Biophysical Journal, Volume *118*

Supplemental Information

NERDSS: A Nonequilibrium Simulator for Multibody Self-Assembly at the Cellular Scale

Matthew J. Varga, Yiben Fu, Spencer Loggia, Osman N. Yogurtcu, and Margaret E. Johnson

Supporting Information
for

NERDSS: a non-equilibrium simulator for multibody self-assembly at the cellular scale

Matthew J. Varga[1], Spencer Loggia[1], Yiben Fu[1], Osman N Yogurtcu[2], and Margaret E. Johnson[1,*]

Supporting Methods

I. Loop closure probability derivation:

Consider a complex A—B, where both protein A and protein B have sites free to bind C. Protein C has two sites that can bind each to A and B, thus being able to form a closed loop. We can write the single step reaction as:

$$A(a).B(b) + C(a,) \rightleftharpoons A(a!1).B(b!2).C(a!1,b!2) \quad \text{Eq. S1}$$

If we treat the association of C to A—B as a single binding event, we can define the K_D due to binding free energy gained by both CA and CB forming, plus an additional free energy gain or loss due to the specifics of the molecules themselves:

$$K_D^{1step} = C_0 \exp\left(\frac{[\Delta G_{CA}+\Delta G_{CB}+\Delta G_{coop}]}{k_B T}\right) = \frac{A(a).B(b)_{eq} * C(a,b)_{eq}}{A(a!1).B(b!2).C(a!1,b!2)_{eq}}, \quad \text{Eq. S2}$$

where C_0 is the standard state concentration of 1M, k_B is Boltzmann's constant, T is the temperature, and ΔG_{CA} is the binding free energy difference of bound minus unbound CA, with the same definition for CB. The free energy does not have to be strictly the sum of the two contributions, hence the addition of the parameter ΔG_{coop}.

In practice, treating the loop closure as a single binding step makes it very difficult and time-consuming to properly evaluate binding probabilities and unbinding probabilities on-the-fly. The rate k_a would change if it were a loop closure relative to a standard binding event. Loop closure events are spatial and not known in advance, so one would have to check for each possible reaction evaluation if the two proteins would spatially orient in a way to allow loop closure. For unbinding, to preserve detailed balance, both links would have to be broken at once, whether A, B, or C unbound.

Because of the algorithmic complexity of single-step closure, we will instead preserve the same free energy difference between the open and closed loop conformations as would occur in a single step closure, by breaking the binding into two steps, a standard bimolecular association with unchanged rates, and a unimolecular loop closure. In BNGL syntax, this looks like:

$$A(a).B(b) + C(a,b) \rightleftharpoons A(a!1).B(b).C(a!1,b) \quad \text{Eq. S3a}$$

$$A(a!1).B(b).C(a!1,b) \rightleftharpoons A(a!1).B(b!2).C(a!1,b!2) \quad \text{Eq. S3b}$$

or

$A(a).B(b) + C(a,b) \rightleftharpoons A(a).B(b!2).C(a,b!2)$ Eq. S4a

$A(a).B(b!2).C(a,b!2) \rightleftharpoons A(a!1).B(b!2).C(a!1,b!2)$ Eq. S4b

For an individual bimolecular association between C and A, we have a standard definition:

$K_D^{CA} = C_0 \exp(\frac{[\Delta G_{CA}]}{k_B T}) = \frac{k_b}{k_f^{CA}}$ Eq. S5

and the same for CB, where k_f and k_b are the binding and unbinding rates for the bimolecular association reaction. If we compare Eq. S5 with Eq. S2, we can state that:

$K_D^{1step} = \frac{k_b k_b \exp(\Delta G_{coop}/k_B T)}{k_f^{CA} k_f^{CB} C_0}$. Eq. S6

We want from our two-step model that:

$K_D^{2step} = \frac{A(a).B(b)_{eq} * C(a,b)_{eq}}{A(a!1).B(b!2).C(a!1,b!2)_{eq}}$. Eq. S7

We assume the first step in the two-step model, the bimolecular association event, is fixed and unchanged, given by:

$\frac{A(a).B(b)_{eq} * C(a,b)_{eq}}{A(a!1).B(b).C(a!1,b)_{eq}} = \frac{k_b}{k_f^{CA}}$. Eq. S8

To recover the desired result of Eq. S7 equivalent to Eq. S6, we must define the second step, the unimolecular step, with equilibrium:

$\frac{A(a!1).B(b).C(a!1,b)_{eq}}{A(a!1).B(b!2).C(a!1,b!2)_{eq}} = \frac{k_b \exp(\Delta G_{coop}/k_B T)}{k_f^{CB} C_0}$. Eq. S9

The forward rate for the unimolecular reaction thus becomes $k_{close} = k_f^{CB} C_0 \exp(-\Delta G_{coop}/k_B T)$. The reaction probability is then evaluated per time-step as the other unimolecular reactions, based on a Poisson process. The same result occurs if CB forms first, followed by CA. Both the on- and off-rates can also be re-scaled by the same constant, since we constrain here only the equilibrium. By default, the scale factor is 1 and k_b is unchanged. We note that in the two-step process, the total A is partitioned amongst four species, A(a).B(b), A(a!1).B(b).C(a!1,b), A(a).B(b!2).C(a,b!2), and A(a!1).B(b!2).C(a!1,b!2), whereas in the one-step process it is only two species (first and last in the list). Hence the equilibrium populations of these species will be slightly different in both models, although the free energy difference between the monomers and closed loops is the same.

In practice, the loop closing frequency using this definition with ΔG_{coop}=0 is much higher than the opening frequency, resulting in loops that rarely disassemble, because the K_D from Eq. S2 is much stronger than the individual binding events. Positive values of ΔG_{coop} thus destabilize the loops, making the open rate more competitive with the close rate, and producing more dynamic and reversible loops. This additional parameter can be specified per reaction pair as the scale factor (loopCoopFactor):

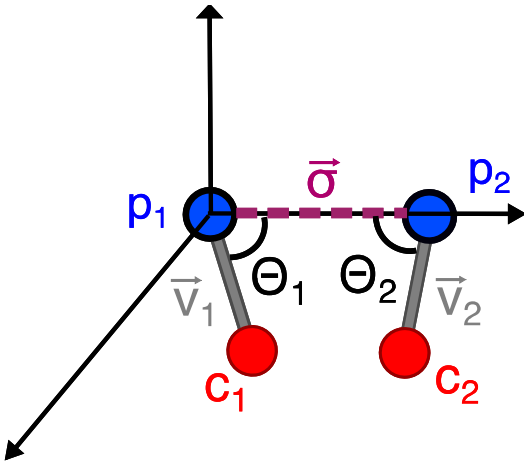
$f = \exp(-\Delta G_{coop}/k_B T)$ Eq. S10

It will be applied to the loop closure rate if those proteins can ultimately form a loop.

II. Orientations of proteins in a bound complex: Angle definitions

Once two molecules have bound to one another, their relative orientations can be specified to 'snap' them into place. This is specified in the model input files via 5 angles: θ_1, θ_2, φ_1, φ_2, and ω, which can be designed using our GUI. Each molecule has two angles defined relative to the binding radius vector (θ_1, φ_1), and ω then defines the orientation of molecule 1 relative to molecule 2. Point particles do not have any orientations to specify, and linear molecules do not have φ values.

