Transforming macromolecular structures into simulations of self-assembly

Yue Moon Ying,[1] Mankun Sang,[1] Gabriel Au, Smriti Chhibber, Yufeng Du, Jonathan A Fischer, Samuel L Foley, Sikao Guo, Ian Herzog-Pohl, Zixiu Liu, Hannah Roscom, Hassan Sohail, Sho S Takeshita, Margaret E Johnson*

[1]These authors contributed equally

TC Jenkins Department of Biophysics, Johns Hopkins University, Baltimore, MD, 21218, USA.

**Corresponding Author*: margaret.johnson@jhu.edu

Abstract

Macromolecular self-assembly is a fundamental step in numerous biological processes, producing molecular machines like the ribosome or highly symmetric viral capsids. Building models of these molecular-scale interactions that bridge to a cell scale environment requires substantial coarse-graining of assembly subunits or intermediates. Balancing multi-valent structural resolution with computational efficiency is challenging. While rule-based or local interactions overcome the often prohibitive enumeration of all possible intermediates, they must ensure global structural constraints are met. We here demonstrate ioNERDSS, a user-friendly Python package that transforms 3D atomic structures into coarse-grained models for immediate simulation with the stochastic reaction-diffusion NERDSS software. NERDSS uses rule-based interactions to produce structural trajectories of assembly dynamics with the microsecond-minutes timescales comparable to experiment. With ioNERDSS, rigid subunits for each protein chain in the assembly contain discrete binding interfaces and explicit geometric constraints to prevent disordered assemblies. Repeated subunits (as in viral capsids) are regularized to preserve the target assembly topology across stochastic association and dissociation events. The ioNERDSS python package links models and outputs to open-source visualization, simulation, and analysis tools that facilitate user-friendly sanity checks, structure validation, and analysis of output for thermodynamic, kinetic, and nonequilibrium properties of macromolecular self-assembly.

1. Introduction

From translation and vesicle trafficking to DNA transcription, living systems rely on multi-subunit macromolecular complexes to function. The generation of the ribosome (>60 subunits, all unique) or a clathrin cage (~200 subunits) requires many successful pairwise association events of monomers into units with correct stoichiometry in the nonequilibrium environment of the cell. While the biogenesis and assembly of ribosomes have been studied in great detail (1, 2), the assembly pathways used by the hundreds of other macromolecular complexes often lack systematic characterization. One factor limiting quantitative measurements of assembly pathways, kinetics, and control mechanisms is access to efficient and structurally accurate models that provide time-resolved simulations to help interpret experiments. The structures of the final assembled complexes are no longer typically unknown, thanks to the protein data bank (PDB) and the predictive powers of AlphaFold(3). Coarse-grained (CG) models offer a computationally tractable, structurally faithful approach to study mechanisms of macromolecular assembly, but it is often a laborious process to construct these models directly from the known 3D structures. To address this, we present ioNERDSS — a “structure-in, simulation-ready” Python package that automates coarse-graining from atomic or PDB-level structural data to rate-based models. The parameterized models and CG subunits are immediately executable with the NERDSS software for stochastic, rigid-body reaction-diffusion simulations(4). In doing so, ioNERDSS bridges the gap between detailed structural information and mesoscale assembly dynamics, making coarse-grained self-assembly tools accessible to scientists in cell biology, synthetic biology, materials design, and soft-matter physics. It supports exploratory modeling, classroom demonstration, and rigorous hypothesis testing without requiring extensive computational expertise.

Current modeling approaches to study pathways and mechanisms of macromolecular self-assembly fall broadly into two classes: rate-based methods (mass-action kinetics or reaction-diffusion) or energy-function based methods (Molecular Dynamics). The most common rate-based methods follow mass-action kinetics. Continuum non-spatial models via ordinary differential equations (ODEs) or reaction-diffusion models (PDEs) are relatively efficient, whereas stochastic non-spatial Gillespie (5, 6) or spatial Gillespie methods(7) capture the discrete nature of molecules. For self-assembly, these methods require enumeration of all subcomplexes or intermediates(8), whether in advance (9) (10, 11) or on-the-fly(12). Rule-based approaches(13-15) eliminate the combinatorial challenge of pre-computing all possible species, but they cannot eliminate the growth in these species during simulations. Further, without molecular structure, they cannot encode orientational and structural constraints on self-limiting

assemblies like virus capsids without explicit enumeration(10), or capture steric and volume excluding effects of multi-valent assemblies (16). This can lead to unphysical topologies or uncontrolled growth. Particle-based reaction diffusion offers the highest-resolution rate-based approach, where the inclusion of coarse molecular structure supports assembly of disordered structures(17) and polymers(18). For the NERDSS software(4), we enforce orientational constraints on assembly, making it uniquely designed to capture assembly of multi-component macromolecular complexes and structured self-limited assemblies like cages and filaments(9, 19-22). While this higher-resolution for particle-based RD comes at increased computational expense, these methods can still capture self-assembly processes over the seconds to minutes timescale.

MD simulations offer a complementary alternative for simulations of self-assembly, but the automated construction of simulation inputs primarily applies only to all-atom MD. At this atomic scale, structure-based simulation is highly automated: researchers can often go from a PDB structure to dynamic trajectories with minimal manual intervention with MD engines like GROMACS (23), NAMD (24), and LAMMPS (25) and toolkits outside MD such as PLUMED (26), CHARMM-GUI (27), and OpenMM (28), and structural prediction advances like AlphaFold (29) and RosettaFold (30). However, while these models are relatively straightforward to construct with parameters from existing force-fields, they are prohibitive to simulate over >ms timescales. Ultra-CG MD offers the only tractable MD tools to study pathways of self-assembly at longer timescales, but the construction of these models still requires substantial user expertise. Sophisticated force-matching techniques are not yet fully automated(31), and model construction for HOOMD or patchy-particle models is still user-specified(32). Nonetheless, the widespread adoption and utility of MD simulations to study molecular mechanisms at all CG levels motivated us to here develop tools to improve the generality and ease-of-use for RD simulations.

Our goal is to eliminate the previous requirement of careful, user-defined manual setup for particle-based RD simulations: translation from atomic structure to rigid-body representation, precise specification of binding interfaces (orientation, geometry), and detailed reaction rules (4). Slight inaccuracies in these specifications can accumulate during assembly, resulting in steric clashes or unphysical assemblies, thereby making the design of inputs the primary bottleneck in mesoscopic reaction-diffusion modeling. Rule-based modeling is a format for generating large systems via only pairwise rules, but it is typically restricted to nonspatial models, like ODEs of the Stochastic Simulation Algorithm (SSA) (33). NERDSS uses this but adds space and thus

structure is required — subunits cannot simply be spheres. Rigid-body subunits must be aligned properly as a result when bound. Another spatial rule-based model, SpringSaLaD (34), does not enforce orientational constraints, and thus does not naturally restrict the structure or the stoichiometry.

