Immunol Res (2012) 53:251–265
DOI 10.1007/s12026-012-8305-7

Systems immunology: a survey of modeling formalisms, applications and simulation tools

Vipin Narang · James Decraene · Shek-Yoon Wong · Bindu S. Aiswarya · Andrew R. Wasem · Shiang Rong Leong · Alexandre Gouaillard

Alexandre Gouaillard

Published online: 13 April 2012

Abstract Immunological studies frequently analyze *individual* components (e.g., signaling pathways) of immune systems in a reductionist manner. In contrast, systems immunology aims to give a synthetic understanding of how these components function together as a *whole*. While immunological research involves in vivo and in vitro experiments, systems immunology research can also be conducted *in silico*. With an increasing interest in systems-level studies spawned by high-throughput technologies, many immunologists are looking forward to insights provided by computational modeling and simulation. However, modeling and simulation research has mainly been conducted in computational fields, and therefore, little material is available or accessible to immunologists today. This survey is an attempt at bridging the gap between immunologists and systems immunology modeling and simulation. Modeling and simulation refer to building and executing an *in silico* replica of an immune system. Models are specified within a mathematical or algorithmic framework called formalism and then implemented using software tools. A plethora of modeling formalisms and software tools are reported in the literature for systems immunology. However, it is difficult for a new entrant to the field to know which of these would be suitable for modeling an immunological application at hand. This paper covers three aspects. First, it introduces the field of system immunology emphasizing on the modeling and simulation components. Second, it gives an overview of the principal modeling formalisms, each of which is illustrated with salient applications in immunological research. This overview of formalisms and applications is conducted not only to illustrate their power but also to serve as a reference to assist immunologists in choosing the best formalism for the problem at hand. Third, it lists major software tools, which can be used to practically implement models in these formalisms. Combined, these aspects can help immunologists to start experimenting with *in silico* models. Finally, future research directions are discussed. Particularly, we identify integrative frameworks to facilitate the coupling of different modeling formalisms and modeling the adaptation properties through evolution of immune systems as the next key research efforts necessary to further develop the multi-disciplinary field of systems immunology.

Keywords Systems immunology · Modeling and simulation · Multiscale modeling · Integration · Modeling formalisms · Software

Introduction

Since the discovery of cell in 1653 [1], significant progress has been made in understanding the fundamental components of biological systems and the mechanisms governing them. These developments are summarized in various "omics" of biology [2, 3]. However, recently, the need to have an integrated understanding of how these components work together in a system is being increasingly realized [4].

V. Narang · J. Decraene · S.-Y. Wong · B. S. Aiswarya · A. R. Wasem · S. R. Leong · A. Gouaillard (✉)
Singapore Immunology Network, BMSI, Singapore, Singapore
e-mail: Alexandre_Gouaillard@immunol.a-star.edu.sg

V. Narang
e-mail: Vipin_Narang@immunol.a-star.edu.sg

Immune systems have a high degree of interdependence and interconnection between components. They have a multihierarchical or *multiscale* organization (Fig. 1), where components at a lower scale are integrated into functional units at the next higher scale [5]. The units are robust and exchange a limited amount of information with their surroundings. Thus, they hide the detailed mechanisms operating at lower scales. At any scale, the system can be studied phenomenologically as long as there are predictable causal relationships between the perturbations and state changes. Indeed, there are separate subdisciplines of biology addressing different phenomenological scales (Fig. 1).

However, there are certain scenarios where phenomenological descriptions are not sufficient and an integrated description of phenomena occurring at various scales is unavoidable. For example, host–pathogen interactions occur on all scales: from the molecular scale (recognition of pathogen molecular patterns by immune cells) to the cellular scale (phagocytosis of the pathogen by macrophages) and up to the population scale (spread of the pathogen through host population and emergence of different strains through different host conditions) [6]. Elements at different scales interact: for example, host–pathogen interactions at the population level determine the selection pressure on both host and pathogen at the genetic level. Thus, a full understanding of host–pathogen interactions requires the integration of phenomena at various scales. Such integration of different spatial and time scales has been called by the term *systems immunology* [7, 8].

Another dimension of integration is *transcale*, which involves aggregating and connecting information from different disciplines of study at the same biological scale. For instance, cells are studied from molecular, biochemical, geometrical, biomechanical and various other perspectives. These properties are interlinked: for example, cytokine TGF-β increases cell stiffness and leads to an elongated cell shape. Reciprocally, cells such as chondrocytes secrete TGF-β in response to mechanical stimulation (mechanotransduction). Transcale integration brings together such interdisciplinary information.

To achieve this integration, systems immunology complements empirical and experimental approaches with *modeling* and *simulation*. A model is an abstract representation of a real system. Simulation refers to operating a model under a configuration of interest to "simulate" the system's behavior. A model has some realism when abstract entities in the model correspond to real components in the system and mathematical or computational rules governing the model correspond to real physical laws [9].

Ideally, a model in systems immunology should give a virtual description of an immune system in the same way as a Google® map describes a physical landscape. The specialty of Google map is its multiscale structure. Zooming in or out of the map not only changes its scale but also displays different features and annotations corresponding to different resolutions. A model of an immunological system should similarly give an integrated description of its components at various scales. Such a model can be easily used to represent the state of the system and infer its dynamics while going from an initial to a final state.

A model therefore serves to synthesize existing knowledge and data about a system via *multiscale and transcale integration*. The knowledge and data could be static (without temporal information) or dynamic. As a consequence, missing information or gaps in understanding may be revealed, which could initiate further experimentation and guide data acquisition policies. A model can also serve to generate new knowledge about the system by simulation,

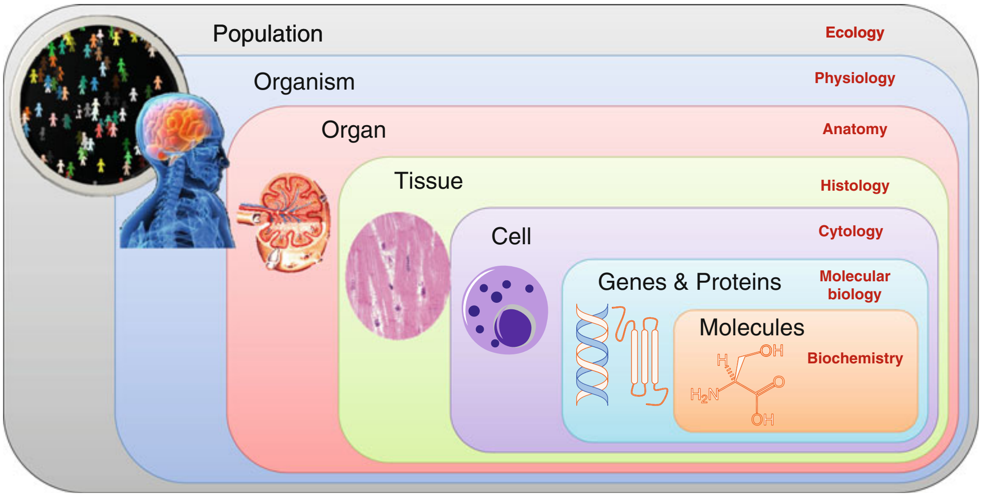

Fig. 1 Multiscale organization of immune systems and the subdisciplines of biology at various scales

which is technically known as *prediction*. Simulation can help in understanding system-level behavior over large ranges of parameter values. This may be difficult to test in vivo due to technological and financial constraints. Modeling and simulation may also demonstrate surprising or counterintuitive consequences of system organization. Indeed, a model can sometimes serve to provide mechanistic explanations about the intricate working of the system [10].

Several formalisms have been proposed in the literature for systems immunology modeling. Each formalism has its particular characteristics that make it suitable for some specific applications. There is no and there will not be any global "best" modeling formalism. Most likely, the only way to model something as complex as the immune system is to blend (compound or aggregate) different formalisms in a common multiscale transcale framework. We can see a trend toward such "hybrid" or multiformalism modeling[11–14].

This paper is an attempt at describing the principal formalisms that currently exist in the literature for systems immunology modeling from an application point of view. It is directed toward immunologists who are interested in using systems immunology modeling but do not have a theoretical or technical exposure to the field. It covers theoretical principles, immunological applications and software tools associated with the formalisms. Some recent reviews have focused on only one or a few of these formalisms and only one or a few biological scales. Thus, an overall view of the modeling field as it exists today is missing. There is no review of available software. Moreover, there are no general guidelines to decide which formalism will be suitable for a given application. This review tries to fill these gaps. The presentation has been kept as much as possible nontechnical for the understanding of immunologists. Technical details are referred to existing relevant literature.

We first provide an introduction to computational modeling and simulation, presenting the key concepts and rationales. Then, formalisms are described in brief, covering their basic principles and modeling strategy. Formalisms covered in this paper include differential equations, cellular automata and agent-based systems. Additional, but less prominent, modeling formalisms are also introduced. For each modeling formalism, we describe typical immunological applications and available computational tools. Finally, we identify and discuss the future research directions where the intricate *adaptive* nature of immune systems would be considered.