When molecules are rotated into position, molecule *i* displaces according to $D_{R,i}/(D_{R,1}+D_{R,2})$. If both molecules have D_R=0, then they displace according to relative D_t values.



Theta Angles: The first two angles, θ_1 and θ_2, determine the orientation [0:π] between the binding radius vector $\vec{\sigma}$ and the site-to-COM vectors

$\overrightarrow{v_1} = p_1 - c_1$ Eq. S11a

$\overrightarrow{v_2} = p_2 - c_2$ Eq. S11b

connecting each molecule center-of-mass (c_1,c_2) to its participating binding site (p_1,p_2).

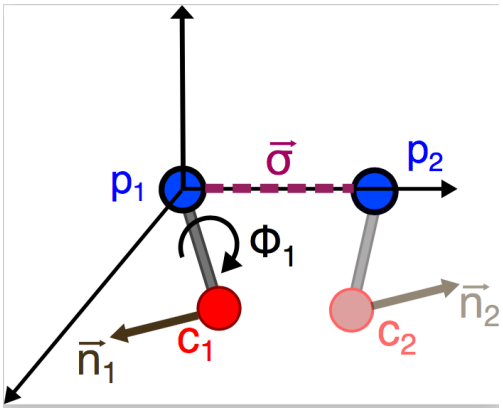
$\theta_1 = \text{acos}\left(\frac{\overrightarrow{v_1}\cdot\overrightarrow{\sigma_1}}{|\overrightarrow{v_1}||\overrightarrow{\sigma_1}|}\right)$ Eq. S12a

and

$\theta_2 = \text{acos}\left(\frac{\overrightarrow{v_2}\cdot\overrightarrow{\sigma_2}}{|\overrightarrow{v_2}||\overrightarrow{\sigma_2}|}\right)$. Eq. S12b

We define the sign of the vector $\vec{\sigma}$ relative to the molecule being moved, such that

$\overrightarrow{\sigma_1} = p_1 - p_2 = -\overrightarrow{\sigma_2}$. Eq. S13



Phi angle: The angles φ_1 and φ_2 are dihedral angles that orient a second axis of each molecule $\overrightarrow{n_1}$ and $\overrightarrow{n_2}$ (not co-linear with the site-to-COM vectors) relative to the binding radius vector $\vec{\sigma}$.

We have that

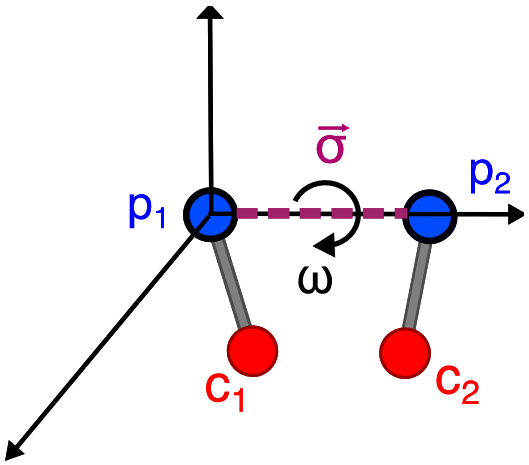
$\varphi_1 = \text{acos}(\hat{t_1} \cdot \hat{t_2})$, Eq. S14a

here $\hat{t_1}$ and $\hat{t_2}$ are the unit normals defined by

$\overrightarrow{t_1} = \overrightarrow{v_1} \times \overrightarrow{\sigma_1}$ Eq. S14b

$\overrightarrow{t_2} = \overrightarrow{v_1} \times \overrightarrow{n_1}$, Eq. S14c

with $\overrightarrow{v_1}$ defined above. The sign of φ_1 is determined by the direction of $\overrightarrow{t_2}$ relative to the right-hand rule of $\overrightarrow{v_1} \times \overrightarrow{t_1}$. The angle φ_2 is defined the same way, with subscripts 2 in the definitions of $\overrightarrow{t_1}$ and $\overrightarrow{t_2}$.



Omega angle: The ω angle is a dihedral angle [-π:π] between the site-to-COM vectors around the binding radius vector σ. This is the only angle that directly constrains the orientations of one molecule to the other (rather than to the binding radius vector $\vec{\sigma}$). We have that

$$\omega = \text{acos}(\hat{t_1} \cdot \hat{t_2}), \qquad \text{Eq. S15a}$$

here $\hat{t_1}$ and $\hat{t_2}$ are the unit normals defined by

$$\overrightarrow{t_1} = \overrightarrow{\sigma_1} \times \overrightarrow{v_1} \qquad \text{Eq. S15b}$$

$$\overrightarrow{t_2} = \overrightarrow{\sigma_1} \times \overrightarrow{v_2}. \qquad \text{Eq. S15c}$$

The sign of ω is determined by the direction of $\overrightarrow{t_2}$ relative to the right-hand rule of $\overrightarrow{\sigma_1} \times \hat{t_1}$. The dihedral is not defined if either $\overrightarrow{v_1}$ or $\overrightarrow{v_2}$ is co-linear with $\vec{\sigma}$. We define a special case that constrains the orientation of molecule 1 relative to molecule 2 when both are co-linear with $\vec{\sigma}$ (e.g. as occurs in the clathrin-clathrin interactions). In that case we use the molecule normals to specify a dihedral, such that $\overrightarrow{t_1} = \overrightarrow{\sigma_1} \times \overrightarrow{n_1}$ and $\overrightarrow{t_2} = \overrightarrow{\sigma_1} \times \overrightarrow{n_2}$.

III. Spherical System

For sphere boundaries, which are treated as reflecting, we must adjust reflection off the curved boundary, diffusion on the curved surface (2D), distances when restricted to the 2D surface, and rigid body motion of complexes on the surface. To calculate the association probability between two particles on the sphere, which is a 2D case, the distance between their sites is calculated as the geodesic distance, not the straight-line distance. All the single particles and complexes on the sphere can diffuse and rotate, but the mathematical description is different from the one in solution.

Diffusion of single point particles on the sphere: Here, a single particle is treated as a point that doesn't have a three-dimensional structure or volume. We suppose that single particles will treat the sphere surface as a planar surface. Therefore, the diffusion on the sphere is the projection of the diffusion on a flat surface.

