Tensegrity representation of microtubule objects using unified particle objects and springs

Arif Pramudwiutmoko¹,2, Gregory Gutmann¹, Yutaka Ueno³, Akira Kakugo⁴,⁵, Masayuki Yamamura¹, Akihiko Konagaya¹*

¹School of Computing, Department of Computer Science, Tokyo Institute of Technology, 4259 Nagatsuta-cho, Midori-ku, Yokohama, Kanagawa, 226-8503, Japan
²Universitas Teknologi Yogyakarta, Jl. Siliwangi (Ringroad Utara), Jombor, Sleman, D.I. Yogyakarta, 55285, Indonesia
³Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology, 2-4-7 Aomi, Koto-ku, Tokyo 135-0064, Japan
⁴Faculty of Science, Hokkaido University, Sapporo 060-0810, Japan
⁵Graduate School of Chemical Sciences and Engineering, Hokkaido University, Sapporo 060-0810, Japan

*E-mail: konagaya.a.aa@m.titech.ac.jp

(Received March 30, 2020; accepted June 28, 2020; published online July 17, 2020)

Abstract

There are limitations in interactions with molecular objects in laboratory experiments due to the very small size of the objects. Common media to show the experimental results of molecular objects is still lack of observer interaction to understand it intuitively. In order to overcome this lack of interaction, this research takes tensegrity representation of molecular objects reproducing experimental results and creates interactive 3D objects to be presented in a virtual reality (VR) environment. The tensegrity representation enables us to enhance the interaction experience with the natural user interface with haptic technology and hand tracking controller. A particle simulation system that utilizes multiple GPUs resources is used to fulfill haptic VR requirements. We developed a unified particle object model using springs and particles which we call anchors which act as tensegrity structure of the object to support conformation of filament-type objects such as microtubules. Some object parameters can be set to match the flexural rigidity of the object with some experimental results. The bending shape of the object is evaluated using the classic bending equation and the results show high compatibility. Viscoelastic behavior also shows similarities with the viscosity reported in other studies. The object's flexural rigidity can be adjusted to match the target value with the direction of the prediction equation. The object model provides a better insight about molecular objects with natural and real-time interactions to provide a more intuitive understanding with the molecular objects presented. The results show that this model can also be applied to any filament-type or rod-like molecular object.
1. Introduction

The mechanics of molecular objects often become a very important property in most molecular systems such as molecular robotics and molecular artificial muscle [1, 2]. Microfilaments (actin filaments), intermediate filaments, and microtubules form the cytoskeleton which perform a multitude of functions in the cellular process. Together they form cellular tensegrity that determines the shape and mechanical resistance of cells [3, 4]. Microtubules also act as tension rods to segregate chromosomes during cellular division. Microtubule is also an important component in biomolecular motor system. Microtubules mediated by a rod-like DNA origami to form an aster structure in the artificial smooth muscle model. A collection of these microtubule asters in the presence of kinesins exhibits fast and dynamic contraction through an energy dissipative process [2].

Measuring the mechanics of molecular object has been studied in many past researches. Flexural rigidity is one of the most studied property because of its significant effect in the system process. For example, the measurement of flexural rigidity of microtubules has been conducted for over a quarter of century using various tools and methods [5], which occasionally yield different results over a wide range of two orders of magnitude. The results differ not only because of various microtubule-associated proteins (MAPs) and different number of protofilaments inside the microtubules. Other selected factors also contribute to it, e.g., static/dynamic process, analysis process, type of working force observed, balance and direction of force, and number of force fulcrums [6].

Conventional ways to show a biomolecular system is by showing photographs or movies. However, it is difficult to get an intuitive insight of the system due to the absence of the observer interaction with it. Interacting with a real molecular object is difficult due to its very tiny size. The object must be treated in certain condition to conserve it and special tools are needed to interact with it, which is not a convenient way to observe. Instead of using the real molecular object, another possible way to understand the mechanical behavior of the object is to create a virtual 3D object using the object’s data and observe it in the simulation. Virtual reality (VR) and haptic technology have the potential to realize it and to provide an intuitive understanding through natural hands interaction.

In order to provide an intuitive hand interaction in the VR environment with molecular elastic objects such as microtubules, we propose tensegrity representation of a microtubule object with a unified particle object of outer anchor particles and inner molecular particles connected by springs, called an anchored spring model in this paper. We refer to the outer particles as anchors because each molecular particle in the object is anchored to several of the outer particles. The formation of the anchors and springs supports the tensegrity of molecular particles which gives shape and rigidity to the molecular object. Several parameters of the object determine its flexural rigidity. How to tune them becomes an objective in this research. Regarding the different measurement results of flexural rigidity from past researches, our research is not in the position of justifying them. Instead, our anchored spring model is intended to give intuitive haptic interface when showing
molecular objects in haptic VR simulation. The haptic VR interface used in this research was reported in a previous study [7, 8].

VR simulation requires to perform at least 90 graphics frame per second (FPS) to avoid user’s motion sickness. The number of frames doubles if we consider the need to provide different view for left and right eyes. Molecular processes can consist of thousands or more objects which will overload the VR system. In order to overcome this obstacle, a robust computing system must be used. Among several alternatives, particle simulation system using multiple Graphics Processing Units (GPUs) resources is the best candidate [7, 8]. Such 3D particle simulation system has been implemented in microtubule gliding assay simulation that utilize multiple GPUs to provide performance power to simulate up to millions of simulated particles [9–11]. The 3D simulation system in this research is derived from the particle simulation system reported in those studies.

All objects in 3D particle simulation system are represented by a collection of particles, that is, a coarse-grained particle model object in order to avoid computational overhead necessary for VR representation. Note that a 10 micrometers microtubule object consists of about 109 million atoms. Considering the massive number of calculations needed in the simulation, atomic microtubule models are difficult to represent with the current VR technologies, although it is a big challenge issue from the viewpoint of high-performance computing.

This paper is organized as follows. First, we explain background knowledge necessary to understand this paper in section 2. Section 3 describes methods, system overview, and model design. We present the results in section 4 followed by discussion in section 5. Finally, we have conclusions and descriptions of our future work in section 6.

2. Background

2.1 Microtubules

Microtubules are hollow tube molecular polymers made from alpha and beta tubulin which are part of the cytoskeleton. They provide structure and shape for eukaryotic cells that give cell shape and keep the organelles in place. Each tubulin has a diameter of 4 nm and a mass of 50 kDa (8.302695333×10^23 kg) [12]. Alpha tubulin and beta tubulin are strongly connected to form heterodimers. The heterodimers polymerize end-to-end to form linear rows of tubulin dimers called protofilaments. Protofilaments are connected laterally to form hollow tubes, which are microtubules. Microtubules are the largest structures in the cytoskeleton having an outer diameter of 23 to 27 nm [13] and an inner diameter of 11 to 15 nm [14]. Their most common form consists of 13 protofilaments of lateral alpha and beta tubulin lattices in a tubular arrangement.

Microtubules have a very important role in numerous cellular processes, including cell movement, cell division, cell transportation, etc. Some artificial molecular systems also depend on the rigidity of microtubules to carry out several processes in the system. Microtubules mostly work individually or in small groups. Therefore, understanding the individual mechanical properties is important.

2.2 Flexural rigidity

The mechanics of microtubules have been the subject of several past researches, especially the flexural rigidity property [5]. All previous researches assumed that microtubules are homogeneous and isotropic elastic rods. In general, there are four types of methods used to measure the flexural rigidity of microtubules: by buckling force, by relaxation, by hydrodynamic flow, and by thermal
fluctuation. These methods differ in the following conditions: static or dynamic process; classical mechanics, statistical mechanics, or hydrodynamics; type of manipulation force; type, balance, and direction of working force; and the number of force fulcrums [6]. These differences cause results to vary by up to two orders of magnitude.

