Harnessing microtubule dynamic instability for nanostructure assembly

Ann M. Bouchard,[1] Christina E. Warrender,[1] and Gordon C. Osbourn[2,*]
[1]*Physical, Chemical, and Nano Sciences Center, Sandia National Laboratories, P.O. Box 5800 MS 1423, Albuquerque, New Mexico 87185-1423, USA*
[2]*Complex Systems Science, Sandia National Laboratories, P.O. Box 5800 MS 1423, Albuquerque, New Mexico 87185-1423, USA*
(Received 29 March 2006; revised manuscript received 31 May 2006; published 3 October 2006)

Intracellular molecular machines synthesize molecules, tear apart others, transport materials, transform energy into different forms, and carry out a host of other coordinated processes. Many molecular processes have been shown to work outside of cells, and the idea of harnessing these molecular machines to build nanostructures is attractive. Two examples are microtubules and motor proteins, which aid cell movement, help determine cell shape and internal structure, and transport vesicles and organelles within the cell. These molecular machines work in a stochastic, noisy fashion: microtubules switch randomly between growing and shrinking in a process known as dynamic instability; motor protein movement along microtubules is randomly interrupted by the motor proteins falling off. A common strategy in attempting to gain control over these highly dynamic, stochastic processes is to eliminate some processes (e.g., work with stabilized microtubules) in order to focus on others (interaction of microtubules with motor proteins). In this paper, we illustrate a different strategy for building nanostructures, which, rather than attempting to control or eliminate some dynamic processes, uses them to advantage in building nanostructures. Specifically, using stochastic agent-based simulations, we show how the natural dynamic instability of microtubules can be harnessed in building nanostructures, and discuss strategies for ensuring that "unreliable" stochastic processes yield a robust outcome.

DOI: 10.1103/PhysRevE.74.041902 PACS number(s): 87.16.Ka, 81.16.Dn, 81.07.Nb, 87.16.Nn

I. INTRODUCTION

Nature makes extensive use of dynamic, nonequilibrium processes for assembling structures and materials from the nano- to the macroscale. Intracellular molecular machines work as "construction crews," synthesizing molecules, tearing apart others, transporting materials where needed, transforming energy into different forms, and a host of other coordinated processes, all dynamic, highly nonequilibrium, and all requiring energy to proceed. In particular, assembly and disassembly processes are coordinated, apparently for multiple purposes. One is to maintain order or structural integrity, by tearing down old parts and replacing them with new ones. Another is to use a scaffold (such as cartilage) as a template for a more permanent structure (bone) and then disassemble the scaffold. Another is to reuse components. For example, in a cell entering mitosis, the microtubules (MTs) normally used as tracks for cargo transport by motor proteins (MPs) are disassembled in order to use the parts (tubulin dimers) in the assembly of the mitotic spindle. When mitosis is complete, the mitotic spindle is disassembled in order to rebuild MT asters for active transport in both daughter cells [1,2]. Yet another purpose of coordinated assembly and disassembly is to dynamically grow and shrink individual structures in order to do work. For example, both polymerization and depolymerization of tubulin subunits are crucial for the functioning of MTs in the mitotic spindle, enabling them to provide force to separate the two sets of chromosomes into the two daughter cells. If either assembly or disassembly of MTs is prevented, the cell cannot divide and eventually dies [1,2].

Many cellular assembly and disassembly processes have been shown to work outside of cells. For example, under suitable conditions, and as long as they are supplied with energy, MTs can form asters on centrosomes [3,4] and undergo dynamic instability [3,5–7]. Similarly, motor proteins can carry cargo, such as pigment granules [8,9], drive stabilized microtubules [10], and help self-organize MTs into asters and spirals [11]. The idea of harnessing these molecular machines to actively build nanostructures is attractive [12]. However, the very behavior of these molecular machines that makes them attractive—their inherent dynamics—also presents a challenge. How does one guide them, turn them on and off at desired times or locations? How does one gain control over these stochastic processes designed for building biological systems, and direct them in a reliable way to build desired nonbiological systems?

A common strategy in attempting to gain this control is to eliminate some dynamic processes in order to focus on others. For example, in an inverted motility assay [10], kinesin motor proteins, which normally "walk" along microtubule filaments in cells, are surface tethered to a microscope slide so that their positions are fixed. The motor domains of the kinesin molecules protrude into the solution, so the interaction of these motor domains with microtubules can be investigated. In addition, by using stabilized microtubule filaments, the complexities of dynamic instability (the dynamic growth and collapse of MTs due to polymerization and depolymerization of tubulin subunits) are avoided. Instead, focus can be placed on strategies for aligning [13] or guiding [14] stabilized MTs as desired. Nonequilibrium assembly and transport of composite nanostructures has been demonstrated in inverted motility systems [15–18].

In this paper, we illustrate a different strategy for building nanostructures, which, as in Nature, harnesses the coordinated, stochastic assembly and disassembly processes of mo-

*Electronic address: gcosbou@sandia.gov

lecular machinery. In particular, we show how the natural dynamic instability of MTs can be used to advantage in building nanostructures, and discuss strategies for ensuring that stochastic processes yield a robust outcome. We describe stochastic agent-based simulations of MTs and motor proteins carrying cargo that illustrate several types of programmable dynamic self-assembly. The first example mimics the dynamic optical properties of melanophores (color-changing skin cells of fish and amphibians) [19]. The second simulation illustrates sorting and patterning of mixtures of two different nanoscale particle types. The third illustrates harvesting of nanoparticles from a mixture, and then delivering the harvested particles at a controlled time to a destination labeled by MT stabilizers. These are just a few examples demonstrating how we might coordinate stochastic assembly and disassembly processes to build nanostructures.

II. STOCHASTIC SIMULATIONS

Computer simulations are applied widely to test conceptual models of cellular processes and predict future behavior [20]. For example, the chemical networks controlling bacterial chemotaxis [21,22], developmental patterning in *Drosophila* [23,24], and infection of *Escherichia coli* by λ phage [25] have been modeled. Stochastic methods [26] are particularly valuable in representing chemical reactions involving relatively small numbers of reactant molecules when the continuous-variation approximation of differential equation methods breaks down.

We model the kinetics of protein interactions in a way that captures the statistical and temporal properties seen in experiments. A brief description follows; more details are provided in Sec. VII. Each molecule is represented by a collection of binding sites that can bind selectively to sites of other molecules. For example, a motor protein has sites to bind cargo, adenosine triphosphate (ATP), and a MT. Binding and dissociation events at such sites change the protein's state and affect subsequent interactions between proteins (through conformational changes that we do not model explicitly). A protein involved in one of these reactions can actuate, such as a motor protein taking a step along a MT, and/or expose, hide, or change the reaction rates of other binding sites.

Our simulation algorithm is similar to the Gibson-modified Gillespie algorithm [27]. To initialize a simulation, pairs of molecules with matching binding sites are selected for reaction. A time for each reaction is randomly selected from an exponential arrival-time distribution $P(t) \sim \exp(-t/\tau)$, where $P(t)$ is the probability of the reaction occurring at time t. Each reaction type has its own characteristic value of τ. The reactions are scheduled on a priority queue with the smallest reaction time t in root position. Each reaction is removed from the priority queue in order and executed. Each reaction may cause subsequent events on the priority queue to be added, modified, or deleted. This process is repeated until there are no more reactions on the queue or the user terminates the simulation.

III. MODEL COMPONENTS

Stochastic simulations can provide valid results only if the inputs—the models of the molecules, their reactions, and reaction times—are accurate at the level of resolution of interest. MT dynamic instability and MP motility are key processes in our simulations, and they depend on accurate models of MTs, MPs, and associated molecules. We developed our molecular models using experimental data from the literature wherever possible, and compared the collective behavior of the simulated molecules with experiments to ensure overall accuracy. The following two sections give an overview of these models; more details are available in Sec. VII.

A. Microtubules

We first describe MTs, their nucleating centers, and their stabilizing proteins. To set the context for comparison to our model, we briefly review here a current well-accepted model of MT growth and dynamics [1,2]. Protein dimers consisting of α- and β-tubulin stack together by noncovalent bonding to form the wall of a cylindrical microtubule. The cylinder is made from 13 parallel protofilaments, each a linear chain of alternating α- and β-tubulin. The microtubule has an overall structural polarity, with α-tubulin exposed at one end, designated as the minus end, and β-tubulin exposed at the plus end. In a living animal cell, the concentration of $\alpha\beta$-tubulin dimers is too low to drive the nucleation of the first ring of a MT. Instead, nucleating sites of γ-tubulin are provided by the centrosome, and $\alpha\beta$-tubulin can bind to and grow a MT from each of these nucleating sites. Each free $\alpha\beta$-tubulin is tightly bound to a guanosine triphosphate (GTP) molecule that is hydrolyzed to guanosine diphosphate (GDP) shortly after the tubulin is added to the growing MT. Whereas GTP is efficient at binding tubulin units together, GDP-tubulin units exhibit weaker intertubulin binding. If GTP-tubulin is added faster than the GTP in the MT is hydrolyzed, a cap of GTP-tubulin holds together the growing end, and the MT continues to grow for some time. However, if, due to the randomness of chemical processes, the GTP is hydrolyzed all the way to the end of the MT before new GTP-tubulin is added, the weakly interacting GDP-tubulin at the end will unravel the MT, often in a catastrophic manner. This alternating MT growth and collapse, due to the random nature of the race between adding GTP-tubulin and hydrolyzing the GTP to GDP, is termed *dynamic instability*. In living cells, dynamic instability is a natural and common behavior of microtubules. Dynamic instability can be suppressed, however, by MT-associated proteins (MAPs) that bind to the ends of MTs, stabilizing them against disassembly [28]. Such stabilized MTs serve as tracks for the transport of intracellular cargo by motor proteins, and help position organelles where needed by the cell.

