

Visualization of Tethered Particle Motion with a Multidimensional Simulation

Khovesh Ramdin,¹ Markus Hackl,² Shishir P.S. Chundawat^{2,*}

1. Department of Physics and Astronomy, Rutgers, The State University of New Jersey, 98 Brett Road, Piscataway, NJ 08854, USA

2. Department of Chemical & Biochemical Engineering, Rutgers, The State University of New Jersey, 98 Brett Road, Piscataway, NJ 08854, USA

Condensed running title: Simulation of Tethered Particle Motion

*Corresponding Author: shishir.chundawat@rutgers.edu

Abstract

The analysis of particles bound to surfaces by tethers can facilitate understanding of biophysical phenomena (e.g., DNA-protein or protein-ligand interactions, DNA extensibility). Modeling such systems theoretically aids in understanding experimentally observed motions and furthermore the limitations of such models can provide insight into modeling complex systems. The simulation of tethered particle motion (TPM) allows for analysis of complex behaviors exhibited by such systems; however, this type of experiment is rarely taught in undergraduate science classes. We have developed a MATLAB simulation package intended to be used in academic contexts to concisely model and graphically represent the behavior of different tether-particle systems. We show how analysis of the simulation results can be used in biophysical research employing single molecule force spectroscopy (SMFS). Students in Physics, Engineering and Chemistry alike will be able to make connections with principles embedded in their field of study and understand how those principles can be used to create meaningful conclusions in a multidisciplinary context. The simulation package can model any given particle-tether system and allows the user to generate a parameter space with static and dynamic model components. Our simulation was successfully able to recreate generally observed experimental trends using Acoustic Force Spectroscopy (AFS). Further, the simulation was validated through consideration of the conservation of energy of the tether-bead system, trend analyses, and comparison of particle positional data from actual TPM *in silico* experiments conducted to simulate data with a parameter space similar to the AFS experimental setup. Overall, our TPM simulator and graphical user interface is primarily for demonstrating behaviors characteristic to tethered particle motion in a classroom setting but can serve as a template for researchers to set up TPM simulations to mimic their specific SMFS experimental setup.

Keywords: Tethered Particle Motion, TPM, Computational Biophysics, Single-Molecule Force Spectroscopy, SMFS, Acoustic Force Spectroscopy, AFS

Introduction

Visualizing monitoring and modeling the complex motion of a particle attached to an extensible tether in a viscous fluid environment (also generally referred to as Tethered Particle Motion or TPM) is relevant to understanding several fundamental biophysical phenomena as well as solving practical engineering problems. Understanding and modeling TPM can enable experimentalists to observe the motion of DNA-scale molecular interactions using immunofluorescence/dark field microscopy (1) or manipulate such small-scale systems with molecular-scale precision using suitable acoustic, magnetic or optical tweezers based TPM imaging tools (2). Further, the advancements in the spatial resolution of optical imaging in the last few decades (3) have made TPM analysis particularly relevant to modern-day theoretical and applied biophysics. Particularly, understanding and modeling TPM is also critical for enabling single molecule experiments that focus on various biopolymers and its relevant molecular properties. For example, DNA polymer properties can be intrinsically studied and experimentally determined using TPM modeling by analyzing the Brownian motion of particles attached to individual double-stranded DNA (4). Schafer et al. was one of the first to devise a TPM assay to directly monitor the movement of single molecule of a processive polymerases acting on a template DNA. Additional notable results that were achieved in subsequent studies include the empirical validation of TPM as a technique to predict tether length (5). Similar TPM and other single-molecule assays have become more commonplace now to provide critical insight into how diverse classes of biological machinery and processive motors (e.g., cellulases/chitinases degrading cellulose/chitin polysaccharides (6–8), cellulose synthases synthesizing cellulose polymer (9), protein/DNA/RNA polymers synthesis/folding/degradation (10–12), and ATP-triggered motility of myosin/kinesin on actin/microtubules (13, 14)) function at the molecular and cellular level to solve diverse biotechnology problems, ranging from developing better enzymes for producing sustainable bioenergy from cellulosic biomass (15) to enabling personalized healthcare using advanced gene editing techniques like CRISPR (16).

While described in the scientific literature, TPM is not typically taught in an academic context although the theory associated with this topic is crucial to understanding observations from many bio/physical experiments. It is particularly necessary to study molecular-scale interactions using single-molecule experiments incorporating TPM methods for comprehension of complex biomolecular and cellular systems which subsequently allow for improved fundamental understanding of living systems and potentially lead to the development of novel biotechnology.

Scientific and Pedagogical Background

Mathematically, tethered particle system behaviors can be approximated through the consideration of Brownian motion. Such motion is a consequence of collisions that occur between the object being tracked and the particles present in a viscous environment (17). In principle, fluids are composed of multiple particles that are constantly colliding. Such uncontrolled and seemingly random small-scale behaviors are better modeled stochastically, since deterministic models often require unfeasible level of complexity for individual particle tracking capabilities (18). The idea associated with such models is to use random fluctuations to account for small scale perturbations that are observed experimentally due to diffusive effects experienced by a particle in a viscous environment (19).

Here, we present a graphical user interface- (GUI) based simulation package for use by students and teachers to perform simulations of a model tether-particle system within a parameter space of their choice (see illustration in Fig. 1). The simulation was developed using a complete

installation of MATLAB version 2022b. The simulation package builds and expands on previous models developed for educational purposes (41). The MATLAB- based model was written in an easily generalizable manner, has a complete user interface and is computationally efficient so data analyses can be easily performed. Simulation features like varying force ramps and constant force application are predefined settings in the simulation package as these are commonly encountered during single-molecule force spectroscopy (SMFS) experiments to mimic real-world scenarios (20). Further, corrections are included from a series of models presented in scientific literature to increase the accuracy of the TPM simulations and allow the user to understand the limitations/uses of the calculations being made. In particular, we have generated experimental data to validate our simulation predictions using Acoustic Force Spectroscopy (AFS) (21, 22). Students will be able to understand the behaviors of tethered-particle systems in general due to the easy-to-follow GUI for model presentation and exportation of several analysis plots/data from the interface to gain an appreciation for how such systems dynamically behave during SMFS experiments.

A biophysics lab course would be ideal for presenting this information because the skills taught are relevant to theoretical and experimental science's alike. Single-molecule studies are associated with and based on concepts from optics, chemical bond theory, cellular machinery, and many other subtopics in physics, chemistry, and biology (23). Biophysics is a highly interdisciplinary field which can benefit significantly from skills typically presented in a specialized manner in other disciplines and TPM is one of such concept that can be used as a template to demonstrate such interdisciplinary connections. In addition to gaining an understanding of theoretical and experimental principles, students with access to this simulation toolkit in their curriculum can gain exposure to computational, statistical, and mathematical knowledge in the context of a useful topic with 'real-world' applications. In its current form, educators can use this TPM toolkit to help students gain an appreciation for basic theory and implementation of theory since all of the code is written and commented in an easily understandable form. Unlike in research contexts where one is assumed to be able to make these connections without prior education, this TPM simulation GUI assumes only basic mathematical and coding knowledge, and little to no background in the theoretical behaviors of the TPM system itself. Overall, this is a toolkit meant for students to have a focused interaction in a short activity (e.g., class assignment or project) and that grants access to easily understandable biophysics concepts without need for complex background knowledge.

Methods

Simulation Overview

Our simulation experiment considers the dynamics of a bead attached to a surface using classical physics-based analysis. All the relevant parameters in this model can be altered by the user to explore alternative scenarios that aid in student learning. Further, parameters that are variable in the actual SMFS experimental setups are designed to be dynamic and can be modified by the user in real-time during the simulation, mimicking an actual experiment being conducted in real-time as well. The static and dynamic parameters associated with a typical single molecule TPM system are summarized in Table 1.

Based on these parameters, a complete description of the tethered bead position, applied force, intrinsic force due to particle collisions, energy and tether extension from equilibrium are provided in the form of continually updated graphical plots in a MATLAB based GUI. These plots are updated at a rate specified by the user in a static field prior to start of the simulation. The interface

where each of these parameters are provided by the user and the key features of the simulation package are summarized in Fig. 1.

Simulation Logic

A single MATLAB function was used that accepts user generated parameter space as well as memory terms. This function is called within a loop used in MATLAB app designer and the outputs are plotted at the user specified rate. Callback functions are utilized in the interface to synchronize the point at which the user makes a change and when that change is reflected in the base code output. The use of memory terms in the app allowed for the computations to be done with continuity as the user inputs are monitored and updated continuously within the base algorithm. Several simulations of the length specified by the user are run consecutively with initial conditions consistent with the end state of the prior simulation. This results in a continuous generation of data until the user specified total number of data points are reached. A flow chart of the simulation logic is found in the supplementary information (SI) Fig. S1. The simulation produces an output file in csv format that contains time (s), planar position (m), net DNA force (N), applied force in z (N), cartesian DNA force components (N) as well as θ/ϕ angular positions (-).

Computational Framework of the Simulation

The notations outlined in Table 1 and Table 2 will be used to reference each variable in this work.

The Modified Marko-Siggia worm-like chain model was considered for our model in Eq. 1 (24), Numerical root finding was used to solve for the approximate magnitude of the force for each direction.

$$F_i = \left(\frac{k_B T}{4L_p} \left[\frac{1}{\left(1 - \frac{R_i}{L_o} + \frac{F_i}{K_o}\right)^2} - 1 + \frac{4R_i}{L_o} - \frac{4F_i}{K_o} \right] \right), i = x, y, z \quad (1)$$

In this model, the $\frac{F}{K_o}$ terms are a correction introduced to the classic worm-like chain model to account for the elasticity of the tether. This modification improves the experimental agreement of the worm-like chain model that only provides an order of magnitude estimate of the persistence and contour lengths (25). The K_o term is a material parameter described by Young's modulus from classical mechanics. In this simulation, the Young's modulus was related to the persistence length of a solid rod with a circular cross section for mathematical simplicity (24). The DNA diameter of $d = 1.6$ nm was chosen, and the Young's modulus depends on the user-defined temperature and persistence length in Eq. 2. Some typical values of this parameter range from 800-1700 pN (26, 27).

$$K_o = \left(\frac{16k_B T}{d^2} \right) L_p \quad (2)$$

A spherical coordinate system was used to describe the particle's 3D-motion. The obtained magnitude was decomposed into x, y, and z components via projection onto a cartesian system using the following elementary trigonometric relations in Eqs. 3-5 below.

$$F_x = F * |\sin\theta * \cos\phi| \quad (3)$$

$$F_y = F * |\sin\theta * \sin\phi| \quad (4)$$

$$F_z = F * |\cos\theta| \quad (5)$$

The signs of these quantities were determined directly through the consideration of the extension of the tether. If the tether was extended in a negative direction, the force would have to be positive to restore the system to its equilibrium position and vice versa. This behavior is consistent with classical spring behavior described by Hooke's Law and serves as a reasonable description for the

behavior of the tether-bead system at any point in its motion due to the elasticity of the tether. All these computations were completed in a MATLAB function named ‘MarkoSiggiaVectorized.m’ and these force computations were continuously updated in a loop from the base code. The supplementary information (SI) documentation of the simulation package provides greater detail on the functional dependencies.