Some parameters in the measurement process are quite difficult to estimate. The hydrodynamic force is not homogenous and can change over time. The thermal fluctuation method involves several uncontrolled factors that cause a tendency to produce a value of one order of magnitude higher than the other methods. It was reported that internal friction exists between two adjacent protofilaments in microtubules which slow down the movement of microtubules. These uncertainties affect the results of the measurement of flexural rigidity. Method that employs fewer number of assumed parameters makes it simpler and more focused on measuring the microtubule material [6].

Kikumoto et al. claim that the buckling force method they use is the most static and most direct method for measuring the flexural rigidity of microtubules. They used two optical traps to bend microtubules. The optical trap technique is also used in the relaxation method to pull one end of the microtubule up while the other is fixed and then release it to observe the relaxation process [15]. The results of several microtubule measurements results using optical trap techniques are listed in table 1.

| Citation | Temp (ºC) | Variation | Flexural rigidity ($\times 10^{-24} \text{Nm}^2$) | Measurement technique |
|----------|-----------|-----------|---------------------------------|----------------------|
| [16]     | 37        | With MAPs (10µm) | 34 ± 17                         | Optical trap buckling |
|          |           | With MAPs (30µm) | 200 ± 60                        | Optical trap buckling |
|          |           | With Taxol (5µm) | 1 ± 0.65                        | Optical trap buckling |
|          |           | With Taxol (20µm) | 20 ± 6                          | Optical trap buckling |
| [15]     | 22-25     | GDP tubulin     | 3.7 ± 0.8                       | Optical trap RELAX   |
|          |           | With Taxol      | 1 ± 0.3                         | Optical trap RELAX   |
|          |           | With MAPs       | 16 ± 3                          | Optical trap RELAX   |
|          |           | GDP tubulin     | 4.7 ± 0.4                       | Optical trap WIGGLE  |
|          |           | With Taxol      | 1.9 ± 0.1                       | Optical trap WIGGLE  |
|          |           | With MAPs       | 18 ± 3                          | Optical trap WIGGLE  |
| [6]      | 33        | GDP tubulin     | 7.9 ± 0.7                       | Optical trap buckling |
|          |           | With Taxol      | 2.0 ± 0.8                       | Optical trap buckling |
| [17]     | -         | With Taxol      | 6.1 ± 1.3                       | Optical trap buckling |
2.3 Biological viscoelasticity model

The most common models for molecular flexural rigidity and biological viscoelasticity models are the Kelvin (Voigt) model and its variants (Fig. 1). The number of dashpots is determined by the length of the object [18, 19]. Increasing the length of a molecular filament object results in an increase in effective viscosity, which is represented by an additional viscous dashpot in the model. The effect of increasing viscosity is that the object becomes slower to move; thus, the time needed for the object to recover from deformation is greater. However, elasticity will not be affected by increasing the length of the object. Therefore, the number of springs in the model remains the same.

![Figure 1](image_url)

**Figure 1.** The rheological model for the viscosity of molecular objects

A single spring represents the elasticity of the object. Dashpots represent viscosity which increases in number as the length of the object increases.

The Kelvin (Voigt) model combines all elements in parallel. The spring will eventually recover to its original position, even though the dashpots hold it back. Therefore, the model ensures that the object will always return to its original form after the deformation forces are being unloaded. However, the model has intrinsic limitation in its representation power with regards to the shape of microtubules. It is applicable to the bending of a single straight microtubule but may not work well for complex forms of microtubules due to the lack of parameters necessary for representing the structures of a microtubule. In order to solve this issue, we focused on tensegrity which have been used for the stability analysis of cellular organisms.

2.4 Tensegrity in cellular biology

Tensegrity was formalized by Richard Buckminster Fuller [20], inspired by the geometric model of his student, Kenneth D. Snelson [4]. Tensegrity structure consists of continuous tension elements and discontinuous compression elements that stabilize the structure to maintain its shape. The structure can withstand external pressure and return to its original shape after the pressure is
unloaded. There are two classes of tensegrity: prestressed tensegrity and geodesic tensegrity [3]. The prestressed tensegrity structure consists of discontinuous compression elements that are bound by a network of continuous tension elements. On the other hand, geodesic tensegrity consists of individual elements that can produce tension or resist compression to adapt the applied external force.

The term biotensegrity is applied to the structure of tensegrity in biological organisms at all levels of the human body [4]. An example of biotensegrity at the cellular level is cytoskeleton network. Microfilaments and intermediate filaments function as tension elements while microtubules act as discontinuous compression elements in cellular biotensegrity. Mechanical forces in extracellular matrix (ECM) adhesion are transmitted to the nucleus via integrins and cytoskeleton as part of the tensegrity system [21]. This allows the cell to sense the environment mechanically.

The concept of tensegrity may also be applicable to realize the flexural rigidity of microtubules in the sense that the surrounding environment affects the strength of the microtubules to withstand the compression load. This is one of our research hypotheses to realize intuitive haptic interface when dealing with biomolecules including microtubules in virtual reality. The tensegrity representation model is so general that it can be used to represent a wide variety of molecular objects in principle.

### 2.5 Virtual reality simulation

Virtual reality (VR) technology has been used as an immersive visualization of molecular object and has a great potential to study biological viscoelasticity models if haptic devices and molecular simulation capabilities are provided. The VR technology has been widely applied for molecular visualization education and research so far: VR model of a breast cancer cell [22], RealityConvert [23], Molecular Rift [24], 3D-Lab [25], ChimeraX, AltPDB, Molecular Zoo [26] and so on. The technology also provides a different perspective by going inside molecules that cannot be done by laboratory observations. Observer may find new insights about molecules that might not have been realized before [26]. Easy navigation and control with head and hand movements help further interaction, as if the observer were in the molecular world.

Haptic technology enhances interaction with the system by providing tactile and force feedback. Molecular docking simulation [27] can make the observer feel the interaction force in the docking process, thereby increasing the understanding of it. HaptiMol ISAS software to explore water accessible biomolecular surfaces [28, 29], HaptiMol ENM to simulate haptic feedback by applying forces to individual atoms [30], and HaptiMol RD to simulate haptic feedback in molecular docking [31–34] are haptic softwares created with the CHAI3D haptic framework that supports a grounded haptic devices with translational force feedback. More sophisticated devices with rotational force feedback also exist [35, 36]. Grabity [37], VRTouch [38], and a custom haptic rendering device with the Leap Motion hand tracking controller [7, 8] are examples of ungrounded haptic devices with advantages in portability. Leap Motion controller [39, 40] and Microsoft Kinect sensor [41] are two well-known devices that can track hand movements to provide natural hand interaction with 3D virtual objects with accuracy up to millimeters.

One of the challenges in a VR simulation system is the demand for performance, especially for complex interaction scenes. The requirement for a VR system to avoid user's motion sickness is 90 FPS performance per eye. Software frameworks that only use the CPU to process the simulation usually struggle to meet the ideal requirements [7]. Gutmann et al. [9–11] has implemented a microtubule gliding assay simulation based on a particle simulation system using a low-level DirectX 12 graphics library with general purpose processing computing units (GPGPU) to utilize
GPU performance, allowing it to simulate up to millions of particles when using multiple GPU cards. Combining with VR and haptic systems, the particle simulation system becomes a system suitable for molecular modeling with haptic VR rendering [8]. Macklin et al. [42] has implemented unified particle physics to simulate fluids, rigid bodies, or deformable objects using matrix polar decomposition. Later on, Lovrovic [43] improved it by adding joints to the rigid body implementation.