In this work, we are interested in MTs for their usefulness in building nanostructures. We chose the resolution of our MT model to capture the overall behavior of MTs—their nucleation, growth, dynamic instability, and stabilization—without unnecessary (and computationally intensive) detail. Rather than treating each tubulin dimer as a separate molecule in the stochastic simulation, we model a unit length of MT (henceforth referred to as a "MT unit") as one object. Each free MT unit has a minus-end binding site ready to

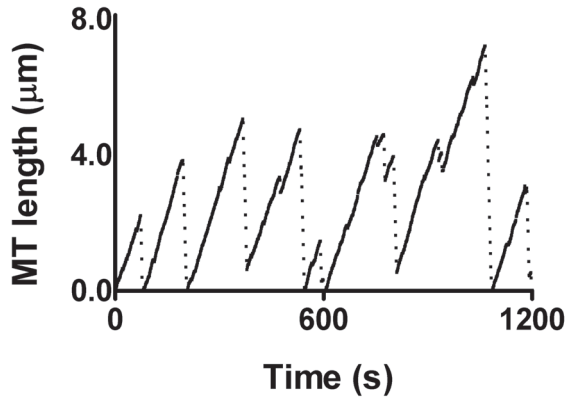

FIG. 1. Simulated microtubule dynamic instability: MT length is plotted as a function of time.

react, and a bound GTP molecule. When the minus end of a MT unit binds, an implicit conformation change exposes the plus end of the MT unit for binding to the minus end of a free MT unit, and the reaction is scheduled. In addition, a hydrolysis reaction of the GTP bound to the MT unit is scheduled. Repeated binding of MT units causes the MT to grow. However, if the GTP on the MT unit at the end of the MT is hydrolyzed, then an unbinding event (depolymerization reaction) of the minus end of the MT unit is scheduled. If the MT unit unbinds from the MT before the next free MT unit is added, the MT shortens.

These three reactions—the binding of a free MT unit to the growing MT, the GTP hydrolysis, and the depolymerization of the (GDP) MT unit from the MT—model the effect of the tubulin-MT interactions at a coarse resolution. By varying the reaction times, we can observe bounded or unbounded MT growth [29,30]. For the simulations that follow, we chose them to reproduce experimentally observed MT behavior [31]. Figure 1 shows a plot of the length of one simulated MT over time. The rates of MT growth and collapse and the frequencies of "catastrophes" (switching from growth to collapse) and "rescues" (switching from collapse to growth) are similar to those seen in experiments [31]. It should be noted that the dynamic instability is not modeled explicitly; it emerges as a result of the collective behavior of the agents in the simulation.

We must also model the molecules that interact with the MT units. A simulated centrosome (or generic MT nucleating center) consists of numerous sites that can bind the minus end of a MT unit. When binding occurs, one of a finite number of growth directions (e.g., 90 equally spaced directions in 360°) is randomly selected for the new MT. If there is already a MT bound to the centrosome growing in the selected direction, a new direction is randomly selected until a free direction is found. MTs grow in a two-dimensional plane about the centrosome. This situation reasonably models an experimental system outside of the cell, in which the depth of the solution above the slide is much smaller than the horizontal dimensions [32]. When a MT unit breaks away from a centrosome site (after total collapse of the MT), a new free GTP-associated MT unit is selected and scheduled for binding to the now-vacant centrosome site. Thus, when one MT collapses completely, a new one quickly becomes nucleated in its place, generally growing in a different direction. This is consistent with experimentally observed MT behavior.

"Stabilizers" model MT-capping proteins [28] and bind the plus end of a MT unit. If a stabilizer binds the end of a MT, then even if all of the MT units have been hydrolyzed, no depolymerization occurs. As long as the MT is bound to the stabilizer, it is stabilized.

The ability to effectively turn on and off centrosomes at will would be very useful in guiding the MTs to build reconfigurable patterns, as will become apparent in the final example in Sec. IV C. To incorporate this capability, we introduce into our simulations "blocker" molecules. (In experiments, anti-γ-tubulin antibodies [33] might function effectively as blockers.) Centrosome blockers bind at the MT-nucleating sites of a centrosome. Unlike MT units, they do not expose a new site to enable polymerization. They simply block the site, preventing the MT from binding. If centrosome blockers are active in the simulation, they compete with MT units for binding to available centrosome sites. Two event times are selected, one from the blocker probability distribution and one from the MT-unit distribution, but only the earlier of the two reactions is scheduled. Thus, if the blocker reaction rate is much faster than that of the MT unit, the blocker is more likely to bind and block the site.

One other feature of a MT unit is that, when it is part of a MT, a motor protein can bind to it in order to walk along the MT. This is described in the next section on motor proteins.

B. Motor proteins

We now describe motor proteins and their associated cargo [1,2], and our models of them. There are two major families of MT-associated motor proteins: kinesins, which usually walk toward the MT's plus end, and dyneins, which usually walk toward the minus end. MP processivity is better understood for kinesins than for dyneins, so we model our MPs on kinesins. These motor proteins have two globular ATP-binding heads that also bind to a MT, and a tail, which binds to cargo. The hydrolysis of ATP drives conformation changes resulting in the movement of one head at a time taking an 8 nm step (the length of one $\alpha\beta$-tubulin dimer) along the MT. The hydrolysis of ATP also leaves adenosine diphosphate (ADP) bound to the motor protein head; this must then be exchanged for a fresh ATP molecule to drive the next step. Except for a few short moments when the second head has bound and the first has not yet released, only one head is bound to the MT at a time. There is a repeated race between the reactions required to take a step (ADP-ATP exchange, ATP hydrolysis, and stepping) and the single bound head falling off the MT. As a result, experimentally measurable features of the motor protein behavior—run length [34], velocity [34–36], and randomness [35,36] (a measure of how close to time periodic the steps are)—are all functions of ATP concentration. When [ATP] is low, ATP arrival is slow, so on average the motor proteins move more slowly and less smoothly, and fall off earlier. When ATP is abundant, the motor proteins tend to move at their optimal speed. Even so, how far the motor protein runs before it falls

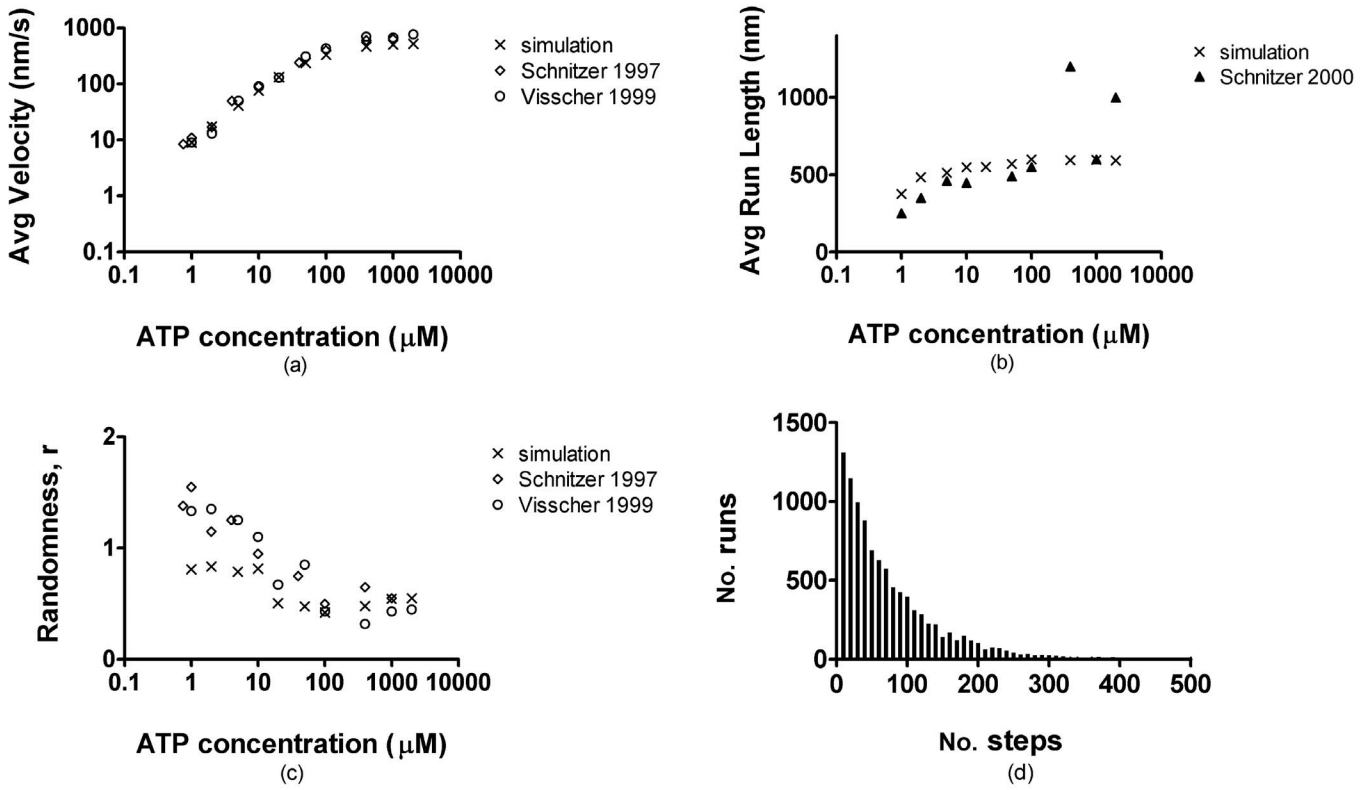

FIG. 2. (a)–(c) Motor protein behavior as a function of ATP concentration (500 runs at each concentration). Experimental data were replotted from [34–36]. (d) Distribution of simulated run lengths at 1 mM ATP (10 000 runs).

off the MT is stochastic, and follows an exponential distribution [37,38]. Even under optimal conditions, the motor is most likely to fall off after only a small number of steps.