Further, NERDSS requires on and off rates to parameterize all pairwise reactions. Direct estimation of interaction free energies from structural data, when available, is an important step toward reliable modeling of biomolecular assemblies. Structure-based estimates — whether derived from physics-based calculations or learned scoring functions — can provide ΔG values that allow one to infer association and dissociation rates (e.g. by assuming a diffusion-limited on-rate). This in turn enables immediate generation of both deterministic (ODE) models and stochastic particle-based simulations (such as NERDSS). The approach thus provides a unified pipeline from structure to energetics and kinetics, and to assembly dynamics. Relying on structural data in this way leverages a growing body of work on binding-affinity prediction from 3D complexes. For example, computational scoring frameworks have long sought to predict protein–protein binding free energies from structural interfaces. Traditional computational prediction of protein–protein binding affinities surveys force-field and knowledge-based methods, including geometric hashing based docking,(35, 36) Fourier-Transform correlation based docking, (37-39) and template based prediction (40, 41). Recent advances in machine learning further strengthen promising binding-affinity prediction directly from 3D geometry, such as ProAffinity-GNN (42) and Boltz-2 (43) offering an new option for fast and accurate prediction of protein-protein affinity prediction.

In summary, ioNERDSS closes a critical gap between crystal structural data and mesoscale modeling of self-assembly. By automating the coarse-graining, parametrization, and simulation setup, it enables researchers to go from a PDB (or structural model) to physically plausible assembly simulations without needing deep expertise in simulation frameworks. By integrating with open-source visualization tools including Simularium (44) and Ovito (45), we support community use and development. ioNERDSS also supports export paths to visualization and modeling platforms: it integrates with Simularium (44) for interactive, shareable 3D visualization of simulation trajectories, and with Virtual Cell (VCell) (46) for reaction-diffusion or compartmental modeling workflows. Furthermore, ioNERDSS can automatically generate a mean-field baseline by converting the same reaction rules into deterministic ODE models as long as the complex has <12 subunits. These ODEs enumerate only geometrically admissible

intermediates, reproducing the steric constraints of the spatial model — though at the cost of combinatorial explosion when applied to very large lattices or shells. By offering both particle-based and mean-field outputs for the same structural specification, ioNERDSS enables users to compare spatial, stochastic behavior against averaged continuum dynamics under identical thermodynamic and kinetic assumptions. ioNERDSS is fully open source at github.com/JohnsonBiophysicsLab/ionerdss and there is also an openly available server for quick trials for the community at nerdssdemo.org.

2. Methods

ionerdss is a Python library that provides user-friendly tools for setting up and analyzing output from the NERDSS software. The package contains a comprehensive pipeline for processing Protein Data Bank (PDB) structures and converting them into coarse-grained molecular models ready for running NERDSS simulations, simple ODE solution to compare with NERDSS simulations, estimating reaction on- and off-rates with Protein-protein Affinity GNN, platonic solid models for users looking for an abstract idealized assembly, and a web server for new users to try out the pipeline.

2.1. Converting PDB structure to ready-to-run NERDSS model

The following section provides a quick overview of the pipeline. See SI section 2 for the detailed implementation.

2.1a. Coarse-graining the PDB structure

Atomic coordinates are read with the Biopython package. Only standard–amino-acid chains are kept. For each chain we compute (i) a geometric center of mass (COM; unweighted) and (ii) a coarse-grained radius (root-mean-square distance of atoms from the COM). To determine interfaces between molecules, putative chain–chain contacts are prefiltered by axis-aligned bounding boxes. Residue-level contacts are then found by KD-tree queries over Cα positions within a distance cutoff. A pair is accepted as an interface if both chains contribute at least a set number of contacting residues. Each side's interface point is the mean of its contacting Cα coordinates. This means that between any two molecules, there is either zero or one pair of interfaces. We record symmetric partner links for every accepted contact.

2.1b. Identifying and regularizing repeated chains

Because proteins are not rigid body, the exact structures imported from PDB can vary. However, we still expect one uniform structure for NERDSS simulation and therefore we need to identify and regularize repeated chains to one uniform geometry. First, Chains are partitioned into repeated groups using, polymer-entity IDs from the mmCIF header (if present, otherwise fall back to sequence alignment); we also offer a backup option of structure-based using Cα superposition.

Similarly, interfaces are categorized, using both sequence and structure signature. If the interface-to-COM distance and the angle between the interface-to-COM vector and the interface-to-interface vector are similar enough between two interfaces, then they pass the structure signature test. Then, the sequence test aligns the indexed sequence at interaction site and output a sequence similarity score. If two interfaces pass the structure signature test and have a sequence similarity score over a threshold then the two interfaces are considered the same type (homotypic). Note an interface can be homotypic to its binding partner, which we call homotypic interaction. This is important because symmetrical interaction must be treated differently in calculating stoichiometry

and thus reaction parameters.

We then create templates for molecule species and interaction types. The first chain in each group is the representative based on which we create a molecule template (COM, radius) and interface templates (offsets relative to the molecule COM). Contacts between the same template pair are de-duplicated by geometric signatures (COM–to–interface distances and angles), rounded to fixed precision to ensure deterministic reuse. After that, the representative geometry is pushed to all group members by a proper rigid transform. With ≥3 matched interfaces, we use Kabsch transformation with the standard determinant fix.

2.1c. Reaction enumeration and geometry calculation

From the templates we synthesize NERDSS molecule types and generate binding reactions. The process is straightforward, unless we have multiple homotypic interfaces on a molecule (see SI):

- **Homodimeric Homotypic case (A + A via symmetrical binding)**: If the interface is repeated on the molecule, we enumerate the unique combinations (i,i)…(i,j) with i ≤ j (total n(n+1)/2), where n is the number of equivalent sites.

  Cross-reactions (i ≠ j) can be assigned a distinct on-rate if desired. The on-rate is doubled to get the right kinetics.
- **Homodimeric heterotypic case** (**A + A via asymmetric binding):** same as below.
- **Heterodimeric case (A + B):** If the interface is repeated on one or both partners, we enumerate them all. Specifically, for *n* equivalent sites on A and *m* on B we enumerate all *n·m* combinations.

Even after previous regularization step, there might be cases where the symmetry is hard to solve and thus there are still discrepancy in bond lengths and angles among interactions of the same kind. However, we still need to decide on one set of bond lengths and angles for the parameter file. Here, we calculate a simple average for bond length and bond angles (angles between interface-to-COM vectors and interface-to-interface vectors) and a circular mean with special handling for high deviation for torsion angles. The exporter writes one molecule file (.mol) per molecule type and a parameter file (.inp) with molecules, reactions, bond lengths, association angles and other parameters, which are ready to run with the NERDSS software.

2.2. Predict binding affinity using ProAffinity-GNN

Our pipeline estimates binding free energies ΔG_{GNN} with the ProAffinity-GNN software (47) which was trained specifically for protein-protein interactions, not just protein-small molecules (Boltz-2(43)), and thus is geared towards our purpose. ProAffinity-GNN estimates interface binding affinities by combining structure aware graph attention neural networks with protein language model embeddings (42, 48). Here, ioNERDSS integrates a simple and user-friendly implementation of ProAffinity-GNN that does not require canonical FASTA sequences for each binding partner to align/renumber PDB chains. In our integration, sequences are reconstructed directly from the observed residues in the PDB (converted to PDBQT using AutoDockFR (49, 50)) files, and ESM-2 embeddings (51) are computed on these structure-derived sequences. As noted by the authors of ProAffinity-GNN, this inference-only path prioritizes ease of use at the cost of a modest reduction in accuracy relative to the full pipeline with FASTA sequences. Note that ioNERDSS does not accept user-supplied curated FASTA files and thus intentionally retains this simplified inference configuration (see SI section 2 for detailed implementation).

