Johansson B, Dunkin A, Ingalls R 2009:908–919.
90. Maus C, Rybacki S, Uhrmacher AM: **Rule-based multi-level modeling of cell biological systems**. *BMC Syst Biol* 2011, **5**:166.
91. Mallavarapu A, Thomson M, Ullian B, Gunawardena J: **Programming with models: modularity and abstraction provide powerful capabilities for systems biology**. *J R Soc Interface* 2009, **6**:257.
92. Lis M, Artyomov MN, Devadas S, Chakraborty AK: **Efficient stochastic simulation of reaction-diffusion processes via direct compilation**. *Bioinformatics* 2009, **25**:2289–2291.
93. Yang J, Monine MI, Faeder JR, Hlavacek WS: **Kinetic Monte Carlo method for rule-based modeling of biochemical networks**. *Phys Rev E* 2008, **78**:031910.
94. Gruenert G, Ibrahim B, Lenser T, Lohel M, Hinze T, Dittrich P: **Rule-based spatial modeling with diffusing, geometrically constrained molecules**. *BMC Bioinformatics* 2010, **11**:307.
95. **RuleBase** [<http://rulebase.org/>].
96. Hu B, Fricke GM, Faeder JR, Posner RG, Hlavacek WS: **GetBonNie for building, analyzing and sharing rule-based models**. *Bioinformatics* 2009, **25**:1457–1460.
97. Clarke EM, Faeder JR, Harris LA, Langmead CJ, Legay A, Jha SK: **Statistical model checking in BioLab: applications to the automated analysis of T-cell receptor signaling pathway**. *Lect Notes Comput Sci* 2008, **5307**:231–250.
98. Koschorreck M, Gilles E: **ALC: automated reduction of rule-based models**. *BMC Syst Biol* 2008, **2**:91.
99. Ollivier JF, Shahrezaei V, Swain P: **Scalable rule-based modeling of allosteric proteins and biochemical networks**. *PLoS Comput Biol* 2010, **6**:e1000975.
100. Fages F, Soliman S, Chabrier-Rivier N: **Modelling and querying interaction networks in the biochemical abstract machine BIOCHAM**. *J Biol Phys Chem* 2004, **4**:64–73.
101. Priami C, Ballarini P, Quaglia P: **BlenX4Bio–BlenX for Biologists**. In *Computational Methods in Systems Biology*, Springer 2009:26 – 51.
102. Vilar JMG, Saiz L: **CplexA: a Mathematica package to study macromolecular-assembly control of gene expression**. *Bioinformatics* 2010, **26**:2060–2061.
103. Tolle D, Novère L: **Meredys, a multi-compartment reaction-diffusion simulator using multistate realistic molecular complexes**. *BMC Syst Biol* 2010, **4**:24.
104. Eker S, Knapp M, Laderoute K, Lincoln P, Talcott C: **Pathway Logic: Executable models of biological networks**. *Electron. Notes Theor. Comput. Sci.* 2004, **71**:125–142.
105. Maiwald T, Timmer J: **Dynamical modeling and multi-experiment fitting with PottersWheel**. *Bioinformatics* 2008, **24**:2037–2043.
106. **The KaSim user manual** [http://cloud.github.com/downloads/jkrivine/KaSim/KaSim_manual.pdf].
107. Bonabeau E: **Agent-based modeling: Methods and techniques for simulating human systems**. *Proc Natl Acad Sci USA* 2002, **99**:7280–7287.
108. Yang J, Hlavacek WS: **Efficiency of reactant site sampling in network-free simulation of rule-based models for biochemical systems**. *Phys Biol* 2011, **8**:055009.

109. Goldstein B: **Desensitization, histamine release and the aggregation of IgE on human basophils.** In *Theoretical Immunology, Part One, SFI Studies in the Sciences of Complexity*. Edited by Perelson AS, Addison-Wesley, Reading, MA 1988:3–40.
