TOPICAL REVIEW

Hybrid computational modeling methods for systems biology

To cite this article: Daniel A Cruz and Melissa L Kemp 2022 *Prog. Biomed. Eng.* **4** 012002

IOP Publishing Prog. Biomed. Eng. 4 (2022) 012002 https://doi.org/10.1088/2516-1091/ac2cdf

Progress in Biomedical Engineering

TOPICAL REVIEW

Hybrid computational modeling methods for systems biology

Daniel A Cruz[1] and Melissa L Kemp[2,*]

[1] School of Mathematics, Georgia Institute of Technology, Atlanta, GA, United States of America
[2] The Wallace H. Coulter Department of Biomedical Engineering, Georgia Institute of Technology & Emory University, Atlanta, GA, United States of America
[*] Author to whom any correspondence should be addressed.

E-mail: melissa.kemp@bme.gatech.edu

RECEIVED
11 June 2021

REVISED
11 August 2021

ACCEPTED FOR PUBLICATION
5 October 2021

PUBLISHED
26 October 2021

Keywords: computational modeling, systems biology, simulation, prediction

Abstract

Systems biology models are typically considered across a spectrum from mechanistic to abstracted description; however, the lines between these forms of modeling are increasingly blurred. Ever-increasing computational power is providing novel opportunities for bridging time and length scales. Furthermore, despite biological mechanisms or network topology often ill-defined, the acquisition of high-throughput data leaves modelers with the desire to leverage available measurements. This review surveys modeling tools in which two or more mathematical forms are blended to describe time-dependent processes in a multivariate system. While most commonly manifested as continuous/discrete description, other forms such as mechanistic/inference or deterministic/stochastic hybrid models can be generated. Recent innovations in hybrid modeling methodologies and new applications illustrate advantages for combining model formats to gaining biological systems level insight.

1. Introduction

Biosystems modeling has been beautifully framed in terms of assignment of mathematical approaches in alignment with level of mechanistic modeling. Ideker and Lauffenburger assigned modeling on a spectrum with abstracted/high-level on one end of statistical/inference modeling and specified/low-level with mechanistic ordinary differential equation (ODE) modeling [1]. Often times, knowledge of a system varies—in terms of timescales, interacting components, or nature of regulation—and hybridization of mathematical description would be useful. We define hybrid modeling for the purposes of this discussion as computational modeling in which the information is passed across common variables to reconcile multiple forms of mathematical description, such as those spanning the axes in figure 1: abstracted/mechanistic; continuous/discrete; steady-state/dynamic. We note that other comprehensive reviews discuss modeling types individually [2, 3] and/or discuss hybridization within a broader discussion of classification [4], computational tools [5], or model simplification and reduction [6]. Here, this review aims to selectively highlight computational implementation of blended modeling forms for biosystems applications and provide diverse, creative hybridized modeling examples in systems biology; see table 1 for an overview of some of these examples.

2. Types of hybrid models

2.1. ODE hybrids

ODEs have historically played an important role in the development of predictive systems biology models, building upon enzyme kinetic theory. These ODE models provide quantitative predictions for a variety of biochemical reaction systems and can be directly compared with measurements from common experimental methods such as flow cytometry, longitudinal microscopy imaging with fluorescent reporter proteins, etc. Soon after the advent of signal transduction modeling (see early landmark examples such as [22–24]) biochemical systems dynamics with mass action kinetics were expanded to larger systems [25, 26] and have continued to grow in complexity. Nonetheless, ODE models are often limited by the need to identify and

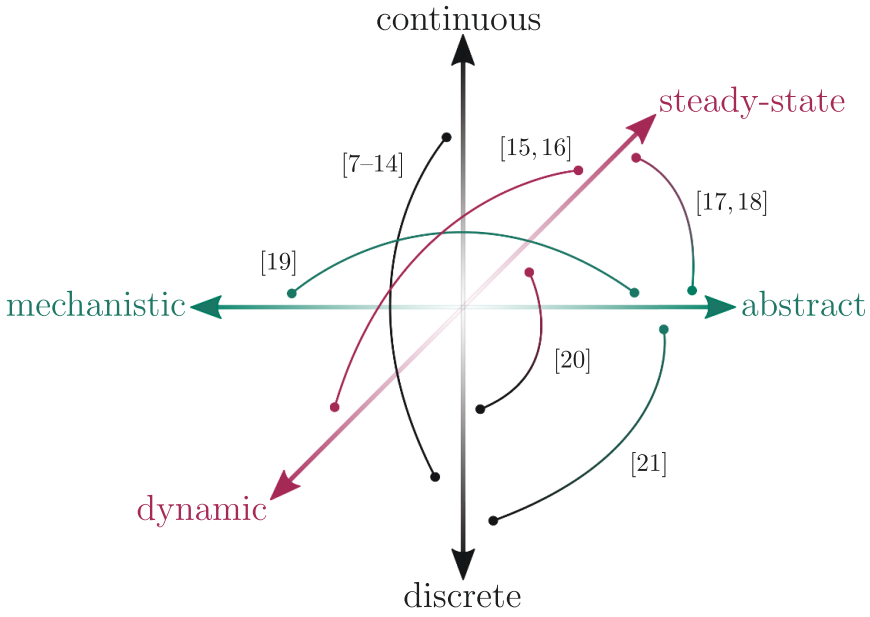

Figure 1. Examples in this review that span across multiple domains of modeling. For simplicity only three axes are depicted, but others categories, such as deterministic/stochastic, could be included. See table 1 for more details.

Table 1. Guide for examples included this review. An asterisk (*) indicates that the computational tool is discussed in this review.

Reference	Hybrid model description	Biological application	Associated tools
[7]	Piecewise linear differential equations	Cyclin synthesis and degradation	
[8]	ODE/Boolean	Bacteria diauxic shift	
[9]	ODE/multi-state logic	E2F1 regulation of genotoxic drug responses,	CellNetAnalyzer*
[10]	ODE/Boolean	Metabolic regulation of hepatocytes	ProMoT, CellNetAnalyzer*, PottersWheel
[11]	SSA/ODE	Budding yeast cell cycle	
[12–14]	ODE/PDE/ABM	Multi-scale model of *Mycobacterium tuberculosis*-immune cell interactions	GranSim
[15]	ABM/constraint-based modeling	Biofilm growth under oxygen limitations	MatNet*
[16]	Boolean/FBA	*Escherichia coli* growth with lac operon	FlexFlux*
[17]	Gene state probabilities/FBA	Regulation of metabolism in *E. coli* and *M. tuberculosis*	PROM*
[18]	Gene regulatory network influence/FBA	Bacteria diauxic shift	CoRegFlux*
[19]	SSA/PLSR	T cell receptor activation	
[20]	Boolean cell cycle/FBA	Budding yeast cell cycle progression	BooleanNet*, COBRApy
[21]	Petri net/Boolean	Transcriptional regulation of glucose metabolism, yeast osmoregulation	

properly estimate parameters such as equilibrium and rate constants [3]. These parameters may be unknown in context-specific situations due to noise in measurements or a general lack of the necessary data.