The computation of the θ and φ positions also come from basic trigonometric relations in Eqs. 6-7. The spatial orientation of the system is initially defined to be along the cartesian z direction alone and the descriptions of the angles are updated as the motion evolves over time.

$$\theta = \tan^{-1} \left(\frac{\sqrt{x^2 + y^2}}{z} \right) \quad (6)$$

$$\varphi = \tan^{-1} \left(\frac{y}{x} \right) \quad (7)$$

Next, the function ‘TetherForce2.m’ base code modifies the implicit force term in the z direction based on the magnitude of external force applied to the system in the user interface. The first option for the user is to choose a modifiable, but constant force at any point in the simulation. When the user makes a modification to the applied force using the force slider built into the app, a constant force is continually applied to the system for the duration that the user leaves the slider in the given position. This force is immediately applied with the chosen magnitude. The second option for the user is to apply a force ramp with a slope pre-determined by the user. The desired force by the end of the simulation is computed to increase in linear increments consistent with the total runtime of the simulation. The third option for the user is to apply a decaying force ramp which is computationally equivalent to the previous case except that a linear decay is considered instead. The projection of this magnitude onto the z direction is added to the force term from the ‘MarkoSiggiaVectorized.m’ function to determine the net force such that $F_{net} = F_z + F_{applied}$, considering the force decomposition based on Newtons 2nd Law.

The net effect experienced by the bead is intended to be consistent with Stokes Law. The bead is assumed to be perfectly spherical, surfaces are all assumed to have no imperfections, components are all assumed to be entirely homogenous, and the flow is constrained to be laminar. This means that the system has a low Reynolds number which is consistent with smooth and constant fluid motion (i.e., laminar flow). When the Reynolds number is low, viscous force is necessarily dominant, meaning perturbations introduced by the bead on the system are not the variable which dominates the overall motion. The liquid viscosity is a crucial variable in determining the scale of such effects and a specific analysis of relevance of the viscosity is presented in the SI Figure S3. A numerical validation of these assumptions is provided in the validation section of this study. Since all these conditions are approximately valid in the considered model, Stokes Law serves as a reasonable approximation to the net effect that is observed. This also means that enough information is available such that the deviations in position within a given timestep can be extrapolated from the simulation as is outlined in Eqs. 8-10 below. A correction factor is introduced to account for the edge effects since the tether-bead system is near the surface throughout the simulation. These corrections are derived based on the boundary condition that tangential flow needs to be zero at the bead surface (28, 29). The x displacement (Δx) is parallel to the surface and is described in Eq. 8.

$$\Delta x \approx \frac{F_x \Delta t}{6\pi\mu R} * \frac{1}{1 - \frac{9}{16}\left(\frac{r}{z}\right) + \frac{1}{8}\left(\frac{r}{z}\right)^3 - \frac{45}{256}\left(\frac{r}{z}\right)^4 - \frac{1}{16}\left(\frac{r}{z}\right)^5} \quad (8)$$

Similarly, the expressions for the y and z directions can also be obtained. The y displacement (Δy) is parallel to the surface so the correction due to surface effects remains the same.

$$\Delta y \approx \frac{F_y \Delta t}{6\pi\mu R} * \frac{1}{1 - \frac{9}{16}\left(\frac{r}{z}\right) + \frac{1}{8}\left(\frac{r}{z}\right)^3 - \frac{45}{256}\left(\frac{r}{z}\right)^4 - \frac{1}{16}\left(\frac{r}{z}\right)^5} \quad (9)$$

The displacement in z (Δz) is perpendicular to the plane so the correction due to surface effects is slightly different. This correction result in the following equation 10:

$$\Delta z \approx \frac{F_z \Delta t}{6\pi\mu R} * \frac{1}{\left(1 - \frac{9}{8}\left(\frac{r}{z}\right) + \frac{1}{2}\left(\frac{r}{z}\right)^3 - \frac{57}{100}\left(\frac{r}{z}\right)^4 + \frac{1}{5}\left(\frac{r}{z}\right)^5 + \frac{7}{200}\left(\frac{r}{z}\right)^{11} - \frac{1}{25}\left(\frac{r}{z}\right)^{12}\right)} \quad (10)$$

Eqs. 8-10 are used to update the x, y and z position at a given timestep. These cartesian position elements are then used in Eqs. 6-7 to update the spatial orientation elements from the initial state due to the viscous motion.

The rearrangement of Stokes law is only an approximation since finite timesteps are used to approximate the velocity of the bead/particle in addition to the numerical approximation of the surface effects. However, an effort was made to more accurately account for time that the particle takes to move between any two given positions through the introduction of a dynamic timestep (29) described in Eq.11.

$$\Delta t = \frac{2\mu\delta R}{|\nabla F|} \quad (11)$$

The implementation of this dynamic timestep allows for the accuracy of the position at any given time-point to be the same, since the deviation is normalized using the force gradient at every datapoint (29). The modified Marko-Siggia model accounts for the extensibility of the tether. This has a direct influence on the dynamic timestep which depends on normalization using the force gradient. The explicit computations are shown in Eqs. 12-13 below.

$$\nabla F = \left\langle \frac{\frac{2}{L_0 d_x} + \frac{4}{L_0}}{\frac{4L_P}{k_B T} + \frac{2}{K_0 d_x} + \frac{4}{K_0}}, \frac{\frac{2}{L_0 d_y} + \frac{4}{L_0}}{\frac{4L_P}{k_B T} + \frac{2}{K_0 d_y} + \frac{4}{K_0}}, \frac{\frac{2}{L_0 d_z} + \frac{4}{L_0}}{\frac{4L_P}{k_B T} + \frac{2}{K_0 d_z} + \frac{4}{K_0}} \right\rangle \quad (12)$$

$$d_i = \left(1 - \frac{i}{L_0} + \frac{F_i}{K_0}\right)^3, i = x, y, z \quad (13)$$

After these predictable effects are accounted for, the last remaining component necessary for accurately describing the particle position is its random motion due to diffusion. The correction factor to the position predicted from the basic force analysis was implemented using a random number generator. The random number generator was set such that the mean value is zero and a standard deviation defined by Eq. 14. The environment is assumed to be approximately isotropic as previously mentioned, which means that the expected standard deviation is independent of direction.

$$\sigma = \sigma_x = \sigma_y = \sigma_z = \sqrt{2\Delta t \frac{k_B T}{6\pi\mu R_b}} \quad (14)$$

The assumptions outlined above allowed for a complete description of the position of the bead-tether system to be generated under applied force. Since the time associated with motion between any two given positions is also available, many useful computations can be done to test the validity of the code, for example the verification of the conservation of energy. The general relation $F = -\nabla U$ was considered. The force acting on the particle is approximated to be constant

and independent for a given timestep. This means that the potential energy can be approximated using Eq. 15 below. The force terms are constant in each interval; thus, the integration only occurs over the position differential.

$$\Delta PE_i \approx - \left(F_x * (x_i - x_{i-1}) + F_y * (y_i - y_{i-1}) + F_z * (z_i - z_{i-1}) \right) \quad (15)$$

The projections of the force on the x, y, and z direction and the displacement as calculated between subsequent timesteps are considered in determining the change in potential energy. The kinetic energy was computed and rewritten using parameters relevant to the constructed system in Eq. 16 below.

$$\Delta KE_i \approx \frac{2\pi\rho R_b^3}{3} * \frac{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2}{(\Delta t)^2} \quad (16)$$

The ρ term in the kinetic energy computation is the density of the bead, the displacement in each direction is determined at subsequent timesteps iterated by variable i and the length of the timestep is denoted Δt .

All the arguments made above are consistent with a probabilistic consideration of the behavior of the tether-particle system. This means that by nature, assumptions of thermal equilibrium are made as can be seen from the applications of the equipartition theorem. These assumptions are not appropriate when a force is instantly applied to the system. The force ramp feature utilized allows for a steady buildup of the force which does not perturb the system by a great extent at any given instant. However, for general applications of large magnitudes of force, this model can break apart. As such, a separate model was implemented as described below, which uses physical constraints to ensure that the system remains stable as expected in reality.

First, in the cases where an external force is applied to the system, it is approximated that $F_{net} \approx F_{applied}$. When a force is applied, the tether will extend, and the tension will increase resulting in limited fluctuations. In accordance with the modified Marko-Siggia model (24), these fluctuations will occur about an equilibrium value which is associated with the applied force. Extracting this equilibrium value allows for a description of the bead position to be made independent of the timescale associated with the instability. Generating a distribution of permissible values about this equilibrium position allows for a complete description of the bead position. It is constrained by the tether length and there is a very small probability the bead will reach a value significantly different from the equilibrium value. These constraints are well described by a normal distribution with 0 mean fluctuations about the equilibrium position. The standard deviation was determined as shown in Eq. 17, where α is an arbitrary parameter meant to describe the resistivity of the environment and z_{eq} is the extracted equilibrium position. This parameter is not strictly defined and can be modified to best fit the data collected. In accordance with the assumption of isotropy, $\alpha = 3$ was assigned as the base setting.

$$\sigma_z = \frac{L_o - z_{eq}}{\alpha} \quad (17)$$

Since the z spatial harmonic behavior is described, the planar region of interest can easily be extrapolated. The system is defined such that magnitude of the position vector corresponds to the tether extension. Since a reasonable approximation to the z position is obtained, the acceptable x and y positions must be approximately consistent with the constraint in Eq. 18 since a force regime in which unwinding of the double stranded DNA occurs (~ 65 pN) is not considered here.

$$x + y = L_o - z_{eq} \quad (18)$$

The distribution of the x and y positions are not expected to have significant bias since an isotropic environment is considered. As such, the weight of the permissible positions will be approximated to be equivalent.