Modeling and simulation

A model is a representation of a real system in mathematical or algorithmic terms. A model cannot possibly include all details of a system. It is an abstraction suitable enough to capture the essential mechanism of the real system. Building a model minimally requires knowledge of the *entities* in the system such as genes, proteins, cells, organs etc., the *characterization* of these entities either by a quantity (such as protein concentration, number of cells, etc.) or by a discrete state (such as on/off state of a gene, an individual being uninfected/infected/chronically infected, etc.), and the *interactions* among different entities such as regulation of genes by proteins, protein–protein interactions, cell–cell interactions, etc. Additional knowledge such as the description of compartments or other subspaces in the system can also be useful.

In addition to knowledge of entities and interactions, a model requires a mathematical or algorithmic structure called *formalism* for its description [15]. Formalism regulates modeling assumptions, model structure, governing laws, nature of interactions, how a model is simulated, etc. For example, one of the simplest formalisms is a static graph which has traditionally been used for visualizing networks of cell–cell interactions, gene expression or signal transduction [16]. In a static graph, entities are represented by nodes, their quantities or states by a weight or color and an interaction between two entities as a (possibly weighted) edge connecting the nodes. Such a graph can provide some insight into the possible behaviors of a system. However, it does not enable quantitative predictions. More descriptive formalisms can enable quantitative analysis.

From an immunologist's point of view, the important question is which formalism is suitable for modeling an immunological application at hand. Although there are no readymade answers to this question, some important considerations, such as the ones below, can guide the choice of an appropriate modeling formalism. A glossary of some technical terms in this connection is given in Table 1.

1. *Objectives of the study*—Different modeling formalisms have different modeling capabilities. The choice of formalism depends on the questions to be answered. For example, when exact numerical quantities of entities and timings of events are of interest, quantitative formalisms are appropriate. On the other hand, when quantitative information is inessential or missing, qualitative models may be appropriate.
2. *Scale of the model*—Some formalisms are more appropriate for modeling at certain biological scales. For example, intracellular networks are suitably modeled by formalisms having a static network structure, whereas tissue, organ or individual scales where the network is usually dynamic or nonexistent are sometimes more appropriately modeled by formalisms supporting a dynamic network or agent-based modeling. Figure 2 gives a summary showing biological

Table 1 Glossary

Term	Meaning
Quantitative formalism	A quantitative formalism is concerned with numerically measurable attributes of the system such as protein concentrations. It establishes relationships between these quantities with the help of mathematical functions
Qualitative formalism	In a qualitative formalism, the exact numerical measurements are not important; rather, the states of the system (such as the presence or the absence of a protein) and the sequence of state transitions are important. The relationships between various entities are defined categorically, such as using combinatorial logic functions (Boolean AND, OR, NOT, etc.), or state transition matrices, etc
Continuous-time formalism	In a continuous-time formalism, time is represented as continuous numerical parameter (a real number). The exact times when events occur in the system are measured
Discrete-time formalism	In a discrete-time formalism, time is represented in integral steps according to when the state transitions occur in the system. Thus, instead of the exact time when an event occurs, the discrete steps that lead to the event are of interest. In general, the time steps are not of equal length. Discrete time formalisms support three different types of updating—synchronous, asynchronous and concurrent
Synchronous update	In synchronous update, all entities update their states at every time step and the current state depends on the previous states of all entities
Asynchronous update	An asynchronous update makes a more realistic assumption that different entities change states at different time steps. The current state of an entity again depends on the existing states of all the other entities
Concurrent update	Concurrent update is most general as it allows entities to change states independently at the same time. The current state of an entity takes into account the current states of the other entities
Compositional formalism	In a compositional formalism, the system is broken down into parts of manageable size and complexity, and these parts are individually modeled. Then, the models for the parts are combined to give the full model. Compositionality is a natural way of modeling multiscale biological systems
Flat formalism	In contrast to a compositional formalism, in a flat formalism, all the entities in the system and their interconnections with each other are represented in a single structure
Deterministic formalism	In a deterministic formalism, the system has well-known numerical properties or states and the system evolves in a predictable manner governed by known laws
Stochastic formalism	In a stochastic formalism, there is unpredictability in the system arising from various factors such as noisy measurements, missing or unknown details, heterogeneity in the system, etc
Network	A set of interconnected entities, where a connection represents a well-defined interaction between a pair of entities
Static network	A network in which entities and connections are fixed
Dynamic network	A network in which entities and connections can appear or disappear in time
No network	When interactions between entities are purely by chance or proximity, it is referred to as a no network situation
Distributed system	A system that consists of independent functioning and possibly physically separated subsystems or processes

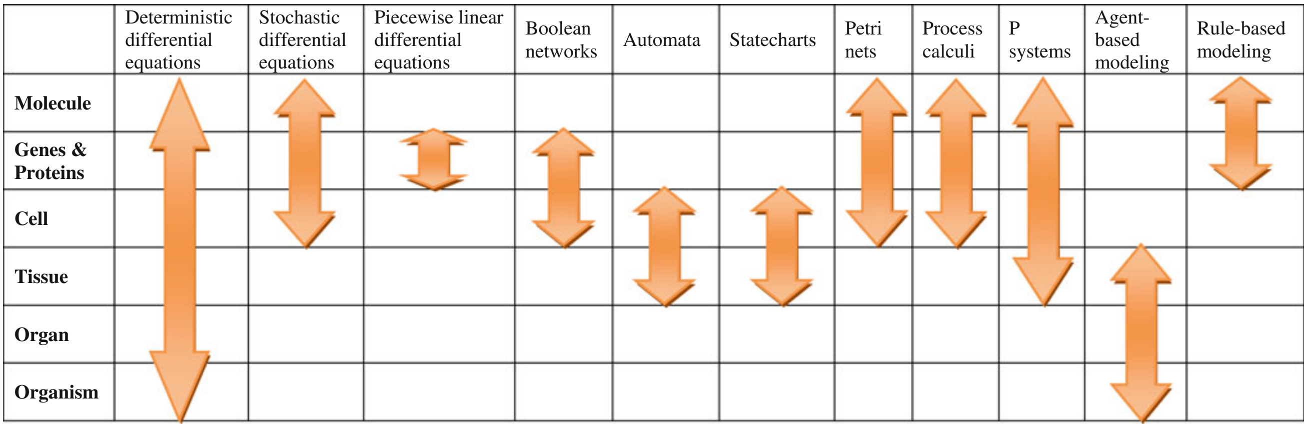

Fig. 2 Typical scales of application of various formalisms

scales at which most applications of a certain formalism are concentrated.

3. *Size of the model*—Models involving large number of entities are easier to describe and use in compositional formalisms which allow hierarchical organization of entities and implicit definition of entity subtypes.
4. *Nature of available data*—The nature and amount of data required to construct a model and learn its

parameters varies for different formalisms. For example, constructing a model in a quantitative formalism may require comprehensive quantitative experimental data. On the other hand, models can be constructed in qualitative formalisms with little or no quantitative data.

5. *Availability of software tools*—Some formalisms are popular for certain modeling applications because high-quality software tools are readily available. For example, Petri nets are frequently used for modeling intracellular processes since a well-developed tool Cell Illustrator [17] is available for the purpose.

This review discusses the basic modeling approach and characteristics of a few important formalisms for systems immunology modeling. Some important characteristics of various formalisms are compared in Table 2. For further technical details, pointers are given in Table 2 to recent reviews and helpful references. The first step toward choosing a suitable modeling formalism is to explore which formalisms have already been used in the literature for modeling similar systems. This review attempts to review salient modeling applications of each formalism.

For implementation of systems immunology models, a wide variety of software tools is available. Models can be implemented by programming in general purpose programming or scripting languages like C++, Matlab, Mathematica, R, Python, etc. Otherwise, one can use specialized tools that are preprogrammed to implement models in a particular formalism. General purpose programming languages provide flexibility of implementing any conceivable model, but programming can be tedious and prone to errors and may involve duplication of efforts in rebuilding what has already been implemented by others. Specialized tools, though limited in functionality, can save time and effort and allow one to focus on modeling rather than programming issues.

For immunologists beginning to take interest in systems immunology, a good starting point would be hands-on experience with existing tools that are user-friendly and well documented and come with a set of example models. This review describes a sample of available software tools, which are in this sense ‘biologist-friendly’.

In terms of usage, software tools can be of various types such as a working environment, an application, a plugin, shell or a library. These terms are explained in Table 3. Usually, working environments are a good starting point as they provide a complete set of tools for a typical modeling workflow including model construction, parameter estimation, model checking, simulation, visualization and typical analyses on a model.

Databases are very important part of modeling software, which organize and curate experimental data and models. However, in this review, we will not discuss about databases. We refer the reader to some recent reviews on this topic [18–20].