2.3. ODE solution for the assembly system

The ODE module provides a lightweight benchmark for NERDSS trajectories. We restrict the construction of the ODEs to systems that contain <12 subunits, due to the combinatorial growth in all possible intermediates (this can be over-ridden). Alternatively, rules can be provided to a rule-based simulator like NFsim(52), without structural constraints.

2.3a. Generating and solving ODEs

Given a reaction network (typically extracted from the coarse-grained graph) and rate constants, it integrates the deterministic kinetics $\frac{d\mathbf{y}}{dt} = \mathbf{S} \cdot \mathbf{r}(\mathbf{y})$, where $\mathbf{y}$ are species concentrations, $\mathbf{S}$ is the reaction stoichiometric matrix, and $\mathbf{r}(\mathbf{y})$ is the vector of reaction propensities. From the CG model, we know the monomer subunits (nodes) and their interactions with one another (edges). We then enumerate the intermediates that can be formed during the assembly (e.g. dimers, trimers, etc.).

2.3b. Defining on and off-rates for reaction for the ODEs

With no reliable tool (yet) to predict on and off-rates, on-rates for all reactions are assumed to be close to but below the diffusion-limit ($k_{\mathrm{on}} = 7.2 \times 10^7$ $M^{-1}s^{-1}$). For the ODE model, we assume that after a subunit associates, all the possible bonds form immediately to reduce the number of possible intermediates and avoid combinatorial explosion. The off-rates are calculated according to the equation below based on how many bonds must be dissociated and the binding affinity inferred from ProAffinity-GNN:

$$k_{\mathrm{off}}^{ij} = k_{\mathrm{on}}^{ij} c_0 \exp\left(\frac{\sum_{n=1}^{m} \Delta G_{\mathrm{GNN}}^{n}}{k_{\mathrm{B}} T}\right).$$

Importantly, ioNERDSS generates inputs for both spatial coarse-grained (CG) simulations and enumerates assembly pathways to write out integrable non-spatial ODE models for self-limiting assemblies. ioNERDSS handles repeated subunits both structurally (for NERDSS) and stoichiometrically (for the ODE models) which can require care(53, 54). The experimental dataset collected in Aim 1 will provide us with more meaningful estimates for k_{on}, which will make ioNERDSS even more powerful(38).

3. Results

3.1. Automated generation of simulation-ready NERDSS models from PDB structures

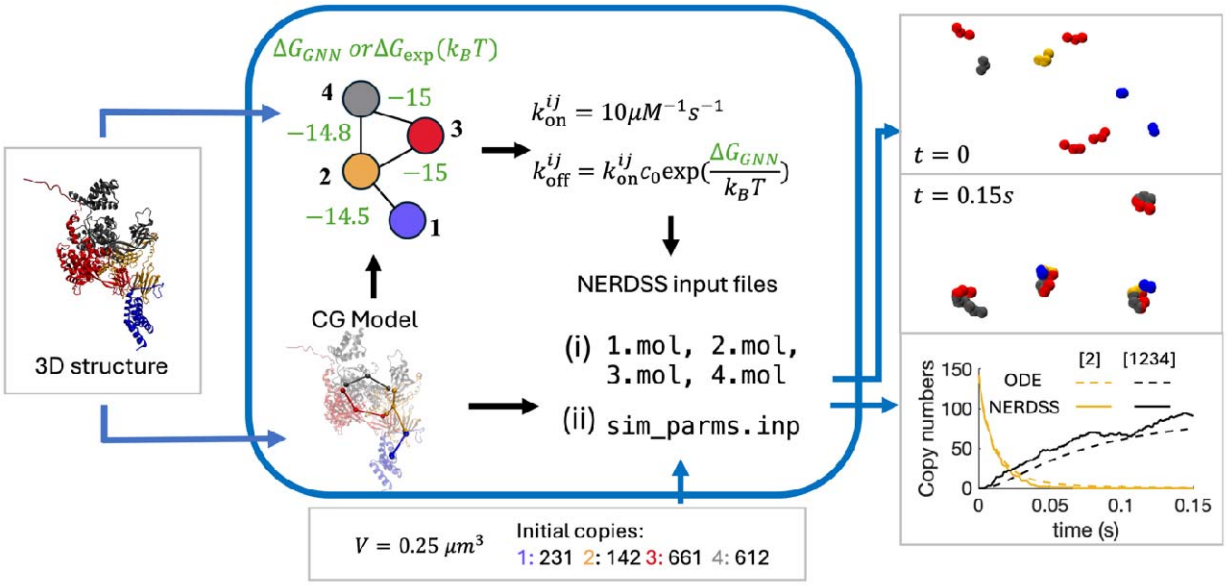

Figure 1. Conversion of Atomic structures into executable models for self-assembly, in NERDSS. The 4-subunit R2TP co-chaperone complex was used as an example from Complex Portal, where because the 3D structure has not been experimentally solved, we constructed it using AlphaFold3. The blue box contains steps performed by ioNERDSS given the structure input. The copy numbers/volume are optional; we otherwise use default values of 10 copies per species in a subvolume of 10^6 nm^3. and are here indexing the subunits. Copy numbers are from SGD, scaled down by 20-fold for the smaller subvolume compared to a yeast cell. Rates are for NERDSS simulations (for ODEs see Methods). is from ProAffinity. Outputs include the time-resolved copies of all intermediates and here we show a monomer and the 4-mer (bottom), and coordinates of all subunits in the volume (top).

In this study, we developed ioNERDSS, a user-friendly software tool which converts protein crystal structure files into fully parameterized NERDSS models that are immediately ready for simulation (Figure 1). Given only a PDB identifier or file, ioNERDSS executes an automated workflow that extracts molecular geometry, identifies

interaction interfaces and stoichiometry, estimates binding energetics, and produces all molecular and system files required for reaction diffusion assembly simulations.

Specifically, ioNERDSS can perform coarse-graining of the input structure, which yields a reduced but geometrically faithful representation of the molecular structure. This tool also infers sets of microscopic rate constants based on free-energy parameters for each detected interface using ProAffinity-GNN, ensuring that reversible interfaces satisfy energy conservation constraints across all pathways. Finally, ioNERDSS automatically generates NERDSS configuration files, including molecular coarse-grained structure files and a parameter file that specifies the interfaces, reaction networks, and global simulation parameters. The resulting files can be executed directly in the NERDSS software to simulate the assembly dynamics of the original PDB complex, from individual chains to high-order structures.

In addition, we constructed a web interface and host a server at nerdssdemo.org, which allows researchers to easily try out our tool via inputing a custom PDB ID and to see the NERDSS simulation result with the coarse-grained structure. Together, these capabilities enable researchers to convert seamlessly a structural data to dynamical simulations, removing the substantial manual effort previously required to construct parameterized NERDSS systems. For visualization, our outputs are compatible with OVITO and Simularium.