Our model closely resembles the kinesin model of Schief and Howard [39], with parameter values taken from Howard [2]. We model the motor protein dimer as two identical molecules that are bound permanently to each other and to a piece of cargo. The cargo is a generic object, which could represent any item a motor protein could be functionalized to carry. For example, it could be a natural cargo, e.g., a vesicle or pigment granule, or something synthetic, such as a quantum dot or magnetic particle that has been chemically linked to the motor protein. Each head has additional sites for binding ATP or ADP and a MT. When included into the simulation, the cargo–motor-protein complex is placed randomly in the xy plane, and both motor protein heads are initially bound to ADP molecules. If a MT passes through the x,y location of the cargo–motor-protein complex, one of the heads is selected for binding to the MT, and the reaction is scheduled. When the head binds to the MT, it releases ADP so it can be replaced by an ATP molecule. When ATP binds to the head, it is hydrolyzed, and this drives the other head to bind to the MT one step length ahead of the bound head. Then the first head releases from the MT. As discussed above, most of the time is spent with only one head bound to the MT, so there is some probability that it will simply fall off. To model this, we schedule an unbinding event at the motor-protein–MT binding site. If the MT unbinding occurs before the step can take place, the motor protein falls off the MT. These model events are explained in more detail in Sec. VII C, along with parameter definitions and values.

To test the model, we ran large numbers of simulations of a motor protein walking on a MT at different ATP concentrations. We measured velocity, run length, and randomness, and compared these results to experiments published in the literature. Agreement was excellent (see Fig. 2 and [34–36]). (The scatter in average run length at high [ATP] seen in Fig. 2(b) is not discussed by Schnitzer *et al.* [34]. Our simulation results closely resemble the fit to the data shown in that work.) It should be noted that these features of the motor protein behavior (velocity, run length, and randomness) were not modeled explicitly. The overall behavior of the motor protein emerges in the simulation from the combination of complicated reactions going on at each of the motor's binding sites.

There are many different kinds of kinesin, each with different characteristics (for example, see Ref. [43]). In particular, motor velocity can vary significantly from one type to another. Also, although most kinesins move in the plus direction along the MT, there is a family of kinesins that move in the minus direction. We included two types of MPs in our simulations, one walking in the plus direction and one in the minus direction. They are modeled on *Drosophila melanogaster* KHC, and *Saccharomyces cerevisiae* Kar3, respectively; we scaled the model MP parameters to produce velocities matching those observed for these kinesins at an ATP concentration of 1 mM. The minus-walking motor is significantly slower than the plus-walking motor.

Tomishige and Vale [40] showed that it is possible to reversibly lock kinesin motors. They used disulfide cross

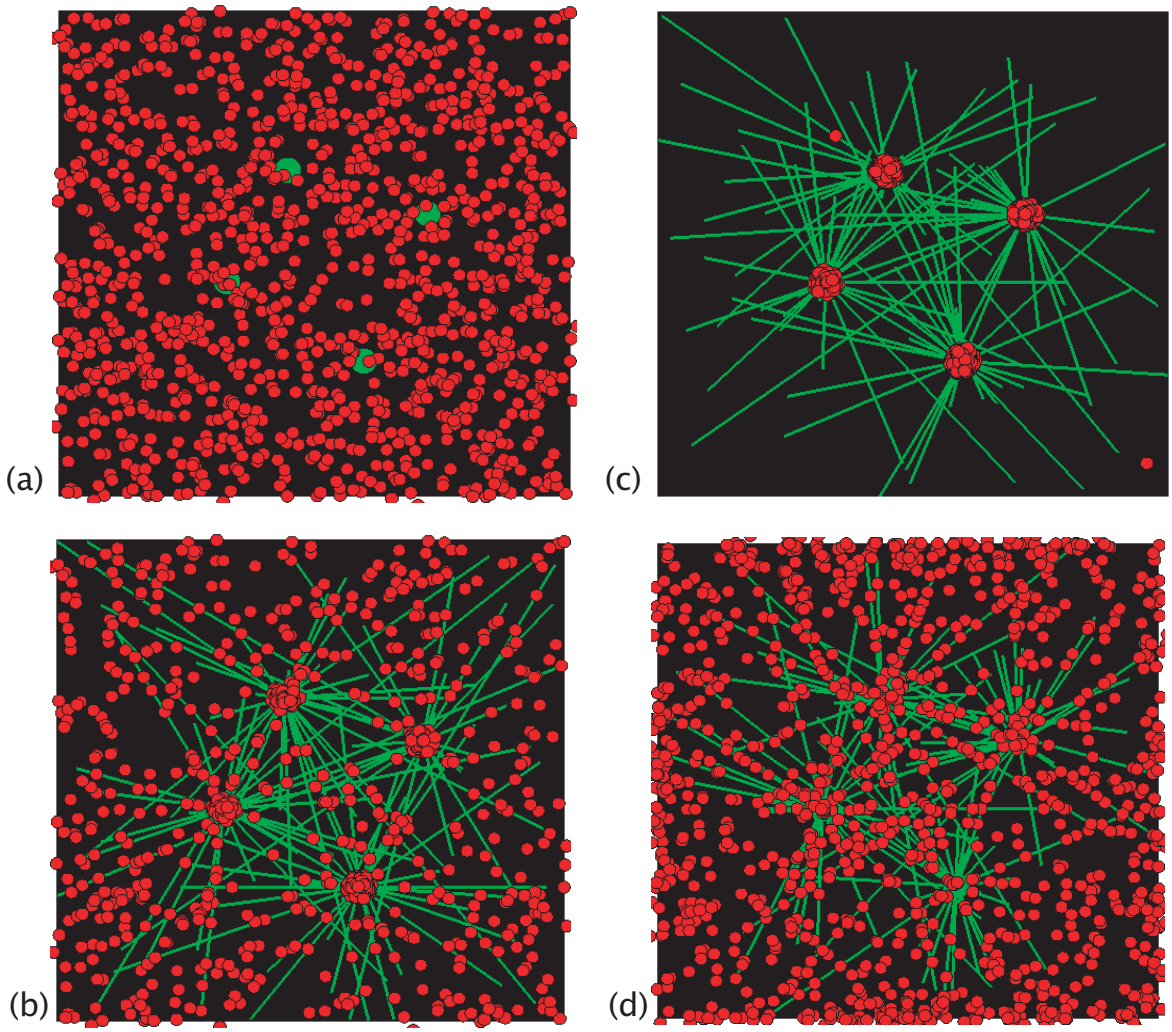

FIG. 3. (Color) Artificial melanophore simulation screenshots. (a) Initial configuration. Elapsed time (b) 100 s, (c) approximately 16 min, and (d) 20 min. The supplementary materials [41] contain four movies showing the beginning, collection phase, dispersal phase, and end of this simulation.

linking of cysteines in the MPs to test models of kinesin motility, but a similar mechanism could also be used to disable or enable cargo transport at will. We added this feature to our model by using a MP-blocker agent that prevents MP stepping when bound to the MP. As we will show in the following sections, this small amount of control over motor behavior enables us to guide cargo delivery to desired locations.

IV. SIMULATIONS

To investigate how the dynamic instability of MTs and cargo-carrying ability of motor proteins can be harnessed to build nanostructures, we mix together these model proteins in different combinations in our stochastic simulations. In all of these simulations, we use multiple centrosomes each supporting many MTs undergoing dynamic instability. The combination of MT dynamic instability and cargo transport by MPs along the MTs allows reversible assembly of many different types of structures. We assume an environment in which diffusion of unbound cargo–motor-protein complexes is negligible. Movies of all the simulations are available with the supplementary materials [41].

A. Artificial melanophore

In color-changing fish skin cells known as melanophores, pigment granules attached to motor proteins enable the cell to change color. One cellular signal drives motors to carry the pigment to the centrosome, leaving the body of the cell free of pigment, thereby making the cell transparent. A different cellular signal stimulates the motors to disperse the pigment, making the cell opaque [19]. In melanophores, three types of motors are involved: the MT-associated kinesin and dynein, as well as myosin, which moves along actin filaments. We show that it is possible to achieve a similar color-changing system with only inward- and outward-walking kinesin motors moving along MTs.

Figure 3(a) shows the initial configuration of the simulation. Several centrosomes (green) are distributed randomly

throughout the environment (black). Cargoes (red), representing pigment granules, are also randomly distributed. Each is bound to both inward- and outward-walking motors, but the outward-walking motors are initially locked.

When MT units are added, MTs nucleate on the centrosome and undergo dynamic instability. When a MT grows out to (or past) a cargo–motor-protein complex, the MT acts as a temporary bridge. The MP hops onto the MT and carries the cargo inward toward the centrosome. When the MT collapses, a new one is nucleated, generally in a different direction, and can act as another temporary bridge. As many MTs rapidly grow and collapse, many directions are sampled, so most of the cargo can be harvested to the centrosomes in a reasonably short time [Figs. 3(b) and 3(c)]. With the cargo condensed in this way, the net effect is to make the area largely transparent. The harvest time depends on the number of MTs and MPs. The simulation shown included a maximum of 30 MTs per centrosome and an average of ten cargo–motor protein complexes per MT; increasing the number of MTs per centrosome to 60 while keeping the number of cargo the same reduces the harvest time from 16 to 9 min. The numbers of MTs would be much larger [42], and the harvest time correspondingly faster, inside a cell.

When the MP blockers are released from the outward-walking motors, cargo moves in both directions along the available MTs. At any given instant, some cargoes are moving toward the centrosomes, some are moving away, and others not currently near MTs are stationary. The net effect is dispersal of the cargo, but in a dynamic fashion [Fig. 3(d) and movie in supplementary materials [41]]. This system is reversible; adding or reenabling the MP blockers again returns the system to the condensed state. Note that even the condensed state is not static; MPs continually bind to the MTs, but fall off again when they are unable to move.

B. Sorting and patterning

The second simulation was designed to illustrate sorting and patterning of mixtures of two nanoscale particle types. One particle type was coated with only outward-walking motor proteins and the other was coated with only inward-walking motor proteins before both types were introduced into the simulation space, creating an approximately uniform mixture. By controlling the placement of centrosomes, we can create areas that exclude one particle type and condense the other.

Figure 4(a) shows the initial disordered state. White particles have inward-walking motors and red particles have outward-walking motors. After about 10 min (simulation time), red particles are excluded from the area and decorate the boundary, while white particles are collected at the centrosomes along the midline, as shown in Fig. 4(b). The left boundary is cleared of particles because the position of the closest centrosome allows MTs to grow vertically along the boundary, providing a path for particles to be transported to the top and bottom boundaries.

C. Harvesting and delivering

The third simulation was designed to illustrate harvesting of nanoparticles and then delivering the harvested particles at a controlled time to a destination labeled by MT stabilizers. This required coating the particles with both inward- and outward-walking motors with switchable locks, and placement of centrosomes in the mixture region.