A hybrid system or hybrid automaton is a model which organizes a set of continuous descriptions of the same dynamical system (e.g. ODE sub-models) into a larger modeling scheme governed by discrete controls [27]. The discrete controls can incorporate non-determinism and are typically formulated as a set of logical conditions that indicate which ODE sub-model is being employed at any given time point. In practice, this model tends to be organized into a set of nodes or 'states' connected by transition edges associated with the discrete controls, hence the association with automata. Because the discrete controls governing a hybrid system can depend on the variables associated to its ODE sub-models and/or other (continuous or discrete) variables under consideration, this model allows for systems to be represented at different scales using both quantitative and qualitative data simultaneously. Hybrid systems have been studied extensively over the last thirty years since their original formalization [28] and include commonly known constructions like systems

of piece-wise linear and piecewise-affine differential equations [2, 5, 7]. We refer the reader to the work by Singhania *et al* for an example of a piece-wise linear differential equation model of mammalian cell cycle regulation [7]. This example demonstrates how a hybrid systems model can be quantitatively built from and compared against experimental data (e.g. flow cytometry measurements) while employing fewer kinetic constants than those typically required in an ODE model. In this way, this type of hybrid model balances the advantages of both quantitative and qualitative modeling, respectively.

To showcase the diversity of hybrid system models, we discuss recent work by Liu and Bockmayr on using a hybrid system framework to study regulatory and metabolic interactions in the context of carbon catabolite repression [8]. Constraint-based models are traditionally used to analyze metabolic networks, and hybrid models involving constraint-based descriptions offer a natural way to incorporating such networks into broader contexts (see section 2.4). Liu and Bockmayr instead propose a hybrid systems framework in which molecular species are described with continuous variables while the discrete states of the framework correspond to all of the gene expression states for the associated proteins within the biological system. The species are thus governed by ODEs within each discrete state while transitions between these states rely on the amounts of each species [8]. The advantages of adopting such a framework for modeling metabolic and regulatory network interplay lie primarily in the insight from using and producing quantitative data and the fact that there are several computational tools for solving resulting hybrid systems models (see section 3). The authors apply their framework to model the diauxic shift in carbon catabolite repression within *E. coli*, extracting some parameters from prior work in the literature. While the model in this study is relatively small and abstract, the simulations are still able to partially capture the experimentally observed relationship between the length of time a cell grows in a preferred carbon source and the lag phase of the shift. However, a larger and more realistic model would need to be developed in order to provide novel insight within this context.

2.2. Boolean and multi-valued logic model hybrids

Fully discrete models such as Boolean networks or Petri nets use different representations for the flow of information or mass through a system. Boolean networks represent biological mechanisms through binary output on/off states as a function of combinatorial binary input values. These models tend to be deterministic and hence can be evaluated for fixed points, sensitivity, and stability [31] using tools such as BoolNet [32], MaBoSS [33], and PlantSimLab [34] even when several mechanisms are at work. The coarse-grained description of Boolean networks is particularly useful in gene regulatory network applications, where logic gates nicely represent how combinatorial transcription factor binding on promoter regions initiates changes in gene expression. On the other hand, Boolean networks tend to be ill-suited for modeling systems involving time-dependent processes (e.g. systems with fast and slow processes) and systems with processes which strongly depend on the concentration of certain biomolecules (e.g. systems with feedback loops) [3]. While some of these limitations can be mitigated by generalizing the binary variables within these models to allow for multiple states and multi-valued logic, mechanistic models tend to be preferred over such generalized logic models when available.

A Petri net is a discrete model whose purpose is to describe the flow of information/resources throughout a network. Petri nets are typically visualized as bipartite, directed graphs with two disjoint sets of nodes, 'places' and 'transitions', through which 'tokens' (e.g. molecules of a species) move in accordance with a set of rules. Though originally used to describe chemical processes, Petri nets have become more frequently used in the context of systems biology over the last three decades given the intuitive way in which they combine qualitative relationships with quantitative analysis (i.e. the change in token distribution with respect to time) [2, 5]. Petri nets may be deterministic or stochastic in nature, though stochastic implementation of this model requires a propensity function that is probabilistic. In this case, model simulations may execute the Gillespie algorithm (see section 2.3) numerous times and average together outputs to represent a single discrete time increment of the next state. Hybridizing deterministic Boolean and stochastic Petri net descriptions together offers advantages of developing extremely large-scale, multi-omics simulations with minimal parameterization. In an early example [21], the integration of two disjoint modeling forms was performed through the assignment of triplets, in which the places in the Petri net were associated with variables in the Boolean system. The drawback of such 'stitching' is that the Petri-to-Boolean conversion requires an assigned threshold for conversion between 1 and 0. Furthermore, the Boolean-to-Petri conversion (largely representing the lag in transcription and translation resulting in changes in metabolite and protein levels) must handle difference in timescales of the networks; Berestovsky *et al* introduced delays by queuing the triplet for evaluation at a delayed time before updating the Boolean variable.

