Local Rules Simulation of the Kinetics of Virus Capsid Self-Assembly

Russell Schwartz,* Peter W. Shor,[#] Peter E. Prevelige, Jr.,[§] and Bonnie Berger*[¶]

*Laboratory for Computer Science and [¶]Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139; [#]AT&T Labs, Florham Park, New Jersey 07932, and [§]Department of Microbiology, University of Alabama at Birmingham, Birmingham, Alabama 35294, USA

ABSTRACT   A computer model is described for studying the kinetics of the self-assembly of icosahedral viral capsids. Solution of this problem is crucial to an understanding of the viral life cycle, which currently cannot be adequately addressed through laboratory techniques. The abstract simulation model employed to address this is based on the local rules theory of Berger et al. (1994. *Proc. Natl. Acad. Sci. USA.* 91:7732–7736). It is shown that the principle of local rules, generalized with a model of kinetics and other extensions, can be used to simulate complicated problems in self-assembly. This approach allows for a computationally tractable molecular dynamics-like simulation of coat protein interactions while retaining many relevant features of capsid self-assembly. Three simple simulation experiments are presented to illustrate the use of this model. These show the dependence of growth and malformation rates on the energetics of binding interactions, the tolerance of errors in binding positions, and the concentration of subunits in the examples. These experiments demonstrate a tradeoff within the model between growth rate and fidelity of assembly for the three parameters. A detailed discussion of the computational model is also provided.

INTRODUCTION

The pathway by which icosahedral virus protein coats assemble from their subunits is an important and incompletely described phenomenon, consisting of the self-assembly of many proteins into a complicated but regular structure. Capsid structure has been described by the quasiequivalence theory of Caspar and Klug (1962), which provides an explanation for the final assembled state of a capsid. However, quasiequivalence offers little predictive power regarding the process of assembly. Although the subunits in a capsid are usually chemically identical, they can adopt distinct conformations in the assembled capsid, with the conformation of any given subunit attained in a pathway-dependent fashion (Rossman, 1984; Johnson and Speir, 1997). It is not currently known what constraints, if any, there are on assembly pathways or the distribution of intermediates. In addition, it is unknown what physical properties of coat proteins would enforce these constraints. Other open questions regarding the assembly process include the points at which conformational switching occurs and the source of the observed nucleation-limited behavior (Prevelige et al., 1993). All icosahedral viruses must have some means of addressing these questions, although the mechanistic answers may differ from one virus to another. Answers to these and other questions may provide insight into viral assembly and have the potential to assist in developing novel approaches to interfering with viral infection.

Experimental techniques for examining these pathway-dependent processes have thus far been hampered by the complexity of the system. However, modeling and simulation-based approaches have the potential to assist in understanding such processes. Prior modeling and simulation work has provided insight into many aspects of virus capsid assembly but has not been directed at answering the specific questions regarding assembly kinetics addressed by the present work. Horton and Lewis (1992) developed a method of using association energies of viral coat proteins to predict assembly intermediates. Reddy et al. (1998) extended this method, applying the CHARMM energy calculation program (Brooks et al., 1983) to the crystallographic structures of three icosahedral viruses to estimate the interaction energies that would be applied to the prediction of intermediates. The method, however, relies on the assumption that the stability and distribution of intermediates can be reliably predicted by examination of the final assembled structure. Given the nature and extent of conformational changes accompanying capsid assembly (Steven et al., 1992; Galisteo and King, 1993; Prevelige et al., 1994), this assumption may not be reasonable for examining some aspects of capsid assembly. Furthermore, as the present work attempts to demonstrate, unexpected behaviors may appear during assembly, even when the energetics of interactions are understood, suggesting that the sort of information provided by the work of Horton and Lewis and Reddy et al. might be supplemented by additional modeling work. Marzec and Day (1993) modeled capsomeres as generic "morphological units" and showed that allowing them to move freely over the surface of a sphere, minimizing a potential energy function, could give rise to observed quasiequivalent patterns. Tarnai et al. (1995) used a similar approach, searching for maximally dense packings of pentagons on a sphere, to model viruses such as SV40 and papilloma virus, which are

composed entirely of pentamers. Although the approaches of Marzec and Day and Tarnai et al. are useful for studying symmetries of the final shell, they provide little insight into its assembly process. Zlotnick (1994) developed a model based on the assumption that fully formed and partially formed capsids exist at equilibrium with free coat proteins, giving rise to kinetics that appear to be nucleation-limited; this approach allowed for a more dynamic simulation of the growth process than other methods. Zlotnick's approach, though, examined interactions at the scale of population distributions of large numbers of intermediates over time, rather than providing detailed information about a small number of capsids, as the present work attempts to do. Furthermore, Zlotnick's simulations were specific to his equilibrium model. It can also be noted that none of the aforementioned approaches can model irregular or malformed structures, an important goal of the present work.

More general simulation techniques are also unable to address the concerns of the present work, because of the computational cost of the problem. Low-level molecular models that work at atomic or even amino acid resolution cannot handle more than a few proteins the size of a typical viral coat protein. Simulations of capsid assembly kinetics must consider hundreds or thousands of coat proteins; the molecular dynamics techniques currently used for protein structure determinations or protein docking simulations are therefore unusable for studying many questions of capsid assembly kinetics. Simpler lattice models that rely on combinatorial optimization would also be unsuitable for this problem, because of their focus on equilibrium values rather than kinetics.

Because the problems examined by the present work cannot currently be adequately addressed experimentally or by prior modeling and simulation approaches, we have attempted to explore them through more abstract modeling techniques based on the principle of local rules (Berger et al., 1994). The local rules theory proposes that coat proteins take on distinct conformations determined by sets of "local rules," in which a protein chooses its conformation and its relative position based on the conformations of its immediate neighbors. The local rules theory makes it possible to describe the high-level symmetries of a completed capsid, using only information available to individual subunits. Fig. 1 shows the simplest local rules set, which describes a $T = 1$ geometry. The rule indicates that a protein in conformation 1 is connected to three other proteins, each of conformation 1. The angles between successive binding sites are ~120°, ~120°, and ~108°. This rule provides an abstract mechanism by which a $T = 1$ capsid could form; each subunit, as it connects to any other subunit, need only ensure that its positioning relative to its neighbors is consistent with the rule to ensure that the completed capsid will have $T = 1$ symmetry. Fig. 2 shows a more complicated $T = 7$ rules set with seven conformations. (One also exists with four conformations.) Local rules can also describe such complicated structures, although that capability is not applied in the present work. The local rules theory has offered
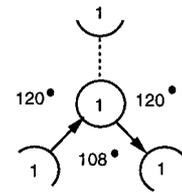

FIGURE 1　Local rule for a $T = 1$ capsid. Angles specify the angle between binding interactions. Arrows distinguish different edge types, so that any edge with an outward arrow on one subunit must connect to an edge with an inward arrow on another subunit, whereas an edge with no arrow on one subunit must connect to an edge with no arrow on another subunit. Assembling subunits consistently with these rules can only produce a $T = 1$ shell or a subset thereof.

explanations for several puzzling observations about capsid assembly. Berger and Shor (1995) showed how only a minor change in a rule set could select between $T = 4$ and $T = 7$ geometries, providing a possible explanation for observed links between the two geometries (Dokland et al., 1992; Earnshaw and King, 1978; Katsura 1983; Thuman-Commike et al., 1998). In addition, Berger et al. (1994) used an early computer model of local rules to explore the robustness of rules sets to small deviations from their ideal binding angles and to demonstrate a possible source of observed shell malformations resulting from an error in applying local rules.

