Quantitative computational models of molecular self-assembly in systems biology

Marcus Thomas\(^1\)\(^2\) and Russell Schwartz\(^3\)\(^4\)

\(^1\) Computational Biology Department, Carnegie Mellon University, 4400 Fifth Avenue, Pittsburgh, PA 15213, United States of America
\(^2\) Joint Carnegie Mellon University/University of Pittsburgh Ph.D. Program in Computational Biology, 4400 Fifth Avenue, Pittsburgh, PA 15213, United States of America
\(^3\) Department of Biological Sciences, Carnegie Mellon University, 4400 Fifth Avenue, Pittsburgh, PA 15213, United States of America
\(^4\) Author to whom any correspondence should be addressed.

E-mail: russells@andrew.cmu.edu

Keywords: biomolecular aggregates, biomolecular dynamics, self-organized systems, computer modeling

Abstract

Molecular self-assembly is the dominant form of chemical reaction in living systems, yet efforts at systems biology modeling are only beginning to appreciate the need for and challenges to accurate quantitative modeling of self-assembly. Self-assembly reactions are essential to nearly every important process in cell and molecular biology and handling them is thus a necessary step in building comprehensive models of complex cellular systems. They present exceptional challenges, however, to standard methods for simulating complex systems. While the general systems biology world is just beginning to deal with these challenges, there is an extensive literature dealing with them for more specialized self-assembly modeling. This review will examine the challenges of self-assembly modeling, nascent efforts to deal with these challenges in the systems modeling community, and some of the solutions offered in prior work on self-assembly specifically. The review concludes with some consideration of the likely role of self-assembly in the future of complex biological system models more generally.

1. Introduction

Self-assembly reactions account for the overwhelming majority of the reaction events occurring in the cell. Most eukaryotic proteins function normally in complexes and self-assembly of these complexes is a key step in nearly all major cellular functions [8]. Examples of processes critically dependent on self-assembly include genome replication [19, 147, 172, 195]; gene transcription and transcript degradation [19, 111, 127]; protein synthesis and degradation [53, 112]; cell movement and shape control [34, 45, 81, 200]; cell-to-cell communication including gap-junction assembly and regulation [188]; formation of membrane complexes such as pore-forming toxins [12]; and mechanotransduction [9, 198, 202]. Through these processes, the assembly and disassembly of molecular complexes and machines plays a crucial role in essentially all regulatory processes in cell biology. Given the centrality of self-assembly to cell biology, one cannot hope to develop truly comprehensive quantitative models of systems biology without tackling self-assembly. Yet self-assembly has until recently been largely absent from major efforts at developing general systems biology modeling tools (e.g. [60, 65, 82, 108, 146, 160, 181, 182, 185]) or handled only with one-off special cases for particular systems of importance (e.g. [59, 96, 199]). Even the most ambitious efforts at large-scale biochemical modeling largely focus on traditional enzymatic chemistry or transcriptional dynamics and only implicitly model the self-assembly reactions involved in those processes (e.g. recent comprehensive models of whole-cell or whole-organism transcriptional and metabolic modeling [18, 189]). This situation is beginning to change as some major systems biology tools (e.g. [54, 68, 69]) and modeling efforts [96] have begun to incorporate methods suitable to complex self-assembly, but major challenges remain.

These challenges of self-assembly modeling largely arise from the extremely large space of possible pathways accessible to the intermediate species of a self-assembly reaction network. The number of possible reaction trajectories by which a set of free monomers can assemble into a complex grows in general exponentially in the complex size, leading to an enormous combinatorial explosion in pathway space for even moderate-sized assemblies and astronomical numbers for large complexes, such as virus capsids or cytoskeletal
networks. This is problematic for experimental study of assembly systems, as it is rarely possible to discriminate experimentally among these pathways except at a coarse level, particularly for highly symmetric or repetitive structures. It likewise creates problems for the most popular modeling methods. Mass action differential equation (DE) models are generally unsuitable for non-trivial assemblies because they require either extensive simplifications [56, 70, 125] or enormous numbers of equations and variables to account for the many possible intermediates [90]. Brownian dynamics (BD) models, even highly coarse-grained [10, 17, 51, 167], are likewise challenged by the large numbers of reactants and long timescales typical of self-assembly systems, requiring themselves great simplifications of reaction processes that generally make them unsuitable for accurate quantitative modeling [58]. Methods based on Gillespie’s stochastic simulation algorithm (SSA) can provide an effective balance between DE and BD, but face their own challenges because the underlying reaction networks are too large to model explicitly [26, 59, 64]. For similar reasons, self-assembly networks are extremely challenging for experimental characterization [27, 29, 93, 100, 141, 222, 223] and model inference as well [104, 210]. For example, the high computational cost and large numbers of intermediate species make it computationally infeasible to learn models via prevailing Bayesian parameter inference schemes [67], which require large numbers of simulation trajectories.

Over the recent decades, however, a specialized literature on self assembly modeling has grown for handling a number of challenging systems of independent importance. Cytoskeletal assembly (i.e. actin and microtubule assembly) has been the subject of extensive modeling work, leading to many seminal results in the basic biophysics of molecular assembly processes. Viral capsid assembly [70] has a long history as one of the primary model systems for macromolecular self assembly, both from an experimental and a computational perspective. Another key model system is amyloid aggregation, the basis for many major public health threats, including Alzheimer’s disease, Huntington’s disease, Parkinsons disease, prion disease, and type II diabetes. Figure 1 shows a few examples of important model systems for self-assembly and models through which they have been studied. The practical importance of these and other systems has led them to attract their own modeling communities to find solutions to the special challenges of molecular self-assembly to computational modeling. In these fields, one can find studies both anticipating the challenges beginning to face broader systems biology efforts and often offering at least partial solutions to these challenges.

The remainder of this review will consider in more detail both the special difficulty of self-assembly modeling and the literature addressing it. It will first discuss some of the important roles of self-assembly in cellular biochemistry as well as the role of systems modeling methods in understanding these systems. It will then discuss some of the successful approaches to self-assembly modeling that have emerged through this literature, as well as continuing challenges. It will conclude with consideration of how quantitative self-assembly modeling may shape future efforts in modeling biological systems more generally.