Aside from hybrid systems (see section 2.1), other works have explored combining ODE models and discrete logic models in order to simultaneously address limitations from both approaches. For example, Khan *et al* developed a hybrid model incorporating ODE and multi-valued logic components in order to

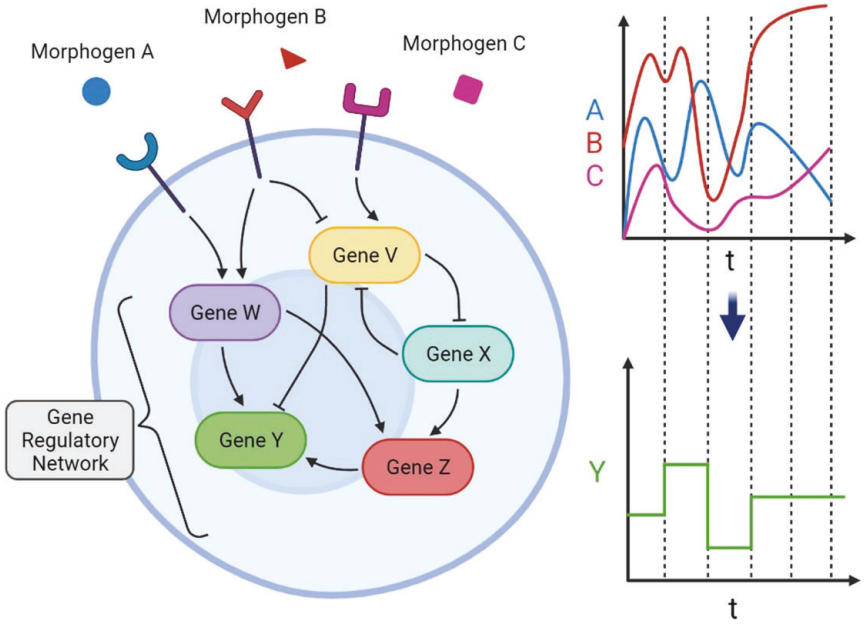

Figure 2. Data for the same network may vary widely in degree of detail depending on the availability of preexisting information, the accessibility of tools for accurate measurement, and/or the use of external controls. For example, the morphogen data in this network can be measured continuously while data for downstream regulation is primarily collected via coarse-grained observations. Rather than attempting to model all of the variables in this network discretely to accommodate for the lack of downstream precision, the output of ODE functions may be converted into input for multi-valued logic functions through the use tools like CellNetAnalyzer [29]. In this manner, the richness of the morphogen data can still be used to analyze the network as a whole. Figure made using BioRender [30].

study the regulation of chemoresistance by a large regulatory network centered on transcription factor E2F1 [9]. Previous work by Vera *et al* [35] and others used ODE models to investigate small-scale regulation circuits (i.e. 'modules') involving or influencing E2F1 in specific contexts (e.g. in cells whose DNA had been damaged). ODE models proved suitable for these investigations because of the presence of non-linear behavior and feedback loops; however, extending such models to include larger, downstream networks would prove difficult due to a lack of necessary data like rate constants for various interactions. To address this issue, Khan *et al* divided their hybrid model into three parts: (a) a core module of a network regulating E2F1 which incorporated experimentally observed feedback loops, (b) a module capturing the expression of target genes downstream from the core module, and (c) a module of target phenotypic responses including chemoresistance. While the core module is an ODE model adapted from prior work, the latter two modules were constructed using several multi-valued logic functions, twenty-four of which regulate the target genes in the second module. In order to connect the core ODE model with the multi-valued logic functions in the second module, the authors developed a pipeline involving MATLAB and CellNetAnalyzer [29] which incorporates an interface for discretizing variables produced by the core module based on manual training. See figure 2 for a visualization of this hybrid modeling approach. While the resulting hybrid model captures a larger network than previous ODE models have, the training required for the discretization rules still requires extensive data acquisition. Nonetheless, the hybrid model is better suited for capture certain feedback behaviors and estimating associated thresholds when compared to a fully discrete model in which the authors replaced the ODE system in the core module with a multi-valued logic system. Others have developed similar hybrid approaches using compartmental models, including the approach by Ryll *et al* in which a kinetic (ODE) sub-model of metabolic processes communicates with a logical sub-model of signalling and/or gene-regulatory networks in order to capture and study information at both levels [10].

2.3. Stochastic approximation model hybrids

The Gillespie algorithm (stochastic simulation algorithm or SSA) is an explicit physical description of discrete stochastic events and thus can represent any time evolution of a chemical reaction. SSA is rarely used for complex networks without approximation methods such as tau leaping due to the shear computational expense. There are many biological processes where stochasticity at a molecular level plays a critical role in higher order behavior. Quantal neurotransmitter release, stochastic gene expression, and fluctuations in calcium transport from endoplasmic reticulum stores are examples of how noise and randomness are

leveraged in biological systems to improve dynamic range of responses or initiate positive feedback dynamics. To streamline simulations, it is desirable to explicitly incorporate stochastic processes for only the mechanisms where necessary; therefore, hybridizing with other types of mathematical description is attractive for large systems models.

A clear gain of hybrid modeling over stochastic simulations of an entire biochemical network is the drastic increase in computational efficiency by streamlining rate laws that are well-described by stochastic differential equations (SDEs) with the chemical Langevin equation (CLE) description. Imposing biological insight into the modeling context is an important step in this case. For example, Liu *et al* used biological intuition to assume that most of the intrinsic noise within cell cycle regulation arises at the gene expression level due to the low numbers of molecules of genes and mRNAs [36]. Scales of reactions and populations were categorized into four classes. Reactions that occur with low propensity and with small numbers of participating species were found to best represented with SSA. Reactions with large numbers of reactants/products that occur on either fast or slow timescales were best approximated with ODEs. Fast reactions that occur with low number of participating species were represented by SDEs without compromising the error of the computation if the species in this category did not react with one another (coupled reactions) and were occurring much faster than the SSA description. The resultant hybrid system with partitioned reactions reduced computational time from 11.6 h to 22 min with minimal compromise in model dynamics. Ahmadian *et al* used a similar approach with SSA and ODE descriptions within their hybrid model of the molecular mechanisms regulating the budding yeast cell cycle [11]. As in [36], the authors separated reactions into categories based propensity and abundance and employed the appropriate sub-model for each category, building on an existing ODE model for the cycle with over sixty species and a hundred reactions [37]. The resulting hybrid model captures important features of the cycle (e.g. cycle time and volume at division), matches experimental data for over a hundred phenotypes of mutant cases, and improves overall computational speed. The systematic approach of binning reaction types and determining contextual appropriateness of computational description is a useful strategy for modelers to take when simplifying computationally expensive simulations.

Hybrid models are also advantageous in blending high-resolution data of biological systems with course-grained knowledge. SSA is typically applied in situations where stochastic reaction mechanisms are necessary and falls on the 'mechanistic' side of figure 1 in the sense that molecular stochasticity of binding/unbinding events are reflected by this formalism. In contrast, course-grained correlative relationships between biological components may be known but not at the level of detail necessary to build mathematical description such as SSA or ODE (abstract side of figure 1). In these latter cases, statistical modeling using partial least squares regression (PLSR) may be suitable for describing multivariate relationships, such as activation of a signaling kinase associated with cell fate decisions such as apoptosis or gene transcription. PLSR relates a dataset of predictor variables to response variables without prior knowledge about the network structure. The associated algorithms rely upon creating latent variables that maximize variance captured the predictor dataset while simultaneously optimizing a linear relationship with one or more dependent variables within a multidimensional space. In systems biology applications [38, 39], the successful use of PLSR to describe cue-signal-response phenomenon of cells may be partially attributed to the sampling of receptor-initiated signaling at 'bowtie' points in biochemical cascades [40] that are highly informative of later downstream events. For example, comprehensive measurements of T cell receptor signaling found that phosphorylation patterns in Erk and Akt better reflected the avidity of antigen peptide than receptor-proximal proteins such as Zap70 [41].