Differential equations

Formalism

Differential equation models give a complete quantitative description of system dynamics in continuous time and space. In a differential equation model, all entities in a system are represented by some real-valued quantitative attributes such as their total numbers, concentrations, masses, etc. All events in the system such as production, degradation and transport of entities and interactions among entities are expressed in the form of chemical

Table 2 Characteristics of various formalisms for systems biology modeling

Formalism	Determinism	Quantities	Time	Type of analysis	Composition (Hierarchy)	Scalability	Review papers
Deterministic differential equations	Deterministic	Continuous	Continuous	Quantitative	No	Low	[137, 138]
Stochastic differential equations	Stochastic	Continuous	Continuous	Quantitative	No	Low	[139]
Piecewise linear differential equations	Deterministic	Continuous	Continuous	Qualitative	No	Medium	[140]
Boolean networks	Deterministic	Discrete	Discrete	Qualitative	No	Medium	[141]
Automata	Deterministic	Discrete	Discrete	Qualitative	No	Low	None
Statecharts	Stochastic	Discrete	Discrete	Qualitative	Yes	High	[142]
Petri nets	Stochastic	Both	Discrete	Quantitative	No	Low	[115]
Process calculi	Stochastic	Continuous	Discrete	Quantitative	Yes	High	[143]
P systems	Stochastic	Discrete	Discrete	Qualitative	Yes	Low	[144]
Agent-based modeling	Stochastic	Discrete	Discrete	Qualitative	No	Low	[145]
Rule-based modeling	Stochastic	Continuous	Continuous	Quantitative	No	Medium	[146]

Table 3 Categories of software tools

Term	Meaning
Programming environment	Software that allows users to define their own tasks by writing a sequence of instructions, called a *code* or *program*, in a computer language. For example, Matlab is a programming environment where users can write programs for performing a variety of computational tasks
Application	An executable software, which can perform a fixed set of tasks. For example, CellDesigner is an application for drawing gene-regulatory and biochemical networks
Plugin	An addition or extension to an application at runtime. For example, SimBoolNet is a plugin for Cytoscape, which adds functionality for simulating Boolean networks
Library	A collection of reusable software code which software developers can include in their own programs. Software developers share pieces of their software code with each other in the form of libraries. For example, libSBML is a library for working with SBML models
Shell	Software that provides a programming interface to an application. It allows the user to write a "script", which is a series of commands that the application will execute. This makes it easier and faster to work with an application. For example, Jarnac is an application for modeling ODEs, which uses a shell for programming modeling tasks
Working environment	An environment (which can be an application, or a shell, or a graphical user interface) that allows using different applications, which could otherwise be used on their own, within a common framework. It allows easy sharing of data and results between different applications. The interactions can be hardwired (in case of an application or a GUI) or programmable (in case of a shell). For example, Systems Biology Workbench (SBW) is a working environment that integrates a number of applications for ODE model creation, analysis and simulation
Module	A general term referring to a unit or an independent component, which can be a piece of code in a program or script, or a plugin to an application, or an application in the context of a working environment

reactions. The rate of change of the quantity of products is represented as a continuous function of the quantities of reactants. The set of differential equations (or *rate equations*) constitutes the model, and its parameters are called *rate constants*. The rate constants are estimated from experimental data. Ordinary differential equations (ODEs) are used when only temporal dynamics are considered, whereas partial differential equations (PDEs) are used for dynamics in both time and space. Delay differential equations (DDE) are used in systems with delays or lags such as gestation times or transport delays.

Differential equation models can be deterministic or stochastic. Deterministic differential equation models are used when quantities of entities can be precisely measured and the network of interactions among entities is static. On the other hand, when the entities are present in small numbers (of the order of hundreds or less) or there are dynamic interactions or external factors such as different initial states, timing variability of responses, random external inputs, etc., stochasticity can no longer be ignored. Stochastic models such as chemical master equation (CME), stochastic differential equations (SDE), continuous-time Markov chains (CTMC) and Bayesian dynamics model (BDM) are used when these effects have to be explicitly accounted [21].

To simulate a differential equation model, the set of equations is integrated with given initial and boundary conditions to yield exact numerical quantities of all entities in continuous time. Analytical solution is seldom feasible; therefore, simulation is usually performed numerically using computational algorithms.

Applications

Differential equations models of the immune system are reviewed in [22–24]. Systems of ODEs have been extensively used for modeling intracellular (genetic, signal transduction and metabolic) networks. Precise continuous-time dynamics of small networks containing up to tens of entities (genes, proteins or metabolites) is frequently studied using ODE models. However, for larger networks, qualitative formalisms such as Boolean networks are preferred due to ease of parameter estimation and analysis. An important analysis performed with an ODE model of an intracellular network is the exploration of its steady states, which is equivalent to understanding its equilibrium states or homeostasis [26, 40, 41].

Models have been constructed to study the signaling events in macrophages triggered by AvCystatin [25], the IFN-related regulatory network [26] and NF-kappaB signaling [27, 28]. Dynamics of gene expression in response to activators and inhibitors is described using usual rules of reaction kinetics along with Michaelis–Menten enzyme kinetic equations and Hill function. These equations can reproduce the characteristic sigmoid shaped dynamics of gene transcription. ODE-based approaches have also been used to study cancer–immune interactions [29–31], natural killer cell response [32], B cell memory [33, 34], the role of inflammation in atherosclerosis [35], interaction of HIV with CD4+ T cells [36], virus-neutralizing immunoglobulin response [37], and cytotoxic T cell proliferation [38] and activation [39].

Table 4 Comparison of software tools for deterministic differential equation modeling

Software	Category	Modeling functions					
		Construction	Calibration	Analysis	Simulation	Visualization	Scripting
Systems Biology Workbench (SBW) Complaint Modules and Application							
BioModels importer	SBW module	X					
CellDesigner	Application	X					
JDesigner	SBW module	X					
JSim	Application	X	X	X	X	X	
Copasi	Application	X	X	X	X	X	
Gillespie Simulator	SBW module				X		
Jarnac	Application	X		X	X	X	X
Roadrunner	SBW module			X	X		
Auto C#	SBW module			X			
Bifurcation discovery tool	SBW module			X			
Frequency analysis	SBW module			X			
Jacobian viewer	SBW module			X			
3D Simulation	SBW module					X	
Autolayout	SBW module					X	
Others							
Gepasi	Application	X	X	X	X	X	
BioUML	Working environment	X	X		X	X	X
Mathematica with MathSBML	Programming environment	X	X	X	X	X	X
MATLAB with SimBiology and SBToolbox2	Programming environment	X	X	X	X	X	X
SBML ODE solver	Library		X		X		
JWS Online	Webserver				X	X	
Cytoscape	Application	X				X	
Kinetics Inference (Klnfer)	Application		X				
Bio-SPICE	Working environment	X	X	X	X	X	
Python with PySCes	Programming environment	X		X	X	X	X
Virtual Cell*	Application	X		X	X	X	
ECell	Application	X		X	X	X	
EnSuite/EnCORE	Not yet mature						
Biodyn	Web Application		X	X			
Oscill8				X	X		

* Virtually Cell supports geometry modeling, which is an additional feature compared to other tools in the list.

Stochastic modeling is common at the intracellular scale for gene expression and signaling [42]. Lysis-lysogenic fate choice of λ-phage-infected Escherichia coli [43] is a classic example of a system where stochastic kinetic modeling of gene expression is required to predict probabilistic outcomes. Here, regulatory proteins Cro and CI compete at low concentrations to control the switching of gene regulatory network between lysogenic and lytic fates, resulting in a probabilistic pathway selection. Other examples of stochastic modeling include terminal differentiation of B lymphocytes [44], variations or oscillations in NFkB pathway [45], optimization of immunoglobulin substitution therapy [46], and viral replication in cells where few virus particles can initiate infection [47].

Simulation tools

Modeling a biological system with differential equations involves multiple related tasks including model construction, verification, calibration (parameter learning), analysis, validation, simulation and visualization. Currently there is no single software tool, which can perform all of these tasks equally well. Therefore use of multiple tools is imperative. Table 4 lists some software tools for working with differential equation models.

A usual problem in using different software tools is that each tool has its own way of handling data, which makes it difficult to share information across different tools. Efforts toward integration of systems biology software tools has resulted in projects such as Systems Biology Markup Language (SBML) [48], CellML [49], Systems Biology Workbench (SBW) [50], and so forth. SBML and CellML define common formats for sharing model structure, parameters and simulation results across different software tools. Almost all software tools today are compatible with SBML. SBW is a working environment, which integrates many software applications and makes it easy to share information across them. Within the SBW, CellDesigner, JDesigner, Copasi and JSIM are useful graphical or scripting tools for constructing and editing models in SBML format. Model parameters can be estimated from an experimental dataset or optimized over a given range using JSim and Copasi. Analyses on a model, such as steady state analysis, sensitivity analysis, mass conservation analysis, etc. are available in JSim, Copasi, Jarnac, Roadrunner, AutoC#, Frequency Analysis, and Jacobian viewer. No tool for model verification exists. However, partial verification is possible in Copasi. Time course simulation can be performed in JSim, Copasi, Gillespie simulator, Jarnac and Roadrunner. Through SBW the same model can be passed around to different tools so that all of these tasks can be performed within a single working session.

Apart from SBW, there are a number of independent software tools that support systems biology modeling and simulation as shown in Table 4. Being independent, each tool tends to support as many features as possible within it. For advanced users, Matlab, Mathematica, Python and C++ are general purpose programming environments. Biology specific capabilities such as support for SBML format, graphical interface for model construction, stochastic simulation, etc. are available through toolboxes such as SimBiology and SBToolBox2 for Matlab, MathSBML for Mathematica, PySCes library for Python and libSBML and SBML ODE solver libraries for C++.

Examples of differential equation models for immunology are stored in the BioModels database. These can be imported into SBW using the Biomodels importer and serve as a useful guide for immunologists.