2.6 Research issues to solve

Researchers have been able to measure some of the mechanical properties of molecular objects through laboratory experiments, although the results are often different due to several factors which cannot be explained yet. Presenting the value of results to others is another challenge. Since presenting it in numbers or in diagram models does not really provide intuitive understanding due to the absent of the direct interaction, presenting molecular objects with the mechanical behavior in VR haptic simulations can be a solution.

The issues we want to solve in this research are: how to develop a 3D tensegrity object model that can represent molecular objects such as microtubules containing large numbers of atoms and how to provide an intuitive natural hand interaction in the system with microtubule objects with their mechanical behavior.

3. Methods

3.1 Design considerations

In order to convey an intuitive understanding of molecular objects, the 3D model in this research was developed to run on a haptic VR simulation system. In order to provide natural hand interaction, Leap Motion hand tracking controller was used. The design of the object model must consider the requirements needed for VR and the haptic system to be executed. The system may simulate a lot of molecular objects that require large numbers of particles. Therefore, the design must ensure that calculations can be processed in parallel so that they can benefit from an increase in the quantity and quality of GPU resources. Coarse-grained particle model object must be used to make the simulation possible.

The anchored spring model utilizes springs that are attached to particles and anchors to form geodesic tensegrity that forms an object and supports its structural strength. Although it is often used in creating flexible 3D objects, the use of springs is known to be very challenging. Setting an incorrect spring constant value with a certain force value can cause an erratic bounce motion that will cause an object explosion. This becomes more complicated by the increasing number of particles and anchors needed to create a longer and larger 3D object. Therefore, to maintain a stable simulation system, the object model that we designed in this research takes the prevention of object explosion due to excessive spring force as an important consideration.

Another consideration is that to provide an intuitive natural hand interaction in the system with the microtubule objects, the 3D object is expected to have reasonable mechanical behavior in accordance with the results of laboratory experiments. Therefore, the object model must be able to be adjusted to fit a number of criteria, namely the bending shape, viscoelastic behavior, and matching of flexural rigidity with the targeted value.
3.2 Particle simulation system overview

Particle simulation system presented in this study is based on particle-based real-time microtubule gliding assay simulation system accelerated by GPGPU [9–11] and haptic rendering system for virtual reality molecular modelling [7, 8]. This system simulates the movements of thousands of particles using the Euler integration for each time step. A damping factor is applied to all particles in the system. The movement velocity of all particles is reduced linearly at each time step to create the impression that all particles are surrounded by viscous solution. A damper value is a multiplier value that is applied to all particles to reduce their movement velocity.

Collision between two particles are calculated with respect to the distance between particle centers and the sum of the particle radii. A shorter distance than the sum of the radii will produce a repulsive force pushing in both directions. The shorter distance will cause greater force. This system allows particles to have different diameters. However, every particle is set to have the same mass in the interaction. Therefore, particle size will not affect the impact force. The collision forces between particles per time step are calculated individually per particle which can be processed in parallel using multiple GPUs or multicore CPU. The resulting force will be added to the calculation of the particle's movement for the next time step.

In order to deploy springs, each particle is given three data lists: a list of particle IDs that are connected to it, a list of default distances with the particles that are connected, and a list of each spring constant. The pairs of particles connected with spring are determined by the object model. At each time step, the distance between a pair of particles can change due to the particle motion, creating a distance difference with the default distance. This distance difference combined with the spring constant is a variable component in the Hooke's Law equation that calculates the spring force. The spring force that occurs in each particle can be calculated individually in parallel using multiple GPUs or multicore CPU.

Anchors are almost identical to particles, but they are not 3D objects. Anchors are invisible and cannot collide with other particles nor with other anchors. Anchors can be connected by springs to particles or other anchors. With different data lists for anchor-to-particle connections and anchor-to-anchor connections, their calculations can be done separately. Therefore, each of the anchor force calculation is concurrent enough to be processed in parallel using multiple GPUs or multicore CPU.

At the end of each time step, the forces that occur on each particle are added to determine each movement. The overall forces in a particle \( F_{\text{particle}} \) derived from the sum of all the forces in the particle:

\[
F_{\text{particle}} = \sum_{i=1}^{a} F_{\text{collision}_i} + \sum_{j=1}^{b} F_{\text{ParticleToParticle}_j} + \sum_{k=1}^{c} F_{\text{AnchorToParticle}_k},
\]

where \( F_{\text{collision}} \) is the collision forces, \( F_{\text{ParticleToParticle}} \) is the spring forces of the connected particles, and \( F_{\text{AnchorToParticle}} \) is the spring forces of the connected anchors.

The movement of each anchor is also determined in a similar way to a particle. The difference is that the forces that occur are different from those in particles. Since the anchor cannot touch other objects, there is no collision force. The overall force at the anchor \( F_{\text{anchor}} \) is derived from the sum of all the forces in the anchor:
where $F_{\text{ParticleToAnchor}}$ is the spring forces of the connected particles, $F_{\text{AnchorToAnchor}}$ is the spring forces of the connected anchor.

How the object model places anchors and springs on the object greatly influences the overall spring force on each particle and anchor. Therefore, further details of each spring force calculation will be explained in paragraph 3.3 along with an explanation of the object model.

3.3 Anchored spring model

In microtubule gliding assay simulation [9–11], microtubule objects are presented as rows of particles connected by springs. The diameter of the particles corresponds to the diameter of the microtubules. Actin filament, DNA strand, collagen, and other rod-like macromolecular object might also be presented in the same way. Since a row of particles do not have a geometric stiffness to maintain its shape, presenting the flexural rigidity of the object by using only springs between particles is not possible. In order to present it, the anchored spring model that we developed creates a tensegrity structure by placing anchors and springs around object particles to support their conformation (Fig. 2). Anchors represent the solution surrounding the object and springs represent the interaction between the solution and the object. Some parameters including spring constants need to be adjusted to match the flexural rigidity with the desired reference value.

From a structural point of view, anchored string model object is a structure consisting of springs with particles acting as joints. These springs try to keep the distance between the particles connected by producing a pull or push force in accordance with the current distance. These forces are analogous to the tension and compression forces in the elements of geodesic tensegrity. Therefore, anchored string model is classified as a geodesic tensegrity structure.

The model places a series of four anchors along the object from end to end. The four anchors are placed on a plane perpendicular to the axis of the object with the same distance and the same angle around the object, thus forming a square-like formation. Each anchor in a set of four is connected to each other by springs to maintain their shape. A group of object particles between two sets of four anchors is called a section. Each set of four anchors holds particles in the sections in front of and behind them. Therefore, each particle is held by two sets of four anchors in front of and behind it. Fig. 3 illustrates the formation of anchors and their connections with object particles in this object model.

The calculation of the spring force basically follows Hooke's Law, which defines it as a spring constant times the change in distance between the two connected particles or anchors ($\Delta d$). Change in distance $\Delta d$ is defined as the difference between the current distance $d$ and the initial distance $d_{\text{init}}$. Below are the spring force equations in the anchored spring model:

$$F_{\text{ParticleToParticle}} = k_{pp} \times \Delta d,$$

$$= k_{pp} \times (d - d_{\text{init}}),$$

$$F_{\text{AnchorToAnchor}} = k_{aa} \times \Delta d,$$

$$= k_{aa} \times (d - d_{\text{init}}),$$
The formation of anchors and particles in anchored spring model

Blue spheres are object particles. Red spheres are illustrations of anchors. Yellow square is a plane that is perpendicular to the axis of the object, which is drawn only to illustrate a square-like formation of the anchors. Red spheres and yellow square are not visible in the simulation.