Ideally, mechanisms of signal initiation due to an external cue detailed in early, upstream events could be combined with the abstracted statistical approaches found to be useful for less refined knowledge of cell fate decisions. Furthermore, many of the 'macro-level' dependent variables such metrics of transcription, proliferation, or apoptosis are measured on long time scales (i.e. from hours to days) and are quantified with a variety of cellular and molecular techniques while initial stochastic signaling events are pertinent on short time scales (i.e. from seconds to hours). Mayalu and Asada described an approach for hybridizing receptor-ligand mass action kinetics with multivariate modeling of cellular response as a 'white-box/black-box' methodology [19]. Outputs from an SSA sub-model of T cell receptor activation and signaling were empirically correlated to downstream behaviors (i.e. the observable output space) by PLSR to yield a lower-order, nonlinear model. First, the input space was augmented with a mechanistic map generated by a stochastic forward Kolmogrov equation model. Multiple stochastic time trajectories of signaling molecules were generated to create a nonlinear dynamic stochastic mapping of external cues of TCR peptide presentation, resulting in a model-generated predictor variable dataset for PLSR using a dependent variable of IL-2 transcription. Akaike's information criterion (AIC) was used as a metric to truncate latent variables. See figure 3 for a visualization of this hybrid modeling approach. While an attractive approach, this hybrid methodology has not been tested in integrated responses from combinatorial cues, a context in which

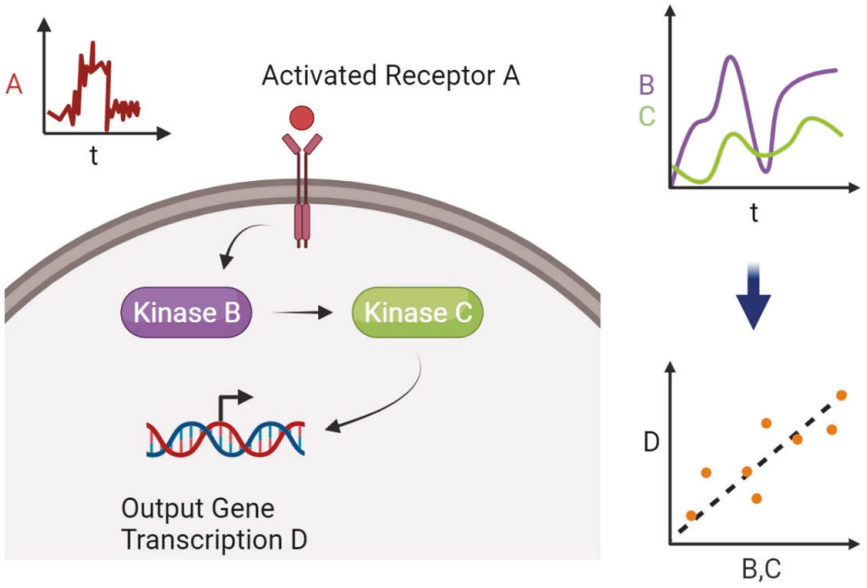

Figure 3. A receptor signaling network can be modeled by stochastic simulation algorithms (SSA) for the detailed mechanism of receptor ligation (A), and/or kinase activation cascades (B), (C). Downstream cellular processes, such as gene transcription, may occur on longer timescales and without knowledge of all molecular players and dynamics. Multivariate statistical modeling can use SSA model features from upstream dynamics (B), (C) as predictor-block variables to fit correlative relationships to dependent outputs (D). Figure made using BioRender [30].

distinct cytokine receptor ligation and common kinase cascade activation (e.g. MAPK) could benefit from hybrid-based discrimination of distinct cellular outputs [42].

2.4. Constraint-based model hybrids

Constraint-based modeling is a widely-used strategy for analyzing metabolic networks, in which knowledge of the stoichiometry matrix of reaction substrates/products to define network architecture is leveraged on a large scale (typically genome-wide). Through assumptions of steady-state, a flux vector is rapidly calculated without the need for kinetic parameterization. The solution to a system of linear equations is typically over-determined; the use of biological objective functions (e.g. maximization of ATP production or biomass) is often used to restrict the solution space. Dynamic or unsteady flux balance analysis (uFBA) relies in piecewise linearization of metabolic time series data in discrete intervals to define ranges of non-zero derivatives [43]. Because FBA and uFBA modeling are limited to metabolism (where stoichiometry and conservation of mass can be defined/upheld), hybrid methods in which gene regulation or dynamic signal transduction can be integrated into model description are desirable. Typically, timescales of metabolic change are assumed to be rapid and reaching steady-state for updates at each discrete time step of the other biological process.

One example of linking a constraint-based model to other cellular processes was executed in the description of *Pseudomonas aeruginosa* biofilm formation. The tool MatNet was used to combine an agent-based model (ABM) with FBA modeling [15]. An ABM is a computational model in which autonomous agents (e.g. molecules, cells, etc) interact with their environment and with one another according to a set of rules. The purpose of an ABM is to study how dynamic, local interactions give rise to complex, system-level behaviors such as the formation of structures or patterns [2, 5, 44, 45]. While the flexibility of defining an ABM's rules allows this model to readily include multi-scale data, the computational cost of running ABM simulations can climb drastically depending on the level of detail with which the model is constructed. In [15], constraints for each cell's (i.e. agent's) extracellular nutrients are limited to local concentrations. These concentrations are used to solve FBA within each cell via MATLAB and then passed back for updating the cell through rules for growth, cell division, and extracellular diffusion. This exchange is made possible using MatNet, a tool which connects MATLAB with the NetLogo platform [46] used to develop and simulate the associated ABM. Thus, the collective development of biofilm as a function of oxygen or nitrate limitations can be evaluated in a tractable manner.