It is well known that the diffusion on a plane is described by two dimensions, Δx and Δy, which follow the Brownian dynamics:

$$\Delta x = \sqrt{2D_x \Delta t} \cdot \Psi_x, \Delta y = \sqrt{2D_y \Delta t} \cdot \Psi_y \qquad \text{Eq. S16}$$

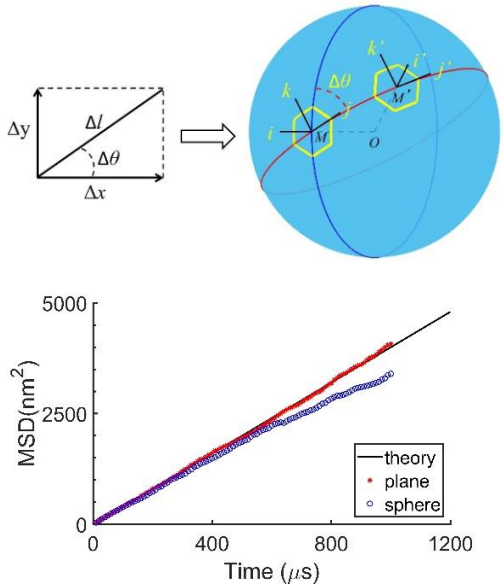
where D_x and D_y are diffusion coefficients, generally $D_x = D_y$, and Ψ_x and Ψ_y are Gauss random numbers, $\langle \Psi \rangle = 0$, $\langle \Psi^2 \rangle = 1$. We transfer them into:

$$\Delta l = \sqrt{(\Delta x)^2 + (\Delta y)^2}, \Delta\theta = \text{acos}(\frac{\Delta x}{\Delta l}) \qquad \text{Eq. S17}$$

where Δl is the displacement distance, and $\Delta\vartheta$ is the movement direction.

The diffusion on the sphere should generate Δl and $\Delta\vartheta$. As shown in the schematic below, M is one point that will diffuse, and O is the center of the sphere, and R is the sphere radius. Firstly, we find the great circle that passes M and the pole points. This great circle is defined as the reference circle (blue circle in the figure). Secondly, rotate this circle around O-M line by $\Delta\vartheta$, we get the new great circle as the trajectory circle (red circle in the figure). Then move M by arc length Δl on the trajectory circle. Through this process, the point M diffuses to

M'. The sign of y is used to decide whether to move up or down the red curve. Our simulation shows that mean squared displacement (MSD) due to diffusion on the sphere matches that expected by theory ($MSD = 4Dt$), up until the trajectories wrap around the sphere. The MSD on the sphere thus drops below the planar diffusion, as the maximum displacement is limited by the geometry.



Top: Illustration of diffusional updates on the sphere. Bottom: For a sphere of R=50nm, D=1nm^2/μs, the mean squared displacement (MSD) on surface matches a plane until the trajectories wrap. 1000 trajectories averaged.

Diffusion of multi-site complexes on the sphere: The diffusion of a rigid complex on the sphere surface is carried out by two steps. First diffuse the center-of-mass (COM) of the complex, then realign the 3D structure of the complex. The diffusion of complex COM is treated as the diffusion of a single particle on the sphere. For example, the point M is the COM of a complex (yellow color in the figure), and following the method above, M moves to M'. To precisely realign the structure, we set up an internal coordinate system of the complex, comprised of three unit-vectors, ***i***, ***j***, ***k***. For example, when the COM of the complex is at M, ***i*** is defined as the normalized vector of $\overrightarrow{OM}$. And ***j*** is defined as the tangent vector of the trajectory circle at point M, and the direction of ***j*** is inclined to M'. ***k*** is defined as $\boldsymbol{i} \times \boldsymbol{j}$. Then every point P in the complex can be expressed of the sum of ***i***, ***j***, ***k***:

$$\overrightarrow{MP} = \alpha \boldsymbol{i} + \beta \boldsymbol{j} + \gamma \boldsymbol{k}. \qquad \text{Eq. S18}$$

When the center-of-mass M moves to M', every point in the complex will also move, such as P moving to P', and the orientation of the internal coordinate system will change into ***i***', ***j***', ***k***'. Since the complex in NERDSS is rigid, we suppose the internal expression of every point in the complex won't change. Thus the new position of P' is:

$$\overrightarrow{M'P'} = \alpha \boldsymbol{i}' + \beta \boldsymbol{j}' + \gamma \boldsymbol{k}'. \qquad \text{Eq. S19}$$

Rotation on the sphere: Due to the constraint of the sphere surface, the complex on the sphere can only rotate around the line of O-M (M is the position of center-of-mass of the complex). Then the rotation is equivalent to a 2-dimentional rotation on a plane.

IV. Reaction rates for molecules with identical, repeated sites

When molecules have identical, repeated sites, such as clathrin (three trimer legs), they must be distinguished by distinct labels, as they have distinct coordinates in space. This labeling results in binding between both self and distinct sites: Cla(c1, c2, c3) can interact with another molecule using all possible pairs, Cla(c1)+Cla(c1)->, or Cla(c1)+Cla(c2)->, or Cla(c2)+Cla(c2)->, etc, with 6 distinct possibilities. The energetics of each of these reactions is (in general) all the same. Importantly, we note this means that the K_D value for distinct reaction sites must be twice as strong as for the self sites, $K_D^{c1c2} = K_D^{c1c1}/2$, i.e. the rates are such that $k_{on}^{c1c2} = 2k_{on}^{c1c1}$.

Only then will the kinetics and equilibrium of the N trimer assembly (excluding all spatial effects), match the behavior of 3*N independent identical sites, with K_D=K_D^{c1c1}. The user must directly specify this increase by a factor of 2, as there are other reactions, e.g. A(a)+A(b)->, where the specific rate is for two truly distinct sites (e.g. actin(barbed)+actin(pointed)->). This modification is performed automatically in NFsim[1] for these reaction types by dividing the user input rates for self sites by 2, and leaving the rates for distinct sites at the input values (we note that this results in a K_D that is half that expected from the input rates). For NERDSS, it is the user's specification when these indices are only distinct due to labeling, and thus the user must introduce the factor of 2 increase to distinct sites in this case. This result was previously validated for a structure-resolved clathrin model[2].

To derive this, we consider a self-binding reaction A(a)+A(a), with equilibrium K_D and initial concentration A_0. The equilibrium value is $A_{eq} = \frac{-K_D}{4} + \frac{\sqrt{K_D^2/4+2A_0K_D}}{2}$. If we now evenly split the sites into *n* distinct labels, we have that $nA_{i0} = A_0$, and at equilibrium, we want that $nA_{ieq} = A_{eq}$, where *i* indexes each of the labels. We know that $A_{i0} = A_{ieq} + 2AA_{iieq} + \sum_{j,j\neq i}^{n} AA_{ijeq}$ and $A_{ieq}^2 = K_D AA_{iieq}$. We then must specify that $A_{ieq}A_{jeq} = \frac{K_D}{2} AA_{ijeq}\ (j \neq i)$ to recover as desired that $A_{ieq} = \frac{1}{n}\left(\frac{-K_D}{4} + \frac{\sqrt{K_D^2/4+2nA_{i0}K_D}}{2}\right) = \frac{A_{eq}}{n}$. We note the equilibrium copy numbers thus generate the distinct products [Cla(c1!1).Cla(c2!1)] twice as numerous as self products [Cla(c1!1).Cla(c1!1)], as order does not matter in listing bound reactants.