2. Why does self-assembly (SA) matter?

2.1. The role of self-assembly in general cell biology

Self-assembly is everywhere in biology, beginning with the most fundamental processes of molecular biology, all of which depend on the self-assembly of specialized complexes, structures, or molecular machines. Examples of self-assembled molecular machines fundamental to molecular biology include DNA polymerases (replication), RNA polymerases (transcription), the spliceosome (splicing), the ribosome (translation), and the proteasome (protein degradation). Each of these processes is critical in different ways to the regulation of complex biological systems and thus has been the focus of specialized modeling efforts. For example, the transcription complex is one of the most well studied systems in molecular biology, with experimental work on the interaction of classic 1D and 3D diffusion of transcription factors [74] inspiring kinetic models of the recruitment process [95]. More specialized examples of self-assembly continue to be elucidated, with prominent recent examples including the RISC complex involved in miRNA [88, 116, 137, 168] and the Cas9-gRNA complex [33] implementing the CRISPR/Cas system [138, 186].

Within eukaryotic biology specifically, a more specialized set of self-assembly systems have evolved critical roles. The cytoskeleton is an unusually large, dynamic, and complicated molecular assembly, making it a crucial target of modeling efforts. The cytoskeleton itself is essential to intracellular transport [150, 152], cell movement and shape control [7, 149], mechanotransduction [201], and cell division [79], among many other functions. Furthermore, the dynamic process of assembly and disassembly is central to each of these functions. Actin and microtubule assembly and disassembly have been key model systems for self-assembly from the early days of molecular biology [20, 49, 61, 62, 91, 124, 134, 170, 204] and have inspired numerous computational models (e.g. [52, 57, 130, 159, 161, 179]). Transport processes in the eukaryotic cell frequently depend on other kinds of specialized self-assemblies, in addition to the cytoskeleton. For example, much eukaryotic transport involves the assembly of specialized machinery for construction and scission of cargo-carrying vesicles, such as the clathrin and COP-I/COP-II coat systems [50, 136], which have inspired their own modeling literature (e.g. [37, 87, 117]).

Beyond its role in general cell and molecular biology, self-assembly is crucial to a number of disease-specific processes. Amyloid diseases are perhaps the prime example of a disease specifically of self-assembly, where aberrant assembly is the mechanism of illness.
Numerous such diseases are known, including many major public health threats. Perhaps best known are Alzheimer’s disease (characterized by aggregates of the Aβ peptide and the Tau protein [94, 122]), Huntington’s disease (characterized by aggregates of the Huntingtin protein [113]), Parkinson’s disease [169], amyotrophic lateral sclerosis [197], type II diabetes, and a variety of known prion diseases such as Creutzfeldt–Jakob [28, 133]. Alzheimer’s and dementia, for example, are strongly associated with aging and affected roughly 36 million people in 2010 [208, 209]. It is becoming increasingly clear that the ability to form the amyloid state is a widespread, generic property of proteins [102] making the process of amyloidogenesis an important topic of theoretical study. From a physical perspective, the main question is what forces stabilize the aggregates into the oligomer (small soluble disordered clusters) and fibrillar (long, many-chain highly structured -sheet-containing aggregates) states associated with neurotoxicity [165]. For a broader discussion of these forces, see [109, 110, 139]. From a computational perspective, the focus is both on identifying the structure of oligomeric intermediates and fibers but also elucidating the kinds of assembly pathways available. This is an especially challenging computational problem due to the intrinsic disorder in the system.

Viral illnesses form another broad class of self-assembly-driven illness, in which assembly of large complexes (i.e. the viruses themselves) are the mechanism of the disease process. Virus assembly is of obvious medical importance, given the millions who die each year from viral illnesses, e.g. 1.5 million from AIDS alone [38]. A fundamental understanding of this crucial aspect of the viral life cycle and infectivity may offer avenues for therapeutics or vaccines [223]. Additionally, there many factors making viruses appealing to the modeling community, including the deep experimental literature on their assembly and a high degree of symmetry in the final structure that allows for large complexes to be produced from small numbers of distinct subunit types. Viral assembly modeling has thus become a subfield in itself. Virus assembly has been a crucial platform for many basic advances in self-assembly modeling, including the use of DE [220], BD [71, 128, 144, 167], and SSA [78, 97, 219] methods. It has likewise been a platform for developing a variety of specialized versions of these modeling methods, such as rule-based approaches to simulating extremely large reaction networks [89] and derivative-free optimization approaches to model inference [104, 210]. Viral capsids have been a focus of intense theoretical study into the basic biophysics of self-assembly [24, 42, 70, 71, 187] as well as for identifying potential new avenues for assembly-mediated treatment [31, 43, 86, 101, 105, 106, 143, 171, 187, 203, 215].

3. Self-assembly modeling and simulation

3.1. The challenge of quantitative modeling of self-assembly reaction networks

At the root of much of the difficulty of modeling self-assembly is the extraordinarily large number of intermediates and pathways potentially accessible to a self-assembly system. Large number of reactants present problems in different ways to most conventional modeling and model inference methods (see section 3.2
below). They likewise present a challenge to experimental characterization of such systems, as there is no practical way to monitor huge numbers of distinct molecular species. While details vary by geometry, in general the number of possible intermediates (partially built structures) one might encounter on the way to a complete assembly will blow up exponentially in the assembly size. This problem has probably been most intensively studied in the virus assembly literature, as it is particularly pronounced for large, highly symmetric structures, of which viral capsids are a prime example. Even a coarse-grained model of an icosahedral virus capsid, consisting of just twelve subunits, has 750 possible intermediate structures [121]. For real viral structures, which typically have several hundred proteins, the numbers of potential intermediates will be astronomical. Similar problems will arise to a lesser degree with large, asymmetric assemblies (e.g. the ribosome [103, 126]) as well as with larger but less symmetric assemblies such as the cytoskeleton. While the number of species possible for a linear filament is small, once one allows for branching [216], numbers of possible branched filaments or networks can blow up exponentially in the structure size as well. Note that this is not a unique challenge of self-assembly, as similar issues arise in other combinatorially explosive systems, such as signaling networks [16, 80].