In another example, metabolic regulation by cell cycle proteins in yeast was analyzed by hybridizing a genome-scale constraint-based model of metabolism with an off-the-shelf Boolean model of interactions among three cyclins (Clb2, Clb3 and Clb5) and an inhibitor of cyclin/Cdk1 kinase activity [20]. The Boolean

model yields the expected sequential activation of cyclins, which drives periodic cell cycles. The metabolic and cell-cycle models are linked because metabolic enzyme activities influence the formation of cyclin complexes and cyclin complexes, in turn, inhibit metabolic reactions. Flux distributions at a given time step are converted into Boolean on/off states and updated in the cyclin network. Likewise, activation status of the cyclin network can inhibit or augment metabolic fluxes. A limitation of this example was the assumption of new connections between metabolism state and cyclin dynamics that were not based upon the authors' literature curation, which only found one such relationship. Despite this, hybridizing cell cycle Boolean nodes with AND/OR logic gates with the yeast metabolic FBA model results in new predictions of metabolic fluctuations specific to G1, S, G2/M phase. Future additions of boundary influx constraints and weighted edges for interactions instead of logical operators may extend the utility of this hybrid approach for systems biology applications. It should be noted that in both of the above examples, previously published models that had been rigorously validated were used as the starting point for blending two models together.

Modeling the interactions between gene regulatory networks and metabolism has often relied upon constraint-based modeling as a common platform. While this approach can hybridize multiple levels of biological processes, constraint-based models typically simplify genes and reaction fluxes to binary on/off representation. Scaling the rules of interaction (which have to be manually curated) across the entire genome/metabolome is currently impractical. Chandrasekaran and Price developed a hybrid framework of probabilistic regulation of metabolism (PROM) to integrate transcriptional regulation with metabolic constraint-based models in an automated fashion [17]. Conditional probability is used in PROM to represent gene states as well as gene-transcription factor interactions. This strategy is advantageous because modelers can easily leverage microarray data to quantify conditional probabilities. Furthermore, gene-knockout mutants are readily simulated; turning off transcription of a metabolic gene will result in a constraint on the enzymatic reaction encoded by the gene. Instead of Boolean representation, PROM's use of conditional probabilities allows for intermediate levels of regulation. Furthermore, specific knowledge of kinetic parameters is not required. Objective functions must satisfy constraints on flux distributions through the metabolic stoichiometric network with the additional 'soft' conditions of penalizing deviations from constraints imposed by the regulation.

Since the development of PROM, subsequent efforts have focused on eliminating the need of *a priori* knowledge of transcription factor:gene targets relationships and the quality of curated networks. Algorithms such as CoRegFlux [18] infer gene regulatory networks as regulators and target sets. Influence scores calculate the effect of regulators on its targets (both activation or inhibition), reducing thousands of gene expression measurements to tens or hundreds of scores. A linear regression model then trains gene expression of targets (and correspondingly metabolic enzyme activity in a reaction flux) as a function of its regulators' influences. Feedback between metabolic processes and gene regulation have subsequently been explored with inclusion of protein-protein interaction data to simulate network failure in metabolic systems due to perturbations at the genetic level [47].

An alternative approach for describing FBA models interactions with transcriptional regulation is by converting discrete, qualitative states of the regulatory networks (e.g. gene regulation and/or protein signal transduction as integer states) into user-defined continuous intervals (upper/lower flux bounds deemed by the modeler to be physiologically relevant). In the FlexFlux software package [16], flux constraints on a metabolic network are realized by mass balance of the system, user-defined objective functions, and translation of the regulatory network steady states into new flux constraints on metabolism. For the latter step, linkages are made by user-defined gene-protein-reaction associations in SBML (see section 3.1). Seeking out attractor states is performed by synchronously updating the regulatory network until a steady state is reached or a cyclic attractor is found. Unfortunately, cyclic attractors are simply averaged together in order to define flux constraints under these conditions. The FlexFlux methodology is useful for small metabolic FBA systems and simple modes of transcriptional regulation, such as bacterial operon which yield concomitant changes of pathway enzymes. In complex genome-scale models, randomized sampling from initial conditions may be applied to seek out dominant steady-state attractors but pose challenges in the size of the search space. Finally, the hybridization of modeling is unidirectional, in that the qualitative description of gene regulation imposes constraints on the metabolic fluxes, but this is not related back to subsequently impact the regulation.

2.5. Multi-format models

In 2012, a monumental achievement in systems biology was accomplished by Covert *et al* through simulation of a simple unicellular organism, *Mycoplasma genitalium*, through an entire life cycle of replication [48]. The complexity of describing whole-cell metabolism, generation of nucleotides and amino acid building blocks, DNA replication and protein translation required interfacing multiple modeling modules together. The whole-cell model is established by defining 16 cellular states (analogous to state variables) and 28 cellular

processes (analogous to ODEs). An example of a cell state would be 'Total Cell Mass = Chromosome + RNA + Protein Monomer + Protein Complex + Transcript + Polypeptide + Metabolite'. Evolution of the cell state at each time step is through evaluation of each cellular process sub-model with allocation of the state variables across the processes, assuming independence for each sub-model update. Common sub-model inputs—such as metabolites like ATP that are utilized across numerous cellular processes—were estimated for expected consumption and computationally allocated as a shared resource. The process of Metabolism serves as the primary interface between the external environment and the other processes. Instantaneous flux of each metabolic reaction is calculated at each point of integration to update the molecule count of each metabolite and collectively provide an instantaneous cellular growth rate. Finally, stochastic processes are used to describe initiation of transcription, DNA damage and repair, protein misfolding, and termination of ribosomal translation. Collectively, the integration of multiple quantitative modeling formats together for whole-cell modeling provided an unprecedented opportunity to identify global distribution of energy and examine the role of interactions across multiple levels of cellular organization, such as DNA-protein binding.

A large body of work by Denise Kirshner and Jennifer Linderman have pushed the frontiers of hybrid modeling in ABMs to describe multi-scale processes in immunology in which discrete processes are blended with continuum equations. A mixed population of immune cell (macrophages, T cells) and *M. tuberculosis* bacterial agents migrate in a lattice field of soluble cytokines described by PDEs. The richness of this multi-format hybrid model allows prediction of granuloma formation and tissue pharmacokinetics of antibiotic therapies [12–14]. Reaction-diffusion kinetics of cytokines such as TNF-α binding to cell surfaces initiates a set of coupled ODE kinetic cascades within each agent, which in turn feeds back into the death, migration, and response of the agents [12]. This ABM/ODE/PDE multi-format model requires fast integration of the ODEs and PDEs nested in the slower processes of discrete agent movement, growth, cell-cell interactions, and state transitions. This multi-format model present challenges in aspects such consistency of units. For example, soluble molecules are described volumetrically (molar) while cell-associated species are in molecules/cell; thus a scaling factor is required assuming cell density within a cubic microcompartment. Temporal operator splitting is performed in which the derivatives of shared biochemical species are decoupled into an extracellular diffusion operator, agent-associated reactions, and extracellular molecular degradation. Advantages of using a non-sequential splitting scheme over executing each operator in series is detailed in [44].