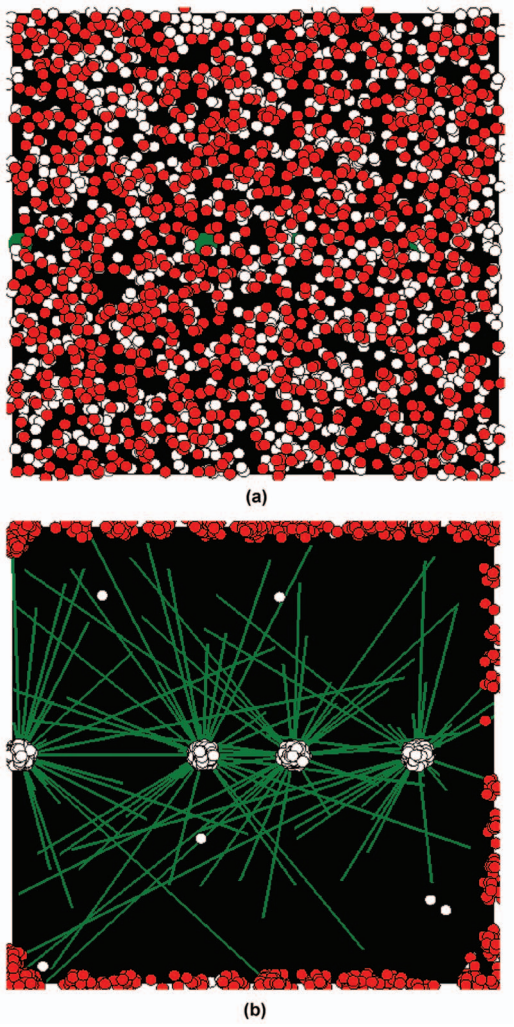

FIG. 4. (Color) Sorting and patterning simulation screenshots. (a) Initial configuration. (b) Elapsed time approximately 13 min. The supplementary materials [41] contain two movies showing the beginning and a later part of this simulation.

Initially, these simulations had particles (red) and centrosomes (green) in the upper portion of space, and MT stabilizers (yellow) along the bottom [Fig. 5(a)]. Particles have lockable inward- and outward-walking motors; initially outward motors are locked. They are harvested by MT dynamic instability in a manner similar to the artificial melanophore. In parallel with the cargo harvesting process, some of the MTs reach the stabilizers and become stabilized [Fig. 5(b)].

When the concentration of cargo at the centrosomes is sufficiently high and enough MTs are stabilized, this stimulates the release of a population of centrosome blockers. When a MT collapses, a blocker is more likely to bind to the centrosome than a new MT is. This eventually shuts off MT growth from the centrosome, because all available MT nucleating sites are blocked. In addition, MT growth is suppressed, so that collapse is far more likely than growth.

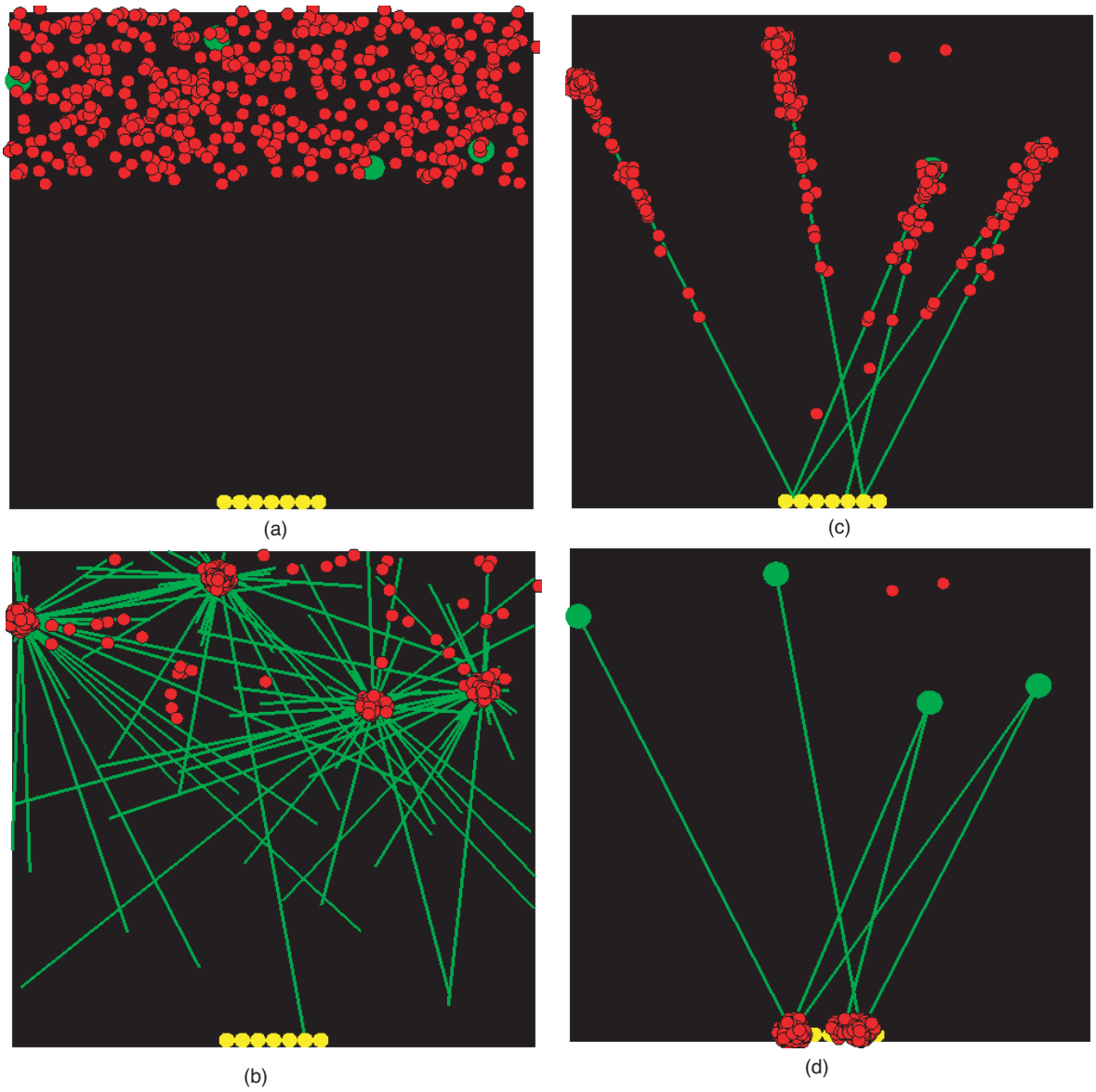

FIG. 5. (Color) Harvesting and delivering simulation screenshots. (a) Initial configuration. Elapsed time approximately (b) 2, (c) 5, and (d) 7 min. The supplementary materials [41] contain a movie of this entire simulation.

Eventually, the only remaining MTs are those in the bridge to the stabilizers [Fig. 5(c)].

When the concentration of centrosome blockers bound to the centrosomes exceeds a threshold, this releases the signal to lock the inward-walking motors and unlock the outward-walking motors. These motor proteins cross the bridge to deliver the particles to their destination at the stabilizers [Figs. 5(c) and 5(d)], building a "wall" of cargo over the original template of stabilizers. The cargo itself could be the nanostructure desired, or it could be used as a template for some other structure.

We should note that in the simulation just described, the molecules and complexes themselves carried or released the signaling molecules to control the sequence of assembly and disassembly processes. An alternative approach would be to manually include signaling molecules at the appropriate times. This was the approach taken in the first example, when specific blockers were activated or deactivated at desired times.

V. USING "UNRELIABLE" STOCHASTIC PROCESSES

Because these molecular processes are stochastic, in the simulation shown in Fig. 5(d), the cargo is not distributed uniformly along the stabilizers. Another simulation with a different random seed would lead to a slightly different result. It may be tempting to conclude that stochastic processes are too unreliable to be used as a general approach to building nanostructures. However, Nature relies on stochastic processes, and yet, is extremely robust. We examine some of Nature's strategies in dealing with stochastic processes, and see how they can be applied to nanostructure assembly.

One of Nature's strategies is to value function over form, so that many imperfect forms can successfully fulfill the re-

quired function. For example, every bird nest is unique and imperfect, with different numbers of twigs, different asymmetries, etc., but when completed, imperfect or not, it serves the function of holding eggs. A human's vascular system does not consist of perfectly straight or perfectly positioned blood vessels, but as long as the two vascular systems carrying blood to and from the heart join up, they serve the function of allowing blood to flow in a complete circuit. Whether or not the structure performs its intended function is the criterion for acceptance.

This strategy is relevant to nanostructure assembly. If one is attempting to build a conducting nanowire, it may not matter if the line of particles making up the wire is straight or bent occasionally, as long as there are no gaps. Alternatively, if the object under construction is structural in function, rather than electrical, imperfect placement of particles may be quite tolerable. In our first example, stochastic errors in the dispersal and condensation phases of the melanophore were irrelevant (i.e., even with errors it was close enough to serve the function of making the area transparent or opaque).

A second strategy is to cycle imperfect stochastic processes repeatedly until success is achieved. Living cells using the dynamic instability of MTs to "search" for stabilizing proteins is a perfect example of this strategy. Each MT repeatedly cycles through growth and collapse until it locks onto a stabilizing protein and becomes fixed. We can also apply this "try, try again" strategy at another level. By adding more cargo–motor-protein complexes, releasing the blockers from the centrosome, and repeating the entire process again, we are very likely to fill in the hole in our nanowall.

A third strategy is to wait for one stage of growth or development to be fully completed, passing a "checkpoint," before initiating the next step. The cell-cycle control system has checkpoints to ensure that cell division occurs only when conditions are right, and that it proceeds successfully [1]. We employed this strategy in the first example we discussed above, in the artificial melanophore. Our checkpoints relied on our own visual inspection of the particle distribution. When most particles had reached the centrosomes, then we manually released the MP blockers to allow particle dispersal. An experimentalist could employ this strategy as well, using microscopy or some other measurement to determine whether the conditions are right to go on to the next step.