In this paper, we apply the local rules theory to the development of an abstract model for subunit-subunit interactions that allows for computationally feasible simulations of the overall self-assembly reaction. By assuming that
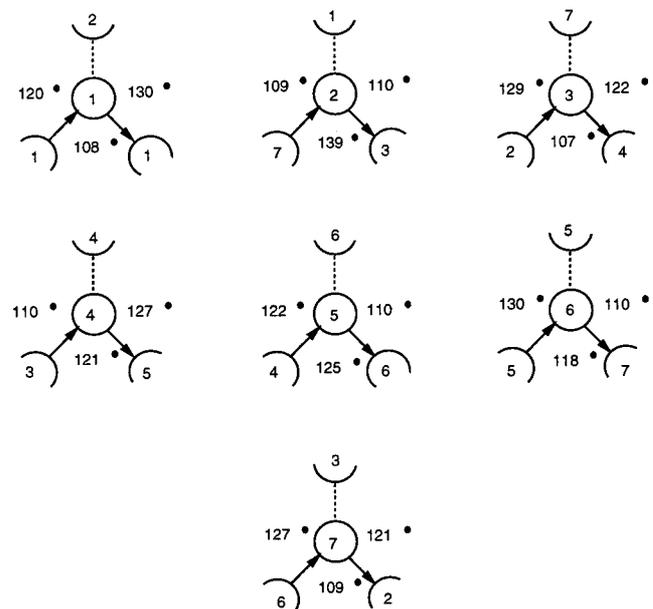

FIGURE 2　One set of local rules for a $T = 7$ capsid. As with Fig. 1, angles specify angles between binding interactions and arrows distinguish types of edges. This rule set specifies seven distinct conformations, each with its own set of neighbors and binding interactions. Assembling subunits according to these rules can only produce partial or complete $T = 7$ shells.

assembly proceeds according to local rules, we can remove many details that then become irrelevant to answering many of the questions that concern us. For example, by assuming that interprotein binding always follows sets of local rules, we can disregard the complex combination of electrostatics, hydrophobic interactions, and other processes that may actually be involved in binding. Other simplifications include the use of probabilistic Boltzmann distributions for handling binding and conformational shifting and the use of a model of Brownian motion for maintaining realistic thermodynamic behavior. By relying on theoretical and statistical models to remove details that do not appear to be relevant to the questions we are asking, we have been able to make the problem computationally tractable. This approach has the additional benefit of allowing for a model that is general with respect to the specific details of assembly and can be adapted to many different assembly models. It remains to be shown that we have retained sufficient detail to use our model as a predictive tool for studying self-assembly behavior.

The purpose of this paper is to describe our approach to the computer modeling of capsid assembly dynamics. To illustrate what can be accomplished with this approach, the paper presents the results of a series of simulation experiments conducted using a simplified version of the simulation model. Although the central focus of this paper is the simulator itself, rather than the results of the particular simulation experiments presented, the simulations nonetheless raise some interesting results relating growth rates and malformation rates of simulated $T = 1$ capsids, which the paper describes. It then draws some conclusions about our results and the applicability of our techniques to future work in biochemical simulation. Finally, the paper describes the simulation model and provides details of how it was implemented for the present work.

MATERIALS AND METHODS

Computer model

Defining a simulated system under the local-rules model requires specifying some basic assembly subunit. The subunit may be an individual coat protein, as in the simulation experiments. However, the model also allows for larger subunits, such as dimers or capsomers, which can themselves assemble according to local-rules binding patterns. It should also be noted that even if a larger subunit is used by the system being modeled, the simulation model described here can capture correct assembly in terms of monomers by biasing monomer kinetics to form the correct subunit more readily than other subsets of a capsid. Simulated subunits are built from unions of spheres, each with a characteristic mass, radius, and binding configuration. In the simplest form, a subunit can be modeled as a single sphere with a set of bonds, each of which represents a potential binding interaction with another subunit. One such subunit is shown in Fig. 7 A. Subunits of different shapes can be made from unions of these simple spheres, modeled as if they were rigidly connected; such a subunit is shown in Fig. 7 B. (This representation is not used in the experiments presented here.)

Binding interactions between subunits are modeled through bonds that form between separate subunits. Each bond has a characteristic length, direction, rotational vector, pair of activation energies for the association and dissociation reactions, set of spring constants used to model the forces exerted by binding interactions, set of tolerances, and set of allowable neighbors. A bond can attach to or detach from another bond probabilistically according to its characteristic energies, provided the two bonds are allowed to be neighbors and their locations are within their permitted tolerances. For a bond with activation energy of association E_r and activation energy of dissociation $-E_d$, the probability of binding to another bond of an allowed type within the correct tolerance is $e^{E_r/k_bT}/(1 + e^{E_r/k_bT})$, and the probability of breaking an existing bond is $(1 + e^{E_d/k_bT})^{-1}$, where k_b is Boltzmann's constant and T is the temperature. The forces bound subunits exert on each other are described in the next section.

Conformational shifting is implemented by allowing subunits to take on several shapes, each with a characteristic energy. A subunit shifts between its allowed shapes probabilistically according to the Boltzmann distribution of the energies. The probability of a subunit shifting to a conformation k on a particular time step is given by

$$P(k) = \frac{e^{E_k/k_bT}}{\Sigma_i e^{E_i/k_bT}}$$

where E_i is the energy of conformation i, T is the temperature, and k_b is Boltzmann's constant. If the subunit has formed any bonds, then the energies of those bonds are added to the energy of the current conformation, as they must be broken to shift conformations. Multiple-conformation subunits can be used as subsets of other subunits, allowing separate domains to shift conformations independently, although this functionality is not used in the experiments presented here.