3.2. Constructing models from complexes with repeated subunits is idiosyncratic

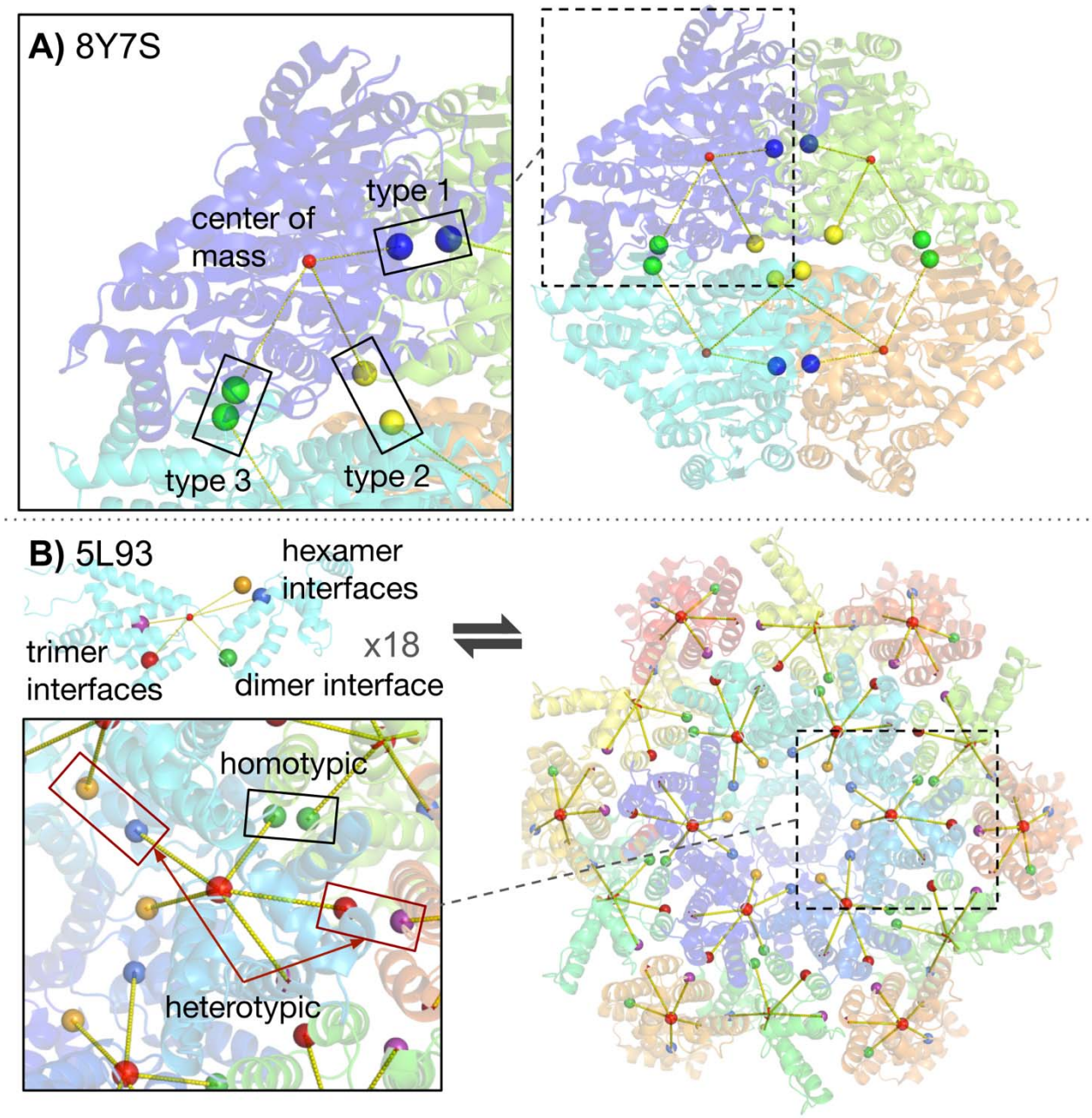

Figure 2: Technically difficult examples to coarse-grain with repeated subunits and variable stoichiometry and interface symmetries. A) Ribbon structure and ioNERDSS coarse-grained structure of a crystal structure of a benzaldehyde lyase mutant M6 from Herbiconiux sp. SALV-R1 (PDB 8Y7S). 8Y7S is an example of a homo-tetramer where each monomer binds with all three other monomers but each with a distinctive type of homotypic interface. B) Ribbon structure and ioNERDSS coarse-grained structure of an atomic model of

HIV-1 CA-SP1 (5L93). 5L93 is an example of a PDB that contains both homotypic and heterotypic interactions between the same type of monomers and that represents only a small part of a larger assembly so that not all the monomers are in the same chemical environment due to this truncation.

Accurately coarse-graining protein complexes that contain repeated subunits remains one of the most technically demanding steps in automated model construction. Despite employing the criteria specified in Methods section 2.1 to detect repeated chains and infer their corresponding reaction-interface networks, we still observe failure modes in which the inferred interfaces do not collectively generate the target higher-order structure. These issues arise most frequently partially symmetric assemblies, where variability in geometry and chemical environment across nominally identical subunits complicates the determination of a unified reaction network.

A major source of error comes from structural variations among subunits that in principle are equivalent. When repeated chains are in slightly different conformations, ioNERDSS may detect distinct sets of interfaces for different copies of the same molecule or interfaces, though these subunits or interactions are expected to share an identical reaction topology. One technically hard example is coarse-graining the structure of a crystal structure of a benzaldehyde lyase mutant M6 from Herbiconiux sp. SALV-R1 (PDB 8Y7S(55), Figure 2A), where four repeated subunits interact with each other via three different types of homotypic interactions. The pipeline must correctly determine that there is only one type of molecule, meaning that the complex is a homo-tetramer, but for each subunit, it interacts with three other subunit each with a different type of subunit and that all of these interactions are homotypic, which is challenging.

Simply averaging over orientations between pairs of interfaces of repeated subunits is typically not sufficient. Averaging over local geometries can produce global issues. Both molecule structures and orientations can vary due to thermal motion. Assembly locally can proceed acceptably, but because it can progress through different sequences of interfaces, small alignment issues can give rise to global variations; in the Gag lattice,

growth through dimers vs hexamers produces spheres with multiple curvatures, that thus do not ‘close’ properly. Correction requires first projecting subunits onto a single sphere. We can define a single subunit geometry via averaging over all subunits, but then these subunits must be tiled on the sphere prior to evaluating orientations between interface pairs.

Another technical difficulty is where the input PDB is a truncated part of a larger assembly. This results in subunits that are expected to be in the same chemical environment in different chemical environment in the provided PDB file. An example is the structure of an atomic model of HIV-1 CA-SP1(56) (PDB 5L93, Figure 2B). Although the Gag monomers in the file are in three different binding states, some of them are bound to 5 neighbors, some are bound to 4, and some are bound to 3, we expect the output NERDSS file to correctly assemble into a larger Gag lattice following the binding pattern specified. Another example is actin filament (Figure 4A-C) where the import PDB only contains an actin filament with 8 subunits but the NERDSS assembled structure grows to a length over 120 subunits. However, such assumption that distinct binding state is an artefact of the PDB file only contains part of the assembly is not always true. In certain complexes, the full complex actually contains the same subunit in two different binding states, such as human GATOR2 complex where the final complex contains both MIOS subunit that are bound to WDR59 and not WDR24, and reverse one. We therefore implement explicit user warnings when repeated subunits do not produce consistent interface sets and force same set of interfaces and corresponding geometry across equivalent subunits.