VI. CONCLUSION

These simulations demonstrate multiple ways in which disordered and mixed nanoparticles can be ordered using energy-expending proteins. Nature uses dynamic, coordinated, stochastic assembly and disassembly processes to maintain order, build temporary structures, recycle components, and do work within biological systems. We have shown how such processes can be harnessed to assemble nonbiological nanostructures. Specifically, we showed how the natural dynamic instability of MTs could be used to assemble and reconfigure patterns, or to serve as dynamic bridges to enable motor proteins to move and deliver cargo to desired locations.

This approach is philosophically quite different from traditional engineering methods, where processes are made as controlled, deterministic, and reproducible as possible. Instead, it is more "hands off," allowing these molecular machines to act in the stochastic, "out of control" manner in which they are programmed, and relying on their collective behavior to yield the desired outcome. The key is to apply some degree of control at the right times, and let the natural behavior of these molecular machines do the rest of the work.

It is often impossible to predict the collective behavior that will result from a population of interacting components. We anticipate that the stochastic simulation capability we have developed will be a powerful tool for exploring what is possible to achieve with these molecular machines, and devising new schemes for building nanostructures in the future. We have emphasized the ability to visualize qualitatively different behaviors resulting from different combinations and geometries of molecular machines. The simulations can also be used to predict quantitative effects of varying individual parameters, such as the number of MTs able to simultaneously nucleate on a centrosome in Sec. IV A. The results presented here represent key components of a much broader set of programmable self-assembly processes that we will explore in the future.

VII. SIMULATION AND MODEL DETAILS

The following sections provide more information on our simulation method and our models of microtubule and motor protein dynamics.

A. Simulation infrastructure

Our simulations use software agents that are abstract representations of proteins. Each agent has one or more binding sites, each of which can bind to at most one other site at a time. Each site has a numeric key, and a state variable indicating whether it is hidden or exposed. An exposed site can bind only to a complementary exposed site—a site with a key of the same absolute value but opposite sign. Simulated bonding between agents is therefore as selective as it is between proteins. Agents are bound by setting their pointers to point to each other and unbound by clearing those pointers. Unmatched exposed sites wait passively until a complementary site becomes available.

Protein binding and unbinding often results in conformational changes that result in actuation (e.g., movement of motor proteins) and/or changes to other binding sites. We do not model the detailed physics and chemistry of these conformational changes. Instead, we directly model the properties of each agent that matter for protein assembly and disassembly. Each binding site has an "event handler" associated with each event type. During the execution of the event handlers, a number of things could happen. (a) Some physical actuation or a calculation could be performed. This could include calculating changes in agent position, size or other characteristics. (b) A binding site could be exposed. If an unbound exposed site with a complementary key is found,

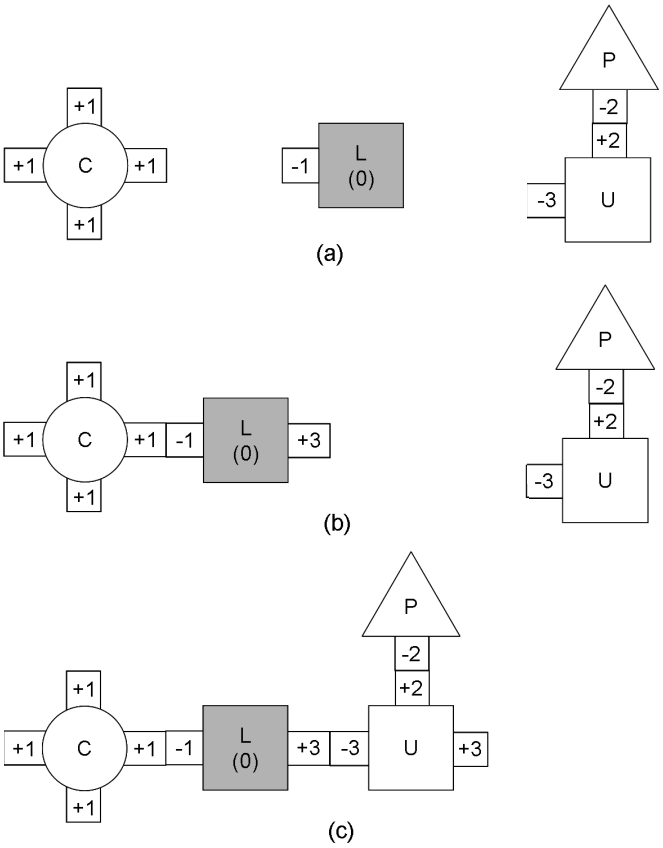

FIG. 6. Illustration of MT nucleation. (a) Centrosome (C), MT length (L, with current length in MT units in parentheses), MT unit (U), and phosphate (P) agents. Numeric keys identify binding sites. Two sites with keys of the same absolute value but opposite sign can bind to each other. The MT unit's (+2) site and phosphate's (−2) site are bound. (b) A MT-length agent (length=0) binds to the centrosome and exposes its (+3) site. (c) A phosphorylated MT unit binds to the MT-length agent and exposes its (+3) site, enabling further polymerization of the microtubule. (Agent sizes in this schematic do not correspond to the areas occupied by agents in the simulation screenshots.)

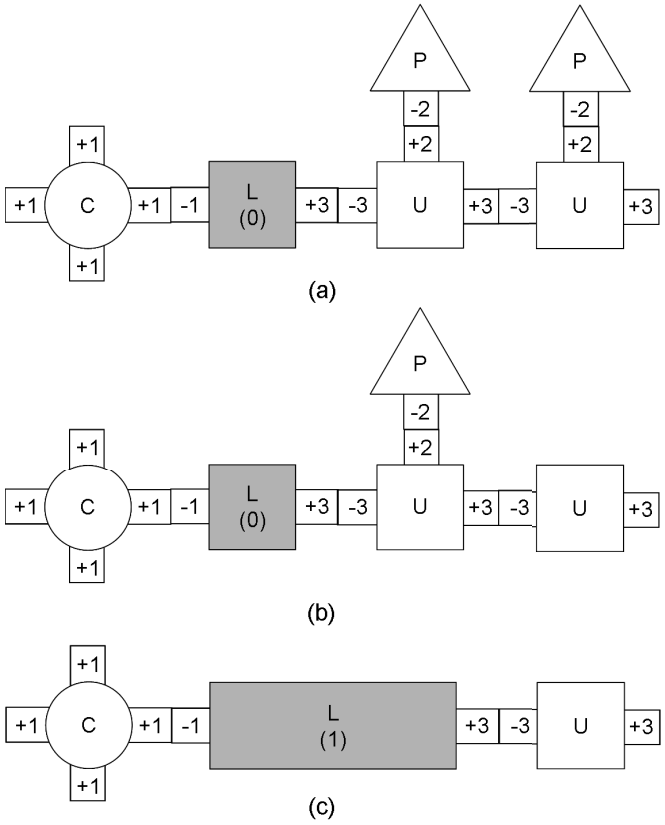

FIG. 7. Initial configuration (a) and hydrolysis of the rightmost (b) and then leftmost (c) MT-unit agents. All but the last hydrolyzed MT-unit agent are removed, and the length stored in the MT-length agent is updated to reflect the number of MT-unit agents removed.

a binding event is scheduled. If no complement is found, the site waits passively for one to become available. (c) A site could be hidden. If the site was bound, the bond is broken automatically. If the site was associated with a scheduled event, that event is canceled. (d) The key of a site could be changed. If the site was bound, the bond is broken automatically. If a binding event was scheduled, it is canceled. If the site is exposed, if a complement to the new key is available, a binding event is scheduled; otherwise, it waits for a complement to become available. (e) An unbinding event could be scheduled.

All of the "action" of the agent, then, is coded in the event handlers. Most of these events are associated with site binding or unbinding, but there are also regularly scheduled "proximity" events that check relative locations of agents and may take further action depending on those locations.

A time for each event is randomly selected from a distribution characteristic of that event type. Usually, this is an exponential arrival-time distribution:

$$P(t) = \frac{e^{-t/\tau}}{\tau}, \tag{1}$$

where $P(t)$ is the probability density function of the event time t, and the parameter τ is the mean event time. However,

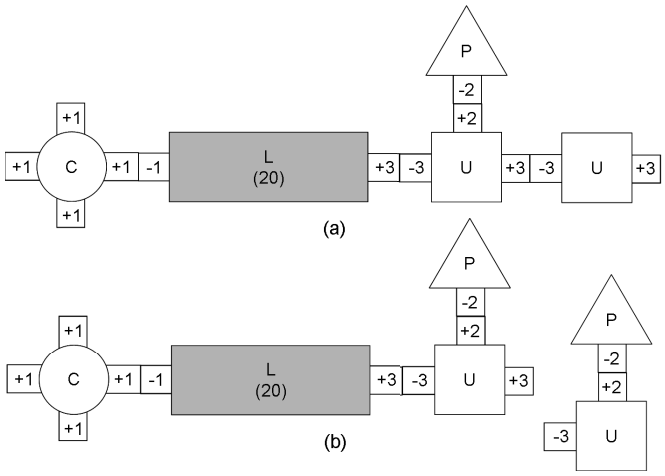

FIG. 8. MT shortening in the case of multiple MT units at the end of the MT-length agent. Initial configuration (a) and MT shortening (b). The MT-unit agent that has become unbound from the end is rephoshorylated to be ready to rebind to this or another MT.