Using these two conditions, a constraint for the x and y position can be obtained. Since the viscous effects also have a contribution to the planar variations, a distribution was generated under the assumption of thermodynamic equilibrium in Eq. 14. The instantaneous application of force in the z direction is accounted for without the consideration of thermodynamic equilibrium. Essentially, this means that the system is forced into a harmonic state which can be described by conditions using thermodynamic equilibrium. The implementation of the previously described method is a boundary condition which restabilizes the environment. This means that the assumption of thermodynamic equilibrium is valid after the system is constrained with this method. This equilibrium is artificial in the sense that constraining the system requires higher force fluctuation magnitudes than is experimentally observed. A spatial resolution of 10% was used to limit the simulation to values similar to experimental fluctuations. Eqs. 19-20 below describe the x and y positions of the bead within this framework. The computations done in the simulation begin by considering the origin of the system in the frame of the bead. All the terms were rescaled such that the result is consistent with these relations where x_{visc} and y_{visc} are Brownian terms that account for the viscous motion and possibilities for extension of the tether.

$$y = \frac{(L_o - z_{eq})}{2} + y_{visc} \quad (19)$$

$$x = \frac{(L_o - z_{eq})}{2} + x_{visc} \quad (20)$$

The assumptions made when generating this simulation are consistent with the assumptions made in a typical Markov process (30). The distributions from which the Brownian fluctuations are determined are normal with a mean of 0 and standard deviation defined in Eq. 14. All Brownian fluctuations are extrapolated from probability distributions governed by the same rules, are time independent and intrinsically constrain how far a particle could be displaced due to a collision with the molecules in the viscous environment. This means that at any given position, the span of reasonable values classically obtainable by the particles is predefined for a given timestep. Last, each state attained by the particle is assumed to be independent of every subsequent state obtained by the particle. This is consistent with the constraints associated with classical Brownian motion (31).

Within the simulation, there are two models implemented. The first, which the simulation is initialized to using the toggle switch, is the “Trend” model. This model is intended only for used for educational purposes. It is a mixture of both model types above wherein the point at which the first model breaks down is when the new model is implemented. In other words, in absence of force, the probabilistic model, described by equations 1-16 and the constraint model, with equations 17-20 are both implemented. The first model is purely probabilistic in nature and gives predictions closer to the equilibrium state on average whereas the second model is much ‘stricter’, and the values accepted tend to be more confined. In transitioning between these models, physically unrealistic results may be occasionally observed. However, there are many benefits to this model, which will later be described in the discussions below. For stricter data collection mode, the toggle switch must be switched to “Data” where the numerical inconsistencies from the force application will not exist.

Results

Implementation and Theoretical Validation

All the data is generated using two primary tiers of code. The base code, TetherForce2, runs the simulation based on the equations previously described using for- loops. Next, a loop in the MATLAB App Designer was used to continually update the arrays containing the parameter space generated. The results generated in the base code based on the updated parameter space are immediately assigned to relevant GUI axes to plot the results as the simulation continues running in real-time. This second tier of code is the most inefficient component of this simulation since it must check for user input at every datapoint, update the parameters the base function calls at every datapoint, and plot the data at a rate specified by the user. A more comprehensive discussion of the simulation efficiency is provided in the SI Appendix. In case the code is being used solely for data generation (and not GUI based results visualization in real-time), the user can set the plot rate equal to the total number of datapoints and this will result in significantly improved runtime. While slightly more computationally intensive, it was found that continual application of a force did not significantly affect the runtime of the simulation.

Aside from the actual implementation, the simulation gives a reasonable approximation to the physical behaviors associated with a typical tethered particle-bead system. The tether particle-bead system will display a wide range of fluctuation in every direction provided that an external force is not applied to the system. The parameter space used to generate the results plots are outlined in Table 3.

Due to the lack of significant tension on the tether in the absence of an applied force, the planar variation will be more prominent since the tether-bead system will not have any rigidity as shown in the upper panel of Fig. 2.

In the presence of an applied force, it is expected that the particle will asymptotically approach maximum extension over a given time and the x-y motion will be confined to smaller scales due to the tension exerted by the tether. The lower panel of Fig. 2 was generated using the trend mode of the simulation. The details of this mode will be discussed in more detail below. In the absence of an applied force, the planar position of the particle is unrestricted and varies with a span of approximately ± 800 nm. In the presence of an applied force, the span in which the planar position varies is limited to approximately ± 200 nm. This behavior becomes more evident as the magnitude of the force increases over time since the tether will experience increasing tension. In presence of an applied force, the system asymptotically fluctuates near maximum extension. All these behaviors are consistent with theoretical expectations.

The final features implemented into this simulation are analysis plots. A useful analysis which validates the results of the simulation is the consideration of the applied and intrinsic DNA forces as a function of the net extension (r), presented in Table 2, which represents the magnitude of the 3-dimensional radius vector of the position for each cartesian components (x , y , z). In this model, the maximum extension should not greatly exceed the combined length of the bead and tether at any timepoint since the force application is limited to a regime in which dsDNA helix unwinding is not relevant as previously discussed. Fig. 3 below confirms this physical restriction both in the presence and absence of an applied force. Further validation using energy analysis is presented in the SI Fig. S3.

Model Validation Via Acoustic Force Spectroscopy Experiments

For the comparison of the simulation results with SMFS experimental data, a TPM experiment was carried out in our laboratory using an Acoustic Force Spectroscopy instrument (LUMICKS, Amsterdam, The Netherlands). A more detailed description of the experimental setup is found in the SI Appendix. Briefly, the surface of the AFS chip is incubated with anti-digoxigenin fab fragments (Sigma Aldrich, St. Louis, MO) for 20 minutes, followed by a surface passivation with bovine serum albumin (BSA, Goldbio, St Louis, MO) protein, casein protein (Sigma Aldrich, St. Louis, MO), and Pluronic F-127 nonionic surfactant (Sigma Aldrich, St. Louis, MO) in 10mM phosphate buffered saline (PBS, Sigma Aldrich, St. Louis, MO) solution for 30 minutes. Next, DNA functionalized on opposite ends with biotin and digoxigenin (Integrated DNA Technologies, Coralville, IA) is mixed with streptavidin-functionalized polystyrene beads (Spherotech Inc, Lake Forest, IL) for 30 minutes, washed twice in PBS containing BSA, casein, and Pluronic F-127 and incubated in the AFS imaging chip for 15 to 30 minutes. Finally, non-bound beads are flushed out and the remaining beads are tracked in 3D. Analysis of bead traces was performed with the software provided by LUMICKS (AFS-Analysis-G2 version, Amsterdam, The Netherlands), with slight modifications (32), and a free academic version can be found in the original publication of the AFS (21).

As can be observed in Fig. 4, the simulated RMS position values of the bead-tether systems are of the same order of magnitude and follow the same trend as the AFS experimental values, based on the simulation parameters presented in Table 4. The RMS value(21) was determined using Eq. 21, where $\bar{x}$ and $\bar{y}$ are the average x and y positions respectively of the position coordinates presented in Table 2.

$$RMS = \langle \sqrt{(x - \bar{x})^2 + (y - \bar{y})^2} \rangle \quad (21)$$

As the magnitude of applied force increases, the equilibrium RMS position the system takes, decreases. The discrepancy in experimental RMS values is due to experimental limitations such as uncertainty in the exact bead size, resolution of the AFS instrument/technique, and model limitations. Namely, an order of magnitude approximation is being conducted based on previously described assumptions so while the trends are captured, the model itself does not have a resolution allowing better certainty to be achieved for all types of analysis. In these simulations, the bead diameter was set at 3110 nm. The bead diameter does not influence the harmonic behavior and only affects the scaling of the time from the earlier described dynamic timestep. The experiments were carried out in phosphate buffer supplemented with low concentrations of proteins and polymers (see Supplementary Information), however it is assumed that those additives do not change the viscosity (33). Thus, the simulation uses the viscosity of pure water.

Case Study Demonstrating the Use of the GUI for TPM Modeling in a Classroom Setting

The following section outlines the question that could be asked by an educator, implementation of the TPM simulation toolkit to address the question, and specific steps taken within the toolkit GUI to obtain a suitable answer generated by the students. One example question posed by the educator to the students could be *“How does the time averaged <RMS> value vary as a function of tether contour length in TPM?”*

Steps taken by the students/instructor to address this specific question are briefly outlined below:

- 1) Open the simulation and show the students each of the variable meanings using the definition tab. Make an emphasis on the importance of fixing all variables except the tether length since that is the variable of interest.
- 2) Assign physically reasonable values (such as the default values) to all the variables except the tether length, consistent with the type of system that is being studied.
- 3) Choice 1: The simulations could be run prior to the lecture and the output files could be displayed in a suitable dataset format to the instructor's preference.
Choice 2. The class can be divided into groups and each group can run a simulation for a particular tether length. The results can then be combined for the final plot.
The simulated data is saved as in csv format in the same folder as the GUI, which contains labeled fields including the time (s), x/y/z position data (m), the total force (N), the applied force (N), the x/y/z components of the force (N) and the θ/ϕ angular positions.
- 4) When the simulation is completed, the script *Example_1_script* included with the sample lesson plan can be used to automatically output the RMS values for a given simulation..
- 5) The general trends could be displayed through the creation of plots as shown in Fig. 5 below. Notes about the nature of Brownian motion and the consequent variation in results among trials could be made. To account for such variability, the trials were conducted 3 times per tether lengths and an average was obtained to generate the figure.
- 6) The characteristic behavior in which the RMS value tapers off as tether length increases (26) should be highlighted by the instructor. Limitations regarding the models in general should be discussed with students.

The raw data used to generate Fig. 5 in csv format can be found on Github (<https://github.com/ChundawatLab/TPM-GUI> (34)) along with an example to create and analyze force-extension curves which is summarized in a lesson plan. The Github material also includes the GUI source code and user guide/manual. Further similar experimental comparisons are included in the SI, Figures S5-S7. Other points of discussion could be how the behavior of such curves depend on the bead radius, which tends to be varied in real SMFS experiments as well. In general, hands-on exploration of the simulation toolkit permits many modes of analysis relevant to actual experiments and provides results that are consistent with the actual trends from real-world experiments such that student learners can gain deeper insight into the concept of TPM.