3. Hybrid computational tools

In this section, we present some of the computational tools which have been developed or expanded in recent years to implement and solve hybrid models. We refer the reader to works by Ghosh *et al* [49] and Tavassoly *et al* [50], among others, for more comprehensive reviews of traditional computational methods, including those involving older integrated platforms.

Following the formalization of hybrid systems [28], several software packages and pipelines have been introduced for implementing and simulating these models [5]. While a number of these tools have been developed to handle the numerical challenges of implementing hybrid systems generically (e.g. [51–53]) we focus here on computational tools that have been specifically developed to facilitate hybrid modeling of biological systems. BooleanNet is an early example of a Python-based tool for simulating Boolean networks which is also capable of blending discrete network descriptions with continuous dynamics through piece-wise linear differential equations [54]. This piece-wise linear formalism is accomplished by associating each node with two variables—a discrete variable (e.g. activity) and a continuous variable (e.g. concentration). A differential equation for change in the continuous variable is calculated by the balance of synthesis and decay; synthesis incorporates the processing of discrete variable based upon threshold criteria, whereas decay is an unregulated first order kinetic process. The output of BooleanNet employing this formalism is smooth, and provides a finer level of detail than a Boolean network could; this is exemplified with a case study modeling the effect of abscisic acid signaling on stomatal closure in plants [54]. Another Python-based package, PyDSTool is a development environment aimed at simulating and analyzing biological models involving dynamical systems. It makes use of hybrid systems as a means of model reduction, simplifying traditional (e.g. ODE) models for systems which already exhibit an ability to be separated into states/modules according to functionality or inherent structure [55]. In particular, PyDSTool allows these modules to be defined as a sub-models and provides users with high-level tools for ease of development, even allowing model descriptions to be imported from other packages which interpret markup languages including SBML (see section 3.1). Moreover, it employs established solvers for sub-models and supports the implementation of hybrid systems models whose discrete controls are more complex than just simple conditional checks.

We discussed a diverse set of computational tools built for hybrid models in section 2 which range from general-purpose software to specialized pipelines. While ready-to-use programs like CoRegFlux [18] allow users to quickly apply hybrid models to their work, pipelines like the ones developed by Khan *et al* [9] and Liu *et al* [36] still serve as framework examples which can be adapted to different contexts. The pipeline by Ahmadian *et al* [11] on the budding yeast cell cycle, which combines SSA and ODE sub-models, not only serves as an example of multi-scale modeling but is itself inspired by the work from Liu *et al* on the eukaryotic cell cycle (see section 2.3). And though the pipelines developed by Khan *et al* and Ryll *et al* [10] make use of CellNetAnalyzer [29] as an interface between ODE and multi-valued logic sub-models (see section 2.2), it is worth noting that CellNetAnalyzer also allows users the flexibility of incorporating constraint-based sub-models as well [56]. In this way, CellNetAnalyzer has the capability of combining metabolic network descriptions with those involving regulatory networks and/or signal transduction. As detailed in section 2.4, the aforementioned fact means that CellNetAnalyzer is part of a growing group of computational tools like CoRegFlux, PROM [17], MatNet [15], FlexFlux [16], and the pipeline by van der Zee and Barberis [20] which include metabolic network descriptions as part of larger, more comprehensive models. The availability of such general-purpose tools highlights the importance of the novel insight provided by constraint-based hybrid models. Separately, we note that the ABM developed by Biggs and Papin [15] is part of a growing number of custom-code ABMs which incorporate multi-scale modeling along with some level of hybridization [44, 45]. And while there are numerous general-purpose platforms for ABM development [57], some platforms have specifically been developed to facilitate hybrid modeling of biological systems. In particular, PhysiBoSS is a recent platform which combines Boolean network descriptions of intracellular signalling using MaBoSS [33] with PhysiCell [58], a multi-scale ABM platform for simulating multicellular systems.

3.1. Markup languages

It is worth considering how both traditional and hybrid models are actually described inside computational platforms given that such tools are constantly in development and that hybrid modeling can naturally involve communication across existing tools. To this end, we discuss markup languages: machine-readable formats for encoding the information associated to biological system models. Markup languages have standardized the formal descriptions of models over the last few decades in order to make implementing and solving these models easier using external computational tools. And while there are several markup languages which are commonly used for systems biology models (e.g. CellML [59, 60], ISML [61, 62], PNML [63, 64], SBRML [65], and SED-ML [66, 67]), the most popular is the Systems Biology Markup Language (SBML) [68, 69] with over 280 associated tools [70]. Because of its ubiquity, SBML has been the natural choice for the development and/or use of several computational tools associated to traditional and hybrid models; it is also often used alongside the Systems Biology Graphical Notation (SBGN) [71]. Computational tools for hybrids models employing SBML include CellNetAnalyzer [29], FlexFlux [16], and pipelines such as the one developed by van der Zee and Barberis [20]; see section 2 for details. There are several other examples of computational tools employing SBML which have been built or suggested for hybrid models involving ODEs, Boolean/multi-valued logic, stochastic approximation, and/or constraint-based descriptions; see [3, 10, 55, 72, 73] for details.

As apparent from the number of tools highlighted in this section and the diversity of how we have defined hybrid modeling in this review, there is not a singular 'one-size-fits-all' platform for spanning the axes depicted in figure 1. Modelers should be cognizant of the challenges in model sharing for specialized code and consider compatibility with SBML and SBGN [71, 74] such that pre-existing model repositories originally designed for curation of ODE systems (such as the EMBL-EBI hosted BioModels database [75]) can be broadened to include the variety of biological models built for hybrid description.

4. Conclusion

New opportunities for systems biology modeling arise with the parallel processing made possible by graphical processing unit (GPU) computing. Originally designed for enhancing player experiences in video gaming, GPUs are now widely used in systems biology research to speed up complex simulations by parallel execution of code. In particular, GPUs allow ABMs to be applied in contexts once confined to single cell scope, enabling description of complex multicellular dynamics of embryo morphogenesis and tissue differentiation [45, 58]. Parallel processing also opens up enhanced opportunities for model optimization by running numerous iterations of parameter assignment simultaneously. We note, however, that speed will ultimately be dictated by the slowest step in sequential updating of a simulation. Furthermore, many of the tools developed for hybrid modeling were not driven by a need for faster simulations, but rather a need to

circumvent limits of knowledge of a network, of time-dependent changes, or of modes of communication across modules or hierarchies in cellular systems.