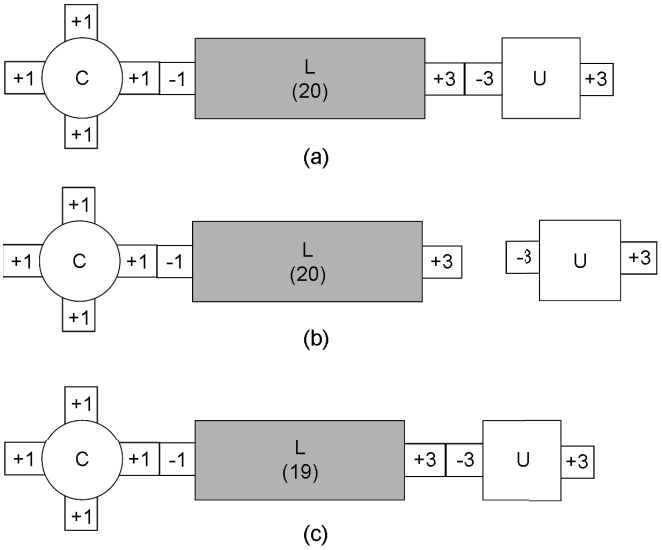

FIG. 9. MT shortening in the case of a single hydrolyzed MT unit at the end of the MT-length agent. (a) Initial configuration. (b) Initial unbinding of the MT unit from the MT length. (c) The length stored in the MT length agent is decremented, and the MT unit is rebound to the MT length, to preserve proper dynamic properties at the end of the microtubule.

unlike most events, proximity events are scheduled either at fixed intervals, or by selecting an event time from a narrow normal distribution centered on a fixed delay Δt. Scheduled events are placed in a priority queue, ordered by the event time. Sometimes an existing event may need to be rescheduled to reflect changing conditions, which is done simply by changing the event time (and the event's location in the priority queue). We do not explicitly distinguish between scheduling new events and rescheduling existing events in the descriptions below.

Each event is pulled off the priority queue and executed in order. For a binding or unbinding event, the two sites' pointers are set or cleared, respectively, and then each site's event handler is executed. In some cases, the two sites' event handlers must be executed in a particular order. In these cases, a flag indicates which event handler must be executed last. For a proximity event, only the associated site's event handler is executed. This process continues until there are no more events on the event queue, or the maximum desired time is reached.

B. Microtubule model

The MT dynamics in our simulations are the result of interactions between centrosome, MT length, MT unit, phosphate, and stabilizer agents. In this section, we describe both common sequences of interactions and the individual event handlers involved in those sequences. MT dynamics as implemented in our model are illustrated schematically in Figs. 6–10.

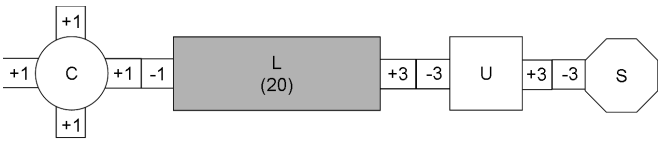

FIG. 10. MT stabilization.

TABLE I. Centrosome agent.

(+1) Site binding
Nothing.

(+1) Site unbinding
If centrosome blockers are active,
choose MT-length binding time;
choose blocker binding time;
schedule the earlier event.
Otherwise,
schedule MT-length binding.

The initial configurations of the agents and their binding sites with their associated numeric keys are shown in Fig. 6(a). A centrosome agent (circle, C) has multiple (+1) sites exposed and ready to bind to MT-length agents (shaded square, L, with the current length shown in parentheses) at their corresponding (−1) sites. MT-unit agents (square, U) each start out bound to phosphate agents (triangle, P); the complex (U-P) represents a GTP-bound MT unit. (Recall

TABLE II. MT-length agent.

(−1) Site binding
Pick a growth direction (length is 0).
Expose and schedule zero-time binding
for (+3) site.

(−1) Site unbinding
Nothing.

(+3) Site binding
Save the address of the bound MT unit so it
can be rebound on events that reduce the
length.

(+3) Site unbinding
If the length is 0,
hide this site;
schedule unbinding for (−1) site.
Otherwise,
for any MPs bound to (+4) sites,
DropCargo (see Table XI);
decrement length;
rebind MT unit at (+3) site;
schedule unbinding for (+3) site.

(+4) Site binding
Nothing.

(+4) Site unbinding
Nothing.

TABLE III. MT-unit agent.

(−3) Site binding
If MT unit has reached a stabilizer,
bind the MT-unit (+3) site to the stablizer (−3) site.
Otherwise, if microtubule has room to grow,
expose and schedule normal binding for (+3) site.
Schedule unbinding for (+2) site (hydrolysis).

(−3) Site unbinding (excute after (+3) unbinding)
For any MPs bound to (+4) sites,
DropCargo (see Table XI).
If (−3) site is bound (this means it was rebound to MT-length agent during execution of its (+3) site unbinding event handler),
If (+2) site is bound,
schedule normal binding for (+3) site.
Otherwise,
schedule slow binding for (+3) site.
Otherwise,
hide (+3) site [cancel (+3) site events];
bind (+2) site to phosphate (−2) site now;
leave (−3) site exposed.

(+3) Site binding
If (−3) site has an unbinding event,
cancel it.

(+3) Site unbinding
If the (+2) site is not bound,
schedule unbinding for (−3) site;
if microtuble has room to grow,
schedule slow binding for (+3) site.
Otherwise,
if microtuble has room to grow,
schedule normal binding for (+3) site.

(+2) Site binding
Nothing.

(+2) Site unbinding (hydrolysis)
If there are contiguous hydrolyzed MT units bound to the MT-length agent,
remove the hydrolyzed MT-unit agents but leave one MT-unit agent at the end;
update the length in the MT-length agent to include the MT-units removed.
If (+3) site is not bound,
If microtubule has room to grow,
schedule slow binding for (+3) site;
Schedule unbinding for (−3) site.

(+4) Site binding
Nothing.

(+4) Site unbinding
Nothing.

TABLE IV. Stabilizer agent.

(−3) Site binding
Nothing.

(−3) Site unbinding
(Never occurs in these simulations)

that a MT unit represents a ring of tubulin dimers, not a single tubulin dimer, and that the rate of hydrolysis was therefore chosen to reproduce experimentally observed microtubule dynamic instability behavior.) The MT unit's (−3) site is exposed and ready to bind.

MT nucleation is a two-step process (Fig. 6). First, a MT-length agent binds to a centrosome agent, which causes the former to expose its (+3) site. This MT-length agent initially is a zero-length placeholder; it chooses the direction in which the microtubule will grow. Second, the MT length's (+3) site binds to the (−3) site of any available U-P complex, which exposes its (+3) site in turn. [Note that the MT-length and MT-unit agents both have sites with the same key (+3), but these sites have different event handlers, as described below.] This last step [Fig. 6(c)] is repeated many times for a growing microtubule. However, blocker agents (not shown) also have (−1) sites that can compete with the MT length (−1) sites for binding to the centrosome's (+1) sites.

When a U-P complex binds at its (−3) site, a hydrolysis event—an unbinding at the (+2) site—is scheduled (Fig. 7). Note that, because event times are stochastic, MT-unit agents do not necessarily hydrolyze in the order in which they were bound to the microtubule. In Fig. 7, the last MT unit to bind is hydrolyzed first. In general, however, there will be more phosphate-bound MT units near the plus end of the microtubule and more hydrolyzed MT units closer to the centrosome. For computational reasons, contiguous chains of hydrolyzed MT-unit agents adjacent to the centrosome are represented by the MT-length agent. In this way, microtubules several hundreds of MT units long can still be represented by just a few agents. In Fig. 7, when the second MT-unit agent is hydrolyzed, it is removed, and the length stored in the MT-length agent is changed to reflect the number of MT units removed. Any adjacent hydrolyzed MT-unit agents are also removed at the same time, except the final agent at the microtubule's plus end. One MT-unit agent is always kept at the end of the microtubule whether it has been hydrolyzed or not in order to preserve correct binding and unbinding dynamics.

If the MT unit at the end of the MT has been hydrolyzed, it may either bind a new MT unit at its (+3) site or unbind

TABLE V. Blocker agent.

(−1) Site binding
Nothing.

(−1) Site unbinding
(Never occurs in these simulations)

TABLE VI. Centrosome proximity event.

```
If #cargos at centrosomes > threshold1
AND #stabilized MTs > threshold2,
    change binding times to inhibit MT growth;
    introduce blockers into the simulation.
If #blockers bound to centrosome > threshold3,
    unlock +motors;
    lock —motors.
Schedule next centrosome proximity event.
```

from the rest of microtubule at its (−3) site. Both events are scheduled; the earliest event is executed first and cancels the later event. If a MT-unit agent detaches from another MT-unit agent at the end of the microtubule (Fig. 8), the detached MT unit hides its (+3) site and exposes its (+2) site to rebind a phosphate agent and return to its initial state. If a MT unit detaches from the MT-length unit (shown in Fig. 9), the length stored in the MT-length agent is decremented, and the previously bound MT unit is rebound to preserve the proper microtubule configuration. However, if the length in the MT-length agent was already 0, then the unbinding at its (+3) site represents complete collapse of the microtubule. In that case, the MT-length agent unbinds from the centrosome. Then the centrosome's exposed (+1) site can renucleate another microtubule.

A MT-unit agent with an exposed (+3) site may also bind a stabilizer agent if their positions are close enough (Fig. 10). A MT unit's position is calculated from the centrosome position, MT-length agent direction and length, and the number of MT-unit agents in the microtubule. A hydrolyzed MT unit bound to a stabilizer will not unbind from the rest of the microtubule [i.e., no unbinding event is scheduled at its (−3) site].

The simulation code does not specify complete sequences such as those illustrated in the last few paragraphs, but individual event handlers associated with each type of binding site. Pseudocode for each of the event handlers modeling microtubule dynamics is given in Tables I–VI. The average times used to schedule new events are shown in Table VII.

In addition to the sites described above that are involved in the MT dynamics, the MT-length and MT-unit agents also have sites for binding MPs. Since one MT-unit agent represents a ring of tubulin dimers, multiple MPs could be walking on it simultaneously, each one bound to a different tubulin dimer in the ring. If we assume cargo of $1-5\ \mu$m in diameter, we estimate that ten cargo could fit around the tubulin ring. Each MT-unit agent therefore has ten (+4) sites (not shown in the figures), making it capable of binding ten different MP agents at a time. The MT-length agent represents many MT units [3] worth of microtubule length. Each MT-length agent has 20 pairs of (+4) sites. A pair of MP agents will bind alternately one then the other of the pair of (+4) sites, as it walks processively along the MT-length agent. Each time the MP takes a step, its position is recomputed, to simulate moving physically along the MT-length agent. When it reaches the end, it steps onto the (+4) site of the adjacent MT-unit agent, then onto the next MT-unit agent, etc. The MP agent (−4) site event handlers, described in the next section, contain all the relevant computation. The (+4) sites of the MT length and MT unit are simply stepping stones for the MP. For completeness, their very simple event handlers are included as pseudocode as well.