Discussion

In its current setup, our simulation package provides an efficient means of generating an estimate for how a tethered particle-bead system behaves in a viscous fluid environment. The introduction of the extensibility of the tether into the worm-like chain model accounts for the elasticity of DNA when forces are applied (26). However, there are some computational limitations associated with this model. First, a basic reduction reveals that the force model has multiple solutions. As the simulation runs, the numerical solver tends to choose a solution which can best numerically minimize the equation. While numerically reasonable, the alternate solution is not physically meaningful. Second, all the descriptions made in the initial model are developed under the assumption of thermal equilibrium. When a force of sufficient magnitude is applied, the timestep calculation which depends inversely on the force gradient becomes infinitesimally small. This results in the computations of the position terms becoming unreasonable as well since position terms depend on the timestep. Consequently, a new model was implemented to preserve the

validity of the assumptions of thermodynamic stability throughout the simulation. Specifically, two models were implemented into our simulation interface: trend analysis and data collection only, as was mentioned in the Computational Framework section.

The trend model captures the overall behaviors exhibited by the system. There are some numerical inconsistencies which exist in the model. In transitioning between a state of stability and artificial stability, the planar position will display a slight increase in magnitude of fluctuations. This is consistent with expectation, since the constraints inherently require the system to achieve a length consistent with the tether length. The values attainable by the particle strictly in the probabilistic approach of the trend model tend to be closer to the equilibrium state but the span is not so strictly constrained. These methods represent different forms of numerical approximations and these discrepancies become evident as the simulation runs. The physical constraints provide an upper limit to the acceptable planar position values attained by the system whereas the probabilistic approach represents the average expected fluctuations. This is optimal for use in a classroom setting as learners can be exposed to the differences in different approximation techniques while simultaneously understanding that models are not exact. Rather, every model has a limitation, and one needs to understand the differences in the nature of models to find the best one for their needs. For the purposes of strict data collection, the data mode is designed only to use the physically constrained model, eliminating the transition point and providing data which can be used for comparisons with previous studies as shown in Supplementary Figures S5-S7 and discussed in the SI document.

A more precise model would require consideration of the bending of DNA beyond the persistence length considered in the worm-like chain model. The bending of the molecular structure of DNA on smaller scales than the persistence length is an experimentally known fact and the worm-like chain model does not account for this (35). Theoretically, this phenomenon can be accounted for through the consideration of the elastic collisions between the molecular bond sites and photons which result in small scale bending, a phenomenon characterized as Raman Scattering which can be modeled with MD and QM/MM modeling techniques (36). However, such algorithms are computationally intensive. The semi-flexible polymer description of the modified worm-like chain model which considers a single persistence length serves as a good approximation of the average of the individual base-pair contributions and is sufficient for a general user trying to understand tethered particle motion.

For general cases, we have shown there is reasonable agreement between the predictions of the TPM simulation model and AFS experimental results. Studies have been conducted which attribute specific protein functionality to the manner in which it biases the Brownian motion that the proteins are undergoing (37). While such effects are experimentally observable, there is no general way of accounting for the binding dynamics of all protein and tether combinations. This simulation is written in a manner where specific systems can easily be considered. For instance, the construction of the bead and tether are done through experimentally modifiable quantities such as material density, stress tolerance, etc. This means an extension can easily be made to transform the algorithm, perhaps through the implementation of a bead subfunction, into a simulation more representative of any given protein-ligand system of interest. As previously mentioned, the 'MarkoSiggiaVectorized.m' subfunction also provides a way for the user to easily modify the expected DNA force fluctuations by implementing corrections that more adequately account for

specific systems of study. Overall, this simulation is a template that can be easily generalized to be made relevant to any specific area of teaching.

In an educational context, the simulation is designed to promote student learning. The code is written such that the student can follow through the logic and go through the derivations done to solve for all the outputs from first principles. All the assumptions made are outlined in the comments of the MATLAB code. Although we have not tested our simulation in an actual classroom setting, several studies have been conducted to determine the effects of using computational sciences and simulations in a classroom setting to promote student STEM learning. Allowing students to work through simulations on their own rather than offering step-by-step guidance is often observed to result in better learning outcomes, although there are indications that the amount of prior knowledge a learner has may have an effect on what they are able to extract from online content (38, 39). In some similar studies, the use of simulations were found to promote knowledge integration processes, which implies that students were able to form a deeper level of understanding of the material due to their exposure to the material (40). The results for these types of studies tend to be diverse due to the extensive number of confounding variables present in such trials. However, these studies tend to compare the effectiveness of simulation-based vs. classical instruction-based learning and it is widely found that both provide similar results for evaluating pedagogical effectiveness. This simulation package with an easy-to-follow GUI was created with the intention of providing students and instructors the opportunity to quickly review a highly relevant topic in modern physics and engineering in a very short amount of time that would otherwise not be covered in a typical undergraduate curriculum.

The simulation is quick to set up and produce results, hence a few minutes are long enough to extrapolate all the noticeable trends associated with a tether particle system. The simulations could be used within a typical 60-80 min instructional lecture period. Further, the students could be tasked with using the code to do more detailed analysis such as model fitting for the data since the code outputs all relevant data. Questions could also be asked about the logic used to develop the model as the manipulations made are clearly defined. Elementary knowledge of Physics and Trigonometry is all that is necessary to follow the logic for early undergraduate students even if they cannot understand the finer details. Upper-level undergraduate students should be able to follow the logic and derive every relation considered using the given models. The use of force-extension curves, RMS position analysis, etc. for data analysis are highly valuable tools for students to gain an understanding of tethered-particle systems.

Conclusions

This work describes a simulation for tethered particle motion which comes with a customizable user interface. By utilization of the modified worm-like chain model with suitable corrections due to geometrical constraints, a simulation capable of predicting trends consistent with experimentally obtained data of tethered particle motion was demonstrated. Using the MATLAB app designer, a user interface was created which allows users to interactively modify parameters in the simulation as they would be able to do if they were conducting an actual SMFS or TPM experiment. Trials were set up to validate our simulation through consideration of behaviors in limits, verification of assumptions made, and comparison to actual experimental data as well as demonstrating a use case for teaching purposes. All these tests indicated that the simulation provides a reasonable classical description of tethered particle motion.

Some fundamental limitations of the simulation are associated with the instability in the probabilistic approach and the consideration of a fixed persistence length. Even so, having a sense of the dynamics of DNA-scale or similar polymer-tether systems will allow students to gain an intuitive understanding and insight into what can be expected from single molecule experiments using advanced techniques like optical tweezers or acoustic force spectroscopy. As advanced imaging tools gain more traction both in the real-world (e.g., point-of-care diagnostics) and academic world (e.g., single-molecule imaging of cellular biophysical phenomena), it becomes imperative to expose students (and future scientists) early on to such techniques in a classroom setting with an appropriate simulation toolkit.

Author contributions: K.R., M.H. and S.P.S.C. designed the research, M.H and K.R. conducted the research, K.R., M.H. and S.P.S.C. wrote the manuscript.

Acknowledgments: S.P.S.C acknowledges support from the Rutgers Aresty Research Center and the National Science Foundation (NSF CBET Award 1846797). K.R. acknowledges Rujuta Mokal for reviewing logic and generally providing support in the development of the code and writing of the paper. M.H. would like to thank Tan Ngo for his help in the initial stages of the study. The DNA-bead sketch in Fig. 1 was generated using Biorender.com.

Conflicts of Interest: The authors have no conflicts to disclose.

References:

1. Brinkers, S., H.R.C. Dietrich, F.H. De Groote, I.T. Young, and B. Rieger. 2009. The persistence length of double stranded DNA determined using dark field tethered particle motion. *J. Chem. Phys.* 130.
2. Heller, I., T.P. Hoekstra, G.A. King, E.J.G. Peterman, and G.J.L. Wuite. 2014. Optical tweezers analysis of DNA-protein complexes. *Chem. Rev.* 114:3087–3119.
3. Jungmann, R., M. Scheible, and F.C. Simmel. 2012. Nanoscale imaging in DNA nanotechnology. *Wiley Interdiscip. Rev. Nanomedicine Nanobiotechnology.* 4:66–81.
4. Schafer, D.A., J. Gelles, M.P. Sheetz, and R. Landick. 1991. Transcription by single molecules of RNA polymerase observed by light microscopy. *Nature.* 352:444–448.
5. Gelles, H.Y.L.R.J. 1994. Tethered Particle Motion Method for Studying Transcript Elongation by. *Biophys. J.* 67:2468–2478.
6. Igarashi, K., T. Uchihashi, T. Uchiyama, H. Sugimoto, M. Wada, K. Suzuki, S. Sakuda, T. Ando, T. Watanabe, and M. Samejima. 2014. Two-way traffic of glycoside hydrolase family 18 processive chitinases on crystalline chitin. *Nat. Commun.* 5:3975.
7. Igarashi, K. 2011. Traffic jams reduce hydrolytic efficiency of cellulase on cellulose surface (Science (1279)). *Science.* 334:453.
8. Brady, S.K., S. Sreelatha, Y. Feng, S.P.S. Chundawat, and M.J. Lang. 2015. Cellobiohydrolase 1 from *Trichoderma reesei* degrades cellulose in single cellobiose steps. *Nat. Commun.* 6:10149. <http://www.nature.com/ncomms/2015/151210/ncomms10149/full/ncomms10149.html>.
9. Hilton, M.A., H.W. Manning, I. Górnjak, S.K. Brady, M.M. Johnson, J. Zimmer, and M.J. Lang. 2022. Single-molecule investigations of single-chain cellulose biosynthesis. *Proc. Natl. Acad. Sci.* 119:e2122770119. <https://pnas.org/doi/full/10.1073/pnas.2122770119>
10. Bustamante, C.J., Y.R. Chemla, S. Liu, and M.D. Wang. 2021. Optical tweezers in single-molecule biophysics. *Nat. Rev. Methods Prim.* 1. <https://www.nature.com/articles/s43586-021-00021-6>
11. Abbondanzieri, E.A., W.J. Greenleaf, J.W. Shaevitz, R. Landick, and S.M. Block. 2005. Direct observation of base-pair stepping by RNA polymerase. *Nature.* 438:460–465.
12. Comstock, M.J., K.D. Whitley, H. Jia, J. Sokoloski, T.M. Lohman, T. Ha, and Y.R. Chemla. 2015. Direct observation of structure-function relationship in a nucleic acid -processing enzyme. *Science.* 348:352–354.
13. Svoboda, K., C.F. Schmidt, B.J. Schnapp, and B.S.M. Block. 1993. Direct observation of kinesin stepping by OT interferometry. *Nature.* 365:721–727.
14. Finer, J.T., R.M. Simmons, and J.A. Spudich. 1994. Single myosin molecule mechanics: Piconewton forces and nanometre steps. *Nature.* 368:113–119.
15. Chundawat, S.P.S., B. Nemmaru, M. Hackl, S.K. Brady, M.A. Hilton, M.M. Johnson, S. Chang, M.J. Lang, H. Huh, S.-H. Lee, J.M. Yarbrough, C.A. López, and S. Gnanakaran. 2021. Molecular origins of reduced activity and binding commitment of processive cellulases and associated carbohydrate-binding proteins to cellulose III. *J. Biol. Chem.* 296:100431. [https://www.jbc.org/article/S0021-9258\(21\)00204-0/fulltext](https://www.jbc.org/article/S0021-9258(21)00204-0/fulltext)
16. Newton, M.D., B.J. Taylor, R.P.C. Driessen, L. Roos, N. Cvetic, S. Allyjaun, B. Lenhard, M.E. Cuomo, and D.S. Rueda. 2019. DNA stretching induces Cas9 off-target activity. *Nat. Struct. Mol. Biol.* 26:185–192.
17. Floyd, K.A., A.R. Eberly, and M. Hadjifrangiskou. 2017. Adhesion of bacteria to surfaces and biofilm