Not only are computing resources rapidly changing, but in contemporary times biological datasets are increasingly easy to acquire. Longitudinal data and broad coverage over many molecular components will only continue to grow in acquisition, mandating analytical tools capable of 'stitching' together dynamic and static detail. Multi-omics data mining is in wide use currently, but imposing prior knowledge with mechanistic detail in the form of systems biology modeling is a more difficult task. Data integration and simulation in microbial systems has pioneered hybrid modeling analyses due to the smaller genome sizes and ability to obtain dynamic growth and response profiles readily in lab settings [15, 17, 48]; now, the application space is expanding. We are now witnessing an exponential increase in publicly available human health data which in turn is driving increasingly sophisticated modeling platforms. While certainly the trend of data mining will continue, machine learning does not preclude the goals of improving hybrid systems models which incorporate varied description of prior biological knowledge into mathematical form. The example applications provide here illustrate the challenges in hybridizing systems analysis but also suggest that the frontier of systems modeling lies at the interfaces of numerical approaches which provide advantageous features of each modeling form.

Data availability statement

No new data were created or analysed in this study.

Acknowledgment

This research was supported by the NSF-Simons Southeast Center for Mathematics and Biology (SCMB) through the grants National Science Foundation DMS1764406 and Simons Foundation/SFARI 594594.

ORCID iDs

Daniel A Cruz https://orcid.org/0000-0001-5832-5228
Melissa L Kemp https://orcid.org/0000-0003-3781-8802

References

[1] Ideker T and Lauffenburger D 2003 *Trends Biotechnol.* **21** 255–62
[2] Machado D, Costa R, Rocha M, Ferreira E, Tidor B and Rocha I 2011 *AMB Express* **1** 45
[3] Le Novère N 2015 *Nat. Rev. Genet.* **16** 146–58
[4] Stéphanou A and Volpert V 2016 *Math. Modelling Nat. Phenom.* **11** 37–48
[5] Bartocci E and Lio P 2016 *PLoS Comput. Biol.* **12** e1004591
[6] Tegner J, Zenil H, Kiani N, Ball G and Gomez-Cabrero D 2016 *Phil. Trans. R. Soc.* A **374** 20160144
[7] Singhania R, Sramkoski R, Jacobberger J and Tyson J 2011 *PLoS Comput. Biol.* **7** e1001077
[8] Liu L and Bockmayr A 2020 *Acta. Biotheor.* **68** 73–85
[9] Khan F, Schmitz U, Nikolov S, Engelmann D, Pützer B, Wolkenhauer O and Vera J 2014 *Biochim. Biophys. Acta* **1844** 289–98
[10] Ryll A, Bucher J, Bonin A, Bongard S, Gonçalves E, Saez-Rodriguez J, Niklas J and Klamt S 2014 *Biosystems* **124** 26–38
[11] Ahmadian M, Wang S, Tyson J and Cao Y 2017 Hybrid ODE/SSA model of the budding yeast cell cycle control mechanism with mutant case study *Proc. 8th ACM Int. Conf. on Bioinformatics, Computational Biology, and Health Informatics* (Boston, MA: Association for Computing Machinery) pp 464–73
[12] Fallahi-Sichani M, El-Kebir M, Marino S, Kirschner D and Linderman J 2011 *J. Immunol.* **186** 3472–83
[13] Warsinske H, Pienaar E, Linderman J, Mattila J and Kirschner D 2017 *Front. Immunol.* **8** 1843
[14] Pienaar E, Sarathy J, Prideaux B, Dietzold J, Dartois V, Kirschner D and Linderman J 2017 *PLoS Comput. Biol.* **13** e1005650
[15] Biggs M and Papin J 2013 *PLoS One* **8** e78011
[16] Marmiesse L, Peyraud R and Cottret L 2015 *BMC Syst. Biol.* **9** 93
[17] Chandrasekaran S and Price N 2010 *Proc. Natl Acad. Sci.* **107** 17845–50
[18] Banos D, Trébulle P and Elati M 2017 *BMC Syst. Biol.* **11** 134
[19] Mayalu M and Asada H 2015 Multi-model selection of integrated mechanistic-empirical models describing T-cell response *2015 American Control Conf. (ACC)* pp 3132–7
[20] van der Zee L and Barberis M 2019 *Methods Mol. Biol.* **2049** 365–85
[21] Berestovsky N, Zhou W, Nagrath D and Nakhleh L 2013 *PLoS Comput. Biol.* **9** e1003306
[22] Ferrell J J and Machleder E 1998 *Science* **280** 895–8
[23] Kholodenko B, Demin O, Moehren G and Hoek J 1999 *J. Biol. Chem.* **274** 30169–81
[24] Bhalla U and Iyengar R 1999 *Science* **283** 381–7
[25] Bhalla U, Ram P and Iyengar R 2002 *Science* **297** 1018–23
[26] Schoeberl B, Eichler-Jonsson C, Gilles E and Müller G 2002 *Nat. Biotechnol.* **20** 370–275
[27] Bortolussi L and Policriti A 2008 Hybrid systems and biology *Formal Methods for Computational Systems Biology (SFM) 2008* vol 5016 pp 424–48