TABLE VII. Average event times for MT dynamics. These values were used to produce the plot shown in Fig. 1 and throughout most of the simulations. To simulate inhibition of MT growth, the normal and slow MT-unit (+3) site binding times were multiplied by 10. Individual event times are chosen from an exponential distribution [Eq. (1)] using the value shown for τ.

Event	Mean time τ (s)
Centrosome (+1), MT length (−1) binding	1.0
Centrosome (+1), blocker (−1) binding	0.01
Centrosome (+1) unbinding	1.0
MT unit (+3) normal binding	0.2
MT unit (+3) slow binding	6.0
MT unit (+2) unbinding (hydrolysis)	1.17
MT unit (−3) unbinding	0.02

A proximity event associated with a centrosome agent is used to simulate molecular sequencing control in the harvesting and delivering simulation. This event controls the MT growth inhibition and changes in MP motor direction described for that simulation. (MP dynamics are explained in the next section). These events are scheduled at regular 5 s

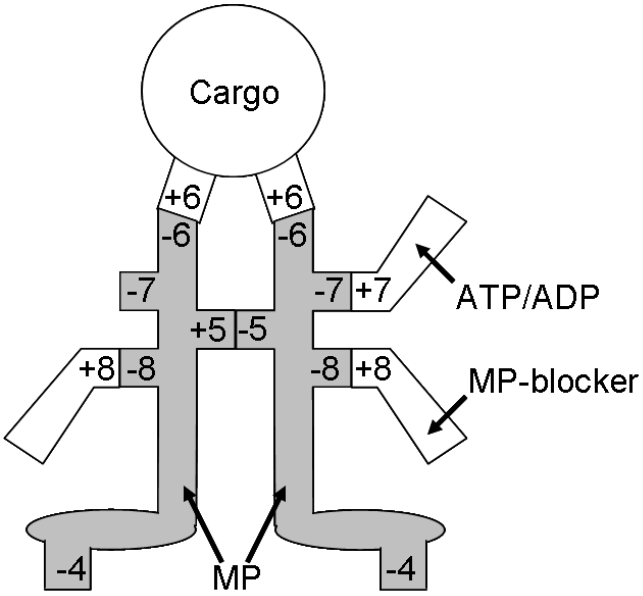

FIG. 11. Schematic of a cargo complex with one plus-walking motor protein pair: cargo, motor protein (MP), adenosine tri- or di-phosphate (ATP/ADP), and motor protein blocker (MP blocker) agents (not drawn to scale; in reality the cargo is orders of magnitude larger than all the other molecules). The MP (±5) sites are permanently bound, as are the (±6) sites between MPs and cargo. The MP (−4) sites (MP heads) bind to MT-length or MT-unit (+4) sites.

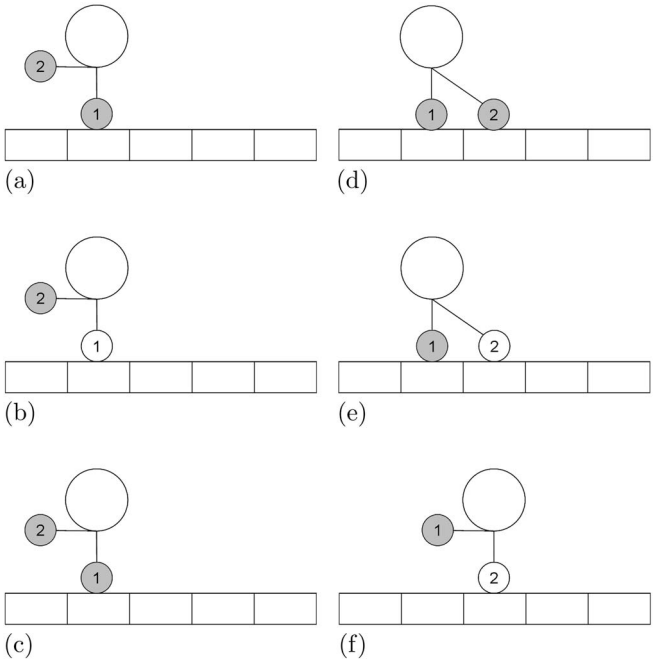

FIG. 12. Schematic of MP motility. Cargo (large circle) is bound to one pair of MPs (small circles). MT is shown as a segmented rectangle, with each rectangle the length of one MT unit (8 nm). Shading on a MP indicates it is bound to ADP. (a) MP1 binds the MT. (b) MP1 releases ADP. (c) MP1 rebinds ATP and immediately hydrolyzes it. (d) MP2 binds to MT. (e) MP2 releases ADP. (f) MP1 unbinds from MT and cargo moves one step. The cycle repeats as in (c)–(f).

intervals, and for this simulation only, the event handler shown in Table VI is added to the centrosome agent.

C. Motor protein model

Cargo movement is the result of motility of the motor protein pair (or pairs) to which the cargo is bound. Each motor protein is represented by a MP agent. Adenosine triphosphate, upon binding to a MP, is rapidly hydrolyzed to adenosine diphosphate. That is, ATP binds, then a phosphate is released, leaving ADP bound to the MP. We represent both forms by a single ATP/ADP agent. MP agents each have five binding sites (see Fig. 11): (−4) binds a MT-length or MT-unit agent, (+5) or (−5) binds the partner MP agent, (−6) binds the cargo, (−7) binds an ATP/ADP agent, and (−8) binds a MP blocker. The (±5) and (−6) site bindings are permanent, so event handlers are not shown for these sites. Binding of a MP-blocker agent at the MP's (−8) site prevents the MP from taking a step. Otherwise, MP motility depends on binding and unbinding of the (−4) and (−7) sites. Interactions between MPs and MTs depend on their relative positions as well as which binding sites are available. Figure 12 illustrates one cycle in our model of MP movement along a MT (binding sites are not shown in this figure to focus attention on the spatial sequence).

Cargo agents are always bound to at least one pair of MP agents. Initially, these MPs are both bound to ADP and not bound to a MT. When a MT (either a MT-length or MT-unit agent) is close enough to a cargo complex, one of the MPs may bind to it [Fig. 12(a)]. MP1-MT binding catalyzes ADP unbinding from MP1 [Fig. 12(b)]. At this point, there are two events scheduled for MP1; it may detach from the MT or recapture ATP. ATP binding to MP1 [Fig. 12(c)] catalyzes binding between MP2 and the MT, and the immediate hydrolysis to ADP catalyzes the unbinding of MP1 from the MT. [We combine both of these catalysis effects into a single event handler, when the MP binds an ATP/ADP agent at the (−7) site.] The most likely next event is the binding between MP2 and the MT [Fig. 12(d)]. Again, MP2-MT binding catalyzes ADP unbinding from MP2 [Fig. 12(e)]. This in turn catalyzes MP1 detachment from the MT. When MP1 unbinds from the MT, the cargo also moves one step [Fig. 12(f)]. The cargo complex is now in a similar configuration to that of Fig. 12(b) and the cycle may repeat as in Figs. 12(c)–12(f).

However, at each step in this cycle, multiple events are possible. Each MP-ATP binding event triggers a subsequent unbinding event (and vice versa) that may occur at a different time than that shown in Fig. 12. ATP concentration,

TABLE VIII. Cargo agent.

```
Proximity event: single MP pair
  If cargo's MPs not bound to MT
  AND cargo is near a MT
  AND a (+4) site is available to bind to;
    schedule binding for (−4) site;
    schedule cargo proximity event at
    same time as (−4) binding event.
  Otherwise,
    schedule cargo proximity event from
    normal distribution.

Proximity event: one winner
  If cargo's MPs not bound to MT
  AND cargo is near a MT
  AND (+4) site is available to bind to,
    choose (−4) binding time for +MP;
    choose (−4) binding time for −MP;
    schedule the earlier event;
    schedule cargo proximity event at
    same time as (−4) binding event.
  Otherwise,
    schedule cargo proximity event from
    normal distribution.

Proximity event: tug of war
  If cargo is near a MT,
    If +MP pair not bound
    AND a (+4) site is available to bind to,
      schedule binding for +MP (−4) site.
    If −MP pair not bound
    AND a (+4) site is available to bind to,
      schedule binding for −MP (−4) site.
    Schedule proximity event at same time
    as earliest scheduled (−4) event.
  Otherwise,
    schedule cargo proximity event from
    normal distribution.
```

TABLE IX. MP agent.

```
(−4) Site binding
  If (−7) site bound,
    schedule fast unbinding for (−7) site.
  Schedule slow unbinding for (−4) site.

(−4) Site unbinding
  If partner-MP (−4) site is bound to MT,
    If partner-MP is in front
    AND MP pair is AbleToMove (see Table XI),
      move cargo one MT unit forward.
    If partner-MP (−7) site is bound to ADP
    AND MP pair is AbleToMove (see Table XI),
      If there is a (+4) site to bind to
      (i.e., not at end of microtubule),
        schedule binding for this MP's
        (−4) site.
      Otherwise,
        DropCargo (see Table XI).
  Otherwise (this MP pair has fallen off MT),
    If no other MP pair is bound to MT,
      DropCargo (see Table XI).
    Otherwise,
      If binding scheduled for partner-MP
      (−4) site,
        cancel it.

(−7) Site binding
  Schedule slow unbinding for (−7) site.
  If (−4) site is bound,
    schedule medium unbinding for (−4) site;
    If partner-MP (−4) site not bound
    AND no binding even is scheduled for it
    AND MP pair is AbleToMove (see Table XI),
      If there is a (+4) site to bind to
      (i.e., not at end of microtubule),
        schedule binding for parftner-MP
        (−4) site.
      Otherwise,
        DropCargo (see Table XI).

(−7) Site unbinding
  If both MPs (−4) sites are bound,
    schedule fast unbinding for partner-MP
    (−4) site.
  Schedule binding for (−7) site.
```

which affects the rate of MP-ATP binding events, may change the "preferred" sequence of events. Also, a MP bound to a MT may fall off at any point in the cycle or may be dropped when the MT shortens.