A related concern for modeling, particularly with respect to self-assembly in cell biology, is the issue of small copy numbers [63, 194] resulting in an inherently discrete and stochastic reaction system. The issue occurs for many cellular systems involving reactants that occur in just a few copies per cell, but is especially an issue for self-assembly because the large number of intermediates guarantees that most are present in zero or one copies at any given time [135]. The issue is exacerbated by the fact that self-assembly reactions are frequently nucleation-limited, meaning that they are characterized by slow and relatively rare nucleation events followed by comparatively rapid polymerization. Nucleation-limited growth is well established for several of the major model systems in self-assembly, such as virus capsids [141, 217], amyloids [107], and actin and tubulin fibers [14]. A large body of theory suggests the nucleation-limited growth is crucial to their robust operation [44, 140–142, 156, 191]. In nucleation-limited systems, nearly every species is populated at most times. Small copy numbers are problematic computationally in part because they mean that discretization errors inherent to efficient continuum models become substantial. In part, they are problematic because they mean that self-assembly must be treated as a stochastic system, forcing the use of less efficient simulation methods than the continuum approximations usable when all species are well populated [41, 63, 194] (see section 3.4).

A second major challenge of self-assembly reactions is their long timescales (see figure 2), and in particular the large gap between timescales of the full assembly reaction and the individual polymerization steps of assembly. Full assembly reactions of large complexes in vitro may have timescales measured in minutes to days (although assembly in vivo may be substantially faster [36, 115, 176]) while individual reaction steps are typically many orders of magnitude faster [180, 222]. In part, this is a side effect of nucleation-limited growth mentioned above: nucleation reactions are necessarily much slower than the subsequent elongation reactions [167, 221]. Furthermore, the nucleation reactions themselves may in fact require extensive trial-and-error involving much faster formation and breakdown of transient partial intermediates [184, 217, 221]. Large timescales, and a large dynamic range of timescales, are challenges for essentially all standard modeling methods, whether that manifests in a need for large numbers of timesteps in a continuum method or large numbers of discrete events for a stochastic simulation.

A third class of challenge arises from the fact that self-assembly reactions are unusually sensitive to the many ways in which the physical biology of the cell differs from that of in vitro models. For example, physical confinement—by the cell membrane, subcellular compartments, or other large structures such as the cytoskeleton or genome—is commonly neglected in modeling reaction systems yet cannot be ignored when dealing with reactions that result in products comparable in size to the spaces in which they form. A related issue is that self-assembly processes are also well known to be unusually sensitive to macromolecular crowding [75, 119, 148], a key distinguishing feature of the cellular environment. Numerous theoretical and experimental studies have suggested both the need for and the challenge to correcting simulation methods to account for the effects of crowding on assembly processes (e.g. [129, 166]). Examples include the effects on several aspects of DNA replication such as helicase activity and the sensitivity of DNA polymerase to salt [1], on protein–protein binding affinity and specificity [99], on the kinetics and morphology of amyloid self-assembly [115], on the stochasticity of gene expression machinery [76], and on viral capsid assembly [36, 176].

3.2. Modeling methodologies
Despite the difficulties they present to modelers, a variety of modeling methods have proven valuable for self-assembly. Table 1 describes a few of the primary methods that have emerged for self-assembly modeling. While most are drawn from older techniques for more general reaction chemistry modeling, in the self-assembly context they often present novel challenges or require specialized adaptations. This section covers three of the most successful methodologies that have been developed for self-assembly, some of the particular challenges they have faced in the self-assembly context, and how they have been adapted to meet those challenges.

3.2.1. Mass action differential equation (DE) models
Much modeling of reaction systems classically has arisen, at least initially, from DE models based on the chemical Law of Mass Action. Such models represent any generic chemical reaction network
in terms of a system of differential equations of the form

\[
\frac{dX_i}{dt} = (k_{i+}b_{ij} - k_{i-}a_{ij})X^n_{i}X^{m_{i}} + \ldots
\]

Accumulating these contributions across a full set of reactions and reactant species defines a system of differential equations modeling the time evolution of all reactants in the system. Such DE models were the basis of many of the earliest cell simulation systems, such as E-cell [193], ProMoT/Diva [65], Virtual Cell [164], GEPASI [118] and others. Later extensions of these models allowed for consideration of spatial heterogeneity via partial differential equation (PDE) reaction-diffusion models:

\[
\frac{\partial X_i}{\partial t} = d_i \nabla^2 X_i + (k_{i+}b_{ij} - k_{i-}a_{ij})X^n_{i}X^{m_{i}} + \ldots
\]

for reactant-specific diffusion coefficients \(d_i\).

DE models provided a basis for some of the first approaches to modeling many self-assembly systems. Classic results on molecular assembly of polymers derived from such models include [131, 132] and they were integral to seminal models of microtubule polymerization [39]. They likewise were used for early attempts at more complex systems, such as the first dynamic models of viral self-assembly [221, 223], where they provided early insights into the parameter space of self-assembly [221]. They continue to prove valuable in that context for such problems as interpreting complex experimental data [27, 70, 173].

The most substantial challenge to DE models on self-assembly systems is computational tractability, as such models need to keep explicit track of all species that might be present in a given simulation. While that number grows only linearly in assembly size for linear polymers, it blows up exponentially in size for more complex structures such as viruses. In practice, the solution to that problem has typically been to simplify: either manually via simplified versions of structures or conflation of subsets of structures [220, 221] or through automated methods for pruning low-usage pathways [48]. While there is good empirical evidence that such strategies can yield quantitatively accurate models [48, 223], degrees of accuracy can be sensitive to structure and pathways used [121]. DE models further provide no good solution for the problem of modeling discretization of small copy number reactions.

3.3. Brownian dynamics (BD) models

The challenges self-assembly modeling presents to DEs led to an alternative approach based on Brownian dynamics (BD) particle models. In a BD approach, we explicitly model a finite set of assembly subunits in three dimensional space. These subunits diffuse through space under a model of Brownian motion, implemented by a variant of damped Langevin dynamics [51]. Models of binding dynamics can be implemented either by discrete reactions occurring upon particle collisions or via short-range binding forces, leading to gradual agglomeration of particles.
over the course of a simulation. BD models have the considerable advantage over DE models that one need not devote computational resources to any species not present at a specific instant in time. Run time thus depends on the number of particles modeled, not the number of species they might in principle form.