Subunit-subunit and subunit-solvent interactions

The most important contributions to the force on a bound subunit are the bond forces exerted by its neighbors. Each bond is forced toward a characteristic angle and length. The forces pushing a bond toward its ideal value are modeled as three springs, representing a translational force, F_t, a rotational torque around the bond, T_r, and a bending torque, T_s, that straightens bonds. These forces are calculated by the following equations:

$$F_t = k_t(\vec{e}_2 - \vec{e}_1)$$

$$T_r = k_r[(\vec{u}_1 \times \vec{u}_2) \cdot \vec{d}_1]\vec{d}_1$$

$$T_s = k_s \sqrt{0.5 + 0.5(\vec{d}_1 \cdot \vec{d}_2)} \frac{\vec{d}_2 \times \vec{d}_1}{\|\vec{d}_2 \times \vec{d}_1\|}$$

Here, k_t, k_r, and k_s are user-supplied spring constants; $\vec{e}_1$ and $\vec{e}_2$ are the end points of bonds one and two; $\vec{u}_1$ and $\vec{u}_2$ are rotational vectors defining the desired relative rotations of the two subunits around the bonds; and $\vec{d}_1$ and $\vec{d}_2$ are the directions of the two bonds. The end points of the bonds define the optimal relative positions of the two bound subunits and need not have any relationship to the sizes of the two particles.

Collisions between two subunits or between a subunit and an artificial boundary around the simulated solution create a second class of forces. When two spheres collide, they exert a force on each other, given by

$$\min\left\{ C\frac{(r_1 + r_2)^2 - (r_1 + r_2)\|\vec{d}\|}{\|\vec{d}\|^2}, 2C \right\}$$

where C is a constant set at 50 kDa · nm/μs^2, $\vec{d}$ is a vector between the centers of the spheres, and r_1 and r_2 are their radii. Likewise, when a sphere collides with the boundary of the simulation, the boundary exerts a force on the sphere, given by

$$\min\left\{ Cm\frac{1 - o/r}{(o/r)^2}, 12CM \right\}$$

where M is the mass of the subunit, r is the radius of the sphere, o is the overlap distance between the edge of the sphere and the edge of the simulation, and C is defined as above. These forces were selected by trial and error to give reasonable collision behavior while being inexpensive to compute and allowing quick convergence of the numerical methods.

The final class of forces results from a model of Brownian motion, intended to keep the average kinetic energy of the particles close to a constant over long periods of time. This model consists of two forces: a damping force and a randomized force of adaptive magnitude designed to increase kinetic energy. The damping force on each time step or partial time step is implemented by the assignments

$$\vec{v} \leftarrow \vec{v} - d\vec{v}M^{2/3}$$

$$\vec{o} \leftarrow \vec{o} - d\vec{o}I^{2/3}$$

where $\vec{v}$ is the velocity, $\vec{o}$ is the angular velocity, M is the subunit mass, I is its moment of inertia about its axis of rotation, and d is a user-supplied damping constant. This damping force alone would tend to reduce the energy of a simulation. The randomized force adds small perturbations to the subunit velocity, whose distributions are adaptively chosen to tend to increase subunit energy. The two forces together prevent small numerical errors from gradually increasing or decreasing the average kinetic energy over the course of a simulation but avoid drastically altering subunit paths over short distances.

Numerical methods

The central computational problem of the simulator is the integration of the equations of motion based on the forces described above. The equations of motion can be described in terms of the vectors of velocities, $\vec{v}$, and positions, $\vec{x}$, as two differential equations:

$$\frac{d\vec{x}}{dt} = \vec{v}$$

$$\frac{d\vec{v}}{dt} = f(\vec{x}, \vec{v})$$

where f computes the accelerations based on the current positions and velocities. The overall approach employed is an adaptive Euler method with Richardson extrapolation. This method is run in parallel, using the Cilk multithreading system (Blumofe et al., 1995).

The Euler method is the simplest numerical integration scheme, computing the simulation state at a future time based on the state at the current time. A forward Euler for integrating velocity is combined with a backward Euler for integrating position, giving the following pair of iterations:

$$\vec{v}_{n+1} = \vec{v}_n + \Delta t f(\vec{x}_n, \vec{v}_n)$$

$$\vec{x}_{n+1} = \vec{x}_n + \Delta t \vec{v}_{n+1}$$

This gives a first-order accurate expression that requires one function evaluation for each time step.

Because of the irregular nature of the problem, this integration scheme is modified with an adaptive step size. When the simulator attempts to advance the time by one unit of time, it first evaluates the problem with a step size of one unit. It then repeats the evaluation for all subunits with step sizes of 0.5 units. It then uses an approximate test of convergence, testing whether the results for these two times are within a user-specified tolerance. If the two values are not within that tolerance, then the evaluation is repeated with those subunits, using half of the previous step size, and all other subunits, using the same step size used on the previous round. This process continues until all parameters have converged. Although this process does not provide any absolute guarantee of bounded errors, it does set an approximate bound on the values of numerical errors.

One further complication in the method is the use of Richardson extrapolation. Richardson extrapolation is a technique for combining lower order representations to generate a higher order representation. This is combined with the adaptive step size, so that the values from different step sizes are extrapolated to produce an nth-order approximation when n different step sizes are used. Because of the possibility of discontinuous forces interfering with the extrapolation, the simulator tries extrapolating with the k smallest step sizes available, for each value of k, and uses any that have converged, favoring higher order values if more than one extrapolated value converges on a given step.

The numerical methods described here are implemented in a parallel, thread-based program. Each time one step of the Euler method is performed, a separate thread is spawned

for each subunit, allowing force calculations to proceed in parallel.

User interface

The simulator uses a graphical user interface running as a front end to an interpreted control language. The graphical interface allows users to enter most common commands through a simple, button-based system. Under this system, users can easily manipulate the graphics display, add or remove subunits, advance the simulation time, or load and save simulations, using only a few button and key presses. Thus most commands can be handled quickly and with little knowledge of simulator details.

At a lower level, all user interactions are handled by the interpreted control language. User commands entered through the graphical interface are translated into text commands, which are passed to the interpreter. In addition, users can type commands directly to the interpreter. The interpreter is also accessible by loading data files, allowing users to create loadable modules that can redefine many aspects of simulator behavior. Users design simulations by creating "rules files"; these are files of commands written in the control language that define the simulation parameters, add the simulated subunits, and create any other data structures needed by a particular simulation. The use of a powerful control language ensures that the simulator is sufficiently versatile to adapt to a wide range of user needs, and the graphical interface maintains ease of use for common commands.

Computational issues

The simulator is written primarily in C. However, the force calculations and most numerical routines are implemented with the Cilk multithreading system (Blumofe et al., 1995), a parallel extension to C portable to several serial and parallel architectures. For the tests described here, the parallel routines ran on an eight-processor Sun Ultra Enterprise 5000 symmetric multiprocessor. The user interface ran separately on a Silicon Graphics Indigo workstation, which generated the graphics contained in this paper.