Additional minor difficulties can arise from overly permissive or restrictive hyperparameters, leading to the detection of spurious interfaces or inability to detect certain interface, and from nonstandard or inconsistently formatted PDB/CIF files, especially those containing nucleic acids, lipids, or large mixed macromolecules. These examples illustrate the nuanced structural factors that complicate automated coarse graining.

3.3. Comparison between NERDSS simulation and ODE result generated by ioNERDSS

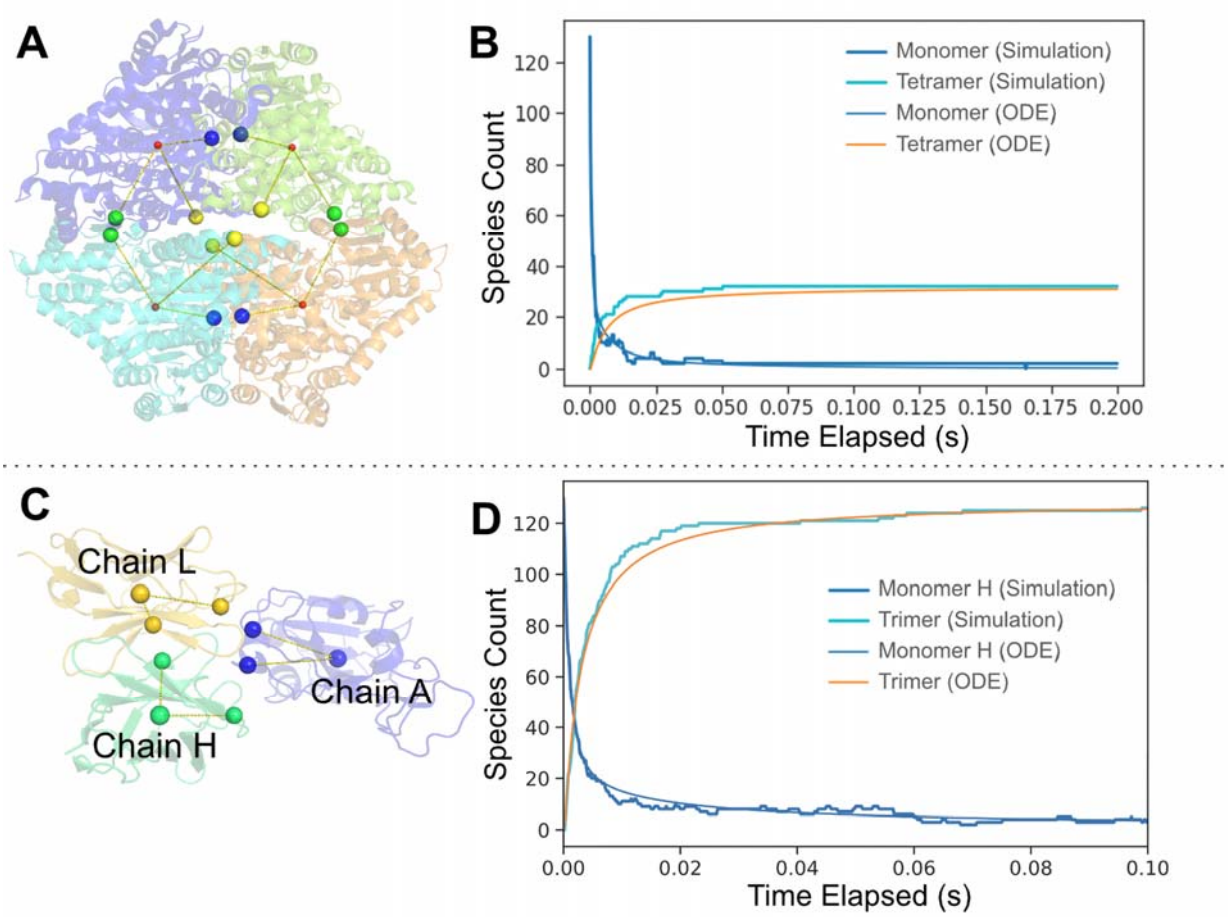

Figure 3. The NERDSS simulation and ODE results that are generated by the ioNERDSS pipeline on model systems (A, B: PDB 8Y7S; C, D: PDB 8ERQ) agree with each other. A) PDB and coarse-grained structure of a crystal structure of a benzaldehyde lyase mutant M6 from Herbiconiux sp. SALV-R1 (PDB 8Y7S); B) Monomer and tetramer count over time from NERDSS simulation and ODE results using ioNERDSS pipeline on self-assembly of 8Y7S; C) PDB and coarse-grained structure of SARS-CoV-2 BA.1 spike ectodomain trimer in complex with the S2X324 neutralizing antibody Fab fragment (PDB 8ERQ); D) Monomer chain H and assembled trimer count over time from NERDSS simulation and ODE result using ioNERDSS pipeline on self-assembly of 8Y7S. ODE solution were calculated assuming initial concentration of 0 for all species except for monomers whose initial concentration were set to a reduced concentration of 1. NERDSS simulations were run with a time step of 0.01 μs, a simulation box

of 600 nm × 600 nm × 600 nm, and an initial species count of 130 for each monomer speices. The reduced concentrations from ODE solution were then scale to the species count from the simulation results.

We compared stochastic NERDSS trajectories with deterministic ordinary differential equation (ODE) solutions generated automatically from the same coarse-grained reaction network. Two representative systems were tested to evaluate whether the pipeline produces consistent dynamical behavior in the ODE model and the reaction diffusion model across distinct structural and stoichiometric regimes (Figure 3).

For the aforementioned model (Fig. 2A) of benzaldehyde lyase mutant M6 (PDB 8Y7S), ioNERDSS uses the automatically extracted reaction schema and rate parameters and computses the corresponding ODE solution under well-mixed assumptions, initializing all species at zero concentration except the monomer, which was assigned a reduced concentration of 1. NERDSS simulations were run in a cubic water box with side length of 600 nm, time step of 0.01 μs, and 130 initial monomer count. The ODE concentrations were then rescaled to match the discrete particle counts used in the simulation. As shown in Figure 3B, the NERDSS and ODE results exhibit decent agreement, confirming that both stochastic and deterministic formulations capture the underlying reaction kinetics.

We next tested an asymmetric system: the SARS-CoV-2 BA.1 spike ectodomain trimer bound to the S2X324 neutralizing antibody Fab fragment(57) (PDB 8ERQ) (Figure 3C), which is a simple assembly of three different monomer species into a heterotrimer. ioNERDSS generated a self-consistent reaction network that accurately reproduced trimer formation dynamics. Figure 3D shows time evolution agress of the monomeric heavy-chain species (chain H) and the assembled trimer population in NERDSS simulations and ODE solutions. These results demonstrate that ioNERDSS produces reaction networks whose deterministic ODE behavior is quantitatively consistent with stochastic 3D NERDSS simulations across both symmetric and asymmetric assemblies.

3.4. ioNERDSS in polymerizing and ideal polyhedral assemblies

ioNERDSS can handle assembly processes across a wide range of geometric and topological regimes, from unbounded polymers that we mentioned as a challenge previously in section 3.2 to ideal polyhedral complexes. To illustrate this versatility, we applied the workflow to two structurally distinct examples: actin filament polymerization and the formation of an idealized dodecahedral cage (Figure 4).