Automata and statecharts

Formalism

In contrast with differential equations, which describe the immune system in *continuous* time and space, automata provide a *discretized* approach where the immune system is regarded as a system having a finite number of states, with one of these states being the current system state. The automaton receives inputs and 'jumps' to the next state depending upon its current state and the input. A transition function defines a mapping to a next state given the current state and an input. An automaton is frequently visualized as a graph in which each state is a node and possible state transitions are indicated by directed edges between the nodes. Edges are marked with the inputs or conditions, which enable that transition.

An automaton, which contains a finite number of states and receives a string of inputs in discrete time is called a *finite state automaton*. A *cellular automaton* [51] is a special finite state automaton where states, or 'cells', are arranged in a grid in any finite number of dimensions. It is useful when spatial organization or patterns are important in a model. Cellular automata have received significant attention due to their ability to exhibit complex *system*-level emerging behavior, which often strikingly reflect real biological phenomena, using relatively simple *local* transition rules. A *hybrid automaton* is a mixed formalism with both discrete states and continuous variables. A *statechart* is an extension of a finite state automaton to include support for additional features such as nested state hierarchy, concurrency and event broadcasting, which makes models more powerful and scalable.

Applications

Cellular automata have been used in several immunological applications for investigating specific mechanisms of the immune system such as affinity maturation and hypermutation in the humoral immune system [52] and tolerance to pathologic rheumatoid factors [53]. Cellular automata are suitable for studying self-organization and colony formation of tumor cells and effect of the environment and stroma on tumor progression. Thus cellular automata and hybrid variant have been extensively used for modeling the dynamics of various diseases such as tumor growth and invasion [54–64]. In these computer simulations, specific diseases' factors (e.g., rate of spread, binding affinities of cytokines) can varied to obtain a better understanding of the diseases' factors and how they may impact the spread dynamics of the disease. Further diseases investigated using cellular automata include: HIV [65, 66], Epstein–Barr [67] and M. tuberculosis [68, 69].

Hybrid automata have been used in two types of applications: which involve a switch like behavior, and those which extend cellular automata to a hybrid (mixed discrete continuous) type. The former are mostly used for modeling gene regulatory and signal transduction networks, while the latter are mostly used for modeling at the tissue level including tumors and immune system.

Kam et al. [70] first applied statecharts for modeling the transition of T cells into active or anergy states depending upon protein tyrosine kinase (PTK) activity, costimulation and inhibition. Subsequently, Efroni et al. modeled T-cell maturation in thymus [71] and thymocyte development [72]. Large numbers of individual T cells expressing different receptors were simulated and their interaction with epithelial cells could be studied in real time by simulation. In another study, Naamah et al. [73] studied how lymph nodes orchestrate the interaction between antigens and various B and T cells bearing receptors for these antigens. States and transitions in the statechart framework were used to simulate the activities in lymph node regions, including immune cell behavior, receptors, interactions and movement of cells, cell proliferation and differentiation.

Simulation tools

Popular applications that supports modeling of various biological systems with cellular automata include DDlab and Cell-Devs. The latter is a programming environment for advanced users. CHARON is a programming environment, which has been extensively used for working with hybrid automata models of biological systems. Bio Sketch Pad is a graphical tool for supporting formal modeling, analysis and simulation of biochemical and cellular networks with CHARON. Since its early introduction in 1992 by Celada and Seiden, the software package IMMSIM-C [74, 75][1] has been employed numerous times for immune systems modeling using hybrid cellular automata. In contrast with Cell-Devs, the utilization of IMMSIM-C does not require expertise in computer programming. As such, this modeling platform has been used for educational purposes for undergraduate immunology courses at various universities (e.g., Princeton University, University of Genoa). Also, IMMSIM-C was ported as a public web application[76][2] where users can readily implement and simulate immune system models online without requiring any computational skills.

Though many programming environments exist for modeling with statecharts, only Rhapsody has been used so far for systems immunology modeling. Rhapsody provides provisions for drawing statechart models and defining transitions and other actions. It translates the statechart into an executable Java/C++ code. Another software called BioCharts, which is specifically meant for biological system modeling with statecharts, is under development [77].

Agent-based models

Formalism

Agent-based modeling is a distinct paradigm compared to the formalisms discussed above in at least two major ways. Firstly, instead of measuring average or ensemble properties of entities such as concentration and total numbers, agent-based models deal with discrete agents, which could be individual molecules, cells, etc. The agents are autonomous, that is, they are not passively manipulated but can make independent decisions. In some models, the agents can learn and adapt their behavior. Secondly, the interaction topology among the agents is a complex dynamic graph. Agents can be free to move within a space. They contact other agents and the environment by chance and, during these interactions, change their state in accordance with a set of well-defined rules. Agent-based models are pertinent for modeling biological systems involving dynamic interactions among heterogeneous components. The model is capable of re-creating macrolevel phenomena by the actions and interactions of microlevel individual agents. This phenomenon is called *emergence*. Agent-based models pay attention to the behavior of individual agents, which is not possible in formalisms that characterize ensemble properties of entities.

Applications

ABMs are suitable for modeling biological phenomena at the multicellular level. Cells as agents are a natural metaphor. Thus, ABMs have been commonly used for modeling inflammatory and immune response, tumor formation, tissue morphogenesis, etc. Systemic inflammatory response syndrome (SIRS) is an early and successful application of ABMs [78–80]. The system has complex, nonlinear dynamics driven by multiple feedback loops, due to which analysis of isolated components fails to explain the systemic responses. For instance, nitric oxide is responsible for lethal drop in blood pressure in SIRS, but paradoxically, treatment with nitric oxide improved survival in animal models [81]. The ABM was intended to reproduce systemic behaviors from simple rules known to operate at the cellular level. Despite several simplifications, the model reproduced diverse clinical outcomes in SIRS and provided practical insights into improving the treatment protocol. The model also addressed a controversial issue whether stimulation of pro- and anti-inflammatory cellular responses is concurrent or separated by a time lag. ABM of inflammation continues to be an active area of research [82–85] and has been extended to other related applications as well, such as modeling of Chagas' disease [86, 87], vocal fold inflammation [88], ischemia [89], leukocyte

[1] http://www.immsim.org/.

[2] http://www.cbs.dtu.dk/services/C-ImmSim-10.1/.

rolling and adhesion [90], etc. ABMs have been recently been combined with ODE and mechanotransduction models in multiscale modeling of human epidermis [91].

Various aspects of immune system response have been modeled using ABMs. Some studies have modeled immune responses to infections, such as formation of granuloma in tuberculosis [92], formation of atheromatous plaque in hypercholesterolemia [93], and the sites of Epstein–Barr virus infection and persistence [94]. Other studies have focused on the internal mechanisms of the immune system such as the recognition of antigens by T cells [95], B-cell activation following ligand presentation [96], T-cell activation and proliferation in lymph nodes [97], B-cell selection in germinal centers [98], unbalanced differentiation of precursor T helper cells to TH1 and TH2 phenotypes [99], formation and recall of antivirus immunological memory [100], etc. Other studies have attempted to build a comprehensive model of the immune system as a whole [101–105].

Simulation tools

Though many software tools exist for agent-based modeling [106, 107], only a handful are useful for systems biology modeling. All agent-based modeling tools require some programming skills. NetLogo is an easy-to-use platform [108, 109] suited for prototyping agent-based models. However, large-scale simulations would require a more powerful platform. Swarm is a historical ABM platform. Swarm models are written in Objective C, which makes it accessible only to experienced programmers, though it now also supports Java. MASON is a Java library with emphasis on cross-platform portability, fast running simulations and reduced space requirements. It is suitable for batch simulations. Repast is one of the most advanced agent-based modeling platforms. It is well suited for working with large and complicated models. Though it is also built in Java, its user-friendly interface called Repast-symphony helps keep programming minimal. Another useful feature is its automated integration with many external tools such as Matlab, R, SQL, VisAD (scientific visualization), JUNG (network modeling), Excel and GIS software. It supports adaptation and learning of agents through evolutionary algorithms, as well as model optimization over Monte Carlo simulations. It supports large-scale simulations though multithreading and distributed computing.

Other formalisms

The discussion above covered some of the primary formalisms that have been used for modeling and simulation of immune systems. However, there is also literature on various other formalisms that have been used for modeling in immunological applications.

Boolean or multilevel network models are used in applications where there is a fixed network and the entities can take on a discrete number of states, ranging from two (boolean) to multiple (multilevel). For example, in a gene regulatory network, each gene may be represented by two states (OFF or ON) or three states (LOW, MEDIUM or HIGH). Using discrete states instead of numerical values makes the model simple and easy to analyze while capturing its essential dynamics. Boolean and multilevel network models have been used for modeling T-cell differentiation [110, 111], T-cell activation [112], inflammatory signaling [113] and host–pathogen signaling [114].