RESULTS

Experimental design

Each simulation involves a number of subunits, representing coat proteins, which move freely throughout a simulated solution. Subunits are capable of forming binding interactions with each other in accordance with local rules specifying allowed binding patterns, provided they are within specified angular and distance tolerances of the ideal values given by the local rules. Although the ideal $T = 1$ rules can only produce correctly formed $T = 1$ capsids, the angular and distance tolerances and the flexibility of existing binding interactions allow for nonoptimal binding, which can lead to malformed capsids. Simulations can be halted at various times to record data on the current state of the population, allowing measurements of the numbers of on- and off-pathway subunits at regular intervals. The reader is referred to Materials and Methods for details of the simulation model.

For these experiments, we did not attempt to model any particular virus, but rather to create a generic model of $T = 1$ capsid assembly. Parameters were initially chosen to be reasonably close to what would be expected for actual virus capsids and then were adjusted empirically, based on simulation runs, to give rapid growth. Subunits were designed to polymerize according to the autostery model of Caspar (1980); all subunits therefore shift between two conformations, a more stable, nonbinding conformation with a potential energy of -2 kcal/mol, and an unstable binding conformation with zero potential energy. Bond angles for the binding conformation were chosen to be ideal for a $T = 1$ capsid. A single sphere was used to represent each coat protein. For this sphere, a mass of 100 kDa and a diameter of 3 nm in the binding conformation and 1 nm in the nonbinding conformation were employed. The different sizes were chosen to make the two conformations visually distinct, rather than for any anticipated effect on the simulation results. Parameters associated with binding interactions were selected empirically to allow a rapid rate of capsid growth while keeping the rate of malformations low. The base binding energies, which control the probability of bonds forming when they are within allowed tolerances and their probability of subsequently breaking, were set at -8500 kcal/mol. An angular tolerance of $30°$ and a base distance tolerance of 10 nm were chosen. Each test used 300 subunits. The base work space size was 125 nm in each dimension, giving a concentration of 2.47×10^{-4} mol/liter, or 24.7 mg/ml. The unusually high concentration was meant to give a rapid growth rate, a constraint made necessary by the high computational costs of simulations. However, the authors believe qualitatively similar results could be achieved at substantially lower concentrations, given sufficient time, by adjusting binding energies or tolerances to compensate for the reduced thermodynamic probabilities of assembly a low concentration would produce. The binding energy, tolerance, and work space size were modified for some simulations, as will be described below. The meanings of the various simulation parameters are described in more detail in Materials and Methods.

Growth and malformation rates were measured by examining, at five multiples of 5000 time steps, the numbers of nucleated on- and off-pathway subunits. For the purposes of the experiment, a cluster of subunits was defined to be nucleated if it was a connected cluster of five or more subunits. A subunit was defined to be on-pathway if it was attached only to a subset of a correctly formed $T = 1$ capsid and off-pathway if it was attached to a cluster, any part of which was inconsistent with a correctly formed $T = 1$ capsid. Thus even a single incorrect bond in an otherwise correctly formed cluster resulted in our classifying all sub-

units in the cluster as off-pathway. The fixed cutoff time of 25,000 time steps was chosen to allow a significant amount of growth. It has not been possible to determine the exact mapping between time steps and physical time in terms of time-dependent phenomena such as the diffusion rate. However, it is expected that the assembly times of the simulated capsids are substantially shorter than the assembly time of an actual capsid, largely because computational limits required biasing the binding parameters toward very rapid growth to perform the experiments in a reasonable time. Each simulation required ~5 h of computer time, so the time required to run them substantially longer would have been prohibitive.

Binding energy

The first experiment compares results of the base simulation to one in which binding energies were altered by −1 kcal/mol and by −2 kcal/mol. The results are shown in Fig. 3.

Fig. 3 A shows the amount of total growth for each of the three simulations, including both on- and off-pathway growth. The graph shows growth in all three plots increasing at each time step, with higher rates of growth resulting from stronger binding interactions.
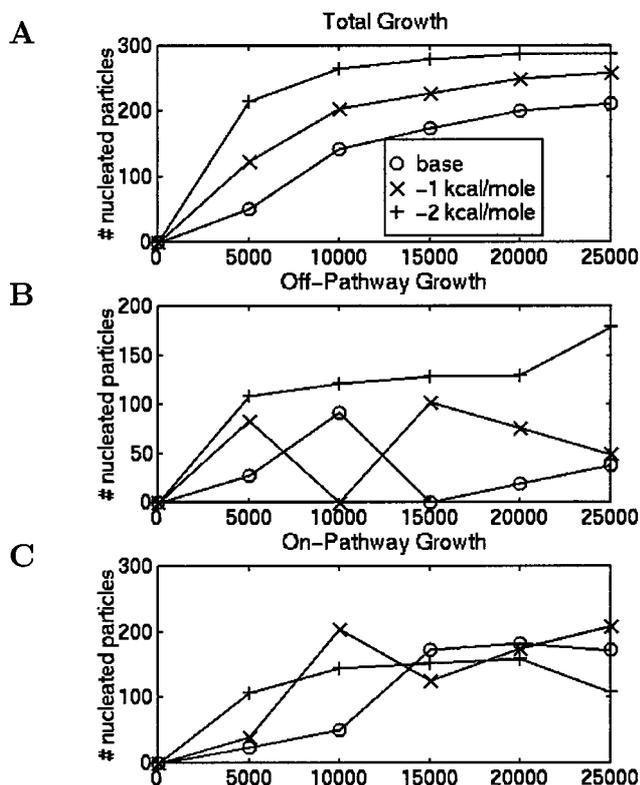

FIGURE 3 Plot of growth as binding energy is varied. These graphs show the numbers of subunits involved in nucleated clusters, in (A) total growth, (B) off-pathway growth, and (C) on-pathway growth. Each plot shows values for the base simulation and for simulations in which binding energies were altered by −1 kcal/mol and −2 kcal/mol. Data are presented at multiples of 5000 time steps.

Fig. 3 B shows the amount of off-pathway growth for the three simulations. With the exception of time step 10,000, stronger binding interactions again consistently produced more growth. The exception at time step 10,000 occurred because at that time the base simulation happened to experience a temporary increase in malformations that were corrected soon afterward, whereas the −1 kcal/mol simulation experienced a temporary reduction, as several malformations were corrected shortly before that time step. One feature to note is that the amount of off-pathway growth for a particular simulation sometimes decreases between two time steps; this is a result of malformations self-correcting during the simulations.