Such models have perhaps been most pronounced in their use with viral capsid systems, perhaps because their exceptionally large space of intermediates makes them especially challenging for DE models. Through viral capsid work, they have been the basis of numerous important insights into the basic biophysics of viral capsid work, they have been the basis of numerical integrated. PDEs must be used when space is explicitly included. BD models were introduced to self-assembly systems biology software packages implementing them, and some notable applications in self-assembly modeling.

### Table 1. Common modeling methodologies for self-assembly. The table lists principal techniques for self-assembly modeling, some systems biology software packages implementing them, and some notable applications in self-assembly modeling.

| Reaction representation | Description | Software packages | Applications |
|-------------------------|-------------|-------------------|--------------|
| Law of mass action (deterministic) | Expresses any well-mixed chemical system as a collection of coupled non-linear first order differential equations which typically must be numerically integrated. PDEs must be used when space is explicitly included | BioNetGen [15], COPASI [82], VCell [146], DBSolve [66] | Virus assembly: [27, 70, 125, 173], metabolomic networks [90] |
| SSA/Gillespie approaches | Provides a way to simulate kinetically correct trajectories consistent with the chemical master equation | Moleculizer [114], BioNetGen [15], VCell [146], DESSA [219] | Virus assembly: [97, 184] |
| Spatial stochastic | Usually combine Gillespie or Stochastic Langevin with diffusion or subunit geometry | MCell [181], StochSim [108], VCell [146], Smoldyn [4], SRSim [69] | Geometric constraints with diffusion: [68], Amyloid-beta: [192] |
| Rule-based | Primarily network-free rule-based methods which may incorporate stochasticity and spatial modeling | RuleMonkey [32], BioNetGen, ML-Space [13], VCell [163], SRSim [69] | Multivalent ligand-receptor interactions: [213], Prion aggregation [154], Virus assembly: [167, 219] |
| Brownian dynamics | An explicitly spatial model where Brownian motion is computed with the Langevin equation | Smoldyn [4], MCell [181] | Multiscale reaction-diffusion [58], virus assembly: [17, 46, 47, 71, 128, 167], crowding/amyloids: [205], clathrin cage formation: [87] |

3.4. Stochastic simulation algorithm (SSA) methods

Just as BD models were introduced to self-assembly modeling to address the weaknesses of DE models, so have models based on the stochastic simulation algorithm (SSA) [63] (also known as ‘Gillespie models’ after their inventor) been adopted to address the weaknesses of BD models. In an SSA model, we represent a system at an instant in time by discrete counts of molecular species (monomers or partial assemblies). Simulation progress proceeds via reaction events, which for a self-assembly system will largely consist of single binding or dissociation reactions.
Classically, one assumes a uniform, well-mixed system, in which reaction times can be approximated with exponential waiting time distributions [63]. The SSA approach can also accommodate spatial heterogeneity through modeling as an array of well-mixed, discretized spatial compartments, a variant known as spatial SSA, e.g. [5, 183].

SSA models offer considerable advantages but also involve important tradeoffs with the previously considered methods. They can be implemented to have run times independent of the number of potential species, unlike DE models, and can thus handle arbitrarily large reaction networks [64]. However, their run time does depend on the number of discrete particles present, limiting them to finite numbers of protein copies, as do BD models. They are, however, typically much more efficient than comparable BD models since they do not need to model diffusion explicitly [64] and are practical over a much broader range of parameter domains [184]. In addition, they provide an explicit quantitative model involving them to finite numbers of protein copies, as do BD models. They are, however, typically much more efficient than comparable BD models since they do not need to model diffusion explicitly [64] and are practical over a much broader range of parameter domains [184]. In addition, they provide an explicit quantitative model yielding kinetically correct samples from a set of reactions and associated rate constants. However, because they do not explicitly model space, they do not easily handle steric constraints that are important to such processes as aberrant assembly [167], interaction of proteins with a flexible genome [47, 214], or any form of continuous flexibility in proteins or complexes [81, 153].

SSA methods have needed some special adaptations to deal with the challenges of self-assembly. Probably the most important advance is the use of rule-based modeling (e.g. [177]), a strategy independently developed for the self-assembly field under the name local rule modeling [167] and later introduced to SSA models under that name [219]. Rule-based models allow one to avoid explicitly constructing the reaction network, an infeasible task for all but trivial self-assembly systems, and rather represent only the current state of the system and its immediate neighbors [55, 219]. This reduces run time from dependence on the size of the network to dependence only on the number of species and reactions present at any instant in time. While steric constraints are a challenge for rule-base models, that challenge has been overcome for some systems, e.g. in modeling multivalent ligand-receptor interactions [123]. Further improvements to queuing methods for discrete event implementations of SSA [40, 89] made it possible to accelerate run time by eliminating quadratic time/memory dependencies in the standard algorithms. Additionally, a set of more specialized theory has been developed to deal with the problem of extreme divergence between timescales of monomeric reactions versus the complete assembly process. Generic methods for accelerating SSA can be helpful, e.g. [25, 145], as well as more specialized variants specifically for self-assembly [120]. Other improvements include hybrid methods combining SSA with ideas from agent-based modeling [3].

While SSA methods have not yet seen as wide use as BD in the self-assembly field, they have proven to have important applications for which neither DE nor BD methods are suitable. Because of their ability to handle complex geometries and long time scales, SSA models have proven valuable for exploring parameter dependence of assembly systems by making it practical to sample large numbers of trajectories over long time scales [211] and to sample trajectories from particularly complex geometries or pathway sets [97]. They have also become a valuable platform for fitting models to experimental data, where their ability to fit an explicit timescale, to function over wide parameter ranges, and to model complex geometries are all crucial features [175, 176, 210, 211].