- formation on medical devices. *Biofilms Implant. Med. Devices Infect. Control.* 47–95.
18. Chandrasekhar, S. 1949. Brownian Motion, Dynamical Friction and Stellar Dynamics. *Dialectica.* 3:114–126.
 19. Kumar, S., C. Manzo, C. Zurla, S. Ucuncuoglu, L. Finzi, and D. Dunlap. 2014. Enhanced tethered-particle motion analysis reveals viscous effects. *Biophys. J.* 106:399–409.
 20. Sitters, G., N. Laurens, E.J. De Rijk, H. Kress, E.J.G. Peterman, and G.J.L. Wuite. 2016. Optical Pushing: A Tool for Parallelized Biomolecule Manipulation. *Biophys. J.* 110:44–50.
 21. Sitters, G., D. Kamsma, G. Thalhammer, M. Ritsch-Marte, E.J.G. Peterman, and G.J.L. Wuite. 2014. Acoustic force spectroscopy. *Nat. Methods.* 12:47–50.
 22. Kamsma, D., R. Creyghton, G. Sitters, G.J.L. Wuite, and E.J.G. Peterman. 2016. Tuning the Music: Acoustic Force Spectroscopy (AFS) 2.0. *Methods.* 105:26–33.
 23. Deniz, A., S. Mukhopadhyay, and L. Edward. 2008. Single-molecule biophysics: at the interface of biology, physics and chemistry. *J R Soc Interface* 5:15–45.
 24. Wang, M.D., H. Yin, R. Landick, J. Gelles, and S.M. Block. 1997. Stretching DNA with optical tweezers. *Biophys. J.* 72:1335–1346.
 25. Epstein, C., and A. Mann. 2012. Measurement of the DNA Spring Constant Using Optical Tweezers. *MIT OpenCourseWare (Optical Trapping Lab Guide).* 14. <https://web.mit.edu/cepstein/Public/CEpsteinDNA.pdf>
 26. Smith, S.B., Y. Cui, C. Bustamante, and S.B. Smith. 1996. Overstretching B-DNA: The Elastic Response of Individual Double-Stranded and Single-Stranded DNA Molecules. *Science.* 271:795–799.
 27. Broekmans, O.D., G.A. King, G.J. Stephens, and G.J.L. Wuite. 2016. DNA Twist Stability Changes with Magnesium(2+) Concentration. *Phys. Rev. Lett.* 116:1–5.
 28. Happel, J., and H. Brenner. 1983. Low Reynolds number hydrodynamics. 3rd ed. The Hague: Martinus Nijhoff Publishers.
 29. Visser, E.W.A. 2017. Biosensing Based on Tethered Particle Motion. PhD Dissertation, Eindhoven University of Technology: the Netherlands. <https://research.tue.nl/en/publications/biosensing-based-on-tethered-particle-motion>
 30. Kampen N.G. Van, Ibe C. Oliver, Pinsky A. Mark, Karlin Samuel, G.T.D. 2007. Markov Process. *Science.* 1–11.
 31. Chandrasekhar, S. 1943. Stochastic Problems in Physics and Astronomy. *Rev. Mod. Phys.* 15:1–87.
 32. Hackl, M., E. V. Contrada, J.E. Ash, A. Kulkarni, J. Yoon, H.-Y. Cho, K.-B. Lee, J.M. Yarbrough, C.A. López, S. Gnanakaran, and S.P.S. Chundawat. 2022. Acoustic force spectroscopy reveals subtle differences in cellulose unbinding behavior of carbohydrate-binding modules. *Proc. Natl. Acad. Sci.* 119:e2117467119. <https://pnas.org/doi/full/10.1073/pnas.2117467119>
 33. Galush, W.J., L.N. Le, and J.M.R. Moore. 2012. Viscosity behavior of high-concentration protein mixtures. *J. Pharm. Sci.* 101:1012–1020.
 34. Ramdin, K.A., M. Hackl, and S. Chundawat. 2023. TPM-GUI. (Version 1.0.0) [Computer software]. <https://github.com/ChundawatLab/TPM-GUI>.

35. Wiggins, P., T. Van Der Heijden, F. Moreno-Herrero, A. Spakowitz, R. Phillips, J. Widom, C. Dekker, and P.C. Nelson. 2006. High flexibility of dna on short length scales probed by atomic force microscopy. *Nat. Nanotechnol.* 1:137–141.
36. Rao, S., S. Raj, B. Cossins, M. Marro, V. Guallar, and D. Petrov. 2013. Direct observation of single DNA structural alterations at low forces with surface-enhanced Raman scattering. *Biophys. J.* 104:156–162.
37. Ishii, Y., Y. Taniguchi, M. Iwaki, and T. Yanagida. 2008. Thermal fluctuations biased for directional motion in molecular motors. *BioSystems.* 93:34–38.
38. Chang, K.E., Y.L. Chen, H.Y. Lin, and Y.T. Sung. 2008. Effects of learning support in simulation-based physics learning. *Comput. Educ.* 51:1486–1498.
39. Al Mamun, M.A., G. Lawrie, and T. Wright. 2022. Exploration of learner-content interactions and learning approaches: The role of guided inquiry in the self-directed online environments. *Comput. Educ.* 178:104398. <https://doi.org/10.1016/j.compedu.2021.104398>
40. Taub, R., M. Armoni, E. Bagno, and M. Ben-Ari. 2015. The effect of computer science on physics learning in a computational science environment. *Comput. Educ.* 87:10–23.
41. Beausang, J.F., C. Zurla, L. Finzi, L. Sullivan, and P.C. Nelson. 2007. Elementary simulation of tethered Brownian motion. *Am. J. Phys.* 75:520–523.

Figure Legends

Fig 1: Overview of Graphic User Interface for TPM simulation developed in this work. The left panel accepts inputs for simulation parameters, type of simulation model and has a simulation start and reset button, which updates and clear the plots. The middle panel includes the simulated data plots which are updated as the simulation runs. At the bottom of the middle panel there is a sketch of the system being simulated. The right panel includes different options for force application which can be modified during the simulation runtime or before it starts. There is also an energy validation plot at the bottom right corner of the GUI.

Fig 2: Simulation model predictions for TPM with or without applied force on the system. Upper Left Panel: Planar position (XY) vs. Time in absence of applied force. Upper Middle Panel: Normal position (Z) vs. Time in absence of applied force. Upper Right Panel: DNA Force vs. Time in absence of applied force. Lower Left Panel: Planar position (XY) vs. Time in presence of 25 pN Applied Force. Lower Middle Panel: Normal position (Z) vs. Time in presence of 25 pN Applied Force. Lower Right Panel: DNA Force vs. Time in presence of 25 pN Applied Force. All panels were generated using the trend mode of the simulation GUI.

Fig. 3: Force extension curves in the absence of force (left) and in the presence of a force ramp up to 25 pN (Right). The contour length was set to 1180 nm as indicated in Table 3.

Fig. 4: Simulation model trends agree well with TPM observed in Acoustic Force Spectroscopy experiments. Left: Experimental and simulated average x/y-RMS position values with application of constant force for 1800 nm DNA strands attached to a polystyrene bead of 3110 nm diameter in buffer at room temperature. Right Panel: Experimental and simulated average XY-RMS position values with application of constant force for a 500 nm DNA strand attached to a polystyrene bead of 3110 nm diameter in buffer at room temperature. The larger discrepancy between experimental and simulated RMS for 500 nm tethers are discussed in the text.

Fig. 5: Case study results demonstrating how the TPM GUI can be used by an educator in a classroom setting. Average RMS position calculated from repeating the simulation 3 times per tether length and taking the average of the obtained results. The red vertical bar represents the standard deviation of 3 experimental replicates.

Tables

Table 1: Parameters considered in the tethered particle motion simulation. Static refers to fields which are assigned before the TPM simulation starts. Dynamic refers to fields which can be modified during the simulation.

Simulation Parameters	Parameter Notation
Length of Simulation (Static)	n
Tether Persistence Length (Static)	L_p
Tether Length (Static)	L_o
Viscosity of Environment (Static)	η
Temperature of Environment (Static)	T
Radius of Bead (Static)	R_B
Density of Bead Material (Static)	ρ
DNA Force -Net (Dynamic)	$F = \langle F_x, F_y, F_z \rangle$

Table 2: Summary of parameter notations used for the MATLAB code and variable descriptions with reference to the defining equation (if applicable). Tether extension (r) and DNA force (net) are the vector quantities of the position and DNA force in x , y and z , respectively.