Fig. 3 C shows on-pathway growth for the three simulations. (This is equivalent to the difference between the values of the two previous plots.) As with off-pathway growth, values occasionally fell between two time points, typically as a result of a correctly formed cluster producing an incorrect bond. However, unlike the previous two graphs, this one does not show the same simple relationship between binding energies and growth rates. At various times, each of the three simulations had the highest total amount of growth. However, by time step 25,000 (the end of the simulation) the intermediate binding energy produced the greatest total amount of on-pathway growth. The other two plots reveal that although the −2 kcal/mol simulation had the most total growth, this was more than offset by its greatly elevated level of malformed growth relative to the other two simulations, giving it ultimately the lowest amount of on-pathway growth. Whereas the base simulation had the lowest amount of off-pathway growth, it also had the least total growth, yielding an intermediate amount of on-pathway growth. The −1 kcal/mol simulation, although it had neither the most total growth nor the least malformed growth, had the highest final amount of on-pathway growth. These plots thus suggest a trade-off between the competing factors of growth rate and malformation rate as functions of binding energy and indicate that producing the most on-pathway growth requires finding a balance between the two.

Binding tolerance

For the second experiment, binding tolerance, which roughly corresponds to the configurational entropy of binding, was increased by two and four times relative to the base simulation. The results are illustrated in Fig. 4. The results show a pattern similar to that described in the previous section, although with some differences.

Fig. 4 A shows the total amount of on- and off-pathway growth. Qualitatively, it shows the same relationship as that seen in Fig. 3 A, with monotonically increasing growth over time for each simulation. Increasingly lenient tolerances yielded increasingly rapid growth.

Fig. 4 B shows the amount of off-pathway growth for the three simulations. It also shows the same qualitative pattern as described in the previous section. Except for time step
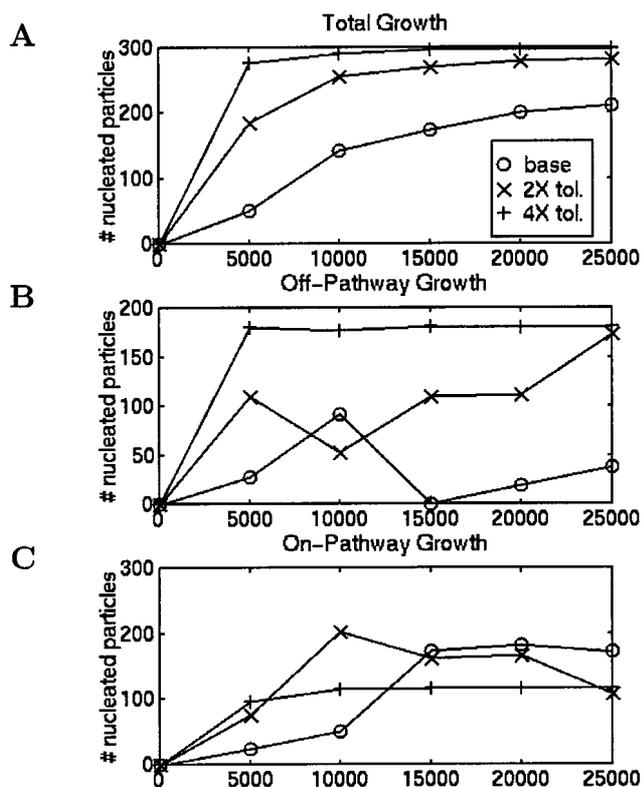
FIGURE 4   Plot of growth as binding tolerance is varied. These graphs show the numbers of subunits involved in nucleated clusters, in (*A*) total growth, (*B*) off-pathway growth, and (*C*) on-pathway growth. Each plot shows values for the base simulation and for simulations in which binding tolerances were increased by two times and four times.

10,000, increasingly lenient tolerances led to increasing off-pathway growth. As with Fig. 3 *B*, the exception at time step 10,000 was caused by the temporary increase in malformation in the base simulation combined with a temporary drop in the 2× tolerance simulation. One contrast to Fig. 3 *B* is that here, the amount of off-pathway growth for the intermediate simulation continues to grow in the final two time steps, yielding an intermediate value very close to the highest value rather than the lowest.

Fig. 4 *C* shows the amount of on-pathway growth for the three simulations. It is notable that despite the similarities between Figs. 3 *A* and 4 *A* and between Figs. 3 *B* and 4 *B*, Fig. 4 *C* is qualitatively very different from Fig. 3 *C*. Again, although all three simulations produce the highest total amounts of on-pathway growth at various times, here the base simulation ultimately yielded the highest amount of on-pathway growth. The double-tolerance simulation, despite its high total growth, also had high off-pathway growth; it therefore finished with the lowest amount of on-pathway growth, although it had neither the least amount of total growth nor the greatest amount of off-pathway growth. Although the quadruple-tolerance simulation had the highest amount of total growth, it also had the highest amount of off-pathway growth and thus yielded an intermediate amount of on-pathway growth. The base simulation,

despite having the least total growth, had considerably less off-pathway growth than the other simulations and thus finished with the most on-pathway growth of the three. This illustrates a trade-off similar to that in the previous section, in which there is a competition between growth and malformation rates as functions of binding tolerance, although the outcome of the competition was different from the outcome described in the previous section.

Concentration

The final simulations explored the effects of doubling and quadrupling the concentration relative to the base simulation, by reducing the volume of the simulated solution. Results are plotted in Fig. 5. The results show many qualitative similarities to the previous two experiments.

Fig. 5 *A* shows the total growth for the three simulations. Again, each simulation displays monotonic growth. In addition, increasing concentration corresponds to increased growth between simulations.

Fig. 5 *B* shows the amounts of off-pathway growth for the three simulations. At each point in time, the amount of off-pathway growth in the double-concentration simulation is at least as high as that in the base simulation. Furthermore, with the exception of time step 5000, the amount of
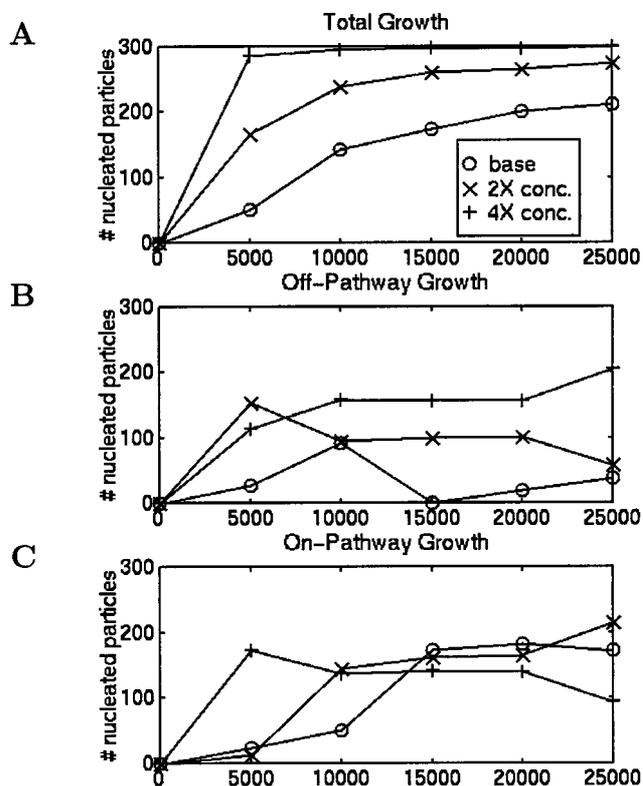

FIGURE 5   Plot of growth as concentration is varied. These graphs show the numbers of subunits involved in nucleated clusters, in (*A*) total growth, (*B*) off-pathway growth, and (*C*) on-pathway growth. Each plot shows values for the base simulation and for simulations in which concentrations were increased by two times and four times.

off-pathway growth in the quadruple-concentration simulation is at least as high as that in either of the other simulations. The exception at time step 5000 occurred because of a large number of malformations in the double-concentration simulation, most of which were corrected by time step 10,000.