Supporting Table

Systems	3D *(A+B)*	$3D \rightarrow 2D$ *(A+B)*	2D *(A+B)*	Self *(A+A)*
Relation to 3D microscopic binding rate	k_a^{3D}	$k_a^{3D\rightarrow 2D} = 2k_a^{3D}$	$k_a^{2D} = k_a^{3D}/h$[a]	$k_a^{self} = 2k_a^{3D}$
Microscopic dissociation rate	k_b	k_b	k_b	k_b
Macroscopic binding rate	$k_{on} = \left(\frac{1}{k_a^{3D}} + \frac{1}{4\pi\sigma D}\right)^{-1}$	$k_{on} = \frac{1}{2}\left(\frac{1}{k_a^{3D\rightarrow 2D}} + \frac{1}{4\pi\sigma D}\right)^{-1}$	$k_{on} = f(k_a^{2D}, \sigma, D, b_\rho)$[b]	$k_{on} = \frac{1}{2}\left(\frac{1}{k_a^{self}} + \frac{1}{4\pi\sigma D}\right)^{-1}$
Macroscopic dissociation rate	$k_{off} = k_{on}\, k_b / k_a^{3D}$	$k_{off} = 2k_{on}\, k_b / k_a^{3D\rightarrow 2D}$	$k_{off} = k_{on}\, k_b / k_a^{2D}$	$k_{off} = 2k_{on} k_b / k_a^{self}$
Equilibrium state	$K_D = k_b / k_a^{3D}$	$K_D = 2k_b / k_a^{3D\rightarrow 2D}$	$K_D = k_b / k_a^{2D}$	$K_D = 2k_b / k_a^{self}$

Table S1. Relationships between microscopic and macroscopic reaction rates. The proper relationship $K_D = k_{off}/k_{on}$ is preserved for all binding reactions. The user always inputs values of k_a^{3D} to the software (or k_{on}). The top row rates are what are input to the Green's function to evaluate binding probabilities, but the software takes care of the conversion from user inputs to these values. [a]The parameter *h* is a lengthscale that by default is set to 2σ. [b]See Ref [3].

Supporting Figures

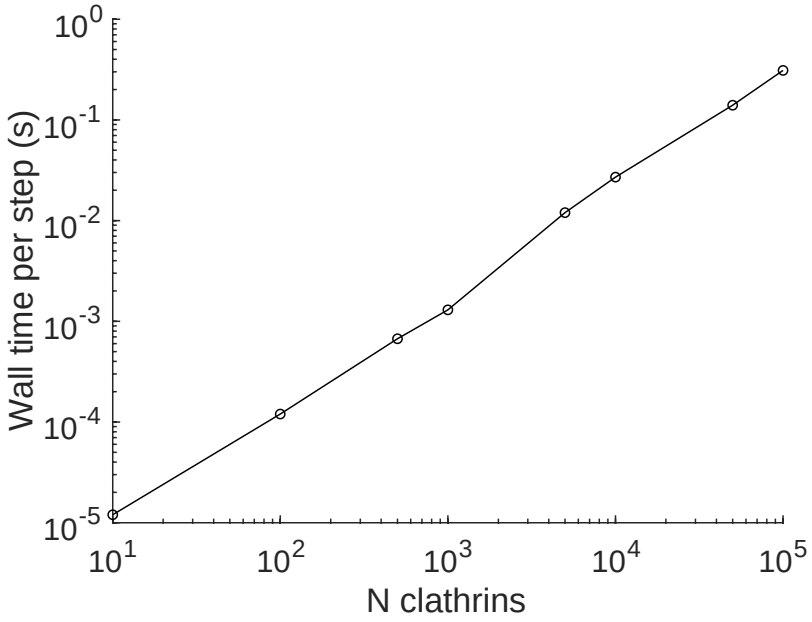


Fig S1: Wall time vs system size for the clathrin assembly is linear. The system here has K_D=1μM, k_{on}=1x10^{6}M^{-1}s^{-1}, k_{off}=1s^{-1}, and a fixed concentration of [Cla]=1.39 μM. The volume is systematically increased with increasing *N*. Exact wall time per system depends on reaction parameters, and diffusion. For smaller diffusion, the system can be partitioned into more sub-volumes, which accelerates pair-wise evaluations. For faster, stronger reactions, fewer sites are left free in solution, generally meaning fewer pairwise evaluations need be evaluated. Timing performed on a Dell workstation running Linux with Intel Core i9-9980XE, with Gnu compiler g++ version 7.4. Times are ~20% faster with Intel compilers.

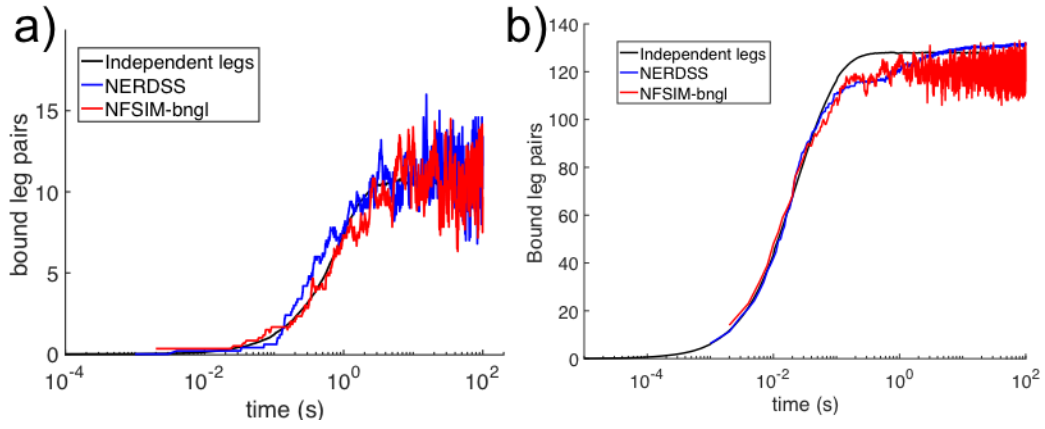


Fig S2: **NERDSS clathrin assembly simulations in solution can be compared with non-spatial stochastic simulations using NFsim software**[1]. a) K_D=100μM, k_{off}=1s^{-1} b) K_D=0.2μM, k_{off}=1s^{-1}.

V=(0.494μm)^3, N_{cla}=100, D_{cla}=13μm^2/s. Blue are NERDSS simulations. Black is the solution with all 300 legs independent—thus no lattice structure forms. Red is solved using the NFSim software package encoded using BNGL syntax. For K_D=100μM, only inter-molecular reactions are specified, that is, only bimolecular association between distinct binding sites. For K_D=0.2μM, intra-molecular reactions are also specified, using the same magnitude of rates as for bimolecular association. Because NFSim is non-spatial, no hexameric structure can be enforced, each clathrin simply has three sites. NFSim is a single trajectory, the NERDSS simulations are averaged over 3-5 trajectories. For NERDSS, f=0.001 (also see Fig S3).