Parameter of Interest	Notation	Equation
Position (cartesian)	x, y, z	-
Displacement (cartesian)	$\Delta x, \Delta y, \Delta z$	8, 9, 10
Tether Extension (net)	r	-
Spherical angles	θ, φ	6, 7
Potential Energy	PE, ΔPE	15
Kinetic Energy	KE, ΔKE	16
DNA Force (cartesian)	F_i	1, 3, 4, 5
DNA Force (net)	F	-
Time-averaged root-mean-square fluctuation	RMS	21

Table 3: Sample parameter space used to generate Fig. 2 result plots. The parameters are approximately consistent with the simulation parameters considered in Beausang’s (41) implementation.

Parameter (unit)	Value
n (-)	10000
L_o (nm)	1180
L_p (nm)	72
η (Pa*s)	0.0089
T (°C)	25
R_B (nm)	50
Plot Rate	10000

Table 4: Sample parameter space used to generate Fig. 4 result plots.

Parameter (unit)	Value
n (-)	10000
L_o (nm)	1800/500
L_p (nm)	50
η (Pa*s)	0.0089
T (°C)	25
R_B (nm)	1550
Plot Rate	10000

Supporting Information (SI)

Visualization of Tethered Particle Motion with a Multidimensional Simulation

Khovesh Ramdin, Markus Hackl, Shishir P.S. Chundawat

SI Table of Contents:

1. Documentation for TPM simulation model package
2. Experimental Setup of TPM using Acoustic Force Spectroscopy (AFS)
3. Supplementary Information of TPM Model
4. Supplementary Information References

Note: See GitHub (<https://github.com/ChundawatLab/TPM-GUI>) for Matlab files/dataset, sample lesson and user guide associated with MATLAB-based TPM app.

1. Documentation for TPM simulation model package

Definitions

Batch Processing: Each simulation length is determined by the parameter n , which represents the total number of data points. The data points are generated in batches continuously and appended into a single output array based on the user specified points per plot until the n^{th} data point is reached.

Points Per Plot: The number of data points being simulated per batch are referred to as points per plot. Each batch is plotted when the data points for that given batch is done being simulated by the base function TetherForce2. This process is continued until the user specified total number of data points are reached.

User Specification: Using the graphic user interface provided, the user can input parameters of their choice to be simulated. Except for the Force Slider, all of the fields must be specified prior to the start of the simulation.

Memory Terms: The overall simulation is run using the user specified parameters. The batches in which each of the simulations occur have initial conditions based on the end state of the prior batch simulated to ensure continuity. For the first batch produced, the initial conditions are set such that the planar position is at 0 and the normal position is at the bead radius, R_b . All of the force contributions and time are also initialized to 0.

Planar Position: This is the plane containing x-y values from a cartesian reference frame.

Normal Position: This is the plane perpendicular to the x-y plane (z direction) from a cartesian frame.

Graphic User Interface: The terminal the user will interact with to produce their desired simulation. This includes fields that allow for user specification in addition to different graphic outputs which summarize the simulation results.

Base Function

TetherForce2: This function accepts user specified parameters from the graphic user interface and the memory terms as input parameters. Each simulation is run for the user specified batch size. This function outputs arrays containing the planar and normal positions for each data point, the time elapsed since the start of the simulation for every given data point, the x, y and z components of the forces acting on the bead due to collisions between the particle and environment and the extension, θ and ϕ positions of the bead from a traditional spherical coordinate system.

Subfunctions

These are used to increase the readability of the code. They require inputs from the base function, tend to perform some form of extended computation and output results needed in the base code for the simulation to continue running.

GetNextTimeStep: This subfunction accepts all parameters necessary for the computation of the time-step. Some of these include user-specified parameters and others include parameters obtained prior within the same simulation. For greater discussion of the logic, specific parameters used and manner of computation, reference the manuscript associated with this work.

MarkoSiggiaVectorized: This subfunction utilizes a numeric solver to obtain force values based on the modified Marko-Siggia worm-like chain model (1) for each direction in a cartesian system. The input parameters for this subfunction are described explicitly in the manuscript. The details of the accepted parameters are presented in the manuscript associated with this work.

getDragCoefXY: This subfunction outputs the planar drag coefficient based on boundary conditions constraining the system. A more thorough discussion can be found in the manuscript.

getDragCoefZ: This subfunction outputs the normal drag coefficient based on boundary conditions constraining the system. A more thorough discussion can be found in the manuscript.

Graphical User Interface:

Callback Functions: These callbacks have user defined functions embedded in them which perform a task based on given input. For instance, the user interface has a start button which initiates the simulation based on the user inputted parameters and a reset button which clears the generated event.

Features: The simulation includes numeric edit field for the static parameters, a toggle switch for the model type, two buttons for the start and reset functions, a set of 3 linked buttons of which only one can be chosen at a time, a slider for the dynamic editing of the force during runtime and plots of relevant parameters. A more in-depth description of how each function was implemented is presented as comments in the code and throughout the paper.

Overall, the flow chart in Fig. S1 below summarizes how these components interact with each other.

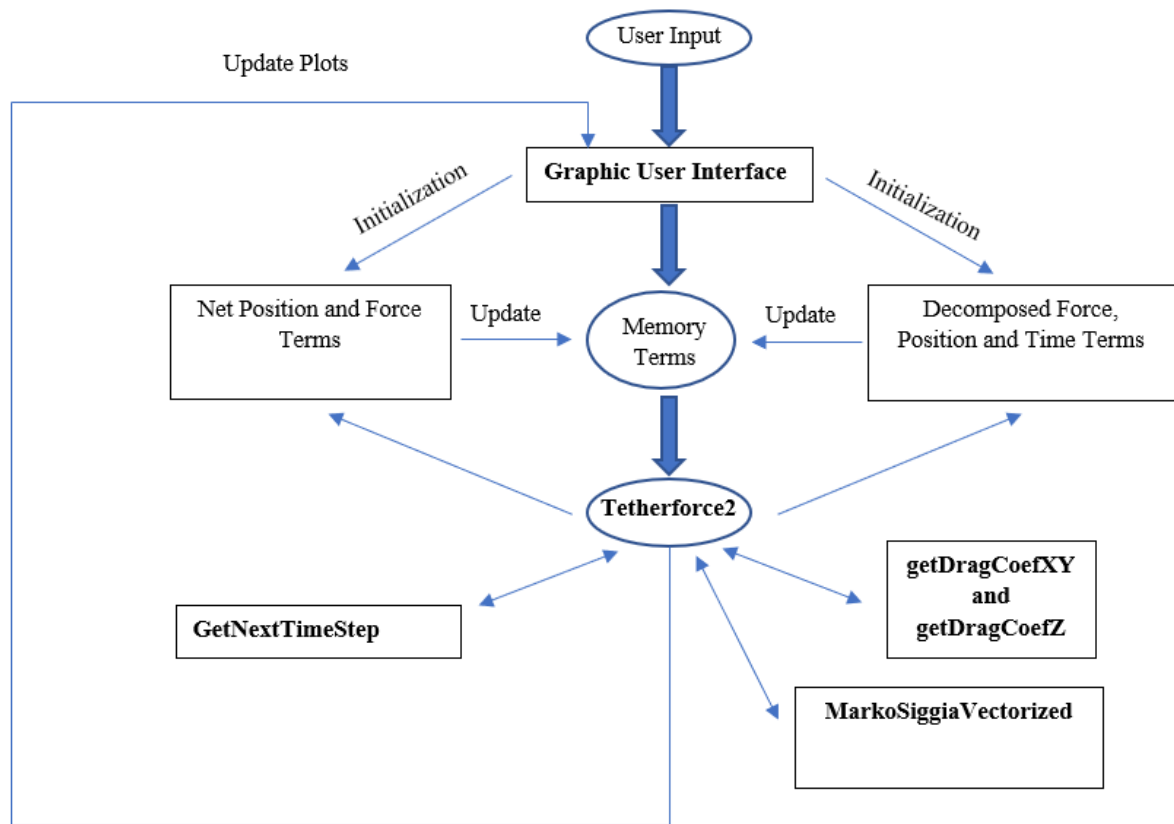


Fig. S1: Flow chart with functional dependencies and model update logic.

2. Experimental Setup of TPM using Acoustic Force Spectroscopy (AFS)

Chemicals: 10x Phosphate buffer saline (PBS), Pluronic F-127, casein technical grade, bleach solution, sodium thiosulfate and anti-digoxigenin Fab fragment antibodies (Roche, 11214667001) were purchased from Sigma Aldrich, USA. Bovine serum albumin, fraction V (BSA) was purchased from Goldbio, USA. Streptavidin-coated polystyrene particles with a nominal diameter of 2.16 μm (SVP-20) and 3.11 μm (SVP-30) were purchased from Spherotech Inc, USA.

Buffers. All AFS experiments were carried out in working buffer (WB) containing 10 mM PBS at pH 7.4 supplemented with 0.31 mg/ml BSA and casein and 0.19 mg/ml Pluronic F-127, respectively. In addition, two blocking buffers were used to passivate the surface before the experiment. Buffer B1 consists of 10 mM PBS supplemented with 2.5 mg/ml BSA and casein. Buffer B2 consists of 10 mM PBS supplemented with 2.2 mg/ml BSA and casein and 5.6 mg/ml Pluronic F-127 respectively. All buffers were degassed in a vacuum (-90 kPa) for 30 minutes.

DNA tethers. Linear double-stranded DNA tethers were synthesized in one step by PCR using the pEC-GFP-CBM3a plasmid(2) as a template and 5' modified primers. The biotin-modified primer (forward primer, 5'-biotin-C6-GGCGATCGCCTGGAAGTA) and digoxigenin modified primer (backward primer, 5'-DIG-NHS- TCCAAAGGTGAAGAACTGTTACC) were purchased from Integrated DNA Technologies, Inc. USA. The whole plasmid (5.4 kb) was amplified, then purified using the PCR Clean-up kit (IBI Scientific USA) resulting in a linear DNA tether of $\sim 1.8 \mu\text{m}$ length with one modification on each end of the DNA. Amplification and product purity was verified by gel electrophoresis.