A limitation of Boolean networks is that the states of entities must be updated one by one. Thus, they cannot perfectly model concurrent or simultaneous events. Immunological systems frequently function as concurrent systems. For example, in humoral immunity B-cell production, antigen presentation, B-cell activation, antibody secretion and virus clearance are all concurrent processes. In signaling networks, protein products of one event dynamically serve as reactants or enzymes for the next set of events, and thus, the production and consumption of proteins are concurrent events. *Petri nets* [115] and *process calculi* [116] are two other formalisms based on fixed network and discrete representations of state and time, which are especially suited for modeling *concurrent* and *distributed* systems. Petri nets have a graphical representation, while process calculi have a textual representation, so Petri nets can be easier to understand for nonspecialists. These formalisms also provide formal analysis and verification techniques (such as model checking [117]), which makes it possible to analyze all possible model behavior and causal and conflicting relationships. So they can answer questions like is it possible to completely eliminate a pathogen, will the parasite be able to complete its life cycle, etc. An integrated Petri net model of the immune system including innate, humoral and cellular immunity has been reported in [118]. Similarly models of immune system in process calculus, such as activation of helper T cells, are reported in [119, 120].

P-systems or membrane computing is an emerging formalism for modeling concurrent and distributed systems. Its specific advantage is that it explicitly models modules or groups of entities separated by compartments and together performing a specific function. Rules specify the evolution and transport of entity groups. This makes it easier to model some phenomena such as selective transport of entities across membranes. P-systems have been used to model the life cycle of infectious viruses [121, 122].

Toward integrative modeling simulation platforms and evolutionary systems immunology

In the previous sections, the principal modeling formalisms and associated simulation tools were presented. In addition, a range of relatively less prominent and more recent modeling approaches were briefly described. Although this survey is far from being exhaustive, it can be readily noted that a wide variety of modeling formalisms exists where there is no single formalism that can be clearly identified as the universal best. This is also reflected in the literature, where new modeling variants are continuously being developed, addressing specific modeling needs and immunological research questions. Although this may appear to be a diverging and ultimately counter-productive research effort, this diversity of modeling formalisms is actually regarded as necessary and desirable by the authors. What we believe is most important, and currently not developed enough in systems immunology, is the access and utilization of modular and integrative frameworks to facilitate the coupling of differing modeling formalisms. This also implies the need for meta-models (and public repository for such models) accounting for the multilevel, hierarchical nature of the immune system. Modularity is here a key feature for such a framework, which would enable the 'plug-in' of the different modeling formalisms and simulation tools while preserving coherency of the different modeled entities in both time and space [123, 124].

We propose an extension to *in silco* systems immunology to investigate the ability of the immune system to adapt to environmental changes through an evolutionary process. Such an evolutionary approach to systems immunology may be of interest due to the natural ability of pathogens (e.g., HIV) to adapt through evolution [125, 126]. To date, a number of theoretical studies have been conducted toward this direction [127–131] where the adaptive properties of *artificial* immune systems have been examined while involving the 'innovation' process through mutations and selections. These algorithms have been shown to be efficient when applied to high-dimensional search or optimization problems [132] and have inspired novel techniques in computer science to realize computer intrusion detection systems [133]. However, when considering the modeling and simulation of *natural* immune systems evolution, relatively less studies have been reported in the literature [134, 135]. To explore further this research direction, significant challenges would have to be addressed. Particularly, preserving biological plausibility would be a hard constraint to ensure that the *in silico* evolutionary exploration of immune systems remains within acceptable and reasonable boundaries. A strong focus should thus be placed on model validation. For instance, tracing the in vivo evolution of pathogens [136] is an example of data useful for validating *in silico* predictions. This proposed evolutionary systems immunology approach may then, ultimately, provide a complementary and potentially useful tool for immunological and epidemiological studies.

Conclusions

We hope that this review lowers the entry level for biologists to systems biology modeling and simulation. We gave pointers to formalisms and also a list of their biological applications to not only illustrate their power and usage but also help biologists with specific problems at hand. This can serve as a reference to quickly identify which formalism could be of interest. Finally, we believe that providing the list of corresponding software is the key to adoption of modeling and simulation within biology research laboratories. This last and practical section makes this review a good entry point for biologists who wish to not just build a theoretical foundation but to start experimenting with computational models.

We gave insights into preliminary questions whose answers can guide the choice of an appropriate formalism. These questions are listed in Sect. 2 (Modeling and Simulation) and further expanded in Tables 1 and 2. Figure 2 gives an overview of which formalism could be used depending upon the biological level or scale of the problem at hand. Finally, the software part (as illustrated by Table 4 for differential equations) guides one to the right software tools to begin modeling and experimenting in a chosen formalism.

We emphasized the fact that there is no universal best modeling formalism. Modeling something as complex as the immune system will require integration of different formalisms in a common multiscale transcale framework. Thus, we outlined move toward integrative modeling as the next challenge for in silico systems immunology. We also proposed modeling of the adaptive properties of the immune system through evolution as a less studied but promising challenge for systems immunologists.

Finally, we would like to emphasize the fact that there is no modeling without data. Therefore, it is of utmost importance to ensure that accurate and appropriate data are available or that data generation and acquisition is included in the modeling strategy.

Acknowledgments The authors thank P.S. Thiagarajan for helpful comments.

References

1. Hooke RC. Micrographia: or some physiological descriptions of miniature bodies made by magnifying glasses. London: Jo. Martyn, and Ja. Allestry; 1665.
2. Ge H, Walhout AJM, Vidal M. Integrating 'omic' information: a bridge between genomics and systems biology. Trends Genet. 2003;19(10):551.
3. Joyce AR, Palsson B. The model organism as a system: integrating 'omics' data sets. Nat Rev Mol Cell Biol. 2006;7(3),198. doi:10.1038/nrm1857.
4. Regenmortel MHVV. Reductionism and complexity in molecular biology. EMBO Rep. 2004;5(11):1016. doi:10.1038/sj.embor.7400284.
5. Schnell S, Grima R, Maini PK. Multiscale modeling in biology. Am Sci. 2007;95(2):134. doi:10.1511/2007.64.1018.
6. Young D, Stark J, Kirschner D. Systems biology of persistent infection: tuberculosis as a case study. Nat Rev Microbiol. 2008; 6(7):520. doi:10.1038/nrmicro1919.
7. Benoist C, Germain RN, Mathis D. A plaidoyer for systems immunology. Immunol Rev. 2006;210(1):229. doi:10.1111/j.0105-2896.2006.00374.x. URL http://www.ingentaconnect.com/content/mksg/imr/2006/00000210/00000001/art00015.
8. Kitano H. Systems biology: a brief overview. Science. 2002;295(5560):1662. doi:10.1126/science.1069492.
9. Bokulich A. How scientific models can explain. Synthese. 2009; eprint 1. doi:10.1007/s11229-009-9565-1.
10. Craver C. When mechanistic models explain. Synthese. 2006; 153(3):355. doi:10.1007/s11229-006-9097-x.
11. Vodovotz Y. Deciphering the complexity of acute inflammation using mathematical models. Immunol Res. 2006;36(1–3):237. doi:10.1385/IR:36:1:237.
12. Materi W, Wishart DS. Computational systems biology in cancer: modeling methods and applications. Gene Regul Syst Bio. 2007;1:91.
13. Guo Z, Sloot PMA, Tay JC. A hybrid agent-based approach for modeling microbiological systems. J Theor Biol. 2008;255(2): 163. doi:10.1016/j.jtbi.2008.08.008.
14. Newman SA, Christley S, Glimm T, Hentschel HGE, Kazmierczak B, Zhang YT, Zhu J, Alber M. Multiscale models for vertebrate limb development. Curr Top Dev Biol. 2008;81:311. doi:10.1016/S0070-2153(07)81011-8.
15. Fisher J, Henzinger TA. Executable cell biology. Nat Biotechnol. 2007;25(11):1239. doi:10.1038/nbt1356.
16. Kanehisa M, Goto S, Kawashima S, Okuno Y, Hattori M. The kegg resource for deciphering the genome. Nucleic Acids Res. 2004;32(Database issue):D277. doi:10.1093/nar/gkh063.
17. Nagasaki M, Doi A, Matsuno H, Miyano S. A versatile petri net based architecture for modeling and simulation of complex biological processes. Genome Inform. 2004;15(1):180.
18. Eils J, Lawerenz C, Astrahantseff K, Ginkel M, Eils R. Computational systems biology (Elsevier, Amsterdam [u.a.]), chap. Databases for systems biology; 2005. p. 15–38.
19. Ng A, Bursteinas B, Gao Q, Mollison E, Zvelebil M. Resources for integrative systems biology: from data through databases to networks and dynamic system models. Brief Bioinform. 2006;7(4):318. doi:10.1093/bib/bbl036.
20. van Gend C, Snoep JL. Systems biology model databases and resources. Essays Biochem. 2008;45:223. doi:10.1042/BSE0450223.
21. Gillespie DT. Stochastic simulation of chemical kinetics. Annu Rev Phys Chem. 2007;58:35. doi:10.1146/annurev.physchem.58.032806.104637.
22. Yates A, Chan CC, Callard RE, George AJ, Stark J. An approach to modelling in immunology. Brief Bioinform. 2001;2(3):245.
23. Andrew SM, Baker CT, Bocharov GA. Rival approaches to mathematical modelling in immunology. J Comput Appl Math. 2007;205(2):669. URL http://www.sciencedirect.com/science/article/B6TYH-4KGPND0-7/2/879347b51f188098ab9e03e0fb0612a8.
24. Kim PS, Levy D, Lee PP. Modeling and simulation of the immune system as a self-regulating network. Methods Enzymol. 2009;467:79. doi:10.1016/S0076-6879(09)67004-X.
25. Klotz C, Ziegler T, Figueiredo A, Rausch S, Hepworth M, Obsivac N, Sers C, Lang R, Hammerstein P, Lucius R, et al. A helminth immunomodulator exploits host signaling events to regulate cytokine production in macrophages. PLoS Pathogens. 2011;7(1):e1001248.
26. Smieja J, Jamaluddin M, Brasier A, Kimmel M. Model-based analysis of interferon-β induced signaling pathway. Bioinformatics. 2008;24(20):2363.
27. Hoffmann A, Levchenko A, Scott ML, Baltimore D. The ikappab-nf-kappab signaling module: temporal control and selective gene activation. Science. 2002;298(5596):1241. doi:10.1126/science.1071914.
28. Lipniacki T, Paszek P, Brasier ARAR, Luxon B, Kimmel M. Mathematical model of nf-kappab regulatory module. J Theor Biol. 2004;228(2):195. doi:10.1016/j.jtbi.2004.01.001.
29. de Pillis L, Radunskaya A, Wiseman C. A validated mathematical model of cellmediated immune response to tumor growth. Cancer Research. 2005;65(17):7950.
30. Pennisi M, Bianca C, Pappalardo F, Motta S. Modeling artificial immunity against mammary carcinoma. In: Proceedings of the 10th International Conference on Mathematical Methods in Science and Engineering (CMMSE 2010); 2010. p. 753–756.
31. Pennisi M, Bianca C, Pappalardo F, Motta S. Compartmental mathematical modeling of immune system—melanoma competition. In: Proceedings of the 10th International Conference on Mathematical Methods in Science and Engineering (CMMSE 2011); 2011. pp. 930–934.
32. Merill SJ. A model of the role of natural killer cells in immune surveillance. J Math Biol. 1981;12:363.
33. Varela F, Stewart J. Dynamics of a class of immune networks i. global stability of idiotype interactions. J Theor Biol. 1990; 144(1):93.
34. De Boer R, Perelson A. Size and connectivity as emergent properties of a developing immune network. J Theor Biol. 1991;149(3):381.
35. Ougrinovskaia A, Thompson RS, Myerscough MR. An ode model of early stages of atherosclerosis: mechanisms of the inflammatory response. Bull Math Biol 2010;72(6):1534. URL http://www.ncbi.nlm.nih.gov/pubmed/20440571.
36. Essunger P, Perelson AS. Modeling hiv infection of cd4+ t-cell subpopulations. J Theor Biol. 1994;170(4):367.
37. Funk G, Barbour A, Hengartner H, Kalinke U. Mathematical model of a virusneutralizing immunglobulin response. J Theor Biol. 1998;195(1):41.
38. Wodarz D, Thomsen A. Effect of the ctl proliferation program on virus dynamics. Int Immunol. 2005;17(9):1269.
39. Pennisi M, Pappalardo F, Chiacchio F, Motta S. A model of cytotoxic t antitumor activation stimulated by pulsed dendritic cells. In: Simos TE, Psihoyios G, Tsitouras C, Anastassi Z, editors. American Institute of Physics Conference Series, American Institute of Physics Conference Series, vol. 1389, American Institute of Physics Conference Series. 2011. p. 1236–1239.
40. Werner S, Kearns J, Zadorozhnaya V, Lynch C, ODea E, Boldin M, Ma A, Baltimore D, Hoffmann A. Encoding nf-κb temporal control in response to tnf: distinct roles for the negative regulators iκbα and a20. Genes Dev. 2008;22(15):2093.