4. Self-assembly in broader systems biology modeling

In recent years, efforts at systems biology modeling have begun increasingly to recognize the importance of self-assembly to comprehensive modeling of complex biochemical systems. For example, a number of general systems biology simulation tools have begun to incorporate handling of self-assembly in various ways. An early example was Moleculizer [114], which incorporated basic models of assembly reactions via a rule-based SSA model with special purpose corrections accounting for altered diffusion rates of growing species. Similar kinds of models have become important more generally in modeling tools, such as RuleBender [212], which have made it possible to integrate similar rule-based SSA models into other tools for systems biology modeling. The Virtual Cell [162] has recently added handling of self-assembly reactions, using a special-purpose extension based on a form of coarse-grained BD models of self-assembly [4, 35], as well as explicit handling of rule-based modeling [163]. The most recent version of the E-Cell [193] simulation environment (ECell4) has also been updated to include capabilities for modeling self-assembly such as a network-free rule model [55] and a spatial SSA method [183]. While none of these systems yet incorporates all of the specializations found in such methods in self-assembly specific contexts, they represent important steps towards generic tools for modeling complex reaction networks that include but are not specific to self-assembly.

This need for handling the kind of combinatorial explosive reaction network that characterizes self-assembly is also beginning to be reflected in systems biology language design. For example, the systems biology markup language (SBML) [84, 85], which has become the de facto standard for specifying models in systems biology, has been updated in more recent versions to accommodate the kind of network-free rule-based models needed for self-assembly work [83]. While it has long been possible to generate SBML from a rule specification through external tools, such as BioNetGen [54], native support of the modeling language is necessary to achieve the benefits of network-free
modeling needed to make complicated self-assembly modeling tractable. Handling of steric constraints that become important in formation of more complicated assemblies remains a hard problem for the field, however, and is so far handled only in more specialized self-assembly simulation languages [219].

Recent years have also seen claims of the first true whole-cell simulations [96, 157], an effort that necessarily involves modeling numerous processes that depend on self-assembly. In practice, such efforts have not relied on a general-purpose simulation engine suitable to both self-assembly and more conventional reaction chemistry, favoring instead general purpose methods ill-suited to self-assembly coupled to special-purpose handling of particular kinds of self-assembly. The landmark work of Karr et al [96] establishing a comprehensive simulation of M. genitalium biochemistry, relied on a series of special-purpose modules, several of which involved ad hoc methods for specific examples of self-assembly, such as macromolecular complexation and ribosome assembly. Nonetheless, even this kind of special-purpose handling remains the exception in similar efforts at comprehensive modeling of whole-cell reaction networks (e.g. [18, 189]).

5. Conclusions

Self-assembly is a greatly important but long neglected issue in the quantitative modeling of biological systems. While it is conventionally seen as a specialized form of chemistry, it is in fact the dominant form of reaction in living systems. It poses distinctive challenges for modeling methods, though, that prevailing methods in systems biology cannot handle. Self-assembly modeling has, however, been studied intensively in many more specialized contexts, leading to an appreciation of these challenges and a variety of ways they can be addressed. As more general systems biology efforts are beginning to embrace the necessity of accommodating self-assembly, this specialized literature can provide guidance and at least partial answers to some of the biggest obstacles these efforts will encounter. This review was intended to provide a brief overview of the particular challenges of self-assembly modeling, how they have been approached to date, and how these methods have been used in the past and are beginning to be incorporated into comprehensive models of systems biology. The hope behind this review is that better awareness of obstacles and solutions already identified by self-assembly modelers can assist the broader systems modeling community in anticipating and navigating the same issues.

An appreciation for the past literature allows us to predict some of the future paths comprehensive systems modeling efforts are likely to follow. For the most part, where general efforts at systems biology modeling have considered self-assembly, it has been as special cases with special-purpose methods for specific systems (e.g. [10, 17, 47, 92, 99, 155, 158, 174, 178, 210, 211, 219]). Given the many examples of self-assembly in cell biology, it is safe to say this is not a sustainable solution; rather general systems biology efforts will need to start to think of self-assembly as the normal case that must be accommodated and integrated into simulation design via both model specifications and simulation algorithms. Modeling methods that will work for both self-assembly and for other kinds of chemistry exist [3, 13, 15, 32, 114, 146, 185, 199], but will need to become the standard for modeling tools and languages. More fore-sighted efforts in a variety of systems modeling contexts can help point the way (e.g. [4, 35, 96, 114, 164, 193]), although most remain behind the state-of-the-art in modeling of self-assembly specifically.

At the same time, there are many challenges for which good solutions do not yet exist. For example model inference [30] remains an extremely difficult problem for self-assembly systems [104, 210, 211, 223], where the Bayesian methods usually favored by the field [67] are unusable in practice, and it is likely advances in both biotechnology and inference algorithms will be needed to address it. The field is beginning to tackle this challenge, e.g. with BioNetFit [190], which uses a genetic algorithm to provide curve fitting capabilities compatible with ODE (BioNetGen) and Network Free (NFSim) model specifications and has proven successful in fitting to steady-state and time-series oligomerization data. There are also, as yet, no universally good methods for modeling hard self-assembly systems. Each of the major approaches covered here—SSA [64], BD [167], and DE [220]—has tradeoffs that make them unsuitable for some questions. It remains to be seen whether more general solutions might arise from advances in one or more of these methods, clever hybrid approaches, or some wholly new ideas. It is also worth noting that self-assembly systems are challenging to characterize experimentally, for similar reasons to their challenge to modelers. The solutions to that issue, as well, are likely to lie in pooled efforts by experimentalists and computational researchers to advance experimental biotechnology and model-fitting algorithms in complementary ways. Indeed, self-assembly may be a particularly valuable test case for addressing the hard problems in building detailed and predictive quantitative models of complex biological systems, where the field can begin to think of modelers and experimentalists not as two communities but as two inseparable pieces of the future practice of biological discovery.

Acknowledgments

MT was supported in part by US National Institutes of Health award T32EB009403. RS was supported in part by US National Institutes of Health award RO1GM117425, US Air Force Office of Scientific Research Award FA9550-13-1-0108, Commonwealth of Pennsylvania CURE grant 1042545, and an award...
from the University of Pittsburgh Medical Center (UPMC). The Pennsylvania Department of Health specifically disclaims responsibility for any analyses, interpretations or conclusions.