Tethered bead preparation for single-molecule force spectroscopy. Single-molecule experiments were carried out on a G1 AFS instrument with G2 AFS chips provided by LUMICKS B.V. After a 0.5 ml rinse of bleach followed by neutralization with 0.5 ml of 0.5M sodium thiosulfate and wash of 2 ml DI water, the chips were incubated for 20 minutes with anti-digoxigenin fab fragments dissolved in PBS (20 $\mu\text{g/ml}$), where they non-specifically bind to the AFS glass surface. Next, the surface was passivated with B1 and B2 buffer for 15 minutes each and rinsed with WB. The dig-DNA-biotin tethers were diluted to 6 pM in WB. The bead-DNA-construct was prepared as follows. First, 15 μl streptavidin-coated beads and DNA tethers were mixed to yield between 5-15 DNA tethers per and incubated on a rotisserie for 30 minutes. Next, the functionalized beads were washed by spinning the sample down on a table-top centrifuge, removing the supernatant, and resuspending in 100 μl WB twice. After the second removal of supernatant, the DNA-bead construct was resuspended in 30 μl WB.

The DNA-bead construct was flushed through the AFS chip and incubated for 15-30 minutes. Non-bound beads were subsequently washed out with WB at a flow rate of 2 $\mu\text{l/min}$ using a syringe pump (New Era Pump Systems Inc., USA). A small force of ~ 0.2 -0.5 pN was applied to speed up the flushing step.

Bead tracking. Tracking and analysis of the beads were accomplished using the software package provided by LUMICKS, with slight modifications to allow efficient export of traces and associated tethers statistics as well as force-distance curves to a spreadsheet. The procedure for identifying a single-molecule tether, force calibration, and rupture force determination is described in detail

elsewhere (3). The beads were tracked at 20 Hz using a 10x magnification objective. The trajectory of the beads without applied force was monitored for 8-10 minutes to determine the point of surface attachment (anchor point). Next, the force on each bead was calibrated by applying a constant amplitude for 2-4 minutes. Typically, 2-3 different amplitude values were used to build the calibration curve between the applied amplitude and effective force on each bead. Single-molecule tethers were identified by the root-mean-square fluctuation (*RMS*) and symmetrical motion (*Sym*) of the bead around the anchor point during the time frame for anchor point determination. Typical values of single-molecule tethers for *RMS* and *Sym* are in the range between 700-1000 nm and 1.0-1.3 respectively for 1800 nm tethers. During force calibration, the diffusion coefficient of the bead and the force were used as fit parameters. This diffusion coefficient was compared to the diffusion coefficient determined by the Stokes-Einstein relation and was in the range between 0.8-1.2 for single tethers.

3. Supplementary Information of TPM Model

A. Efficiency Analysis

Fig. S2 below outlines the efficiency of the simulation with and without the application of force by comparing the overall runtime to the rate of plotting.

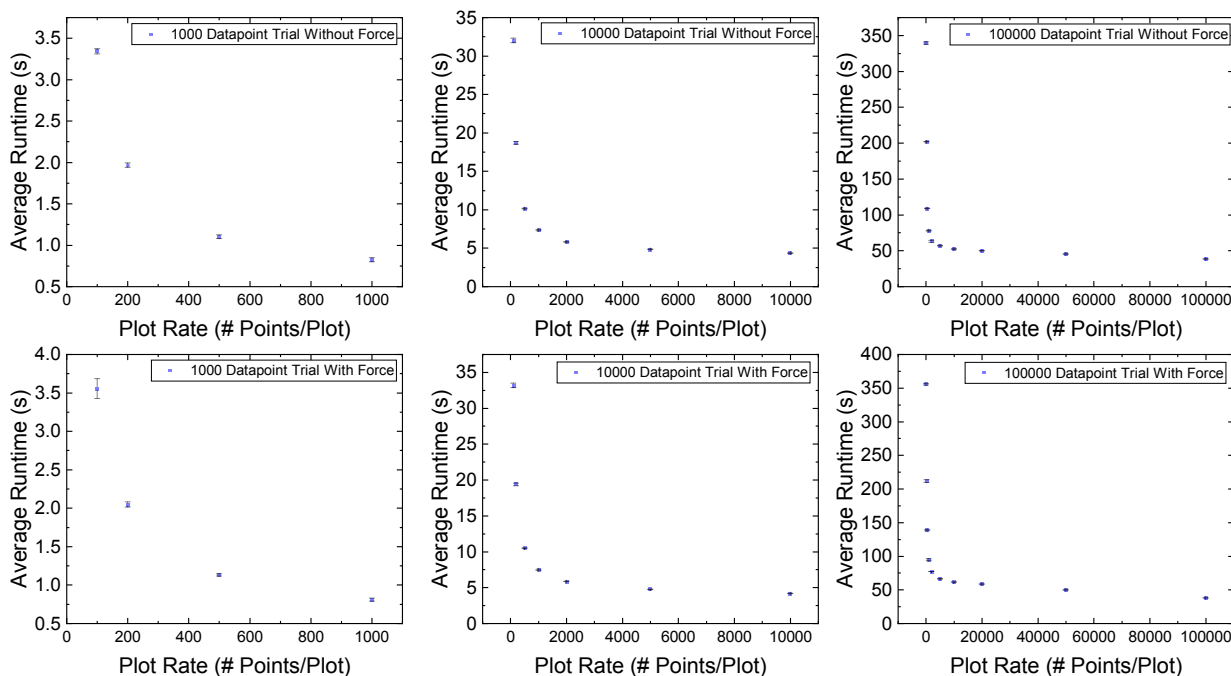


Fig. S2: Runtime vs. Plot-Rate without Force Ramp (Upper 3 plots) and with Force Ramp (Lower 3 plots)

The amount of time it takes for data to be generated increases linearly as can be observed from the runtimes that approach the horizontal asymptote in each of the trials above. As the number of plots that must be generated decreases for a specific trial, the runtime can be modeled with an exponentially decaying curve.

Another analysis of the efficiency can be considered using Big-O notation in computer science. This formalism characterizes how efficient an algorithm is through the consideration of the space and time complexities of the algorithm. This simulation is designed using sequential for loops. That is, the for loop which generates the data occurs prior to the for loop which plots the data in the app designer. This means that there are 2 first order complexities in this code since all the implicit computations are computationally constant in time (4). Eq. (i) below provides a general description of the complexity of sequential first order for loops (4).

$$f(n, m) = n * O(1) + m * O(1) \quad (i)$$

The measure of complexity is done in the limit that the spatial resources approach infinity meaning that some elementary reductions can be made (5):

$$f(n, m) = O(n + m) \quad (ii)$$

$$f(n, m) = O(\max(n + m)) \quad (iii)$$

The first loop in the sequence generates the data and the second loop plots all the data. This means that $n = m$ yielding Eq. (iv):

$$f(n, n) = O(2n) \quad (iv)$$

Since the big O notation is a measure of complexity in a limit, pre-factors can be omitted yielding the result (6):

$$f(n) = O(n) \quad (v)$$

To verify that there is a linear dependence between the data points and runtime, a series of simulations were considered, and runtimes were extrapolated using the inbuilt MATLAB stopwatch. Five repeated trials for each number of data points were conducted and the average runtime was obtained. Verification of linearity can be achieved through the consideration of Pearson's correlation coefficient (7). A linear regression analysis yielded a correlation coefficient of 0.99997. This means that there is a very strong, positive linear relationship between the number of data points in a trial and the runtime in a trial. This indicates that the Big-O analysis was appropriately conducted. A linear dependence in the Big-O formalism corresponds to a decent efficiency.

Overall, there is a point in which the resources necessary to generate the plots continuously greatly exceed the resources necessary to generate the data points. Finding the ideal simulation speed can be done by identifying the data point generation time and finding the point in which the exponential behavior begins dominating. The ideal plot rate will typically exist near this turning point although the total runtime will depend on the specifications of the device used to run the simulation.

B. Further Validation and AFS Experimental Comparisons

Energy analysis was used to validate the simulation. Although the total energy of the system is conserved, the energy imparted onto the environment is not trivially obtainable. Instead, a consideration of the distribution of the kinetic and potential energy gives insight into the validity of the chosen parameter space. This model is generated under the assumption that the viscous forces dominate the forces exerted due to the particle on the environment. That means that this model serves as a reasonable approximation if the potential energy of the tether-bead system dominates the kinetic energy. For the system considered in this set of validation runs, this expectation is verified in the left panel of Fig. S3 below. The right panel of Fig. S3 presents a case

where the simulation results are not valid due to the low viscosity. Users can use these plots to ascertain if the simulation provides a reasonable approximation for their parameter space.

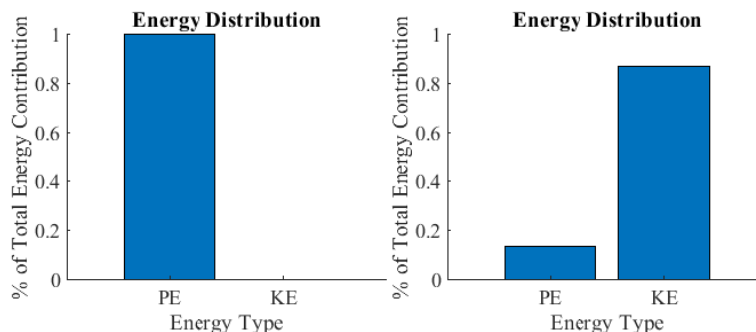


Fig. S3: Potential and kinetic energy of the tether bead system based on the TPM simulations. Left Panel: Energy distribution for viscosity = $8.9e-4$ Pa*s. Right Panel: Energy distribution for viscosity = $2e-6$ Pa*s. All other simulation parameters remained identical.

An additional AFS experiment was conducted to verify the validity of the simulation for a bead diameter of 2160 nm. In this case, there was a greater asymmetry in the direction of the force application. The generated model only accounts for a force application in the z-direction so it cannot explicitly predict the experimental results with high confidence for an XY RMS analysis. Even so, the general trend is captured as is depicted in Fig. S4 below.

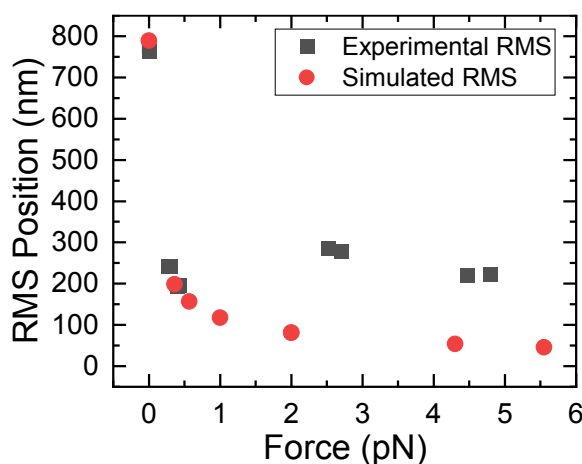


Fig. S4: Experimental and simulated average XY-RMS position values with application of constant force for 1800 nm DNA strands attached to 2160 nm polystyrene bead.