In addition, motors may be locked by the addition of MP-blocker agents that bind to corresponding sites on the MP agents. If a MP is bound to a MP blocker, then ATP binding by a MP bound to the MT will not trigger MT binding by the second MP (physically, this represents the second MP not being able to reach the next binding site because of the MP lock). We use two types of MP blockers distinguished by different numeric keys on their MP-binding sites. For plus-walking motors, the MP-blocker key is (+8), as shown in Fig. 11. For minus-walking motors, the MP-blocker key is (+9), and the MP has a corresponding (−9) site in place of the (−8) site.

TABLE X. Average event times for MP motility. [ATP] represents the concentration of ATP (μM). Most values were obtained from experiments (references given); one parameter was fitted to match experimental run lengths. These values were used unscaled to produce Fig. 2; they were multiplied by 0.63 to simulate the outward-walking kinesin KHC and by 17 to simulate the inward-walking kinesin Kar3. Individual event times are chosen from an exponential distribution [Eq. (1)] using the value shown for τ.

Event	Mean time τ (s)	Reference
Initial (−4) binding	0.01	[2], Chap. 14
Subsequent (−4) binding	0.0026	Fit
Fast (−4) unbinding	0.001	[2], Chap. 14
Medium (−4) unbinding	0.26	[2], Chap. 14
Slow (−4) unbinding	100.0	[2], Chap. 14
Fast (−7) unbinding	0.01	[2], Chap. 14
Slow (−7) unbinding	100.0	[2], Chap. 14
(−7) binding	$(1.1[\mathrm{ATP}])^{-1}$	[35]

As was the case for the MT dynamics, so too the simulation code does not specify complete MP sequences as just described, but individual event handlers associated with each type of binding site. Pseudocode for each of the event handlers modeling MP motility is given in Tables VIII and IX. The average times used to schedule new events are shown in Table X.

The sequence illustrated above applies when a single MP pair is active in each cargo complex. Some of our simulations use two MP pairs, one trying to move in the plus direction and the other in the minus direction. We have implemented two models for how different MP types interact. In the "one-winner" model, the first MP to bind to a MT deter-

TABLE XI. Functions used by event handlers.

```
AbleToMove:  single MP pair
  This MP is not bound to MP blocker.

AbleToMove:  one winner
  This MP is not bound to MP blocker.

AbleToMove:  tug of war
  This MP is not bound to MP blocker
  AND other MP pair is not bound to MT.

DropCargo
  Cancel scheduled cargo proximity event.
  Cancel any scheduled events for all of
  this complex's MP and ADP sites.
  Unbind any bound (−4) sites of this
  complex's MPs.
  Bind ATP/ADP agents to all this complex's
  MP (−7) sites.
  Schedule next cargo proximity event from
  normal distribution.
```

mines the direction in which the cargo complex will move; the second MP pair is unable to bind to the MT. In the "tug-of-war" model, the second MP pair may bind to the MT, but no movement is possible when both pairs are bound. A run in one direction may be interrupted when the opposite MP pair binds to the MT; movement in either direction may resume when one MP pair detaches from the MT. Note that, even if a MP is locked by a MP blocker, it can still bind to a MT in both the one-winner and tug-of-war models. In either case, the cargo complex can only move if the unlocked MP pair alone is bound to the MT. The one-winner and tug-of-war models result in the same qualitative behavior in all our simulations, but cargo movement is generally slower in the tug-of-war model.

In experiments, particles may be coated with hundreds or even thousands of MPs, and the exact number is not easily controlled. Our simplified models are not intended to capture the complex details of a large number of interacting MPs. Rather, the intent is to shed light on the expected limiting cases. At one extreme, all the plus or all the minus motors "win." At the other extreme, the forces exerted by the plus and minus motors exactly cancel out. Motility observed in experiments is expected to be somewhere between these two extremes (perhaps with increased run lengths for multiple same-direction motors). The fact that our simulation results show the same qualitative behavior for both extremes leads us to expect that the same qualitative behavior would be observed in experiments.

The different models are implemented by different versions of two event handlers. In the MP (−4) unbinding event, different methods are used to determine whether a MP pair is able to move (see Table XI). In the cargo proximity event (Table VIII), different criteria determine how initial binding of the cargo complex to the microtubule is scheduled. For a cargo complex not currently walking along a MT, cargo proximity event times are selected from a Gaussian distribution with a mean of 1 s and a standard deviation of 0.01. For a cargo complex already moving along a MT, the next proximity event is scheduled immediately after the next MP-MT binding event (in case the MP-MT binding event gets canceled or the MP falls off the MT).

ACKNOWLEDGMENTS

We thank George Bachand and Bruce Bunker for useful discussion and suggestions. This work was supported by the Division of Materials Sciences and Engineering, Office of Basic Energy Science, U.S. Department of Energy. Sandia National Laboratories is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract No. DE-AC04-94AL85000.

[1] B. Alberts, D. Bray, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter, *Essential Cell Biology* (Garland, New York, 1998).
[2] J. Howard, *Mechanics of Motor Proteins and the Cytoskeleton* (Sinauer, Sunderland, MA, 2001).
[3] T. Mitchison and M. Kirschner, Nature (London) **312**, 232 (1984).
[4] R. E. Palazzo, J. B. Brawley, and L. I. Rebhun, Zoolog Sci. **5**, 603 (1988).
[5] T. Mitchison and M. Kirschner, Nature (London) **312**, 237 (1984).
[6] T. Horio and H. Hotani, Nature (London) **321**, 605 (1986).
[7] T. J. Mitchison, J. Cell Biol. **109**, 637 (1989).
[8] S. L. Rogers, I. S. Tint, P. C. Fanapour, and V. I. Gelfand, Proc. Natl. Acad. Sci. U.S.A. **94**, 3720 (1997).
[9] H. Nilsson, W. Steffen, and R. E. Palazzo, Cell Motil. Cytoskeleton **48**, 1 (2001).
[10] J. Howard, A. Hunt, and S. Baek, Methods Cell Biol. **39**, 137 (1993).
[11] T. Surrey, F. Nédélec, S. Leibler, and E. Karsenti, Science **292**, 1167 (2001).
[12] H. Hess and G. D. Bachand, Mater. Today **8**, 22 (2005).
[13] L. Limberis, J. J. Magda, and R. J. Stewart, Nano Lett. **1**, 277 (2001).
[14] J. Clemmens, H. Hess, R. Lipscomb, Y. Hanein, K. F. Böhringer, C. M. Matzke, G. D. Bachand, B. C. Bunker, and V. Vogel, Langmuir **19**, 10967 (2003).
[15] G. D. Bachand, S. B. Rivera, A. K. Boal, J. Gaudioso, J. Liu, and B. C. Bunker, Nano Lett. **4**, 817 (2004).
[16] M. Bachand, A. M. Trent, B. C. Bunker, and G. D. Bachand, J. Nanosci. Nanotechnol. **5**, 718 (2005).
[17] H. Hess, J. Clemmens, C. Brunner, R. Doot, S. Luna, K.-H. Ernst, and V. Vogel, Nano Lett. **5**, 629 (2005).
[18] A. K. Boal, G. D. Bachand, S. B. Rivera, and B. C. Bunker, Nanotechnology **17**, 349 (2006).
[19] H. N. Sköld, S. Aspengren, and M. Wallin, Microsc. Res. Tech. **58**, 464 (2002).
[20] D. Endy and R. Brent, Nature (London) **409**, 391 (2001).
[21] U. Alon, M. G. Surette, N. Barkai, and S. Leibler, Nature (London) **397**, 168 (1999).
[22] D. Bray, M. D. Levin, and C. J. Morton-Firth, Nature (London) **393**, 85 (1998).
[23] Z. J. Burstein, J. Theor. Biol. **174**, 1 (1995).
[24] G. Marnellos and E. Mjolsness, Pac. Symp. Biocomput **3**, 30 (1998).
[25] H. H. McAdams and L. Shapiro, Science **269**, 650 (1995).
[26] D. T. Gillespie, J. Comput. Phys. **22**, 403 (1976).
[27] M. A. Gibson and J. Bruck, J. Phys. Chem. A **104**, 1876 (2000).
[28] N. Galjart, Nat. Rev. Mol. Cell Biol. **6**, 487 (2005).
[29] M. Dogterom and S. Leibler, Phys. Rev. Lett. **70**, 1347 (1993).
[30] M. Dogterom and B. Yurke, Phys. Rev. Lett. **81**, 485 (1998).
[31] D. K. Fygenson, E. Braun, and A. Libchaber, Phys. Rev. E **50**, 1579 (1994).
[32] T. E. Holy, M. Dogterom, B. Yurke, and S. Leibler, Proc. Natl. Acad. Sci. U.S.A. **94**, 6228 (1997).

[33] B. R. Oakley, C. E. Oakley, Y. Yoon, and M. K. Jung, Cell **61**, 1289 (1990).
[34] M. J. Schnitzer, K. Visscher, and S. M. Block, Nat. Rev. Mol. Cell Biol. **2**, 718 (2000).
[35] M. J. Schnitzer and S. M. Block, Nature (London) **388**, 386 (1997).
[36] K. Visscher, M. J. Schnitzer, and S. M. Block, Nature (London) **400**, 184 (1999).
[37] S. M. Block, L. S. B. Goldstein, and B. J. Schnapp, Nature (London) **348**, 348 (1990).
[38] R. D. Vale, T. Funatsu, D. W. Pierce, L. Romberg, Y. Harada, and T. Yanagida, Nature (London) **380**, 451 (1996).
[39] W. R. Schief and J. Howard, Curr. Opin. Cell Biol. **13**, 19 (2001).
[40] M. Tomishige and R. D. Vale, J. Cell Biol. **151**, 1081 (2000).
[41] See EPAPS Document No. E-PLEEE8-74-025609 for model details and simulation movies. For more information EPAPS, see http://www.aip.org/pubservs/epaps.html.
[42] M. Schliwa, U. Euteneuer, W. Herzog, and K. Weber, J. Cell Biol. **83**, 623 (1979).
[43] See http://www.proweb.org/kinesin/