Certain images in this publication have been obtained by the author(s) from the Wikipedia/Wikimedia website, where they were made available under a Creative Commons licence or stated to be in the public domain. Please see individual figure captions in this publication for details. To the extent that the law allows, IOP Publishing [and full partner name (if applicable)] disclaim any liability that any person may suffer as a result of accessing, using or forwarding the image(s). Any reuse rights should be checked and permission should be sought if necessary from Wikipedia/Wikimedia and/or the copyright owner (as appropriate) before using or forwarding the image(s).

References

[1] Akabayov B, Akabayov S R, Lee S J, Wagner G and Richardson C C 2013 Impact of macromolecular crowding on dna replication Nat. Commun. 4 1615
[2] Alioscha-Perez M, Benadiba C, Goossens K, Kasas S, Dietler G, Willaert R and Sahli H 2016 A robust actin filaments image analysis framework PLoS Comput. Biol. 12 e1005063
[3] Amar P and Paulevé L 2015 Hium: a hybrid stochastic simulation system for systems biology Electron. Notes Theor. Comput. Sci. 313 3–21
[4] Andrews S S, Addy N J, Brent R and Arkin A P 2010 Detailed chemical reactions with spatial resolution and single molecule detail Phys. Biol. 1 137
[5] Andrews S S and Bray D 2004 Stochastic simulation of chemical reactions with spatial resolution and single molecule detail PLoS Comput. Biol. 6 e1000705
[6] Apetri M M, Harkes R, Subramaniam V, Canters G W, Schmidt T and Aartsma T J 2016 Direct observation of α-synuclein amyloid aggregates in endocytic vesicles of neuroblastoma cells—licensed under cc by 4.0 PLoS One 11 e0153020
[7] Ausmees N, Kuhn J R and Jacobs-Wagner C 2003 The bacterial cytoskeleton: an intermediate filament-like function in cell shape Cell 115 705–13
[8] Bai L, You Q and Liu J 2016 Protein self-assembly via supramolecular strategies Chem. Soc. Rev. 45 2756–67
[9] Bailey J, Critser P, Whittington C, Kuske J, Yoder M C and Vostrik-Harbin S 2011 Collagen oligomers modulate physical and biological properties of three-dimensional self-assembled matrices Biomaterials 95 77–93
[10] Baschek J E, Klein H C and Schwarz U S 2012 Stochastic dynamics of virus capsid formation: direct versus hierarchical self-assembly BMC Biophys. 5 22
[11] Bieler N S, Knowles T R, Frenkel D and Vácha R 2012 Connecting macroscopic observables and microscopic assembly events in amyloid formation using coarse grained simulations PLoS Comput. Biol. 8 e1002692
[12] Biscoberger M, Lacovache I, Boss D, Naf E, van der Goot F G and Molina N 2016 Revealing assembly of a pore-forming complex using single-cell kinetic analysis and modeling Biophys. J. 110 1574–81
[13] Bittig A and Uhrmacher A 2016 Mi-space: hybrid spatial Gillespie and particle simulation of multi-level rule-based models in cell biology IEEE/ACM Trans. Comput. Biol. Bioinform. (https://doi.org/10.1109/TCBB.2016.2598162)
[14] Blanchion L, Amann K J, Higgs H N, Marchand J B, Kaiser D A and Pollard T D 2000 Direct observation of dendritic actin filament networks nucleated by arp2/3 complex and wasp/scar proteins Nature 404 1007–11
[15] Blinov M L, Faeder J R, Goldstein B and Hlavacek W S 2004 BioNetGen: software for rule-based modeling of signal transduction based on the interactions of molecular domains Bioinformatics 20 3289–91
[16] Blinov M L, Faeder J R, Goldstein B and Hlavacek W S 2006 A network model of early events in epidermal growth factor receptor signaling that accounts for combinatorial complexity Biosystems 83 136–51
[17] Boettcher M A, Klein H C and Schwarz U S 2013 Role of dynamic capsomere supply for viral capsid self-assembly PLoS Biol. 12 016014
[18] Bordbar A, Monk M J, King Z A and Palsson B O 2014 Constraint-based models predict metabolic and associated cellular functions Nat. Rev. Gen. 15 107–20
[19] Burgess R J and Zhang Z 2013 Histone chaperones in nucleosome assembly and human disease Nat. Struct. Mol. Biol. 20 14–22
[20] Buxbaum R and Heidemann S 1988 A thermodynamic model for force integration and microtubule assembly during axonal elongation J. Theor. Biol. 134 379–90
[21] By Awapuhi Lee (Own work) 2013 (CC BY-SA 4.0 (http://creativecommons.org/licenses/by-sa/4.0)) via Wikimedia Commons 1x9j
[22] By Doc. RNDr. Josef Reischig, CSc. (Author’s archive) 2014 (CC BY-SA 3.0 (http://creativecommons.org/licenses/by-sa/3.0)), via Wikimedia Commons. Chromatin fibers (261 19) mitosis; xenopus egg
[23] By Nepron (Own work) 2010 (CC BY-SA 3.0 (http://creativecommons.org/licenses/by-sa/3.0) or GFDL (www.gnu.org/copyleft/fdl.html)), via Wikimedia Commons. Cerebral amyloid angiopathy -2b- amyloid beta—very high mag
[24] Cadena-Nava R D, Comas-Garcia M, Garmann R F, Rao A, Knobler C M and Gelbart W M 2012 Self-assembly of viral capsid protein and rna molecules of different sizes: requirement for a specific high protein/rna mass ratio J. Virol. 86 3315–26
[25] Cao Y, Gillespie D T and Petzold L R 2006 Efficient step size selection for the tau-leaping simulation method J. Chem. Phys. 124 044110
[26] Cao Y, Li H and Petzold L 2004 Efficient formulation of the stochastic simulation algorithm for chemically reacting systems J. Chem. Phys. 121 4059–67
[27] Casini G, Graham D, Heine D, Garcea R L and Wu D T 2004 In vitro papillomavirus capsid assembly analyzed by light scattering Virology 325 328–37
[28] Cescati M, Saverion D, Capellari S, Tagliavini F, Kitamoto T, Ironside J, Giese A and Parchi P 2016 Analysis of conformational stability of abnormal prion protein aggregates across the spectrum of creutzfeldt-jakob disease prions J. Virol. 90 6244–54
[29] Chen C, Cao C C and Dragnea B 2008 Self-assembly of brome mosaic virus capsids: insights from shorter time-scale experiments J. Phys. Chem. A 112 9405–12
[30] Chou I C and Voti E O 2009 Recent developments in parameter estimation and structure identification of biochemical and genomic systems Math. Biosci. 219 57–83
[31] Cole A G 2016 Modulators of hiv capsid assembly as an approach to treating hepatitis b virus infection Curr. Opin. Pharmacol. 30 131–7
[32] Colvin J, Monine M I, Gutenkunst R N, Hlavacek W S, Von Hoff D D and Posner R G 2010 Baalemonkey: software for stochastic simulation of rule-based models BMC Bioinform. 11 1
[33] Cong L et al 2013 Multiplex genome engineering using crispr/cas systems Science 339 819–23
[34] Cosgrove D J 2014 Re-constructing our models of cellulose and primary cell wall assembly Curr. Opin. Plant Biol. 22 122–31
[35] Cowan A E, Moraru I I, Schaff J C, Slepechenko B M and Loew L M 2012 Spatial modeling of cell signaling networks Methods Cell Biol. 110 195
[83] Hucka M, Bergmann F T, Hoops S, Keating S M, Sahle S, Schaff J C, Smith L P and Wilkinson D J 2015 The systems biology markup language (SBML): language specification for level 3 version 1.0. J. Bioinform. Biochem. 41 533–6