41. Shih VF, Kearns JD, Basak S, Savinova OV, Ghosh G, Hoffmann A. Kinetic control of negative feedback regulators of NF-kappaB/RelA determines their pathogen- and cytokine-receptor signaling specificity. Proc Nat Acad Sci. 2009;106(24):9619. doi:10.1073/pnas.0812367106.
42. Kepler TB, Elston TC. Stochasticity in transcriptional regulation: origins, consequences, and mathematical representations. Biophys J. 2001;81(6):3116. doi:10.1016/S0006-3495(01)75949-8.
43. Arkin A, Ross J, McAdams HH. Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected escherichia coli cells. Genetics 1998;149(4):1633.
44. Zhang Q, Bhattacharya S, Kline DE, Crawford RB, Conolly RB, Thomas RS, Kaminski NE, Andersen ME. Stochastic modeling of b lymphocyte terminal differentiation and its suppression by dioxin. BMC Syst Biol.2010;4:40. doi:10.1186/1752-0509-4-40.
45. Lipniacki T, Paszek P, Marciniak-Czochra A, Brasier AR, Kimmel M. Transcriptional stochasticity in gene expression. J Theor Biol. 2006;238(2):348. doi:10.1016/j.jtbi.2005.05.032.
46. Figge M. Optimization of immunoglobulin substitution therapy by a stochastic immune response model. PloS one. 2009;4(5): e5685.
47. Srivastava R, You L, Summers J, Yin J. Stochastic vs. deterministic modeling of intracellular viral kinetics. J Theor Biol. 2002;218(3):309.
48. Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC, Kitano H, Arkin AP, Bornstein BJ, Bray D, Cornish-Bowden A, Cuellar AA, Dronov S, Gilles ED, Ginkel M, Gor V, Goryanin II, Hedley WJ, Hodgman TC, Hofmeyr JH, Hunter PJ, Juty NS, Kasberger JL, Kremling A, Kummer U, Novre NL, Loew LM, Lucio D, Mendes P, Minch E, Mjolsness ED, Nakayama Y, Nelson MR, Nielsen PF, Sakurada T, Schaff JC, Shapiro BE, Shimizu TS, Spence HD, Stelling J, Takahashi K, Tomita M, Wagner J, Wang J, Forum SBML. The systems biology markup language (sbml): a medium for representation and exchange of biochemical network models. Bioinformatics. 2003;19(4):524.
49. Lloyd CM, Halstead MDB, Nielsen PF. Cellml: its future, present and past. Prog Biophys Mol Biol. (2004);85(2–3):433. doi:10.1016/j.pbiomolbio.2004.01.004.
50. Bergmann FT, Sauro HM. Sbw—a modular framework for systems biology. In: WSC '06: Proceedings of the 38th conference on Winter simulation (Winter Simulation Conference); 2006. p. 1637–1645.
51. Wolfram S. Cellular automata as models of complexity. Nature. 1984;311(5985):419.
52. Celada F, Seiden P. Affinity maturation and hypermutation in a simulation of the humoral immune response. Eu J Immunol. 1996;26(6):1350.
53. Stewart J, Agosto H, Litwin S, Welsh J, Shlomchik M, Weigert M, Seiden P. A solution to the rheumatoid factor paradox: pathologic rheumatoid factors can be tolerized by competition with natural rheumatoid factors. J Immunol. 1997;159(4):1728.
54. Patel AA, Gawlinski ET, Lemieux SK, Gatenby RA. A cellular automaton model of early tumor growth and invasion. J Theor Biol. 2001;213(3):315. doi:10.1006/jtbi.2001.2385.
55. Dormann S, Deutsch A. Modeling of self-organized avascular tumor growth with a hybrid cellular automaton. In Silico Biol. 2002;2(3):393.
56. Gevertz JL, Torquato S. Modeling the effects of vasculature evolution on early brain tumor growth. J Theor Biol. 2006;243(4): 517. doi:10.1016/j.jtbi.2006.07.002.
57. Mallet DG, Pillis LGD. A cellular automata model of tumor-immune system interactions. J Theor Biol. 2006;239(3):334. doi:10.1016/j.jtbi.2005.08.002.
58. Bankhead A, Magnuson NS, Heckendorn RB. Cellular automaton simulation examining progenitor hierarchy structure effects on mammary ductal carcinoma in situ. J Theor Biol. 2007; 246(3):491. doi:10.1016/j.jtbi.2007.01.011.
59. Gerlee P, Anderson ARA. An evolutionary hybrid cellular automaton model of solid tumour growth. J Theor Biol. 2007;246(4):583. doi:10.1016/j.jtbi.2007.01.027.
60. Gerlee P, Anderson ARA. A hybrid cellular automaton model of clonal evolution in cancer: the emergence of the glycolytic phenotype. J Theor Biol. 2008;250(4):705. doi:10.1016/j.jtbi.2007.10.038.
61. Basanta D, Strand DW, Lukner RB, Franco OE, Cliffel DE, Ayala GE, Hayward SW, Anderson ARA. The role of transforming growth factor-beta-mediated tumor-stroma interactions in prostate cancer progression: an integrative approach. Cancer Res. 2009;69(17):7111. doi:10.1158/0008-5472.CAN-08-3957.
62. Mallet DG, Heymer KJ, Rank RG, Wilson DP. Chlamydial infection and spatial ascension of the female genital tract: a novel hybrid cellular automata and continuum mathematical model. FEMS Immunol Med Microbiol. 2009;57(2):173. doi:10.1111/j.1574-695X.2009.00596.x.
63. Gerlee P, Anderson ARA. Diffusion-limited tumour growth: simulations and analysis. Math Biosci Eng. 2010;7(2):385.
64. Smallbone K, Maini PK, Gatenby RA. Episodic, transient systemic acidosis delays evolution of the malignant phenotype: Possible mechanism for cancer prevention by increased physical activity. Biol Direct. 2010;5:22. doi:10.1186/1745-6150-5-22.
65. Zorzenon dos Santos R, Coutinho S. Dynamics of hiv infection: a cellular automata approach. Phys Rev Lett. 2001;87(16): 168102.
66. Strain M, Richman D, Wong J, Levine H. Spatiotemporal dynamics of hiv propagation. J Theor Biol. 2002;218(1):85.
67. Castiglione F, Duca K, Jarrah A, Laubenbacher R, Hochberg D, Thorley-Lawson D. Simulating epstein-barr virus infection with c-immsim. Bioinformatics. 2007;23(11):1371.
68. Warrender C, Forrest S, Koster F. Modeling intercellular interactions in early mycobacterium infection. Bull Math Biol. 2006;68(8):2233.
69. Marino S, Linderman J, Kirschner D. A multifaceted approach to modeling the immune response in tuberculosis. Wiley Interdisciplinary Reviews: Systems Biology and Medicine; 2010.
70. Kam N. The immune system as a reactive system: Modeling t cell activation with statecharts. In: Human-centric computing languages and environments, IEEE CS International Symposium on, vol. 0, ed. by Cohen IR, Harel D. 2001;vol. 0, p. 15–15. doi: 10.1109/HCC.2001.995228.
71. Efroni S, Harel D, Cohen IR. Toward rigorous comprehension of biological complexity: modeling, execution, and visualization of thymic t-cell maturation. Genome Res. 2003;13(11), 2485. doi: 10.1101/gr.1215303.
72. Efroni S, Harel D, Cohen IR. Emergent dynamics of thymocyte development and lineage determination. PLoS Comput Biol. 2007;3(1):e13. doi:10.1371/journal.pcbi.0030013.
73. Naamah S, David CIRH. The lymph node b cell immune response: dynamic analysis in-silico. Proc IEEE. 2008;96(8): 1421.
74. Celada F, Seiden P. A computer model of cellular interactions in the immune system. Immunol Today. 1992;13(2):56.
75. Puzone R, Kohler B, Seiden P, Celada F. Immsim, a flexible model for in machina experiments on immune system responses. Fut Gen Comput Syst. 2002;18(7):961.
76. Rapin N, Lund O, Castiglione F. Immune system simulation online. Bioinformatics. 2011;27(14):2013.
77. Kugler H, Larjo A, Harel D. Biocharts: a visual formalism for complex biological systems. J R Soc Interface. 2009. URL http://www.biomedsearch.com/nih/Biocharts-visual-formalism-complex-biological/20022895.html.