The spring force of two connected particles $F_{\text{ParticleToParticle}}$ in Eq. 3 and 4 has the same spring constant $k_{pp}$ for both particles, and so does the spring force of two connected anchors $F_{\text{AnchorToAnchor}}$ in Eq. 5 and 6 with their spring constant $k_{aa}$. However, it is not the same as the spring force between a particle and an anchor. While the spring force on the particle $F_{\text{AnchorToParticle}}$ in Eq. 7 and 8 uses the spring constant value $k_{ap}$ given by the object parameter, the spring force at the anchor $F_{\text{ParticleToAnchor}}$ in the Eq. 9 and 10 is given a spring constant value $k_{pa}$ of one per number of particles connected to it (Eq. 11). The reason for this is that the anchor may have lots of particles connected to it using springs when creating a large object. As a result, the anchor might receive too much spring force which can cause the object to explode. By using the number of connected particles as a divider in the spring constant, excessive spring force can be avoided. The different spring constants used between the pair of spring forces of the anchor and the particle seem uneven. However, Euler's
integration allows the two spring forces to balance each other in the following time steps. Therefore, the object remains stable.

In the case of cytoskeleton objects such as microtubules and actin filament, they are almost solid objects. Many researchers consider them as rigid objects. Increasing the spring constants too much can cause the object to suffer excessive forces and explode. In order to avoid that, the anchored spring model implements a bridge connection between anchor sets. Bridge connection helps stiffen the object and speed up the distribution of forces by connecting each anchor set to the other. Bridge connection 0 means there is no bridge connection between anchor sets, bridge connection 1 means that each anchor is connected to adjacent anchor in a row, bridge connection 2 means that each anchor is connected to the next second anchors in a row, and so on for the next (Fig. 3).

![Figure 3. Anchor spring model connections scheme in 2D illustration](image)

Each set of anchors connects to particles in the sections in front of and behind it. In bridge connection 0, there is no connection between anchor sets. In bridge connection 1, anchor \( A_1 \) is connected to \( A_2 (BC_{1,2}) \), \( A_2 \) is connected to \( A_3 (BC_{2,3}) \), \( A_3 \) is connected to \( A_4 (BC_{3,4}) \), and so forth. In bridge connection 2, anchor \( A_1 \) is connected to \( A_3 (BC_{1,3}) \), \( A_2 \) is connected to \( A_4 (BC_{2,4}) \), \( A_3 \) is connected to \( A_5 (BC_{3,5}) \), and so forth.

The parameters that can be adjusted in the anchored spring model are:

- Particle-to-particle constant.
- Anchor-to-anchor constant.
- Anchor-to-particle constant.
- Anchor distance, which is the distance between each anchor to the center of the anchor set.
- Number of sections in the object.
- Section length, which is the number of particles per section.
- Bridge connection.

Of the seven parameters, only anchor distance and section length parameters can have units, namely meters. The value of the two parameters in meters depends on the particle diameter and its scale in the simulation.
3.4 Flexural rigidity measurements

Flexural rigidity can be defined as the resistance of an object when it is bent. It is denoted by $EI$ coming from the Young's modulus $E$ and the second moment of area $I$. Previous researches often yield significantly different flexural rigidity results because each method used involves different observation forces. The more static and more direct the method used to estimate flexural rigidity, the more precise and consistent in getting results [6].

This research used classical mechanics analysis to directly measure the flexural rigidity of the anchored spring model object. All previous measurement methods assumed that microtubules are isotropic elastic rods, and so did this research. The object with the length of $L$ was attached to one end to make it immovable by making the position of four anchors at this end fixed. The object was then bent perpendicularly by applying a force $F_{up}$ at the other end towards the top. The force was applied by moving the particle with mass at that end progressively upward with the specified acceleration $a$:

$$F_{up} = ma$$  \hspace{1cm} (12)

The acceleration is defined as:

$$a = \frac{v^2}{t^2} = \frac{r}{t^2}$$  \hspace{1cm} (13)

where $V$ is the velocity added to the moving particle, $t$ is the time to complete one computing cycle, and $r$ is the range the particle is moved in each computing cycle.

![Figure 4. Measuring the flexural rigidity by bending the object](image)

The bending will slowly decrease until steady state with no further increase in deflection. Deflection in the steady state will be used to measure the maximum deflection of $y$ ($L$).

The object was slowly bent until it reached a steady state and stopped bending (Fig. 4). In the simulation, the endpoint of the object never really stops moving which causes small fluctuations in the value of the deflection. The maximum deflection at the endpoint of the object $y$ ($L$) is measured from the average of the first 100 – 300 iterations that have no increase (Fig. 5). Using the classical mechanics bending of rods equation, the value of the flexural rigidity can be obtained:
The maximum deflection of the anchored spring model object is measured from the average endpoint particle deflection of the first 100 – 300 iterations which has no increase.

The mass $m$, the moving range of the particle $r$, and the length of the object $L$ in Eq. 13 depend on the type of molecular object presented. Time $t$ is also a simulation environment parameter derived from the number of computing cycles per second $cps$ and the simulation time scale $t_{sim}$:

$$ t = \frac{t_{sim}}{cps}. $$  

(17)

In order to make results that apply to general objects, object-specific parameters and simulation environment parameters are separated from general result values as:

$$ EI = \frac{n_{particles}r_{sim}(n_{section} \times section_{length} - 1)}{3y(L)} \times m_{particle} \times d_{particle}^3 \times \frac{cps^2}{r_{sim}^2}. $$  

(18)

where $n_{particles}$ is the number of moving particles, $r_{sim}$ is the moving range in unit length of

**Figure 5.** The maximum deflection of the anchored spring model object is measured from the average endpoint particle deflection of the first 100 – 300 iterations which has no increase.
simulation, $d_{\text{sim}}$ is the particle diameter in unit length of simulation, $n_{\text{section}}$ is the object’s parameter that define the number of section in the object, $\text{section length}$ is the object’s parameter that define the number of particles per object’s section, $m_{\text{particle}}$ is the mass of a particle in kilogram, and $d_{\text{particle}}$ is the diameter of a particle in meter. The result values presented in the next section are the simulated flexural rigidity $E_{I_{\text{sim}}}$:

$E_{I_{\text{sim}}} = \frac{n_{\text{particles}} d_{\text{sim}}^2 (n_{\text{section}} \times \text{section length} - 1)^3}{3 y(L)}$.  

(19)

By adding Eq.19 into Eq. 18, $E_{I}$ becomes:

$E_{I} = E_{I_{\text{sim}}} \times m_{\text{particle}} \times d_{\text{particle}}^3 \times \frac{c_{ps}^2}{e_{\text{sim}}^2}$.  

(20)

By placing $E_{I_{\text{sim}}}$ from Eq. 20 to the left and others to the right, the relationship between $E_{I_{\text{sim}}}$ and $E_{I}$ is shown by:

$E_{I_{\text{sim}}} = \frac{E_{I}}{m_{\text{particle}} \times d_{\text{particle}}^3 \times \frac{c_{ps}^2}{e_{\text{sim}}^2}}$.  

(21)

3.5 Bending evaluation

Bending shape which reflects elasticity was analyzed by comparing all the particle deflections in the model with the classical mechanics bending equation to evaluate the bending mechanics of the model object. According to classical bending mechanics, deflection along the rod object varies depending on the $x$ distance from the fixed end. The shape of the object is obtained as:

$y(x) = \frac{F_{\text{app}} x^2}{6 E_{I}} \left[ - \left( \frac{x}{L} \right)^3 + 3 \left( \frac{x}{L} \right)^2 \right]$.  