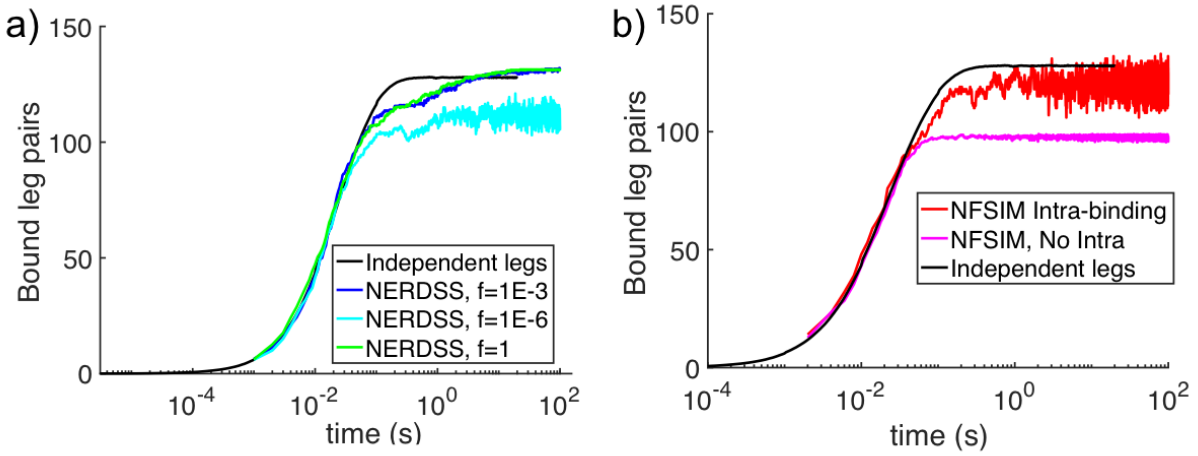


Fig S3. Loop closure events in rate-based models impact total numbers of 'bonds' formed. a) NERDSS simulations of clathrin assembly in solution with varying loop closure factors $f = \exp(-\Delta G_{coop}/k_B T)$. Simulations were performed with leg dimerization strengths of K_D=2 x10^{-7} M, k_{on}=5 x10^6 M^{-1}s^{-1}, k_{off}=1s^{-1}, V=(0.494μm)^3, N_{cla}=100, D_{cla}=13μm^2/s, $D_{R,cla}$=0.03 rad^2/s, Δt=0.2μs. Until the loop factor is comparable to $\exp(-\Delta G_{AB}/k_B T)$, (for this reaction 2 x10^{-7}), loops that form are difficult to break. For comparison, we simulated 300 independent legs, where no lattice forms and all events are dimerization. b) In non-spatial NFSim[1] simulations, the structural geometry is not specified (no hexagonal lattice), but sites that are within the same complex can form intra-molecular bonds, which are effectively loop closure events. For purely inter-molecular, or bimolecular association, the bound pairs plateaus relatively low, and becomes independent of K_D (data not shown), due to formation of a single component, where no internal links are allowed. With internal or intra-molecular events, here defined with the same magnitude rate, more events can occur. For non-spatial models, aside from each clathrin having 3 sites, no geometric or spatial constraints limit where loop closure occurs.

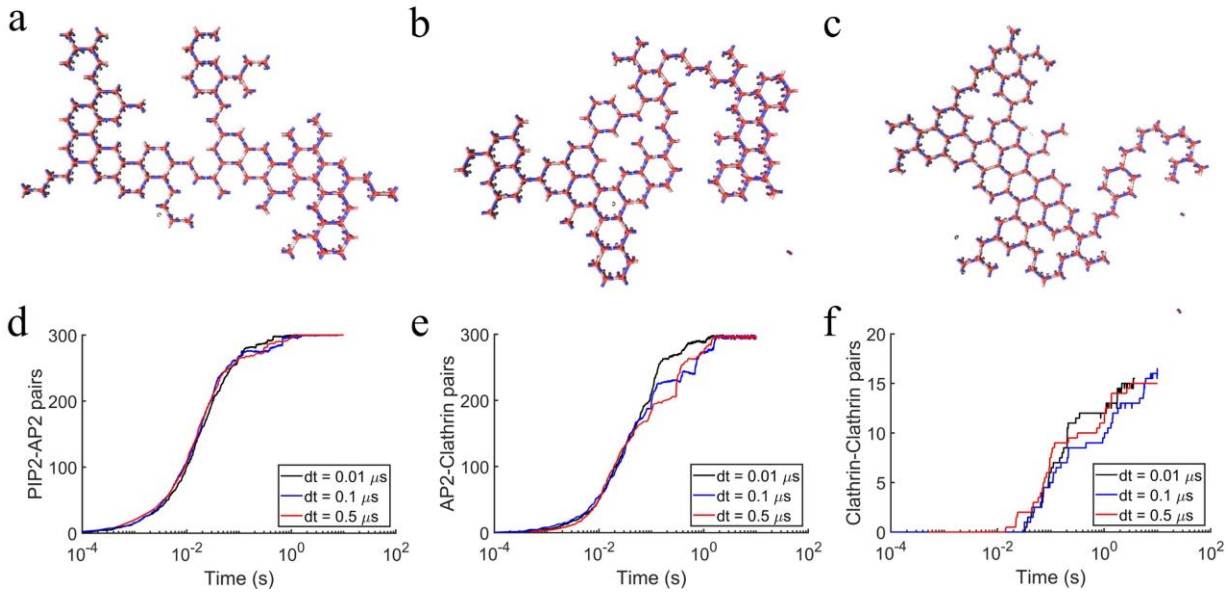


Fig S4. Clathrin assembly on the membrane surface is independent of time-step. Verification that when clathrin forms assemblies in 2D, the kinetics and structures are the same as Δt changes. Example structures from a) Δt=0.01μs, b) Δt=0.1μs, c) Δt=0.5μs. Kinetics of d) recruitment of adaptor protein (labeled AP2) to membrane lipids, e) Binding of clathrin to the adaptor protein, f) binding of clathrin to clathrin. Each clathrin has a reflecting site at its center, to prevent fully bound clathrin from not 'seeing' each other. Each trace is a single trajectory.