[28] Alur R, Courcoubetis C, Henzinger T and Ho P H 1993 Hybrid automata: an algorithmic approach to the specification and verification of hybrid systems *Hybrid Systems* ed R Grossman, A Nerode, A Ravn and H Rischel (Berlin: Springer) pp 209–29
[29] Klamt S, Saez-Rodriguez J and Gilles E 2007 *BMC Syst. Biol.* **1** 2
[30] BioRender Developers 2021 BioRender (available at: https://biorender.com/) (Accessed 11 August 2021)
[31] Layne L, Dimitrova E and Macauley M 2012 *Bull. Math. Biol.* **74** 422–33
[32] Müssel C, Hopfensitz M and Kestler H 2010 *Bioinformatics* **26** 1378–80
[33] Stoll G, Viara E, Barillot E and Calzone L 2012 *BMC Syst. Biol.* **6** 116
[34] Ha S *et al* 2010 *BMC Bioinform.* **20** 508
[35] Vera J, Schmitz U, Lai X, Engelmann D, Khan F, Wolkenhauer O and Pützer B 2013 *Cancer Res.* **73** 3511–24
[36] Liu Z, Pu Y, Li F, Shaffer C, Hoops S, Tyson J and Cao Y 2012 *J. Chem. Phys.* **136** 034105
[37] Chen K, Calzone L, Csikasz-Nagy A, Cross F, Novak B and Tyson J 2004 *Mol. Biol. Cell* **15** 3841–62
[38] Janes K, Albeck J, Gaudet S, Sorger P, Lauffenburger D and Yaffe M 2005 *Science* **310** 1646–53
[39] Janes K, Gaudet S, Albeck J, Nielsen U, Lauffenburger D and Sorger P 2006 *Cell* **124** 1225–39
[40] Csete M and Doyle J 2004 *Trends Biotechnol.* **22** 446–50
[41] Kemp M, Wille L, Lewis C, Nicholson L and Lauffenburger D 2007 *J. Immunol.* **178** 4984–92
[42] Gaudet S, Janes K, Albeck J, Pace E, Lauffenburger D and Sorger P 2005 *Mol. Cell Proteom.* **4** 1569–90
[43] Bordbar A, Yurkovich J, Paglia G, Rolfsson O, Sigurjonsson O and Palsson B 2017 *Sci. Rep.* **7** 46249
[44] Cilfone N, Kirschner D and Linderman J 2015 *Cell. Mol. Bioeng.* **8** 119–36
[45] Glen C, Kemp M and Voit E 2019 *PLoS Comput. Biol.* **15** e1006577
[46] Wilensky U 1999 NetLogo *Center for Connected Learning and Computer-Based Modeling* (Evanston, IL: Northwestern University) (available at: http://ccl.northwestern.edu/netlogo/)
[47] Liu X, Maiorino E, Halu A, Glass K, Prasad R, Loscalzo J, Gao J and Sharma A 2020 *Nat. Commun.* **11** 6043
[48] Karr J, Sanghvi J, Macklin D, Gutschow M, Jacobs J, Bolival B J, Assad-Garcia N, Glass J and Covert M 2012 *Cell* **150** 389–401
[49] Ghosh S, Matsuoka Y, Asai Y, Hsin K-Y and Kitano H 2011 *Nat. Rev. Genet.* **12** 821–32
[50] Tavassoly I, Goldfarb J and Iyengar R 2018 *Essays Biochem.* **62** 487–500
[51] Donzé A 2010 Breach, a toolbox for verification and parameter synthesis of hybrid systems *Computer Aided Verification* vol 6174, ed T Touili, B Cook and P Jackson (Berlin: Springer) pp 167–70
[52] Annpureddy Y, Liu C, Fainekos G and Sankaranarayanan S 2011 S-TaLiRo: a tool for temporal logic falsification for hybrid systems *Tools and Algorithms for the Construction and Analysis of Systems* vol 9035, ed P Abdulla and K Leino (Berlin: Springer) pp 254–7
[53] Kong S, Gao S, Chen W and Clarke E 2015 dReach: δ-reachability analysis for hybrid systems *Tools and Algorithms for the Construction and Analysis of Systems* vol 9035, ed C Baier and C Tinelli (Berlin: Springer) pp 200–5
[54] Albert I, Thakar J, Li S, Zhang R and Albert R 2008 *Source Code Biol. Med.* **3** 16
[55] Clewley R 2012 *PLoS Comput. Biol.* **8** e1002628
[56] von Kamp A, Thiele S, Hädicke O and Klamt S 2017 *J Biotechnology* **261** 221–8
[57] Abar S, Theodoropoulos G, Lemarinier P and O'Hare G 2017 *Comput. Sci. Rev.* **24** 13–33
[58] Ghaffarizadeh A, Heiland R, Friedman S, Mumenthaler S and Macklin P 2018 *PLoS Comput. Biol.* **14** e1005991
[59] Lloyd C, Halstead M and Nielsen P 2004 *Prog. Biophys. Mol. Biol.* **85** 433–50
[60] Miller A, Marsh J, Reeve A, Garny A, Britten R, Halstead M, Cooper J, Nickerson D and Nielsen P 2010 *BMC Bioinform.* **11** 178
[61] Asai Y *et al* 2008 *J. Physiol. Sci.* **58** 447–58
[62] Asai Y, Abe T, Okita M, Okuyama T, Yoshioka N, Yokoyama S, Nagaku M, Hagihara K and Kitano H 2012 Multilevel modeling of physiological systems and simulation platform: PhysioDesigner, Flint and Flint K3 service *2012 Symp. on Applications and the Internet (Piscataway, NJ)* (IEEE) pp 215–9
[63] Weber M and Kindler E 2003 The petri net markup language *Petri Net Technology for Communication-Based Systems: Advances in Petri Nets* ed H Ehrig, W Reisig, G Rozenberg and H Weber (Berlin: Springer) pp 124–44
[64] Hillah L M, Kordon F, Lakos C and Petrucci L 2012 Extending PNML scope: a framework to combine petri nets types *Transactions on Petri Nets and Other Models of Concurrency VI* ed K Jensen, W van der Aalst, M Ajmone Marsan, G Franceschinis, J Kleijn and L Kristensen (Berlin: Springer) pp 46–70
[65] Dada J, Spasić I, Paton N and Mendes P 2010 *Bioinformatics* **26** 932–8
[66] Waltemath D *et al* 2011 *BMC Syst. Biol.* **5** 198
[67] Bergmann F *et al* 2018 *J. Integr. Bioinform.* **15** 20170086
[68] Hucka M *et al* 2003 *Bioinformatics* **19** 524–31
[69] Hucka M *et al* 2018 *J. Integr. Bioinform.* **15** 20170081
[70] SBML Contributors 2021 The systems biology markup language (available at: http://sbml.org/Main_Page) (Accessed 11 August 2021)
[71] Le Novère N *et al* 2009 *Nat. Biotechnol.* **27** 735–41
[72] Chiang H, Fages F, Jiang J and Soliman S 2015 *ACM Trans. Model. Comput. Simul.* **25** 14
[73] Herajy M, Liu F, Rohr C and Heiner M 2017 *BMC Syst. Biol.* **11** 71
[74] SBGN Contributors 2021 The systems biology graphical notation (available at: https://sbgn.github.io/) (Accessed 11 August 2021)
[75] Malik-Sheriff R *et al* 2020 *Nucleic Acids Res.* **48** D407–15