(22)

The bending shape of the anchored spring model object can be evaluated by comparing the $y$ axis position of each particle in the object with the expected deflection according to Eq. 22. The closer the result of the comparison means the bending shape of the object is closer to the real bending shape according to classical mechanics.

3.6 Viscoelasticity behavior

Viscoelasticity was investigated by observing the retardation time, which is the time taken for the object to reach a steady state in the bending process. It can be determined by how many iterations are needed for the deflection of the endpoint of the object to stop or not have a significant increase (Fig. 6).

The object's viscoelasticity behavior is expected to be similar to Fig. 1, the number of viscous dashpots in the model increases with increasing object length. Increasing the object length is expected to cause the viscosity to also increase. Therefore, longer objects are expected to have a longer time to reach a steady state.
4. Results

In this research, one particle represents a segment in a microtubule containing three cycles of tubulin dimers. The particle diameter is 24 nm which is the same as the diameter of the microtubules and also the length of one segment. In this simulation, a particle has a diameter of one unit of simulation length. Therefore, the value of anchor distance and section length parameters are multiplied by 24 nm. One segment contains 78 monomers that lead to the mass of a particle of $6.4761023597 \times 10^{-21}$ kg. The computing cycle is assumed to be identical to the graphics framerate, which is 90 frames per second for VR performance. The simulation time scale is 1: 0.000004, which one second in the simulation means 4 microseconds in reality. Therefore, one iteration represents 44.44 nanoseconds. Meanwhile, the length of the object is determined by the multiplication between number of sections, section length, and particle diameter, minus particle diameter.

4.1 Bending evaluation results

The evaluation of the bending shape of the anchor spring model resulted in a very high match between the deflection of each particle and the expected deflection according to the classical bending equation (Fig. 6). This means that the object's bending behavior is in accordance with mechanical law. This can happen because the anchored spring model successfully distributes the force applied to one end to each particle in all parts of the object.

![Particle Deflection Comparison](image)

Figure 6. The graph shows the evaluation of bending by comparing the y axis of each particle and the expected deflection according to the classical bending equation.

4.2 Viscoelasticity observation results

As mentioned in section 3.6, the object's viscoelastic behavior can be observed from the retardation time. The parameters that determine the length of the object are the number of sections...
and section length parameters. Increasing the number of sections parameter means adding more groups of particles to the object, which directly makes the object longer. Increasing the section length parameter means adding more particles to each part of the object, which also makes the object longer. The number of iterations needed in various values of these parameters are shown in Table 2 and Table 3.

### Table 2. Retardation time of objects with various number of sections in default parameter values

| Number of section | Microtubule length (nm) | Retardation time | Number of iterations | Time (µs) |
|-------------------|-------------------------|------------------|----------------------|----------|
| 5                 | 576                     | 153              |                      | 6.80     |
| 6                 | 696                     | 254              |                      | 11.29    |
| 7                 | 816                     | 603              |                      | 26.80    |
| 8                 | 936                     | 989              |                      | 43.96    |
| 9                 | 1056                    | 1712             |                      | 76.09    |
| 10                | 1176                    | 2581             |                      | 114.71   |

The results in Table 2 and Table 3 indicate that increasing the length of the anchored spring model object does make the retardation time longer; therefore, it does increase viscosity. This is identical to the model in Fig. 1, which shows that the number of viscous dashpots increases with increasing object length. Therefore, the anchored spring model is in line with the experimental results that increasing the length of biomolecular filament objects causes an increase in viscosity behavior.

### Table 3. Retardation time of objects with various section length in default parameter values

| Section length | Microtubule length (nm) | Retardation time | Number of iterations | Time (µs) |
|----------------|-------------------------|------------------|----------------------|----------|
| 2              | 456                     | 363              |                      | 16.13    |
| 4              | 936                     | 1303             |                      | 57.91    |
| 6              | 1416                    | 3136             |                      | 139.38   |
| 8              | 1896                    | 5217             |                      | 231.87   |
| 10             | 2376                    | 7446             |                      | 330.93   |

### 4.3 Flexural rigidity measurement results

Finding the object's flexural rigidity can be done by running the simulation and measuring the deflection as described in section 3. 4. However, finding the combination of object parameters to obtain the desired flexural rigidity value is not simple. The number of possible combination values of the seven parameters is infinite because some values have continuous values. In order to find the correlation between each parameter and the flexural rigidity of the anchored spring model object, measurements in various values of each individual parameter were performed.

In order to find the correlation between parameters and flexural rigidity, a set of default
parameters was determined. For each parameter, flexural rigidity was measured several times by changing the parameter values within a certain range of values. The results show a trendline for flexural rigidity according to variations of this parameter. In order to find correlations with other parameters, the steps to find the trendline were repeated several times by also modifying the values of other parameters one by one. Table 4 shows the default values, variation values, modified values and that we use. As a result, several curves were obtained with each representing the baseline and the influence of other parameters.

**Table 4. Parameters and values for correlation analysis**

| Parameter                  | Default value | Variation value | Modified value |
|----------------------------|---------------|-----------------|----------------|
| Particle-to-particle constant | 0.1           | 0.05, 0.01, 0.15, 0.2, 0.25 | -              |
| Anchor-to-anchor constant   | 0.2           | 0.05, 0.01, 0.15, 0.2, 0.25 | 0.15           |
| Anchor-to-particle constant | 0.1           | 0.05, 0.01, 0.15, 0.2, 0.25 | 0.15           |
| Anchor distance             | 8             | 5, 10, 15, 20, 25 | 15             |
| Number of sections          | 10            | 5, 6, 7, 8, 9, 10 | 7              |
| Section length              | 5             | 2, 4, 6, 8, 10 | 2              |
| Bridge connection           | 0             | 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 | -              |

Some parameters show linear correlations with flexural rigidity, some have power trendline correlations, and one parameter shows random results. The results of the analysis are then used to determine the fitting function for regression to find the predictive function of the object's flexural rigidity.

The particle-to-particle constant parameter analysis was performed three times by only modifying its value from the default parameter set. The results of various particle-to-particle constant parameter value settings in Fig. 7A show random flexural rigidity values with small magnitude of differences. It can be concluded that this parameter may not have a significant contribution to the object’s flexural rigidity. It makes sense because springs in the array of particles are in 1-dimensional arrangement, thus they do not produce geometric stiffness for the object. Therefore, this parameter can be excluded from correlation with the flexural rigidity value.

Positive linear correlation with flexural rigidity was shown by the anchor-to-anchor constant parameter, anchor-to-particle constant parameter, and number of sections parameter (Fig. 7B, 7C, and 7D). The correlation results in linear curves as:

$$y = m(x + a) + b$$

where $y$ is the flexural rigidity and $x$ is the parameter. Changes in the value of other parameters affect the slope $m$ of the trendline in a positive direction. The slope $m$ increases when the value of the other parameters increases, and vice versa. It can be concluded that the relationship between these parameters and others in the predictive equation is in the form of multiplication.

Fig. 7E and 7F show that the anchor distance parameter and section length parameter have a
power trendline in correlation with flexural rigidity as:

\[ y = cx^b \]  

(24)

While it does not seem to change the power \( b \), changing the value of other parameters causes the scaling factor \( c \) to change in the positive direction. Therefore, the relationship between these parameters and others in the predictive equation is also in the form of multiplication.