Actin filaments represent the class of open-ended, anisotropic polymers whose growth has no inherent geometric termination point. Using the high-resolution structure of actin monomers(58) (PDB 6BNO), ioNERDSS identified four distinct interfaces per monomer (Figure 4A), which together support the characteristic helical arrangement of F-actin. Simulated assembly trajectories with NERDSS reproduced the expected nucleation mechanism: an initial nucleus forms from a small cluster of monomers (Figure 4B) to a long polymer (Figure 4C). Importantly, the model does not impose a predetermined filament length strictly according to the provided PDB, demonstrating that the same coarse-graining and connectivity principles extend to systems with much longer growth than the provided PDB file.

In contrast, the dodecahedral cage exemplifies an ideal polyhedral assembly with a fixed target geometry (Figure 4D). Here, for ideal polyhedral homo-n-mer assembly, ioNERDSS does not require the user the input any PDB file or ID, instead ioNERDSS have pre-defined customizable ideal model for each polyhedral case. Together, these two contrasting examples show that ioNERDSS works across fundamentally different assembly regimes, accurately capturing distinctive architectures within a unified framework.

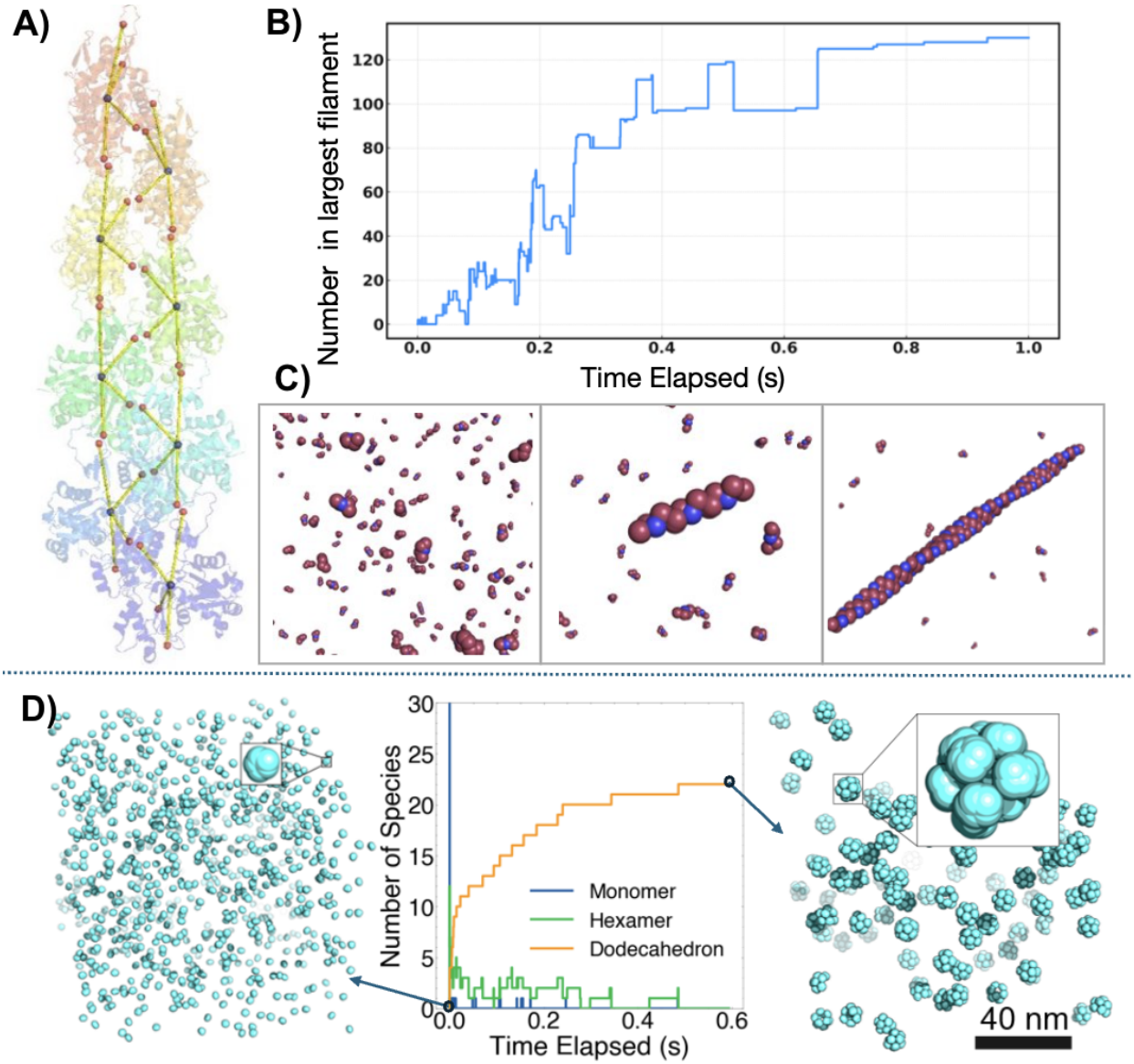

Figure 4. ioNERDSS resolves monomer structures that can form both polymers and fixed-geometry complexes. A) Actins (6BNO) assemble into filaments without a fixed geometry. The ioNERDSS model of actin monomers have four interfaces allowing them to assemble a helical filament. B) NERDSS simulated the assembly process an actin polymer showing monomers C) nucleates and the seed grows to form a long filament. D) Complexes like dodecahedrons have a fixed geometry and the regularization function of ioNERDSS ensures that the monomer structure is correct. The plot shows the dynamics of monomers assembling into dodecamers. Hexamer is shown as an example of intermediate in this assembly process. See SI Methods section 2 for simulation details.

4. Discussion

Our results demonstrate that ioNERDSS successfully converts high-resolution structural data into a mesoscale self-assembly model. The pipeline transforms PDB-derived molecular structures into coarse-grained reaction networks, which then yield consistent behavior when simulated as both stochastic particle-based assemblies and deterministic ODE systems. In test cases (e.g., PDB 8Y7S and 8ERQ), we observed good agreement between NERDSS simulation trajectories and ioNERDSS-generated ODE predictions — validating that the coarse-grained geometries, binding interfaces, and inferred kinetics. The bottleneck the mesoscale has been in translating atomic structures into rigid-body coarse-grained models with accurate interface geometry and appropriate reaction rules — a process that until now required substantial manual intervention. By automating that translation, ioNERDSS lowers the barrier to entry for mesoscale modeling, opening the door for use by cell biologists, materials scientists, and synthetic-biology researchers without deep computational expertise.

Moreover, by incorporating structure-derived binding free energies into a reaction-network formalism — converting them into on and off rates — ioNERDSS makes it possible to predict global assembly behavior from local interactions. This unifies structure, binding affinity prediction, kinetics, and mesoscale dynamics in a single pipeline. Because of this integration, one can now directly test whether predicted pairwise affinities and binding geometries suffice to drive the formation of the intended higher-order assembly, bridging a key conceptual gap in mesoscale modeling.