78. An G. Agent-based computer simulation and sirs: building a bridge between basic science and clinical trials. Shock. 2001; 16(4):266.
79. An G. Concepts for developing a collaborative in silico model of the acute inflammatory response using agent-based modeling. J Crit Care. 2006;21(1):105. doi:10.1016/j.jcrc.2005.11.012.
80. An G, Hunt CA, Clermont G, Neugebauer E, Vodovotz Y. Challenges and rewards on the road to translational systems biology in acute illness: four case reports from interdisciplinary teams. J Crit Care. 2007;22(2):169. doi:10.1016/j.jcrc.2006.12.011.
81. Cauwels A, Buys ES, Thoonen R, Geary L, Delanghe J, Shiva S, Brouckaert P. Nitrite protects against morbidity and mortality associated with tnf- or lps-induced shock in a soluble guanylate cyclase-dependent manner. J Exp Med. 2009;206(13):2915. doi:10.1084/jem.20091236.
82. Bailey AM, Thorne BC, Peirce SM. Multi-cell agent-based simulation of the microvasculature to study the dynamics of circulating inflammatory cell trafficking. Ann Biomed Eng. 2007;35(6):916. doi:10.1007/s10439-007-9266-1.
83. Mi Q, Rivire B, Clermont G, Steed DL, Vodovotz Y. Agent-based model of inflammation and wound healing: insights into diabetic foot ulcer pathology and the role of transforming growth factor-beta1. Wound Repair Regen. 2007;15(5):671. doi:10.1111/j.1524-475X.2007.00271.x.
84. An G. Introduction of an agent-based multi-scale modular architecture for dynamic knowledge representation of acute inflammation. Theor Biol Med Model. 2008;5:11. doi:10.1186/1742-4682-5-11.
85. Dong X, Foteinou PT, Calvano SE, Lowry SF, Androulakis IP. Agent-based modeling of endotoxin-induced acute inflammatory response in human blood leukocytes. PLoS One. 2010;5(2): e9249. doi:10.1371/journal.pone.0009249.
86. Galvo V, Miranda JGV, dos Santos RR. Development of a two-dimensional agent-based model for chronic chagasic cardiomyopathy after stem cell transplantation. Bioinformatics. 2008;24(18):2051. doi:10.1093/bioinformatics/btn362.
87. Galvo V, Miranda JGV. A three-dimensional multi-agent-based model for the evolution of chagas' disease. Biosystems. 2010; 100(3):225. doi:10.1016/j.biosystems.2010.03.007.
88. Li NYK, Verdolini K, Clermont G, Mi Q, Rubinstein EN, Hebda PA, Vodovotz Y. A patient-specific in silico model of inflammation and healing tested in acute vocal fold injury. PLoS One. 2008;3(7):e2789. doi:10.1371/journal.pone.0002789.
89. Bailey AM, Lawrence MB, Shang H, Katz AJ, Peirce SM. Agent-based model of therapeutic adipose-derived stromal cell trafficking during ischemia predicts ability to roll on p-selectin. PLoS Comput Biol. 2009;5(2):e1000294. doi:10.1371/journal.pcbi.1000294.
90. Tang J, Hunt CA. Identifying the rules of engagement enabling leukocyte rolling, activation, and adhesion. PLoS Comput Biol. 2010;6(2):e1000681. doi:10.1371/journal.pcbi.1000681.
91. Adra S, Sun T, MacNeil S, Holcombe M, Smallwood R. Development of a three dimensional multiscale computational model of the human epidermis. PLoS One. 2010;5(1):e8511. doi:10.1371/journal.pone.0008511.
92. Segovia-Juarez JL, Ganguli S, Kirschner D. Identifying control mechanisms of granuloma formation during m. tuberculosis infection using an agent-based model. J Theor Biol. 2004; 231(3):357. doi:10.1016/j.jtbi.2004.06.031.
93. Pappalardo F, Musumeci S, Motta S. Modeling immune system control of atherogenesis. Bioinformatics. 2008;24(15):1715. doi:10.1093/bioinformatics/btn306.
94. Duca KA, Shapiro M, Delgado-Eckert E, Hadinoto V, Jarrah AS, Laubenbacher R, Lee K, Luzuriaga K, Polys NF, Thorley-Lawson DA. A virtual look at epstein-barr virus infection: biological interpretations. PLoS Pathog. 2007;3(10):1388. doi:10.1371/journal.ppat.0030137.
95. Casal A, Sumen C, Reddy TE, Alber MS, Lee PP. Agent-based modeling of the context dependency in t cell recognition. J Theor Biol. 2005;236(4):376. doi:10.1016/j.jtbi.2005.03.019.
96. Nudelman G, Weigert M, Louzoun Y. In-silico cell surface modeling reveals mechanism for initial steps of b-cell receptor signal transduction. Mol Immunol. 2009;46(15):3141. doi:10.1016/j.molimm.2009.03.027.
97. Bogle G, Dunbar PR. Agent-based simulation of t-cell activation and proliferation within a lymph node. Immunol Cell Biol. 2010;88(2):172. doi:10.1038/icb.2009.78.
98. Meyer-Hermann ME, Maini PK, Iber D. An analysis of b cell selection mechanisms in germinal centers. Math Med Biol. 2006;23(3):255. doi:10.1093/imammb/dql012.
99. Santoni D, Pedicini M, Castiglione F. Implementation of a regulatory gene network to simulate the th1/2 differentiation in an agent-based model of hypersensitivity reactions. Bioinformatics. 2008;24(11):1374. doi:10.1093/bioinformatics/btn135.
100. Cheng Y, Ghersi D, Calcagno C, Selin LK, Puzone R, Celada F. A discrete computer model of the immune system reveals competitive interactions between the humoral and cellular branch and between cross-reacting memory and nave responses. Vaccine. 2009;27(6):833. doi:10.1016/j.vaccine.2008.11.109.
101. Baldazzi V, Castiglione F, Bernaschi M. An enhanced agent based model of the immune system response. Cell Immunol. 2006;244(2):77. doi:10.1016/j.cellimm.2006.12.006.
102. Folcik VA, An GC, Orosz CG. The basic immune simulator: an agent-based model to study the interactions between innate and adaptive immunity. Theor Biol Med Model. 2007;4:39. doi:10.1186/1742-4682-4-39.
103. Mitha F, Lucas TA, Feng F, Kepler TB, Chan C. The multiscale systems immunology project: software for cell-based immunological simulation. Source Code Biol Med. 2008;3:6. doi:10.1186/1751-0473-3-6.
104. Halling-Brown M, Pappalardo F, Rapin N, Zhang P, Alemani D, Emerson A, Castiglione F, Duroux P, Pennisi M, Miotto O, Churchill D, Rossi E, Moss DS, Sansom CE, Bernaschi M, Lefranc MP, Brunak S, Lund O, Motta S, Lollini PL, Murgo A, Palladini A, Basford KE, Brusic V, Shepherd AJ. Immunogrid: towards agent-based simulations of the human immune system at a natural scale. Philos Transact A Math Phys Eng Sci. 2010;368(1920):2799. doi:10.1098/rsta.2010.0067.
105. Rapin N, Lund O, Bernaschi M, Castiglione F. Computational immunology meets bioinformatics: the use of prediction tools for molecular binding in the simulation of the immune system. PLoS One. 2010;5(4):e9862. doi:10.1371/journal.pone.0009862.
106. Wikipedia, http://en.wikipedia.org/wiki/comparison_of_agent-based_modeling_software.
107. Swarm, http://www.swarm.org/index.php/tools_for_agent-based_modelling.
108. Wilensky U. Netlogo. http://ccl.northwestern.edu/netlogo/. (1999).
109. An G, Wilensky U. Artificial life models in software. (Springer, London), chap. From artificial life to in silico medicine: NetLogo as a means of translational knowledge representation in biomedical research; 2009. p. 183–214.
110. Remy E, Ruet P, Mendoza L, Thieffry D, Chaouiya C. From logical regulatory graphs to standard petri nets: Dynamical roles and functionality of feedback circuits. In: Priami C, Ingolfsdottir A, Mishra B, Riis Nielson H, editors. Transactions on Computational Systems Biology VII, Lecture Notes in Computer Science, vol. 4230. Springer, Berlin/Heidelberg; 2006. p. 56–72.
111. Mendoza L. A network model for the control of the differentiation process in th cells. Biosystems. 2006;84(2):101. doi:10.1016/j.biosystems.2005.10.004.