Bridge connection parameter shows a unique curve (Fig. 7G). The curve shows a significant increase initially but suddenly it starts to decrease at a certain value. The presence of springs in bridge connection significantly increase object stiffness. Connecting each anchor with adjacent anchors, an object with bridge connection 1 has nearly twice the flexural rigidity compared to that with bridge connection 0. Increasing the bridge connection value reinforces the structure of the object's frame. Figure 3 illustrates the anchor bridge connections. In bridge connection 1, almost every anchor is connected to two anchors each — one on the left and one on the right. For example: \( A_2 \) is connected to \( A_1 \) (with connection \( BC_{1,2} \)) and \( A_2 \) (\( BC_{2,3} \)), \( A_3 \) is connected to \( A_2 \) (\( BC_{2,3} \)) and \( A_4 \) (\( BC_{3,4} \)), and so forth. However, two anchors at the end point of the object have only one side of the adjacent anchor to connect to. \( A_1 \) is connected to \( A_2 \) (\( BC_{1,2} \)), but there is no \( A_0 \) to connect to. At the other end, \( A_n \) is only connected to \( A_{n-1} \) (\( BC_{n-1,n} \)) because there is no \( A_{n+1} \). Therefore, the number of bridge-connecting springs is \( n-1 \). In bridge connection 2, because each bridge-connecting spring skips one anchor in between, there are four anchors — two anchors at each end — that only have one side of the neighboring anchor to connect to. \( A_1 \) is only connected to \( A_3 \) (\( BC_{1,3} \)), \( A_2 \) is only connected to \( A_4 \) (\( BC_{2,4} \)), \( A_{n-1} \) is only connected to \( A_{n-3} \), and \( A_n \) is only connected to \( A_{n-2} \). Therefore, the number of bridge-connecting springs is \( n-2 \). This reduction continues for the next bridge connection value. Increasing the bridge connection value reduces the number of bridge-connecting springs in the object. Therefore, at some point, the flexural rigidity begins to decrease.

The results indicate that the flexural rigidity of the object stops increasing when the bridge connection value reaches about one third of the object's number of sections. Therefore, we only consider bridge connection values less than one third of number of sections values. This adjusted bridge connection curves are shown in Fig 7H. The correlation of the bridge connection with flexural rigidity is then considered to be linearly positive (Eq. 22). Similar as the previous parameters, this parameter is multiplied by the others in the predictive equation for flexural rigidity.

The results of the correlation analysis are then used to determine the fitting function for regression to find the prediction function of the object's flexural rigidity. The predictive function is explained as:

\[ EI_{sim} = 0.29 \cdot (x_1 + 0.04) \cdot x_2 \cdot x_3^{1.7} \cdot (x_4 + 3.27) \cdot x_5^{2.2} \cdot (x_6 + 0.97) \]  

(25)

where \( x_1 \) is the anchor-to-anchor constant parameter, \( x_2 \) is the anchor-to-particle constant parameter, \( x_3 \) is the anchor distance parameter, \( x_4 \) is the number of sections parameter, \( x_5 \) is the section length parameter, and \( x_6 \) is the bridge connection parameter.
Figure 7. Correlation between object parameters with the flexural rigidity
(A) Particle-to-particle parameter analysis. (B) Anchor-to-anchor constant parameter analysis.
(C) Anchor-to-particle parameter analysis. (D) Number of sections parameter analysis.
(E) Anchor distance parameter analysis. (F) Section length parameter analysis.
(G) Bridge connection parameter analysis. (H) Adjusted bridge connection parameter analysis.
The predictive function provides a closer calculation to obtain the object's flexural rigidity, while the actual measurement of flexural rigidity must be carried out as in section 3.4. The predictive function helps set initial parameters to get the object's flexural rigidity closer to the target value. After the actual flexural rigidity is measured, parameters can be reset to meet the target value of the flexural rigidity.

As mentioned earlier, flexural rigidity $E_{I\text{sim}}$ is a general flexural rigidity in the simulation world that can be applied to any molecular object. In order to get $E_{I\text{sim}}$ from the flexural rigidity of the simulated object ($E_I$), some object-specific and the simulation environment parameters must be defined. Particle mass $m_{\text{particle}}$, particle diameter $d_{\text{particle}}$, number of computing cycles per second $cps$, and simulation time scale $t_{\text{sim}}$ are needed for Eq. 21. As explained earlier at the beginning of section 4, the particle mass $m_{\text{particle}}$ for this microtubule model object is $6.4761023597 \times 10^{-21}$ kg, particle diameter $d_{\text{particle}}$ is 24 nm, number of computing cycles per second $cps$ is 90, and simulation time scale $t_{\text{sim}}$ is 44.44 nanoseconds.

Two values of flexural rigidity of the microtubules that we chose to present in this research were $3.7 \times 10^{-24}$ Nm$^2$ from pure microtubules (GDP tubulin) and $16 \times 10^{-24}$ Nm$^2$ from microtubules with MAPs. Both values were obtained from the results of experiments using optical tweezers with the RELAX method [15]. The reason for this choice is because both values originate from the same source and the same method with a marked difference in magnitude. Those flexural rigidity values ($E_I$) were used to get the simulated flexural rigidity values $E_{I\text{sim}}$ using Eq. 24. The results of $E_{I\text{sim}}$ were 81637.42 for the pure microtubules and 353026.7 for the microtubules with MAPs. The object parameters were adjusted according to Eq. 25 to get the value of $E_{I\text{sim}}$ close to the reference. These parameters were used as initial parameter values to create objects. The object was tested to measure its flexural rigidity. Because many approximations were taken in each parameter analysis, the value of the flexural rigidity that results from Eq. 25 can differ from the value of the flexural rigidity measured from the object. The difference between the two can be used to adjust parameters to get the final parameter value which will produce an object with a close matching flexural rigidity with the reference value. The simplest way to adjust parameters is to adjust anchor-to-anchor constant and / or anchor-to-particle constant values because both parameters have a linear correlation with the flexural rigidity. The initial parameter values and the final parameter values from the two microtubule representations are shown in table 5.
**Table 5.** Parameter values for microtubule objects presented in this research  
The time scale is 1: 0.000004. The anchor distance and section length parameters are multiplied by 24 nm to get the real unit value; therefore, the value of each parameter is 720 nm.

| Parameter                              | Microtubules | Pure          | Final value | Final value |
|----------------------------------------|--------------|---------------|-------------|-------------|
| Reference flexural rigidity ($EI$)     |              | $3.7 \times 10^{-24}$ Nm$^2$ | $16 \times 10^{-24}$ Nm$^2$ |
| Reference simulation flexural rigidity ($EIsim$) | 81637.42 | 353026.7 |
| Particle-to-particle constant          | 0.1          | 0.1           | 0.1         | 0.1         |
| Anchor-to-anchor constant              | 0.17         | 0.196         | 0.18        | 0.172       |
| Anchor-to-particle constant            | 0.1357       | 0.145         | 0.137       | 0.137       |
| Anchor distance                        | 30           | 30            | 30          | 30          |
| Number of sections                     | 14           | 14            | 14          | 14          |
| Section length                         | 30           | 30            | 30          | 30          |
| Bridge connection                      | 0            | 0             | 3           | 3           |
| Length of the object                   | 10.056 μm    | 10.056 μm     | 10.056 μm   | 10.056 μm   |
| Simulation flexural rigidity ($EIsim$) | 79805.74     | 81638.99      | 366449.437  | 352210.6    |
| Flexural rigidity ($EI$)               | $3.617 \times 10^{-24}$ Nm$^2$ | $3.7 \times 10^{-24}$ Nm$^2$ | $16.61 \times 10^{-24}$ Nm$^2$ | $15.96 \times 10^{-24}$ Nm$^2$ |

**5. Discussion**

Particle collisions, particle springs, anchor springs, and all movements of each iteration are processed individually in parallel. These calculations are concurrent enough to allow the simulation system to utilize multiple GPUs and multicore CPUs for massive computing tasks such as simulating millions of particles. Therefore, the system can benefit from increasing GPU resources in quality and quantity.