Beyond biological complexes, these features place ioNERDSS in the emerging paradigm of programmable, addressable self-assembly. In particular, recent work on semi-addressable and multifarious self-assembly demonstrates that by carefully designing binding interactions (and not simply maximizing species diversity), one can

optimize both assembly yield and efficiency while suppressing off-pathway structures (59, 60). Our pipeline mirrors the same statement that structural data define the geometry and binding specificity, while the reaction network enforces the thermodynamic and kinetic constraints, enabling robust assembly of the target complex — or revealing intrinsic kinetic/thermodynamic barriers that must be overcome.

However, important limitations remain. First, we do not yet handle non-protein components, so the ribosome is not correctly handled—it leaves out the nucleic acids. We also currently have an upper limit on the number of edges, beyond which it does not write the ODE model. NERDSS models are still generated, and it can be converted to a nonspatial model using BioNetGen for simulations using NFSim, for example. Also the web version is more limited than the command line one. The ODE model is more limited because NERDSS does not require full enumeration of all substructures, whereas the ODE does. We enforce monomer growth, this could be ameliorated by switching to rule-based solvers like NFSim. such as that ioNERDSS cannot currently capture conformational flexibility, inducible fit, or large-scale rearrangements, as because models are built using rigid body. This restricts applicability to systems whose assembly does not rely on dramatic conformational remodeling. Some structures, including the clathrin cage and the mature HIV lattice, are assembled with varying bending angles between otherwise identical subunits. While ioNERDSS can identify and define monomer subunits for the mature HIV lattice, they will not assemble the correct final structure observed in the PDB structure due to the rigid body assumptions of a single set of bond orientations. Clathrin trimers have a further distinction, what we think of as the assembly subunits, the clathrin trimers, are actually 6-mers themselves. ioNERDSS would therefore identify light chains and heavy chains only as the assembly subunits, not trimers. A future direction would be to allow users to merge chains to create larger subunits for self-assembly. Similarly, structures stabilized by membrane mechanics, filament rigidity, or weak transient interactions — such as disordered-protein aggregates, or liquid–liquid phase separated condensates — may lie beyond its current design scope. Finally, while ODE-based mean-field models derived from the reaction network

provide a useful baseline, enumeration of all intermediates becomes prohibitively expensive for large lattices or shells, limiting scalability.

Despite these limitations, ioNERDSS provides a valuable module in a larger multiscale modeling scheme. By combining structural realism, kinetic modeling, and mesoscale assembly simulation, it helps researchers interpret large-scale spatial and temporal biological datasets in mechanistic terms — extending the vision of computational cell biology beyond molecular dynamics. As structural databases continue to grow rapidly (especially through cryo-EM), and as binding-affinity prediction improves, ioNERDSS provides a practical, reproducible, and transparent framework for hypothesis testing, model sharing, and community-wide adoption.

In conclusion, ioNERDSS addresses a foundational barrier in mesoscale modeling by delivering a “structure-in simulation-ready” workflow. Its fully automated coarse-graining, interface detection, and kinetics assignment makes self-assembly simulation accessible, even to non-experts. By integrating with visualization tools and offering both stochastic and deterministic output formats, it supports rigorous, reproducible studies and broad dissemination. In doing so, ioNERDSS helps investigation and simulation of the addressable self-assembly for biological systems, synthetic biology, and materials design — enabling researchers to build, share, and test assembly models grounded in atomic structure, physical constraints, and realistic kinetics.

ACKNOWLEDGMENTS: MEJ gratefully acknowledges funding support from an NIH R35GM133644 and NIAID R01AI186663.

1. S. Klinge, J. L. Woolford, Jr., Ribosome assembly coming into focus. *Nat Rev Mol Cell Biol* **20**, 116-131 (2019).

2. A. Vanden Broeck, S. Klinge, Eukaryotic Ribosome Assembly. *Annu Rev Biochem* **93**, 189-210 (2024).
3. J. Abramson *et al.*, Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature* **630**, 493-500 (2024).
4. M. J. Varga, Y. Fu, S. Loggia, O. N. Yogurtcu, M. E. Johnson, NERDSS: A Nonequilibrium Simulator for Multibody Self-Assembly at the Cellular Scale. *Biophysical Journal* **118**, 3026-3040 (2020).
5. D. T. Gillespie, General Method for Numerically Simulating Stochastic Time Evolution of Coupled Chemical-Reactions. *Journal of Computational Physics* **22**, 403-434 (1976).
6. D. T. Gillespie, Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* **81**, 2340-2361 (1977).
7. E. Roberts, J. E. Stone, Z. Luthey-Schulten, Lattice Microbes: High-Performance Stochastic Simulation Method for the Reaction-Diffusion Master Equation. *Journal of Computational Chemistry* **34**, 245-255 (2013).
8. T. M. Earnest, K. Chen, J. Lai, Z. Luthey-Schulten, Towards a Whole-Cell Model of Ribosome Biogenesis: Kinetic Modeling of SSU Assembly. *Biophysical Journal* **108**, 390a-390a (2015).
9. A. Jhaveri, S. Loggia, Y. Qian, M. E. Johnson, Discovering optimal kinetic pathways for self-assembly using automatic differentiation. *Proc Natl Acad Sci U S A* **121**, e2403384121 (2024).
10. M. Thomas, R. Schwartz, Quantitative computational models of molecular self-assembly in systems biology. *Phys Biol* **14**, 035003 (2017).
11. M. Feig, Y. Sugita, Whole-Cell Models and Simulations in Molecular Detail. *Annual Review of Cell and Developmental Biology* **35**, 191-211 (2019).
12. M. W. Sneddon, J. R. Faeder, T. Emonet, Efficient modeling, simulation and coarse-graining of biological complexity with NFsim. *Nature Methods* **8**, 177-U112 (2011).
13. M. L. Blinov, J. R. Faeder, B. Goldstein, W. S. Hlavacek, BioNetGen: software for rule-based modeling of signal transduction based on the interactions of molecular domains. *Bioinformatics* **20**, 3289-3291 (2004).
14. A. Sorokin, K. F. Heil, J. D. Armstrong, O. Sorokina, Rule-based modelling provides an extendable framework for comparing candidate mechanisms underpinning clathrin polymerisation. *Sci Rep* **8**, 5658 (2018).
15. L. A. Harris *et al.*, BioNetGen 2.2: advances in rule-based modeling. *Bioinformatics* **32**, 3366-3368 (2016).
16. D. O. Holland, M. E. Johnson, Stoichiometric balance of protein copy numbers is measurable and functionally significant in a protein-protein interaction network for yeast endocytosis. *PLoS Comput. Biol.* **14**, e1006022 (2018).
17. P. J. Michalski, L. M. Loew, SpringSaLaD: A Spatial, Particle-Based Biochemical Simulation Platform with Excluded Volume. *Biophys J* **110**, 523-529 (2016).
18. M. Hoffmann, C. Frohner, F. Noe, ReaDDy 2: Fast and flexible software framework for interacting-particle reaction dynamics. *PLoS Comput Biol* **15**, e1006830 (2019).
19. S.-K. Guo, A. J. Sodt, M. E. Johnson, Large self-assembled clathrin lattices spontaneously disassemble without sufficient adaptor proteins. *PLOS Computational Biology* **18**, e1009969 (2022).
20. Y. Qian *et al.*, Temporal control by cofactors prevents kinetic trapping in retroviral Gag lattice assembly. *Biophys J* **122**, 3173-3190 (2023).