112. Saez-Rodriguez J, Simeoni L, Lindquist JA, Hemenway R, Bommhardt U, Arndt B, Haus UU, Weismantel R, Gilles ED, Klamt S, Schraven B. A logical model provides insights into t cell receptor signaling. PLoS Comput Biol. 2007;3(8):e163. doi:10.1371/journal.pcbi.0030163.
113. Saez-Rodriguez J, Alexopoulos LG, Epperlein J, Samaga R, Lauffenburger DA, Klamt S, Sorger PK. Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction. Mol Syst Biol. 2009;5:331. doi:10.1038/msb.2009.87.
114. Franke R, Mller M, Wundrack N, Gilles ED, Klamt S, Khne T, Naumann M. Hostpathogen systems biology: logical modelling of hepatocyte growth factor and helicobacter pylori induced c-met signal transduction. BMC Syst Biol. 2008;2:4. doi:10.1186/1752-0509-2-4.
115. Chaouiya C. Petri net modelling of biological networks. Brief Bioinform. 2007;8(4):210. doi:10.1093/bib/bbm029.
116. Regev A, Silverman W, Shapiro E. Representation and simulation of biochemical processes using the pi-calculus process algebra. Pac Symp Biocomput; 2001:459–470.
117. Clarke EM, Grumberg O, Peled DA. Model checking. MIT Press; 2000.
118. Na D, Park I, Lee KH, Lee D. Integration of immune models using petri nets. In: Nicosia Giuseppe, Cutello Vincenzo, Bentley Peter J, et al, editors. Proceedings of Artificial Immune Systems: Third International Conference, ICARIS 2004, Catania, Sicily, Italy, September 13–16, 2004—Vol 3239 of Lecture Notes in Computer Science. Berlin: Springer; 2004. p. 205–216.
119. Monroy R. A process algebra model of the immune system. In: Proceedings of the 8th Knowledge-Based Intelligent Information & Engineering Systems, KES 2004. Lecture Notes in Artificial Intelligence; 2004.
120. Guerriero ML, Prandi D, Priami C, Quaglia P. Process calculi abstractions for biology. Tech. Rep. Technical Report TR-13-2006, CoSBi (Center for Computational and Systems Biology), University of Trento; 2006.
121. Spicher A, Michel O, Cieslak M, Giavitto JL, Prusinkiewicz P. Stochastic p systems and the simulation of biochemical processes with dynamic compartments. Biosystems. 2008;91(3):458. doi:10.1016/j.biosystems.2006.12.009.
122. Corne DW, Frisco P. Dynamics of hiv infection studied with cellular automata and conformon-p systems. Biosystems. 2008;91(3):531. doi:10.1016/j.biosystems.2007.01.007.
123. Chopard B, Falcone J, Hoekstra A, Borgdorff J. A framework for multiscale and multiscience modeling and numerical simulations. Unconvent Comput; 2011:2–8.
124. Eissing T, Kuepfer L, Becker C, Block M, Coboeken K, Gaub T, Goerlitz L, Jaeger J, Loosen R, Ludewig B, et al. A computational systems biology software platform for multiscale modeling and simulation: Integrating whole-body physiology, disease biology, and molecular reaction networks. Front Physiol. 2011;2.
125. Kawashima Y, Pfafferott K, Frater J, Matthews P, Payne R, Addo M, Gatanaga H, Fujiwara M, Hachiya A., Koizumi H., et al. Adaptation of hiv-1 to human leukocyte antigen class i. Nature. 2009;458(7238):641.
126. Obbard D, Welch J, Kim K, Jiggins F. Quantifying adaptive evolution in the drosophila immune system. PLoS Genet. 2009;5(10):e1000698.
127. Farmer J, Packard N, Perelson A. The immune system, adaptation, and machine learning. Phys D: Nonlinear Phenomena. 1986;22(1–3):187.
128. Farmer J, Kauffman S, Packard N, Perelson A. Adaptive dynamic networks as models for the immune system and autocatalytic sets. Ann N Y Acad Sci. 1987;504(1):118.
129. Forrest S, Perelson A. Genetic algorithms and the immune system. Parallel Problem Solving from Nature; 1991. p. 319–325.
130. Oprea M, Forrest S. Simulated evolution of antibody gene libraries under pathogen selection. In: Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on, vol 4. 1998;4:3793–3798.
131. Kim J, Bentley P. Immune memory and gene library evolution in the dynamic clonal selection algorithm. Genet Program Evolvable Mach. 2004;5(4):361.
132. De Jong K. Evolutionary computation; 2002.
133. Forrest S, Beauchemin C. Computer immunology. Immunol Rev. 2007;216:176. doi:10.1111/j.1600-065X.2007.00499.x.
134. Izmailian N, Papoyan V, Priezzhev V, Hu C, et al. Self-organizing behavior in a lattice model for co-evolution of virus and immune systems. Phys Rev E, Stat Nonlinear Soft Matter Phys. 2007;75(4 Pt 1):041104.
135. Guttenberg N, Ali Tabei SM, Dinner AR. Short-time evolution in the adaptive immune system. Phys Rev E. 2011;84(3):031932.
136. Lieberman T, Michel J, Aingaran M, Potter-Bynoe G, Roux D, Davis M Jr, Skurnik D, Leiby N, LiPuma J, Goldberg J, et al. Parallel bacterial evolution within multiple patients identifies candidate pathogenicity genes. Nat Genet; 2011.
137. Aldridge BB, Burke JM, Lauffenburger DA, Sorger PK. Physicochemical modelling of cell signalling pathways. Nat Cell Biol. 2006;8(11):1195. doi:10.1038/ncb1497.
138. Alves R, Antunes F, Salvador A. Tools for kinetic modeling of biochemical networks. Nat Biotechnol. 2006;24(6):667. doi:10.1038/nbt0606-667.
139. Wilkinson DJ. Stochastic modelling for quantitative description of heterogeneous biological systems. Nat Rev Genet. 2009.10(2):122. doi:10.1038/nrg2509.
140. de Jong H, Ropers D. System modeling in cellular biology: from concepts to nuts and bolts, chap. Qualitative approaches towards the analysis of genetic regulatory networks. Cambridge, MA: MIT Press. 2006. p. 125–148.
141. Albert R, Wang RS. Discrete dynamic modeling of cellular signaling networks. Methods Enzymol. 2009;467:281. doi:10.1016/S0076-6879(09)67011-7.
142. Cohen IR, Harel D. Explaining a complex living system: dynamics, multi-scaling and emergence. J R Soc Interface. 2007;4(13):175. doi:10.1098/rsif.2006.0173.
143. Fisher J, Piterman N. The executable pathway to biological networks. Brief Funct Genomics. 2010;9(1):79. doi:10.1093/bfgp/elp054.
144. Romero-Campero FJ, Twycross J, Camara M, Bennett M, Gheorghe M, Krasnogor N. Modular assembly of cell systems biology models using p systems. Int J Found Comput Sci. 2009;3:427 doi:10.1142/S0129054109006668.
145. Swarm, http://www.swarm.org/wiki/agent-based_models_in_biology_and_medicine.
146. Hlavacek WS, Faeder JR, Blinov ML, Posner RG, Hucka M, Fontana W. Rules for modeling signal-transduction systems. Sci STKE. 2006;2006(344):re6. doi:10.1126/stke.3442006re6.