The computational power gives the simulation system the ability to run on a VR interface that allows users to see molecular objects in a 3D perspective. This system even allows user interaction with objects naturally using hands as if interacting with common sized objects. Haptic rendering technology presents a more intuitive interaction by providing tactile sense when touching an object. The combination of easy navigation, freedom of exploration, and intuitive understanding gives the possibility for the observer to discover something that may not have been discovered before by laboratory experiments.

The anchored spring model that we developed has demonstrated flexural behavior that is consistent with the classical bending mechanics described in Eq. 22. Springs and anchors help external forces to be distributed along all the particles of the object. Every movement of any particle causes all other particles of the object to react according to mechanical laws. This allows the object to make movements in real time interactions with the observer. The observer can manipulate objects in a way that is difficult to do in laboratory experiments and see the object's reaction from any desired position.

The anchored spring model approves the experimental model's viscosity by showing an increase in retardation time when the object being made is longer. The method of determining the retardation time is indeed difficult to produce accurate results, given the iteration where the endpoint of the object stops deflecting is not clearly visible. However, a rough interpretation of the deflection data clearly shows that the time needed to achieve steady state increases with the length of the measured object. In practice, the simulation damper value — which controls the velocity reduction of all particles in each time step — can be changed to adjust the viscosity of the object.
The formation of anchors and springs that surround the particles of an object provide structural strength to the objects. Even for objects without geometric stiffness such as a row of particles, the anchored spring model provides the object with adjustable rigidity. The results of the parameter analysis in section 4.3 also reinforce the assumption that the strength of the inter-particle joints does not have a significant effect in determining flexural rigidity. This gives the idea that the anchored spring model can be used to create any rod-like or filament type molecular objects regardless of the structural form of the object. Several molecular objects have been observed by researchers and their mechanical properties have been measured; however, the causes of mechanical behavior are not fully explained by existing theories. An anchored spring model object can be adjusted to make the mechanical property values match with any reference values. Therefore, the anchored spring model can be a suitable tool for representing the mechanics of these objects without having to fully understand the theoretical physics behind it. Having adjusted the parameters, the anchored spring model can be eventually related to some physical model.

6. Conclusion and future work

The particle simulation system and anchored spring model that we developed in this research execute computations in parallel which allows multiple GPUs to provide computing capability to carry out large-scale simulations in VR. The immersive VR user interface and haptic technology make it easy for observers to interact with molecular objects intuitively. Observers are able to manipulate objects better than in laboratory experiments, examine from various angles, and gain an intuitive understanding of objects. The anchored spring model provides a geodesic tensegrity structure of the object that can be adjusted, allowing it to be used to create any object regardless of the intermolecular structure of the object.

In the future, we would like to cover the body of the object with finer meshes such as cylinders and apply textures to the surface to create a better visual appearance of objects in the simulation. Textures can be used to distinguish the appearance of different objects with their surface images to make VR simulation more realistic. Therefore, the simulation is expected to provide the observer an intuitive interaction with more immersive experience as if he were in the molecular world.

Financial Support

This research was partially supported by the Strategic Advancement of Multi-Purpose Ultra-Human Robot and Artificial Intelligence Technologies of the New Energy and Industrial Technology Development Organization (NEDO, P15009) of Japan, MEXT/JSPS KAKENHI Grant Number 18H03673.

Acknowledgements

This study is based on results obtained from an Artificial Molecular Muscle project commissioned by the New Energy and Industrial Technology Development Organization (NEDO) Japan. The authors would like to thank the Indonesia Endowment Fund for Education (LPDP) for supporting the author with a scholarship.
References

[1] Inoue, D.; Gutmann, G.; Nitta, T.; Kabir, A. M. R.; Konagaya, A.; et al. Adaptation of Patterns of Motile Filaments under Dynamic Boundary Conditions. ACS Nano. 2019, 13(11), 12452–12460. https://doi.org/10.1021/acs.nano.9b01450

[2] Matsuda, K.; Kabir, A. M. R.; Akamatsu, N.; Saito, A., Ishikawa, S.; et al. Artificial Smooth Muscle Model Composed of Hierarchically Ordered Microtubule Asters Mediated by DNA Origami Nanostructures. Nano Lett. 2019, 19(6), 3933–3938. https://doi.org/10.1021/acs.nanolett.9b01201

[3] Ingber, D. E. Tensegrity I. Cell structure and hierarchical systems biology. J. Cell Sci. 2003 , 116, 1157–1173. https://doi.org/10.1242/jcs.00359

[4] Swanson, R. L. Biotensegrity: A unifying theory of biological architecture with applications to osteopathic practice, education, and research-a review and analysis. J. Am. Osteopath. Assoc. 2013, 113, 34–52. https://doi.org/10.7556/jaoa.2013.113.1.34

[5] Hawkins, T.; Mirigian, M.; Yasar, M. S.; Ross, J. L. Mechanics of microtubules. J. Biomech. 2010, 43(1), 23–30. https://doi.org/10.1016/j.jbiomech.2009.09.005

[6] Kikumoto, M.; Kurachi, M.; Tosa, V.; Tashiro, H. Flexural rigidity of individual microtubules measured by a buckling force with optical traps. Biophys. J. 2006, 90, 1687–1696. https://doi.org/10.1529/biophysj.104.055483

[7] Pramudwiatmoko, A.; Tsutoh, S.; Gutmann, G.; Ueno, Y.; Konagaya, A. Haptic Rendering Applied to Hand Tracking 3D User Interface for a Molecular Modeling Environment, in: 24th Int. Symp. Artif. Life Robot., International Society of Artificial Life and Robotics, Beppu, Oita, Japan, Jan 10–13, 2019; 109–114

[8] Pramudwiatmoko, A.; Tsutoh, S.; Gutmann, G.; Ueno, Y.; Konagaya, A. A high-performance haptic rendering system for virtual reality molecular modeling. Artif. Life Robot. 2019, 24, 542–549. https://doi.org/10.1007/s10015-019-00555-9

[9] Gutmann, G.; Inoue, D.; Kakugo, A.; Konagaya, A. Real-time 3D microtubule gliding simulation accelerated by GPU computing. Int. J. Autom. Comput. 2016, 13, 108–116. https://doi.org/10.1007/s11633-015-0947-1

[10] Gutmann, G.; Inoue, D.; Kakugo, A.; Konagaya, A. Using a master and slave approach for GPGPU computing to achieve optimal scaling in a 3D real-time simulation, 2016 IEEE 11th Annu. Int. Conf. Nano/Micro Eng. Mol. Syst., Sendai, Japan, April 17–20, 2016. https://doi.org/10.1109/NEMS.2016.7758208

[11] Gutmann, G.; Inoue, D.; Kakugo, A.; Konagaya, A. Parallel Interaction Detection Algorithms for a Particle-based Live Controlled Real-time Microtubule Gliding Simulation System Accelerated by GPGPU. New Gener. Comput. 2017, 35, 157–180. https://doi.org/10.1007/s00354-017-0011-5

[12] Desai, A.; Mitchison, T. J. Microtubule polymerization dynamics. Annu. Rev. Cell Dev. Biol. 1997, 13, 83–117. https://doi.org/10.1146/annurev.cellbio.13.1.83

[13] Ledbetter, M.; Porter, K. R. A “microtubule” in plant cell fine structure. J. Cell Biol. 1963, 19(1), 239–250. https://doi.org/10.1083/jcb.19.1.239

[14] Chalfie, M.; Thomson, J. N. Organization of neuronal microtubules in the nematode Caenorhabditis elegans. J. Cell Biol. 1979, 82, 278–289. https://doi.org/10.1083/jcb.82.1.278