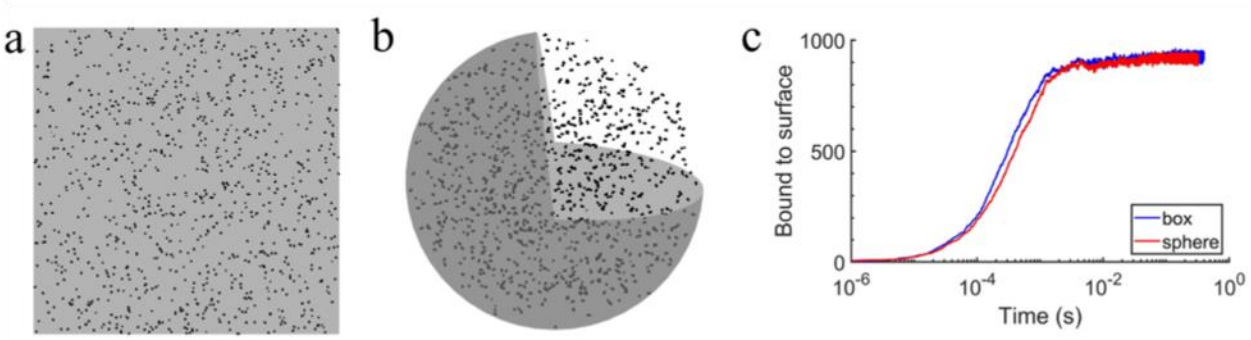


Fig S5. Reversible recruitment of interacting protein pairs to a surface in a spherical system. This model has two species A(a,m) and M(m), and 6 reactions: 1) A(a)+A(a) ⇌A(a!1).A(a!1), 2) A(m)+M(m) ⇌A(m!1).M(m!1), 3) A(a!1).A(a!1,m)+M(m) ⇌A(a!1).A(a!1,m!2).M(m!2), 4) A(a,m!1).M(m!1)+A(a) ⇌A(a!2).A(a!2,m!1).M(m!1), 5) A(a,m!1).M(m!1)+A(a,m!2).M(m!2) ⇌ M(m!1).A(a!3,m!1).A(a!3,m!2).M(m!2) and 6) A(a!1).A(a!1,m!2).M(m!2)+M(m) ⇌ M(m!3).A(a!1,m!4).A(a!1,m!2).M(m!2). The last two reactions are in 2D. The kinetics of binding to the surface of a) a box vs b) a sphere, with the same V/A ratio. Lipids are modeled using the implicit lipid model. c) Number of solution particles bound to the surface agrees well. Sphere R=100nm, Box=[354.4908, 354.4908, 33.333]nm, k_a=8.3056nm^3/μs, k_b=1000s^{-1} for both A+A and A+M binding reactions. σ=1nm. N_A=1000, N_{IL}=2000, D_A=12nm^2/μs, D_{IL}=1nm^2/μs.

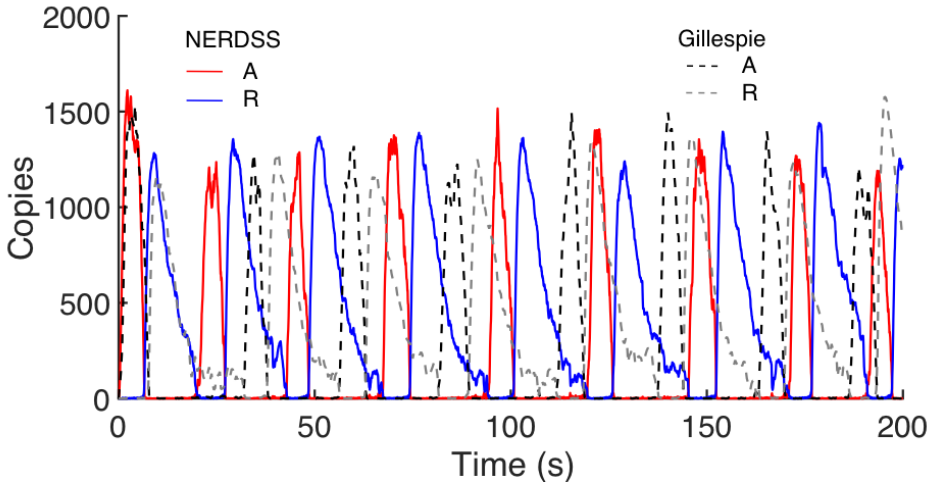


Fig S6: NERDSS simulations of the circadian clock model[4] agree with Virtual Cell[5] simulations. Virtual Cell simulations were solved using either deterministic partial differential equations (Fig 3 main text) (PDE) or stochastic non-spatial algorithms (Gillespie). To quantitatively compare stochastic simulations, which are noisy, we calculate the average period of oscillations for A and R over 200 s. For PDEs (or ODEs—data not shown) the oscillation times are 25.1s and 25.1s, with a lag-time of 6.55s. For Gillespie, averaged over 10, 200s trajectories, the times are 24.8 and 25.1s, with a lag-time of 6.63s. For NERDSS, we have comparable results, indicating that the system remains well-mixed and largely rate-limited, with times of 24.5s and 24.75s for a 200s trajectory, and lag-time of 6.45s. We also ran a set of simulations with increased diffusion from 10 to 20 $\mu m^2/s$, which alters the intrinsic rates for bimolecular reactions, and then measured times of 25.1, 25.2, and 6.4 over 325s. To calculate these times, we performed a discrete Fourier transform of the signals (the signal was zero-padded out to 5000s to increase resolution of the frequencies). The reported oscillation time is the maximum amplitude frequency observed. The lag-time was evaluated using cross-correlation, again reporting the time with maximum amplitude. MATLAB was used. Similar results are obtained by simply finding the peaks, calculating the time-separation between them, and averaging.