[84] Hucka M et al 2004 Evolving a lingua franca and associated software infrastructure for computational systems biology: the systems biology markup language (sbml) project. J. Biol. 1 41–53

[85] Hucka M et al 2003 The systems biology markup language (sbml): a medium for representation and exchange of biochemical network models Bioinformatics 19 524–31

[86] Hueging K et al 2014 Apolipoprotein e codetermines tissue tropism of hepatitis c virus and is crucial for viral cell-to-cell transmission by contributing to a postenvelopement step of assembly. J. Virol. 88 1433–46

[87] Ilie I M, den Otter W K and Bries W J 2014 Rotational brownian dynamics simulations of clathrin cage formation J. Chem. Phys. 141 065101

[88] Iwasaki S, SASaki H M, Sakaguchi Y, Suzuki T, Tadakuma H 2013 Discovery of novel small-molecule hiv-1 replication inhibitors that stabilize capsid complexes. Antimicrobial Agents Chemotherapy 57 4622–31

[89] Lampel A, Bram Y, Ezer A, Shaltiel-Kario R, Saad J S, Bacharach E and Gazit E 2015 Targeting the early step of building block organization in viral capsid assembly ACS Chem. Biol. 10 1785–90

[90] Lazo N D, Grant M A, Condron M C, Rigby A C and Teplov D B 2005 On the nucleation of amyloid β-protein monomer folding. Protein Sci. 14 1581–96

[91] Le Novere N and Shimizu T S 2001 Stochsim: modelling of stochastic biomolecular processes Bioinformatics 17 575–6

[92] Leckband D and Israelachvili J 2001 Intermolecular forces in biology Q. Rev. Biophys. 34 105–263

[93] Leckband D and Sivasankar S 1999 Forces controlling protein interactions: theory and experiment Cell Surfaces B 14 83–97

[94] Lee N N et al 2013 Mtr4-like protein coordinates nuclear rna processing for heterochromatin assembly and for telomere maintenance Cell 155 1061–74

[95] Li G W, Burkhart D, Gross C and Weissman J S 2014 Quantifying absolute protein synthesis rates reveals principles underlying allocation of cellular resources Cell 157 624–35

[96] Li H and Li X J 2014 Huntingtin-protein interactions and the pathogenesis of huntington’s disease Trends Genet. 20 146–54

[97] Lok I and Brent R 2005 Automatic generation of cellular reaction networks with molecular 1.0 Nat. Biotechnol. 23 131–6

[98] Ma B, Xie J, Wei L and Li W 2013 Macromolecular crowding modulates the kinetics and morphology of amyloid self-assembly by n-lactoglobulin Int. J. Biol. Macromol. 53 82–7

[99] Mahnkenych T and Houry W A 2012 The role of hsp90 in protein complex assembly Biochim. Biophys. Acta Mol. Cell. Res. 1823 674–82

[100] Matthews R and Li S N 2005 Structures and pathways for clathrin self-assembly in the bulk and on membranes Soft Matter 1 5794–806

[101] Mendes P 1993 Gepasi: a software package for modelling the dynamics, steady states and control of biochemical and other systems Comput. Appl. Biosci. 9 563–71

[102] Minton A P 2000 Implications of macromolecular crowding for protein assembly Curr. Opin. Struct. Biol. 10 34–9

[103] Misra N and Schwartz R 2008 Efficient stochastic sampling of first- passage times with applications to self-assembly simulations J. Chem. Phys. 129 204109

[104] Misra N, Lees D, Zhang T and Schwartz R 2008 Pathway complexes of model virus capsid assembly systems Comput. Math. Methods Med. 9 277–307

[105] Mondragón-Rodríguez S, Perry G, Luna-Muñoz J, Acevedo-Aquino M and Williams S 2014 Phosphorylation of tau protein at sites ser396–404 is one of the earliest events in alzheimer’s disease and down syndrome Neuropathol. Appl. Neurobiol. 40 121–35

[106] Munroe M I, Posner R G, Savage P B, Faeder J R and Hlavacek W S 2010 Modeling multivalent ligand-receptor interactions with steric constraints on configurations of cell-surface receptor aggregates Biophys. J. 98 48–56

[107] Mooren O L, Galletta B J and Cooper J A 2012 Roles for actin assembly in endocytosis Biochemistry 81 661

[108] Morozov A Y, Bruinsma R F and Rudnick J 2009 Assembly of viruses and the pseudo-law of mass action J. Chem. Phys. 131 155101