21. S. Guo, I. Saha, S. Saffarian, M. E. Johnson, Structure of the HIV immature lattice allows for essential lattice remodeling within budded virions. *eLife* **12**, e84881 (2023).
22. S. Guo *et al.*, Parallelization of Particle-Based Reaction-Diffusion Simulations Using MPI. *J Comput Chem* **46**, e70132 (2025).
23. M. J. Abraham *et al.*, GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX* **1-2**, 19-25 (2015).
24. J. C. Phillips *et al.*, Scalable molecular dynamics on CPU and GPU architectures with NAMD. *J Chem Phys* **153**, 044130 (2020).
25. A. P. Thompson *et al.*, LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Physics Communications* **271**, 108171 (2022).
26. The PLUMED cosortium, Promoting transparency and reproducibility in enhanced molecular simulations. *Nature Methods* **16**, 670-673 (2019).
27. S. Jo, T. Kim, V. G. Iyer, W. Im, CHARMM-GUI: A web-based graphical user interface for CHARMM. *Journal of Computational Chemistry* **29**, 1859-1865 (2008).
28. P. Eastman *et al.*, OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLoS Comput Biol* **13**, e1005659 (2017).
29. J. Jumper *et al.*, Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583-589 (2021).
30. M. Baek *et al.*, Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **373**, 871-876 (2021).
31. Y. Peng *et al.*, OpenMSCG: A Software Tool for Bottom-Up Coarse-Graining. *J Phys Chem B* **127**, 8537-8550 (2023).
32. J. A. Anderson, J. Glaser, S. C. Glotzer, HOOMD-blue: A Python package for high-performance molecular dynamics and hard particle Monte Carlo simulations. *Computational Materials Science* **173**, 109363 (2020).
33. D. T. Gillespie, Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry* **81**, 2340-2361 (1977).
34. P. J. Michalski, L. M. Loew, SpringSaLaD: A Spatial, Particle-Based Biochemical Simulation Platform with Excluded Volume. *Biophys J* **110**, 523-529 (2016).
35. E. Mashiach *et al.*, An integrated suite of fast docking algorithms. *Proteins: Structure, Function, and Bioinformatics* **78**, 3197-3204 (2010).
36. D. Schneidman-Duhovny, Y. Inbar, R. Nussinov, H. J. Wolfson, PatchDock and SymmDock: servers for rigid and symmetric docking. *Nucleic Acids Research* **33**, W363-W367 (2005).
37. M. Zacharias, Accounting for conformational changes during protein–protein docking. *Current Opinion in Structural Biology* **20**, 180-186 (2010).
38. S. Vajda, D. Kozakov, Convergence and combination of methods in protein–protein docking. *Current Opinion in Structural Biology* **19**, 164-170 (2009).
39. E. Katchalski-Katzir *et al.*, Molecular surface recognition: determination of geometric fit between proteins and their ligands by correlation techniques. *Proceedings of the National Academy of Sciences* **89**, 2195-2199 (1992).
40. P. J. Kundrotas, Z. Zhu, J. Janin, I. A. Vakser, Templates are available to model nearly all complexes of structurally characterized proteins. *Proceedings of the National Academy of Sciences* **109**, 9438-9441 (2012).

41. P. J. Kundrotas *et al.*, Dockground: A comprehensive data resource for modeling of protein complexes. *Protein Science* **27**, 172-181 (2018).
42. Z. Zhou *et al.*, ProAffinity-GNN: A Novel Approach to Structure-Based Protein–Protein Binding Affinity Prediction via a Curated Data Set and Graph Neural Networks. *Journal of Chemical Information and Modeling* **64**, 8796-8808 (2024).
43. S. Passaro *et al.*, Boltz-2: Towards Accurate and Efficient Binding Affinity Prediction. *bioRxiv* 10.1101/2025.06.14.659707 (2025).
44. B. Lyons *et al.*, The Simularium Viewer: an interactive online tool for sharing spatiotemporal biological models. *Nat Methods* **19**, 513-515 (2022).
45. A. Stukowski, Visualization and analysis of atomistic simulation data with OVITO-the Open Visualization Tool. *Modelling Simul. Mater. Sci. Eng.* **18** (2010).
46. Moraru, II *et al.*, Virtual Cell modelling and simulation software environment. *IET Syst Biol* **2**, 352-362 (2008).
47. Z. Zhou *et al.*, ProAffinity-GNN: A Novel Approach to Structure-Based Protein-Protein Binding Affinity Prediction via a Curated Data Set and Graph Neural Networks. *J Chem Inf Model* **64**, 8796-8808 (2024).
48. P. A. Ravindranath, S. Forli, D. S. Goodsell, A. J. Olson, M. F. Sanner, AutoDockFR: Advances in Protein-Ligand Docking with Explicitly Specified Binding Site Flexibility. *PLOS Computational Biology* **11**, e1004586 (2015).
49. P. A. Ravindranath, S. Forli, D. S. Goodsell, A. J. Olson, M. F. Sanner, AutoDockFR: Advances in Protein-Ligand Docking with Explicitly Specified Binding Site Flexibility. *PLoS Comput Biol* **11**, e1004586 (2015).
50. Y. Zhao, D. Stoffler, M. Sanner, Hierarchical and multi-resolution representation of protein flexibility. *Bioinformatics* **22**, 2768-2774 (2006).
51. Z. Lin *et al.*, Language models of protein sequences at the scale of evolution enable accurate structure prediction. *BioRxiv* **2022**, 500902 (2022).
52. M. W. Sneddon, J. R. Faeder, T. Emonet, Efficient modeling, simulation and coarse-graining of biological complexity with NFsim. *Nat Methods* **8**, 177-183 (2011).
53. A. I. Curatolo, O. Kimchi, C. P. Goodrich, R. K. Krueger, M. P. Brenner, A computational toolbox for the assembly yield of complex and heterogeneous structures. *Nat Commun* **14**, 8328 (2023).
54. M. C. Hubl, C. P. Goodrich, Accessing Semiaddressable Self-Assembly with Efficient Structure Enumeration. *Phys Rev Lett* **134**, 058204 (2025).
55. Y. Zhang *et al.*, Manipulating Activity and Chemoselectivity of a Benzaldehyde Lyase for Efficient Synthesis of α-Hydroxymethyl Ketones and One-Pot Enantio-Complementary Conversion to 1,2-Diols. *ACS Catalysis* **14**, 9687-9700 (2024).
56. F. K. M. Schur *et al.*, An atomic model of HIV-1 capsid-SP1 reveals structures regulating assembly and maturation. *Science* **353**, 506-508 (2016).
57. Y.-J. Park *et al.*, Imprinted antibody responses against SARS-CoV-2 Omicron sublineages. *Science* **378**, 619-627 (2022).
58. P. S. Gurel *et al.*, Cryo-EM structures reveal specialization at the myosin VI-actin interface and a mechanism of force sensitivity. *eLife* **6**, e31125 (2017).
59. C. P. Goodrich, E. M. King, S. S. Schoenholz, E. D. Cubuk, M. P. Brenner, Designing self-assembling kinetics with differentiable statistical physics models. *Proc Natl Acad Sci U S A* **118** (2021).

60. M. C. Hübl, C. P. Goodrich, Accessing Semiaddressable Self-Assembly with Efficient Structure Enumeration. *Physical Review Letters* **134**, 058204 (2025).