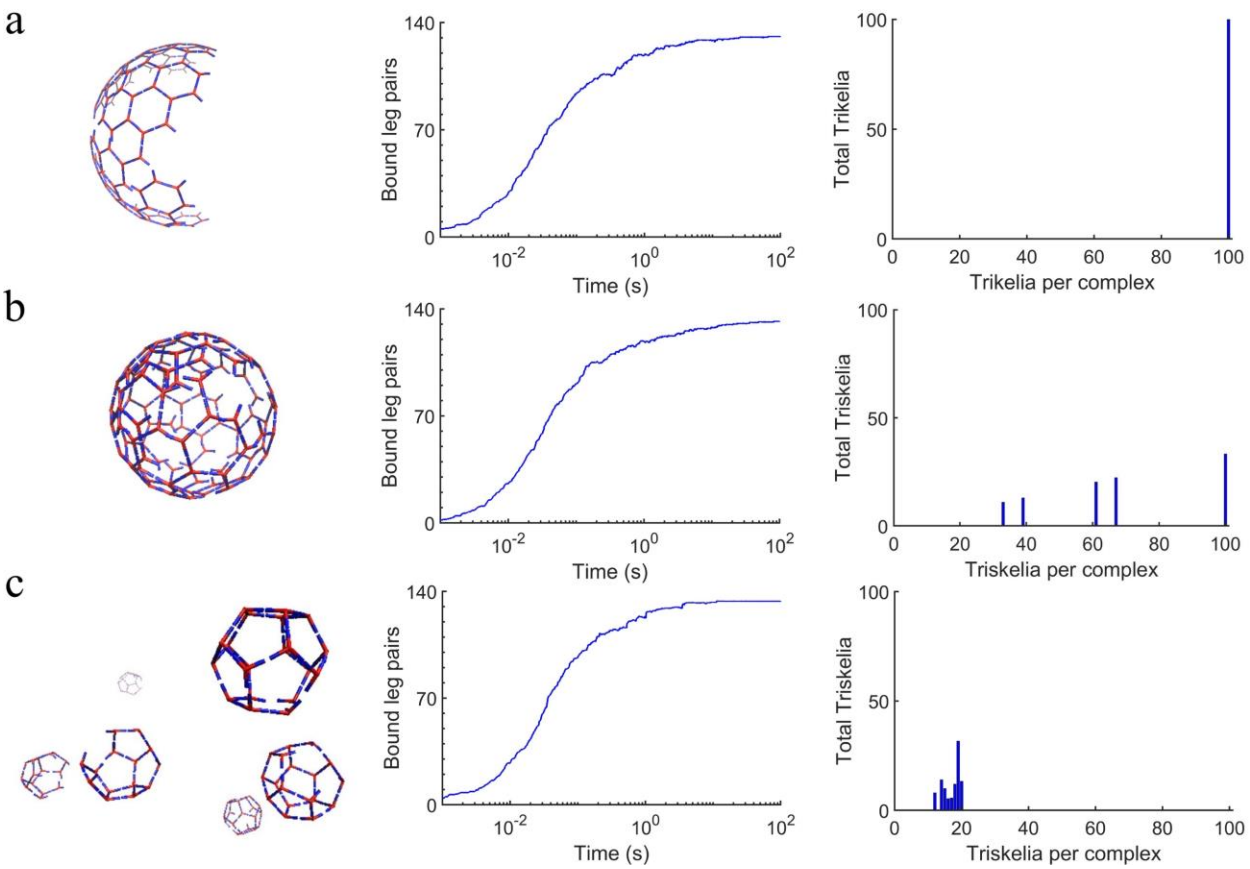


Fig S7. Clathrin assembly with varying pucker angles. Flat planar clathrin has a pucker angle of 90°. Here we increase it to a) 95° b) 100° c) 110°. As the angle increase, the curvature of the cages increases and accommodates fewer trimers. The largest can accommodate ~500 trimers, the smallest ~20, which orient into pentagonal faces not hexagonal faces, despite binding with the same angles. For simulations here and in S8-S10, V=(0.494μm)^3, N_{cla}=100, D_{cla}=13μm^2/s, $D_{R,cla}$=0.03 rad^2/s, Δt=0.2μs. K_D=0.2μM, l_{leg}=7.5 nm, f=0.001, bindRadSamCom = 5nm, overlapSepLimit = 7nm.

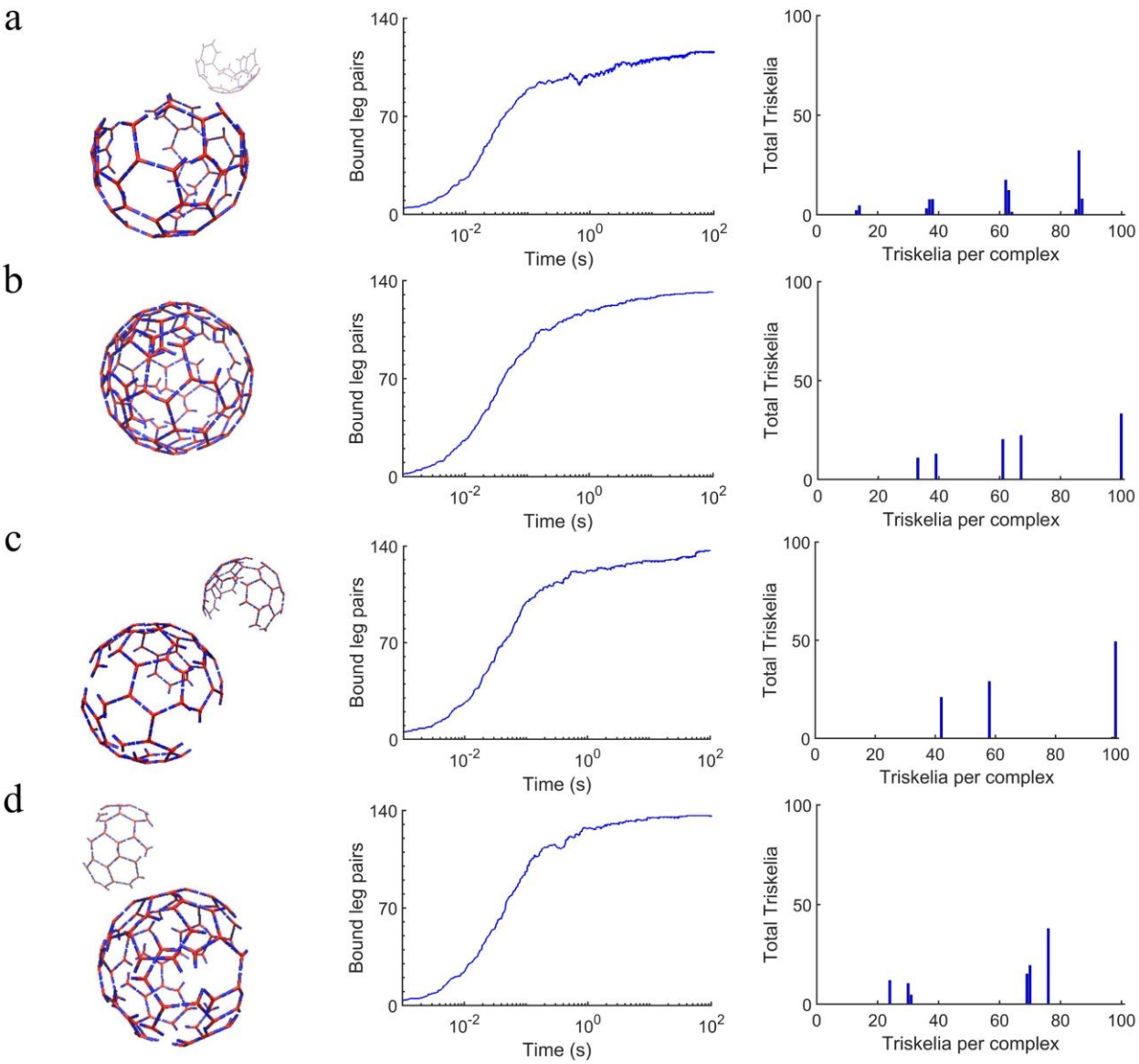


Fig S8. Clathrin cage simulations can form defects that can still stabilized interactions. By allowing free leg sites that are in close by not perfect contact to still form 'bonds', the cages are stabilized further against dissociation. This parameter, bindRadSameCom, is increased to allow bond formation at longer distances, from cutoffs at a) 3nm b) 5nm c) 8nm d) 10nm. When the distances are relatively large compared to the molecule size (l_{leg}=7.5 nm), the bonds are less physical, and prevent recruitment of additional trimers. Pucker is 100°, other parameters match Fig S7.