Fig. 5 C shows the total amounts of growth for the three simulations. As with the previous two experiments, each of the three simulations has the most total growth at some point in time. By time step 25,000, the intermediate simulation yielded the greatest amount of on-pathway growth. As with varying the binding energies, the simulation with the greatest total growth also had the greatest malformed growth, resulting in the least on-pathway growth for the quadruple-concentration simulation. Furthermore, the base simulation had the smallest amount of off-pathway growth but also the least total growth, yielding an intermediate amount of on-pathway growth. The double-concentration simulation, although it had neither the most total growth nor the least malformed growth, finished with the most on-pathway growth of the three simulations. This again shows a trade-off between growth and malformation rates, in which maximizing on-pathway growth requires finding a parameter value that strikes a balance between the two.

Mechanisms of malformation rates

Although it is not always possible to determine the source of a malformation without saving and analyzing data for many time steps, sources can be determined for many malformations observed in the simulations. These data may be helpful in interpreting other results. Overall, three mechanisms were found to affect the number of malformations observed in a simulation.

One source of observed malformations was interactions of incomplete capsids. Occasionally, the growing edges of two partially formed capsids came sufficiently close to each other that it was possible to form a binding interaction between subunits in the two capsids. The two capsids thus become connected into a single malformed structure. Malformed structures of this type can continue to add subunits after they have formed. An example of such a malformation is shown in Fig. 6 A.

Another source of malformations is the development of small local errors in a growing capsid, which can lead to significant structural deformities. An example is shown in Fig. 6 B, in which a group of subunits that should have formed a pentamer have instead incorrectly bonded into a tetrameric structure. The capsid continued to grow after the malformation appeared, producing a closed structure in which all other binding interactions were locally consistent with a correct capsid, but in which the overall capsid was abnormally small.

A final factor affecting malformation rate is the ability of malformed capsids to correct themselves after a malformation has appeared. Fig. 6 C shows a local malformation similar to that shown in Fig. 6 B, in which subunits have formed a tetrameric structure where a pentamer would be expected. However, unlike in Fig. 6 B, the malformation in Fig. 6 C broke apart later in the simulation, allowing the insertion of another subunit to yield a correct pentamer. The cluster later went on to produce a correct, fully formed capsid. Malformations were frequently corrected in the simulations through similar local rearrangements of binding patterns. This was true not only for local errors, but also for errors resulting from interactions of partially formed capsids, which occasionally broke apart to again yield two on-pathway partial capsids.

DISCUSSION

We have presented a new technique for simulating self-assembly systems, described an implementation of that technique, and discussed the results of three sample simulation experiments. The technique allows computationally tractable simulations of the complex process of self-assembly of virus capsids from individual subunits. These simulations are configurable to different growth models and different values for a variety of subunit parameters. In addition, they allow the gathering of qualitative and quantitative information on the behavior of the simulations under different models.