110. Blue JL, Beichl I, Sullivan F: **Faster Monte Carlo simulations.** *Phys Rev E* 1995, **51**:R867–R868.
111. Gibson MA, Bruck J: **Efficient exact stochastic simulation of chemical systems with many species and many channels.** *J Phys Chem A* 2000, **104**:1876–1889.
112. Slepoy A, Thompson AP, Plimpton SJ: **A constant-time kinetic Monte Carlo algorithm for simulation of large biochemical reaction networks.** *J Chem Phys* 2008, **128**:205101.
113. Danos V, Feret J, Fontana W, Krivine J: **Scalable simulation of cellular signalling networks.** *Lect Notes Comput Sci* 2007, **4807**:139–157.
114. Shimizu TS, Bray D: **Computational cell biology - the stochastic approach.** In *Foundations of Systems Biology*. Edited by Kitano H, MIT Press 2001.
115. Houtman JCD, Barda-Saad M, Samelson LE: **Examining multiprotein signaling complexes from all angles.** *FEBS J* 2005, **500**:5426–5435.
116. Schulze WX, Deng L, Mann M: **Phosphotyrosine interactome of the ErbB-receptor kinase family.** *Mol Syst Biol* 2005, **1**:2005.0008.
117. Zhang Y, Wolf-Yadlin A, Ross PL, Pappin DJ, Rush J, Lauffenburger DA: **Phosphotyrosine interactome of the ErbB-receptor kinase family.** *Mol Cell Proteomics* 2005, **4**:1240–1250.
118. Jones RB, Gordus A, Krall JA, MacBeath G: **A quantitative protein interaction network for the ErbB receptors using protein microarrays.** *Nature* 2006, **439**:168–174.
119. VanMeter AJ, Rodriguez AS, Bowman ED, Jen J, Harris CC, Deng J, Calvert VS, Silvestri A, Fredolini C, Chandhoke V, Petricoin EF, Liotta LA, Espina V: **Laser capture microdissection and protein microarray analysis of human non-small cell lung cancer: differential epidermal growth factor receptor (EGFR) phosphorylation events associated with mutated EGFR compared with wild type.** *Mol Cell Proteomics* 2008, **7**:1902–1924.
120. Ciaccio MF, Wagner JP, Chuu CP, Lauffenburger DA, Jones RB: **Systems analysis of EGF receptor signaling dynamics with microwestern arrays.** *Nat Methods* 2010, **7**:148–155.
121. Cox J, Mann M: **Quantitative, high-resolution proteomics for data-driven systems biology.** *Annu Rev Biochem* 2011, **80**:273–299.
122. Blinov ML, Faeder JR, Golstein B, Hlavacek WS: **A network model of early events in epidermal growth factor receptor signaling that accounts for combinatorial complexity.** *Biosystems* 2006, **83**:136–151.
123. Houtman JCD, Houghtling RA, Barda-Saad M, Toda Y, Samelson LE: **Early phosphorylation kinetics of proteins involved in proximal TCR-mediated signaling pathways.** *J Immunol* 2005, **175**:2449.
124. Sawyers C: **Targeted cancer therapy.** *Nature* 2004, **432**:294–297.
125. Birtwistle MR, Hatakeyama M, Yumoto N, Ogunnaike BA, Hoek JB, Kholodenko BN: **Ligand-dependent responses of the ErbB signaling network: experimental and modeling analyses.** *Mol Syst Biol* 2007, **3**:144.
126. Thomson T, Benjamin KR, Bush A, Love T, Pincus D, Resnekov O, Yu RC, Gordon A, Colman-Lerner A, Endy D, Brent R: **Scaffold number in yeast signaling system sets tradeoff between system output and dynamic range.** *Proc Natl Acad Sci USA* 2011, **13**:20265–20270.
127. Calzone L, Fages F, Soliman S: **BIOCHAM: an environment for modeling biological systems and formalizing experimental knowledge.** *Bioinformatics* 2006, **22**:1805–1807.
128. Gong H, Zuliani P, Komuravelli A, Faeder JR, Clarke EM: **Analysis and verification of the HMGB1 signaling pathway.** *BMC Bioinformatics* 2010, **11**:S10.