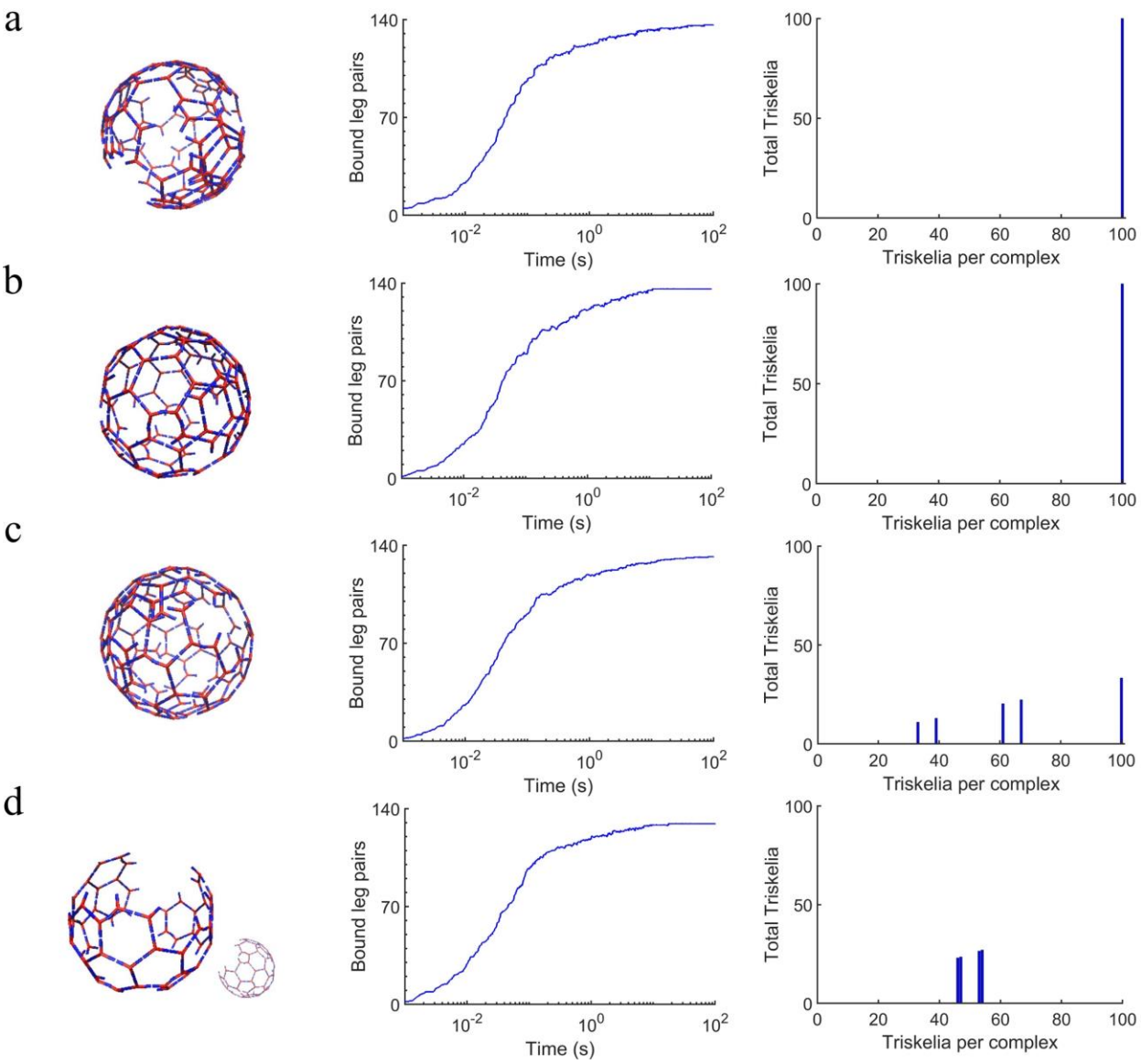


Fig S9. Clathrin cage simulations can form defects, and evaluating steric overlap impacts structure. When binding events occur between clathrin, the events are rejected if components of the two complexes have steric overlap, which is true if any COM-COM distances are less than overlapSepLimit. As this limit is increased from a) 1.5nm b) 5nm c) 7nm d) 10nm, more association moves with any trimer overlap are rejected, resulting in more prevention of defects and fewer trimers per cage. Pucker is 100°, bindRadSameCom=5nm, other parameters match Fig S7.

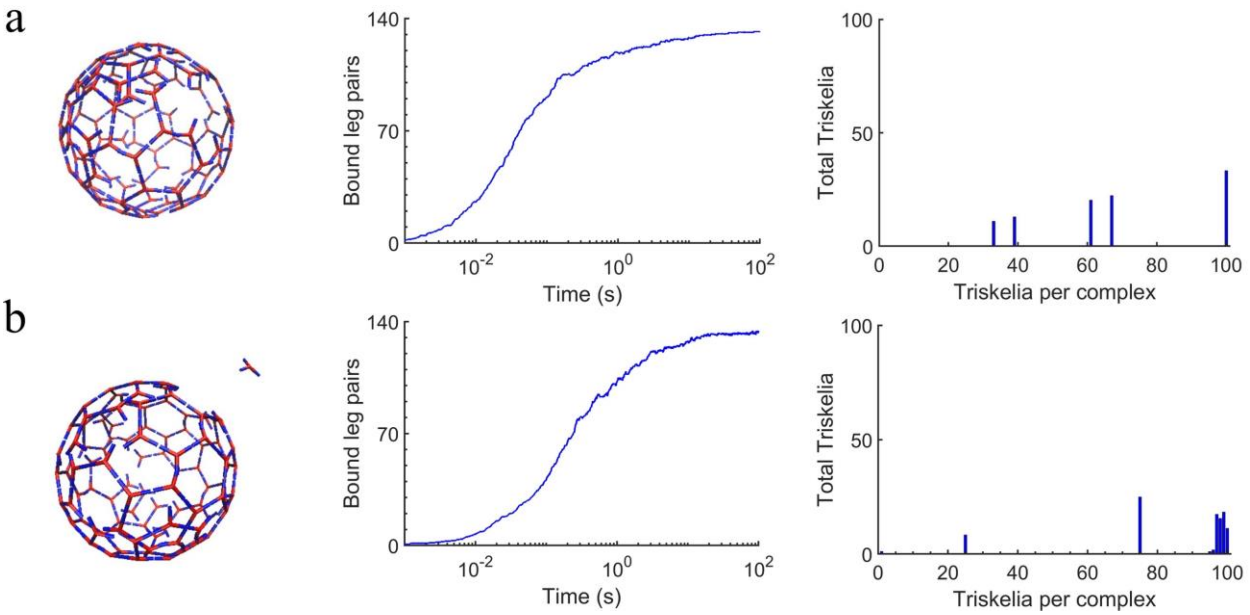


Fig S10. Clathrin cages nucleate once K_D is sufficiently strong. Here we compare results for a) K_D=0.2μM. b) 1μM. Parameters are the same as Fig S7. Here, we note the K_D in (b) is the same as in Fig 2d from the main text. The final equilibrium of bound leg pairs is higher here because we have increased *f* to 0.001, from $5x10^{-6}$, resulting in more stable loops.

SI References

1 Sneddon, M. W., Faeder, J. R. & Emonet, T. Efficient modeling, simulation and coarse-graining of biological complexity with NFsim. *Nature methods* **8**, 177-U112 10.1038/Nmeth.1546 (2011).

2 Johnson, M. E. Modeling the Self-Assembly of Protein Complexes through a Rigid-Body Rotational Reaction-Diffusion Algorithm. *J Phys Chem B* **122**, 11771-11783 (2018).

3 Fu, Y. *et al.* An implicit lipid model for efficient reaction-diffusion simulations of protein binding to surfaces of arbitrary topology. *J Chem Phys* **151**, 124115 10.1063/1.5120516 (2019).

4 Vilar, J. M., Kueh, H. Y., Barkai, N. & Leibler, S. Mechanisms of noise-resistance in genetic oscillators. *Proc Natl Acad Sci U S A* **99**, 5988-5992 10.1073/pnas.092133899 (2002).

5 Moraru, I. I. *et al.* Virtual Cell modelling and simulation software environment. *Iet Syst Biol* **2**, 352-362 Doi 10.1049/Iet-Syb:20080102 (2008).