The specific experiments described above were designed to illustrate some of the value of simulation models in general and of local-rules simulation in particular. The experiments involved changing various parameters of a physical system. Whereas some parameters, such as concentration, may be easily adjusted in a laboratory setting, others, such as binding energy or the entropic penalty of binding, may be difficult or impossible to alter with precision and without affecting other binding properties. However, in a simulation, parameters such as these can be easily and precisely adjusted in isolation from any other factors. Furthermore, the experiments involved examining low-level details of the experimental state at specific time intervals. Many aspects of the state of a laboratory experiment cannot be precisely examined by existing experimental techniques; these include the data on distributions of on- and off-pathway subunits examined in the present work. However, in a computer simulation, the complete state of the system is available to the user. In addition, it may not be possible to quench an actual biochemical reaction at a specific point in time without possibly altering its state. A simulation, though, can easily be frozen, examined, and resumed at any time. Purely theoretical work is unlikely to be a valid replacement for simulation work for applications such as this, which involve the interplay of many parameters in a complicated and incompletely understood system. For the reasons given above, the simulation experiments presented illustrate how simulation work may provide insights that cannot be found either through laboratory work or through purely theoretical analysis. Such insights, in turn, can sug-
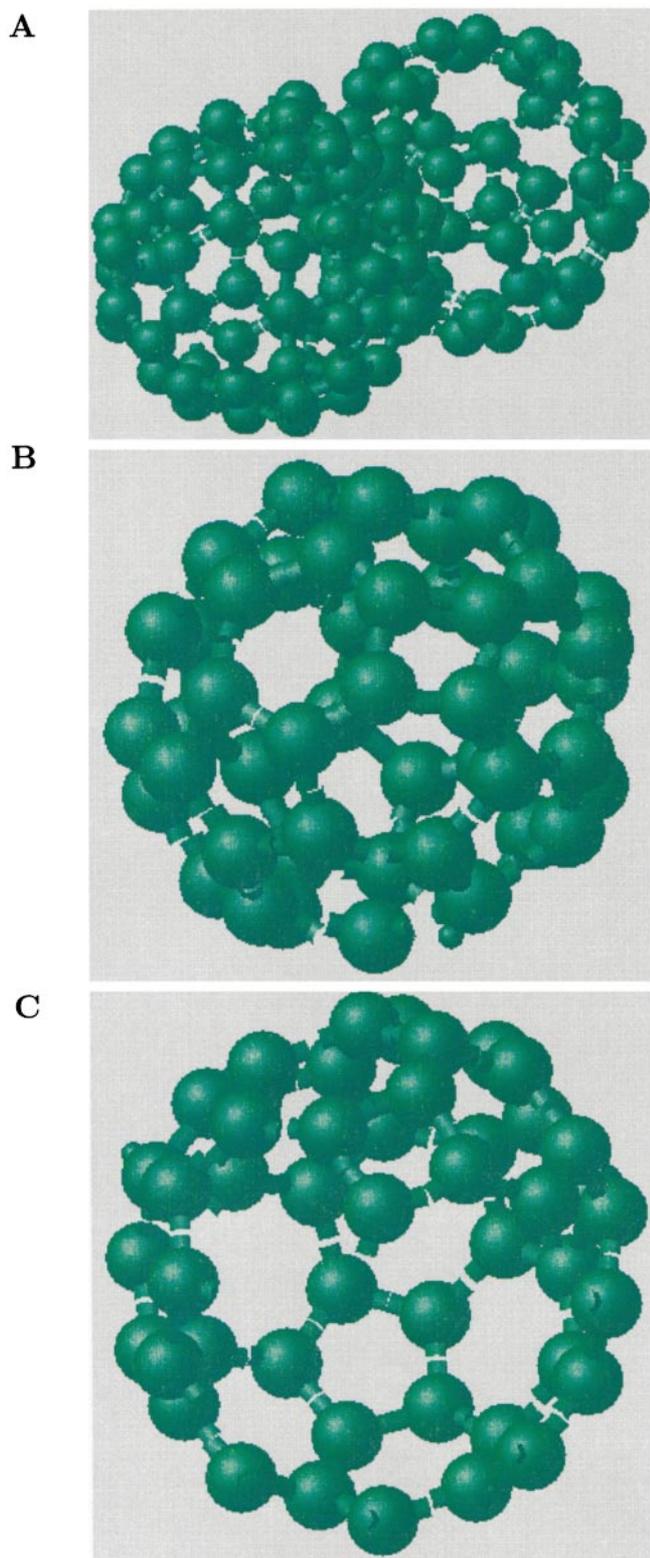
FIGURE 6 Demonstration of sources of increased malformations in simulations with elevated growth rates. (*A*) A simulated malformation produced when a binding interaction formed between the growing edges of two incomplete capsids. (*B*) A malformed capsid produced when a group of subunits incorrectly formed a tetrameric structure where they should have formed a pentamer. The capsid was able to continue growing to produce a closed particle with an incorrect size. (*C*) A malformation produced by local a binding error that is later corrected. The capsid
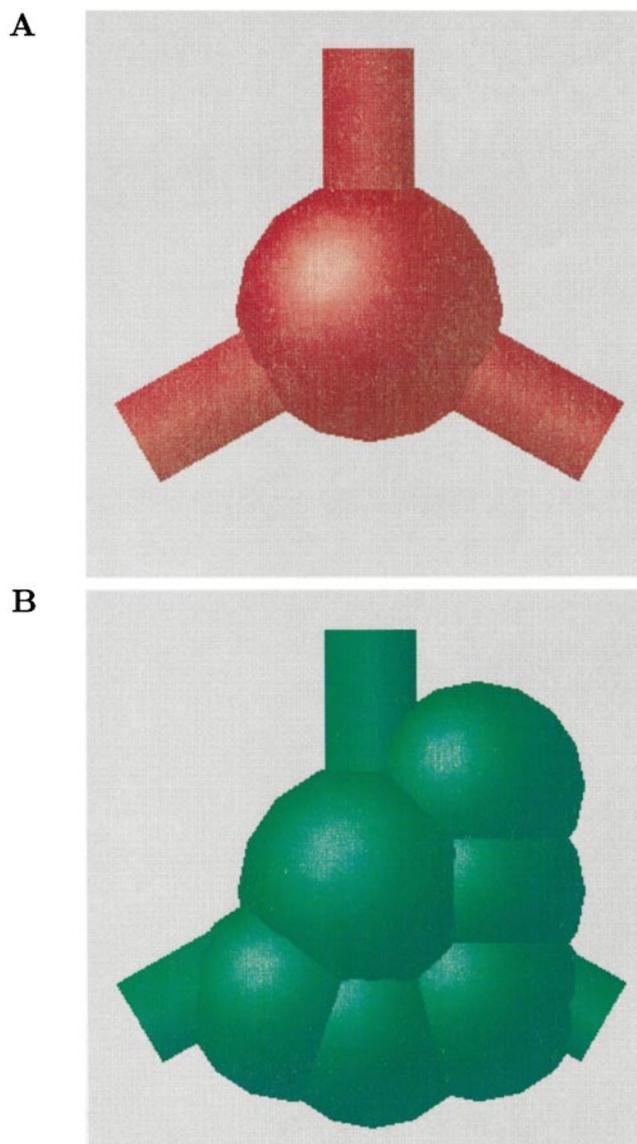

FIGURE 7 Two examples of simulated proteins. (*A*) A simple subunit model consisting of a single sphere with three edges projecting from it, representing binding sites. (*B*) A more complicated subunit built from a union of spheres. The separate spheres are treated as if they are rigidly connected for the purpose of calculating the effects on the entire subunit of forces acting on any part of it.

gest hypotheses that might themselves be independently confirmed in the laboratory.

Our particular experience suggests that the abstract local-rules framework can be useful for simplifying otherwise intractable problems involving the assembly of repeating or symmetrical structures. Virus capsids represent an excellent application of local-rules simulation due to the relative complexity of the problem and the consequent difficulty of

contains a tetramer in place of a pentamer, as in (*B*). However, the incorrect tetramer breaks apart later in the simulation and is corrected into a pentamer before the mistake can lead to larger structural errors. The partial shell goes on to form a correct, complete capsid.

creating predictive mathematical models. However, local-rules simulation may also be used for simulating other self-assembly systems. An earlier local-rules simulator has already been adapted to the study of carbon-silicon compounds (Hobbs et al., 1998). A particularly useful aspect of local-rules-based simulations that was demonstrated by the sample experiments is their ability to model malformations, which have proved difficult to analyze by both experimental techniques and purely mathematical models.

Although our central result is the simulator itself rather than our specific simulation experiments, we can also consider possible implications of those experiments. The experimental results suggest a trade-off between growth and malformation rates as functions of the parameters we examined. For all three parameters, modifications that resulted in increased growth rates also led to increased incidence of malformations. This implies that achieving a maximum number of on-pathway subunits required striking a balance between these competing factors. If these results accurately reflect the biological system they model, then they may have several implications. First, if the observed competition between growth and malformation rates is found in actual viruses, then this may provide an evolutionary argument for malformations observed in real virus growth. Specifically, if viruses can only suppress their malformation rate by sacrificing growth rate, then it may be that the optimum evolutionary strategy for a virus is to allow some small number of malformations in exchange for an accelerated growth rate over a similar virus that produces no malformed capsids. Such a trade-off might also have implications for the design of capsid assembly-targeted antiviral drugs. If virus capsids have evolved to find a balance between competing growth and malformation rates, then this suggests the possibly counterintuitive strategy of introducing agents that will promote capsid growth as a means of attacking that growth. For example, an agent that catalyzes capsid nucleation or stabilizes binding interactions might increase malformation rates more than enough to offset the increased growth rates it would produce and thus could be an effective antiviral.