[109] Mulder A M, Yoshioka C, Beck A H, Bunner A E, Milligan R A, Potter C S, Carragher B and Williamson J R 2010 Visualizing ribosome biogenesis: parallel assembly pathways for the 30s subunit Science 330 673–7

[110] Müller-McNicoll M and Neugebauer K M 2013 How cells get the message: dynamic assembly and function of mrna-protein oligomers Mol. Cell. Biol. 14 384–96

[111] Murr E A and Buehler M J 2008 The interaction between surface roughness and processing for heterochromatin assembly and for telomere maintenance Cell 155 1061–74

[112] Nicolai D V and Burrage K 2008 Stochastic simulation of chemical reactions in spatially complex media Comput. Math. Appl. 55 1007–18
Phys. Biol. 14 (2017) 015033

(2017) 035003

1. Introduction

2. Conclusion

3. Methods

4. Results

5. Discussion

6. References

7. Acknowledgments

8. Supplementary Information

9. Author Information

10. References

11. Supporting Information

12. Supplementary Data

13. Appendix

14. Figure Legends

15. Table Legends

16. Footnotes

17. Notes

18. Abbreviations

19. Review

20. Conclusion

21. References

22. Supplementary Information

23. Author Information

24. References

25. Supporting Information

26. Supplementary Data

27. Appendix

28. Figure Legends

29. Table Legends

30. Footnotes

31. Notes

32. Abbreviations

33. Review

34. Conclusion

35. References

36. Supporting Information

37. Supplementary Data

38. Appendix

39. Figure Legends

40. Table Legends

41. Footnotes

42. Notes

43. Abbreviations

44. Review

45. Conclusion

46. References

47. Supporting Information

48. Supplementary Data

49. Appendix

50. Figure Legends

51. Table Legends

52. Footnotes

53. Notes

54. Abbreviations

55. Review

56. Conclusion

57. References

58. Supporting Information

59. Supplementary Data

60. Appendix

61. Figure Legends

62. Table Legends

63. Footnotes

64. Notes

65. Abbreviations

66. Review

67. Conclusion

68. References

69. Supporting Information

70. Supplementary Data

71. Appendix

72. Figure Legends

73. Table Legends

74. Footnotes

75. Notes

76. Abbreviations

77. Review

78. Conclusion

79. References

80. Supporting Information

81. Supplementary Data

82. Appendix

83. Figure Legends

84. Table Legends

85. Footnotes

86. Notes

87. Abbreviations

88. Review

89. Conclusion

90. References

91. Supporting Information

92. Supplementary Data

93. Appendix

94. Figure Legends

95. Table Legends

96. Footnotes

97. Notes

98. Abbreviations

99. Review

100. Conclusion

101. References

102. Supporting Information

103. Supplementary Data

104. Appendix

105. Figure Legends

106. Table Legends

107. Footnotes

108. Notes

109. Abbreviations

110. Review

111. Conclusion

112. References

113. Supporting Information

114. Supplementary Data

115. Appendix

116. Figure Legends

117. Table Legends

118. Footnotes

119. Notes

120. Abbreviations

121. Review

122. Conclusion

123. References

124. Supporting Information

125. Supplementary Data

126. Appendix

127. Figure Legends

128. Table Legends

129. Footnotes

130. Notes

131. Abbreviations

132. Review

133. Conclusion

134. References

135. Supporting Information

136. Supplementary Data

137. Appendix

138. Figure Legends

139. Table Legends

140. Footnotes

141. Notes

142. Abbreviations

143. Review

144. Conclusion

145. References

146. Supporting Information

147. Supplementary Data

148. Appendix

149. Figure Legends

150. Table Legends

151. Footnotes

152. Notes

153. Abbreviations

154. Review

155. Conclusion

156. References

157. Supporting Information

158. Supplementary Data

159. Appendix

160. Figure Legends

161. Table Legends

162. Footnotes

163. Notes

164. Abbreviations

165. Review

166. Conclusion

167. References

168. Supporting Information

169. Supplementary Data

170. Appendix

171. Figure Legends

172. Table Legends

173. Footnotes

174. Notes

175. Abbreviations

176. Review

177. Conclusion

178. References

179. Supporting Information

180. Supplementary Data

181. Appendix

182. Figure Legends

183. Table Legends

184. Footnotes

185. Notes

186. Abbreviations

187. Review

188. Conclusion

189. References

190. Supporting Information

191. Supplementary Data

192. Appendix

193. Figure Legends

194. Table Legends

195. Footnotes

196. Notes

197. Abbreviations

198. Review

199. Conclusion

200. References

201. Supporting Information

202. Supplementary Data

203. Appendix

204. Figure Legends

205. Table Legends

206. Footnotes

207. Notes

208. Abbreviations

209. Review

210. Conclusion

211. References

212. Supporting Information

213. Supplementary Data

214. Appendix

215. Figure Legends

216. Table Legends

217. Footnotes

218. Notes

219. Abbreviations

220. Review

221. Conclusion

222. References

223. Supporting Information

224. Supplementary Data

225. Appendix

226. Figure Legends

227. Table Legends

228. Footnotes

229. Notes

230. Abbreviations

231. Review

232. Conclusion

233. References

234. Supporting Information

235. Supplementary Data

236. Appendix

237. Figure Legends

238. Table Legends

239. Footnotes

240. Notes

241. Abbreviations

242. Review

243. Conclusion

244. References

245. Supporting Information

246. Supplementary Data

247. Appendix

248. Figure Legends

249. Table Legends

250. Footnotes

251. Notes

252. Abbreviations

253. Review

254. Conclusion

255. References

256. Supporting Information

257. Supplementary Data

258. Appendix

259. Figure Legends

260. Table Legends

261. Footnotes

262. Notes

263. Abbreviations

264. Review

265. Conclusion

266. References

267. Supporting Information

268. Supplementary Data

269. Appendix

270. Figure Legends

271. Table Legends

272. Footnotes

273. Notes

274. Abbreviations

275. Review

276. Conclusion

277. References

278. Supporting Information

279. Supplementary Data

280. Appendix

281. Figure Legends

282. Table Legends

283. Footnotes

284. Notes

285. Abbreviations
13