[15] Felgner, H.; Frank, R.; Schlüwa, M. Flexural rigidity of microtubules measured with the use of optical tweezers. J. Cell Sci. 1996, 109, 509–516

[16] Kurachi, M.; Hoshi, M.; Tashiro, H. Buckling of a single microtubule by optical trapping forces: Direct measurement of microtubule rigidity. Cell Motil. Cytoskelet. 1995, 30, 221–228. https://doi.org/10.1002/cm.970300306
[17] Mameren, J. V.; Vermeulen, K.C.; Gittes, F.; Schmidt, C. F. Leveraging single protein polymers to measure flexural rigidity. J. Phys. Chem. B. 2009, 113(12), 3837–3844. https://doi.org/10.1021/jp808328a

[18] Ghodsi, H.; Darvish, K. Investigation of mechanisms of viscoelastic behavior of collagen molecule. J. Mech. Behav. Biomed. Mater. 2015, 51, 194–204. https://doi.org/10.1016/j.jmbbm.2015.07.015

[19] Ghodsi, H.; Darvish, K. Characterization of the viscoelastic behavior of a simplified collagen micro-fibril based on molecular dynamics simulations. J. Mech. Behav. Biomed. Mater. 2016, 63, 26–34. https://doi.org/10.1016/j.jmbbm.2016.06.006

[20] Fuller, R. B. Tentional Integrity Structures, US patent 3,063,521, 1962.

[21] Matthews, B. D.; Overby, D. R.; Alenghat, F. J.; Karavitis, J.; Numaguchi, Y. Ingber, Mechanical properties of individual focal adhesions probed with a magnetic microneedle. Biochem. Biophys. Res. Commun. 2004, 313(3), 758–764. https://doi.org/10.1016/j.bbrc.2003.12.005

[22] Johnston, A. P. R.; Rae, J.; Ariotti, N.; Bailey, B.; Lilja, A.; et al. Journey to the centre of the cell: Virtual reality immersion into scientific data. Traffic 2018, 19, 105–110. https://doi.org/10.1111/tra.12538

[23] Borrel, A.; Fourches, D. RealityConvert: A tool for preparing 3D models of biochemical structures for augmented and virtual reality. Bioinformatics 2017, 33(23), 3816–3818. https://doi.org/10.1093/bioinformatics/btx485

[24] Norrby, M.; Grebner, C.; Eriksson, J.; Boström, J. Molecular Rift: Virtual Reality for Drug Designers. J. Chem. Inf. Model. 2015, 55(3), 2475–2484. https://doi.org/10.1021/acs.jcim.5b00544

[25] Grebner, C.; Norrby, M.; Enström, J.; Nilsson, I.; Hogner, A. 3D-Lab: a collaborative web-based platform for molecular modeling. Future Med. Chem. 2016, 8, 1739–1752. https://doi.org/10.4155/fmc-2016-0081

[26] Goddard, T. D.; Brilliant, A. A.; Skillman, T. L.; Vergenz, S.; Tyrwhitt-Drake, J.; et al. Molecular Visualization on the Holodeck. J. Mol. Biol. 2018, 430(21), 3982–3996. https://doi.org/10.1016/j.jmb.2018.06.040

[27] Férey, N.; Nelson, J.; Martin, C.; Picinali, L.; Bouyer, G.; et al. Multisensory VR interaction for protein-docking in the CoRSAIRe project. Virtual Real. 2009, 13, 273–293. https://doi.org/10.1007/s10055-009-0136-z

[28] Stocks, M. B.; Hayward, S.; Laycock, S. D. Interacting with the biomolecular solvent accessible surface via a haptic feedback device. BMC Struct. Biol. 2009, 9, 1–7. https://doi.org/10.1186/1472-6807-9-6

[29] Laycock, S. D.; Stocks, M. B.; Hayward, S. Navigation and exploration of large data-sets using a haptic feedback device, ACM SIGGRAPH 2010 Posters, SIGGRAPH ’10. Los Angeles California, July, 2010, 4503. https://doi.org/10.1145/1836845.1837007

[30] Stocks, M. B.; Laycock, S. D.; Hayward, S. Applying forces to elastic network models of large biomolecules using a haptic feedback device. J. Comput. Aided. Mol. Des. 2011, 25, 203–211. https://doi.org/10.1007/s10822-010-9410-0

[31] Iakovou, G.; Hayward, S.; Laycock, S. A real-time proximity querying algorithm for haptic-based molecular docking. Faraday Discuss. 2014, 169, 359–377. https://doi.org/10.1039/c3fd00123g

[32] Iakovou, G.; Hayward, S.; Laycock, S. D. Adaptive GPU-accelerated force calculation for interactive rigid molecular docking using haptics. J. Mol. Graph. Model. 2015, 61, 1–12. https://doi.org/10.1016/j.jmgm.2015.06.003

[33] Iakovou, G.; Laycock, S.; Hayward, S. Determination of locked interfaces in biomolecular
complexes using Haptimol_RD, Biophys. *Physicobiology*. **2016**, *13*, 97–103. https://doi.org/10.2142/biophysico.13.0_97

[34] Iakovou, G.; Hayward, S.; Laycock, S. D. Virtual Environment for Studying the Docking Interactions of Rigid Biomolecules with Haptics. *J. Chem. Inf. Model.* **2017**, *57*(5), 1142–1152. https://doi.org/10.1021/acs.jcim.7b00051

[35] Daunay, B.; Micaelli, A.; Force, A.; Regnier, S. 6 DOF haptic feedback for molecular docking using wave variables, Robot. Autom. 2007 IEEE Int. Conf., Roma, Italy, Apr 10–14, 2007; 840–845. https://doi.org/10.1109/ROBOT.2007.363090

[36] Hou, X.; Sourina, O. Six degree-of-freedom haptic rendering for biomolecular docking. *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics).* **2011**, *6670LNCS*, 98–117. https://doi.org/10.1007/978-3-642-22336-5_6

[37] Choi, I.; Culbertson, H.; Miller, M. R.; Olwal, A.; Follmer, S. Gravity: A Wearable Haptic Interface for Simulating Weight and Grasping in Virtual Reality, Proc. 30th Annu. ACM Symp. User Interface Softw. Technol.– UIST 2017, Québec City, Canada, Oct 22–25, 2017; 119–130. https://doi.org/10.1145/3126594.3126599

[38] Go Touch VR: Touch the Virtual Reality with VR Touch, (n.d.). https://www.gotouchvr.com/ (accessed September 4, 2018)

[39] Weichert, F.; Bachmann, D.; Rudak, B.; Fisseler, D. Analysis of the accuracy and robustness of the Leap Motion Controller. *Sensors (Switzerland)* **2013**, *13*(5), 6380–6393. https://doi.org/10.3390/s130506380

[40] Guna, J.; Jakus, G.; Pogačnik, M.; Tomažič, S.; Sodnik, J. An analysis of the precision and reliability of the leap motion sensor and its suitability for static and dynamic tracking. *Sensors (Switzerland)* **2014**, *14*(2), 3702–3720. https://doi.org/10.3390/s140203702

[41] Khoshelham, K.; Elberink, S. O. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors* **2012**, *12*, 1437–1454. https://doi.org/10.3390/s120201437

[42] MacKlin, M.; Müller, M.; Chentanez, N.; Kim, T. Y. Unified particle physics for real-time applications, *ACM Trans. Graph.* **2014**, *33*, 1–12. https://doi.org/10.1145/2601097.2601152

[43] Lovrović, B.; Mihajlović, Z. Rigid Body Joints in Real-Time Unified Particle Physics, in: Eurographics 2018, Delft, the netherlands, Apr 16–20, 2018, https://doi.org/10.2312/egs.20181031