The nature of malformations observed in the simulations suggests possible mechanisms by which changes in binding parameters of actual viruses may affect growth and malformation rates in ways similar to those described above. Although it was not possible to reliably determine the sources of all malformations observed in the simulations, the available data make it possible to propose reasons why changing parameters that influence growth rate might also affect rates of malformation. Increased binding energies might increase the probability of binding interactions occurring when two coat proteins are in close proximity, even if they are both already bound to separate partial capsids or to nonadjacent points on the same partial capsid. Furthermore, increased binding energies will stabilize such errors when they do occur. More lenient binding tolerances would allow coat proteins to bind in positions farther from optimal, potentially promoting local errors, and could yield a greater probability that a collision between partial capsids would

cause coat proteins on the two capsids to move within their binding tolerances of each other. More lenient binding tolerances might also stabilize errors, by ensuring that if an incorrect binding interaction is broken, the proteins involved will have a greater opportunity to reestablish that interaction. Finally, greater concentrations could increase the probability of collisions and the probability that a subunit bound in a slightly nonoptimal position will be able to form a new binding interaction that stabilizes that position before it can be corrected. Thus, if actual viruses exhibit mechanisms for producing and correcting errors similar to those observed in local-rules simulations, then there are theoretical arguments for proposing that the relationship between growth and malformation rates observed with local-rules simulations might hold for actual viruses as well.

Avenues for continuing this work include several areas in which our techniques can be applied and our model refined. Exploring models of kinetics for more complicated rule sets may allow examination of a wider range of virus growth properties. Simulations could also be improved by developing more realistic models of simulated proteins that better approximate known structures and measured physical properties as the relevant biochemical data become available and as increasing computer speeds reduce the run times of our simulations; more detailed models may assist in determining the dependence of assembly properties on those details. The approach described by Reddy et al. (1998) offers a possible means of deriving some of this data and may therefore provide a valuable complement to the simulation approach described in this paper. A potentially significant focus for future work is exploring the feasibility of different avenues for blocking or misdirecting capsid growth to assist in the development of capsid assembly targeted antivirals. This work will require studying the sources of naturally occurring malformations, the circumstances under which a malformed capsid can recover, and the dependence of these events on different binding properties, to gain clues to the aspects of binding an antiviral agent should affect to achieve maximum effectiveness.

REFERENCES

Berger, B., and P. W. Shor. 1995. Local rule switching mechanism for viral shell geometry. Technical Memo 527. MIT Laboratory for Computer Science.

Berger, B., P. W. Shor, L. Tucker-Kellog, and J. King. 1994. Local rule-based theory of virus shell assembly. *Proc. Natl. Acad. Sci. USA.* 91:7732–7736.

Blumofe, R., C. Joerg, B. Kuszmaul, C. Leiserson, K. Randall, and Y. Zhou. 1995. Cilk: an efficient multithreaded runtime system. *In* Proceedings of the Fifth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming.

Brooks, B., B. Buccoleri, D. Olafson, D. States, S. Swaminathan, and M. Karplus. 1983. CHARMM: a program for macromolecular energy, minimization and dynamics calculation. *J. Comp. Chem.* 4:187–217.

Caspar, D. L. D. 1980. Movement and self-control in protein assemblies: quasi-equivalence revisited. *Biophys J.* 232:103–138.

Caspar, D. L. D., and A. Klug. 1962. Physical principles in the construction of regular viruses. *Cold Spring Harb. Symp Quant. Biol.* 27:1–24.

Dokland, T., B. H. Lindqvist, and S. D. Fuller. 1992. Image reconstruction from cryoelectron micrographs reveals the morphopoietic mechanism in the P2–P4 bacteriophage system. *EMBO J.* 11:839–846.

Earnshaw, W., and J. King. 1978. Structure of phage P22 coat protein aggregates formed in the absence of the scaffolding protein. *J. Mol. Biol.* 126:721–747.

Galisteo, M. L., and J. King. 1993. Conformational transformations in the protein lattice of phage P22 procapsids. *Biophys. J.* 65:227–235.

Hobbs, L. W., C. E. Jesurum, V. Pulim, and B. Berger. 1998. Local topology of silica networks. *Philos. Magazine.* 78:679–711.

Horton, N., and M. Lewis. 1992. Calculation of the free energy of association for protein complexes. *Protein Sci.* 1:169–181.

Johnson, J. E., and J. A. Speir. 1997. Quasi-equivalent viruses: a paradigm for protein assemblies. *J. Mol. Biol.* 269:665–675.

Katsura, I. 1983. Structure and inherent properties of the bacteriophage lambda and head shell IV: small-head mutants. *J. Mol. Biol.* 171:297–317.

Marzec, C. J., and L. A. Day. 1993. Pattern formation in icosahedral virus capsids: the papova viruses and nudaurelia capensis β virus. *Biophys. J.* 65:2559–2577.

Prevelige, P. E., J. King, and J. L. Silva. 1994. Pressure denaturation of the bacteriophage P22 coat protein and its entropic stabilization in icosahedral shells. *Biophys. J.* 66:1631–1641.

Prevelige, P. E., D. Thomas, and J. King. 1993. Nucleation and growth phases in the polymerization of coat and scaffolding subunits into icosahedral procapsid shells. *Biophys. J.* 64:824–835.

Reddy, V. S., H. A. Giesing, R. T. Morton, A. Kumar, C. B. Post, C. L. Brooks, and J. E. Johnson. 1998. Energetics of quasiequivalence: computational analysis of protein-protein interactions in icosahedral viruses. *Biophys. J.* 74:546–558.

Rossmann, M. G. 1984. Constraints on the assembly of spherical virus particles. *Virology.* 134:1–13.

Steven, A. C., H. L. Greenstone, F. P. Booy, L. W. Black, and P. D. Ross. 1992. Conformational changes of a viral capsid protein: thermodynamic rationale for proteolytic regulation of bacteriophage T4 capsid expansion, co-operativity, and super-stabilization by Soc binding. *J. Mol. Biol.* 228:870–884.

Tarnai, T., Z. Gáspár, and L. Szalai. 1995. Pentagon packing models for "all-pentamer" virus structures. *Biophys. J.* 69:612–618.

Thuman-Commike, P. A., B. Greene, B. Malinski, J. A. King, and W. Chiu. 1998. Role of the scaffolding protein in P22 procapsid size determination suggested by $T = 4$ and $T = 7$ procapsid structures. *Biophys. J.* 74:559–568.

Zlotnick, A. 1994. To build a virus capsid: an equilibrium model of the self-assembly of polyhedral protein complexes. *J. Mol. Biol.* 241:59–67.