Next, a comparison to previous simulations and theoretical fits were considered. Using the parameter space outlined in Table S1 below, simulations attempting to replicate results obtained by Nelson et. al. (8) and from Pouget et. al. (9) Figure 7 from Nelson et. al. (8) is particularly considered since it summarizes results from multiple studies, providing a reference for the use cases of this simulation.

Table S1: Sample parameter space used to generate Fig.S5 and Fig.S6 result plots.

Parameter (unit)	Value
n (-)	10000
L_o (nm)	Variable
L_p (nm)	43
μ (Pa*s)	0.0089
T (°C)	25
R_B (nm)	115/100
Plot Rate	10000

As can be seen from both Figures S5 and S6, within the error tolerances of the values extracted from the plots in Nelson et al (8) and Pouget et al (9) and considering a fixed error for the developed simulation, the data generated is consistent with the predictions in terms of trend. Further, it tends to follow the theoretically predicted curve as expected in both the rigid body approximation and flexible polymer spans which are presented in Eq (vi) and (vii (9)), respectively.

$$R \sim L_o \quad (vi)$$

$$R \sim N^{3/5}, N = L_o/L_p \quad (vii)$$

An appropriate fit parameter was used to scale the curves in the plots. Eq. (vi) is a natural consequence of the rigid chain model considered since little bending occurs for extensions less than persistence length. Eq. (vii) presented by Pouget was initially determined by Flory (10). In these cases, a fixed error is considered to account for the uncertainty in generated fit parameters in the mentioned papers and simulation uncertainties present due to a lack of direct comparison in the case of Figure S5.

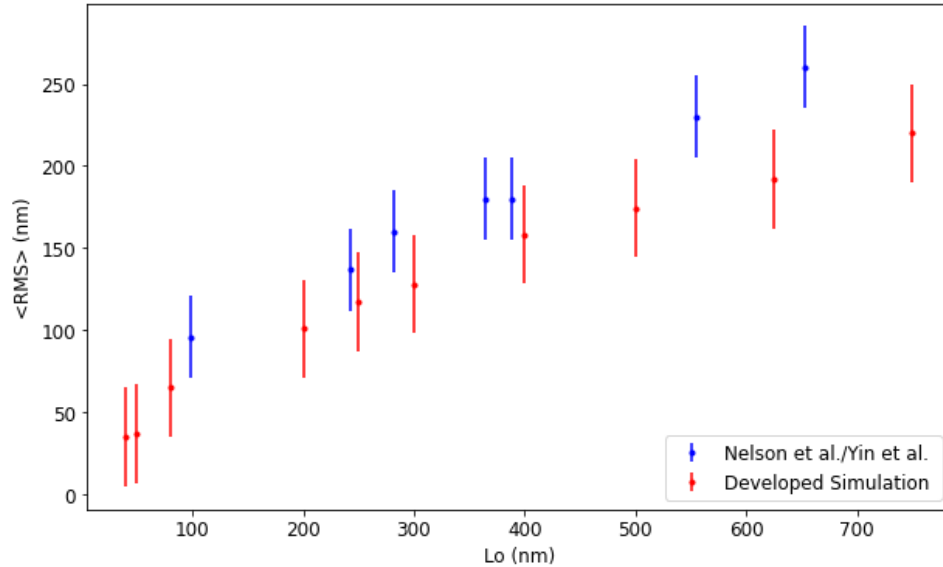


Figure S5: Simulation considering a bead radius of 115 nm at various tether lengths with an persistence length of 43 nm(8). Results from this study are shown in red while results from previous work are shown in blue.

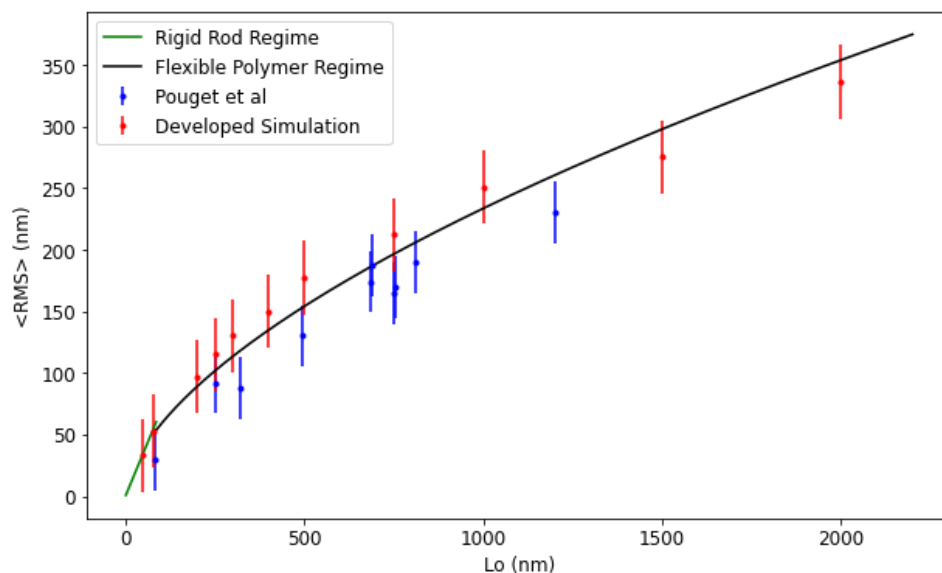


Figure S6: Simulation considering a bead radius of 100 nm at various tether lengths with an persistence length of 43 nm(8). Results from this study are shown in red while results from previous work are shown in blue/blue/black.

Figure S7 was generated using the same parameters in Table S1 except persistence length was varied instead of tether length and the bead radius was fixed at 50 nm. As the persistence length increases, it is found that the $\langle \text{RMS} \rangle$ position value increases. This makes sense within the limitations of this model since the persistence length provides a description of the bending point of the tether. A higher persistence length means that the scales at which the tether bends becomes larger for a given tether length. Consequently, less bending from the initialized states will occur for larger tether lengths and therefore a larger $\langle \text{RMS} \rangle$ position value is observed.

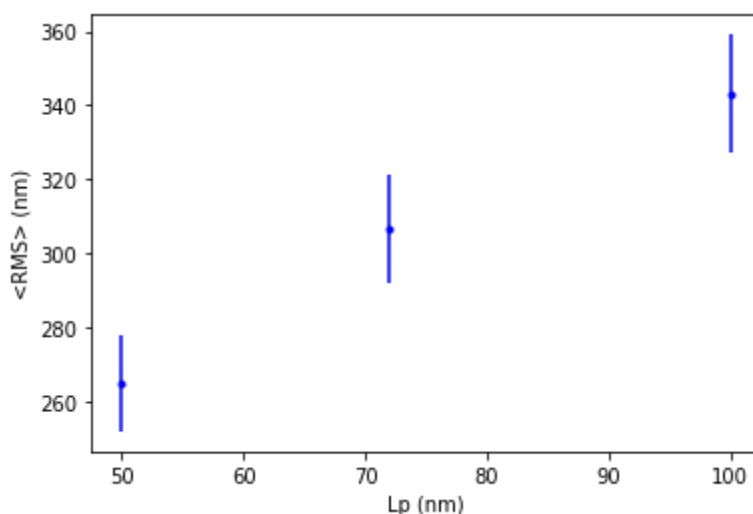


Figure S7: Simulation considering a bead radius of 50 nm at various persistence lengths with a tether length of 1180 nm(8). Results from this study are shown in blue.

4. Supplementary Information References

1. Wang, M.D., H. Yin, R. Landick, J. Gelles, and S.M. Block. 1997. Stretching DNA with optical tweezers. *Biophys. J.* 72:1335–1346.
2. Hackl, M., E. V. Contrada, J.E. Ash, A. Kulkarni, J. Yoon, H.-Y. Cho, K.-B. Lee, J.M. Yarbrough, C.A. López, S. Gnanakaran, and S.P.S. Chundawat. 2022. Acoustic force spectroscopy reveals subtle differences in cellulose unbinding behavior of carbohydrate-binding modules. *Proc. Natl. Acad. Sci.* 119:e2117467119.
3. Sitters, G., D. Kamsma, G. Thalhammer, M. Ritsch-Marte, E.J.G. Peterman, and G.J.L. Wuite. 2014. Acoustic force spectroscopy. *Nat. Methods.* 12:47–50.
4. Bae, S. 2019. JavaScript Data Structures and Algorithms: An Introduction to Understanding and Implementing Core Data Structure and Algorithm Fundamentals. <https://doi.org/10.1007/978-1-4842-3988-9>
5. Danziger, P. 2010. Big O Notation Basics. 2–7.
6. Gayathri Devi, S., K. Selvam, and S.P. Rajagopalan. 2011. An abstract to calculate big o factors of time and space complexity of machine code. *IET Conf. Publ.* 2011:844–847.
7. Schneider, A., G. Hommel, and M. Blettner. 2010. Lineare regressionsanalyse - Teil 14 der serie zur bewertung wissenschaftlicher publikationen. *Dtsch. Arztebl.* 107:776–782.
8. Nelson, P.C., C. Zurla, D. Brogioli, J.F. Beausang, L. Finzi, and D. Dunlap. 2006. Tethered particle motion as a diagnostic of DNA tether length. *J. Phys. Chem. B.* 110:17260–17267.
9. Pouget, N., C. Dennis, C. Turlan, M. Grigoriev, M. Chandler, and L. Salomé. 2004. Single-particle tracking for DNA tether length monitoring. *Nucleic Acids Res.* 32.
10. Flory, J.P. 1967. Principles of Polymer Chemistry. Ithaca, NY: Cornell University Press.

User-guide for TPM GUI

Brief overview

The GUI was developed using MATLAB® version 2022b. Compatibility with other versions was not tested. Some routines run into errors if the user input is incorrect. Commonly encountered errors and how to avoid them are described in this document. The GUI simulates the x,y, and z position of a tethered particle as found in tethered particle motion (TPM) experiments. In addition to the position, the forces acting on the system (external applied force and total force) are simulated. The plot update rate (PPP) affects the performance significantly. It is advised to begin using the GUI with the default values. Basic MATLAB skills are needed to use the GUI.

Graphical User Interface

GUI layout

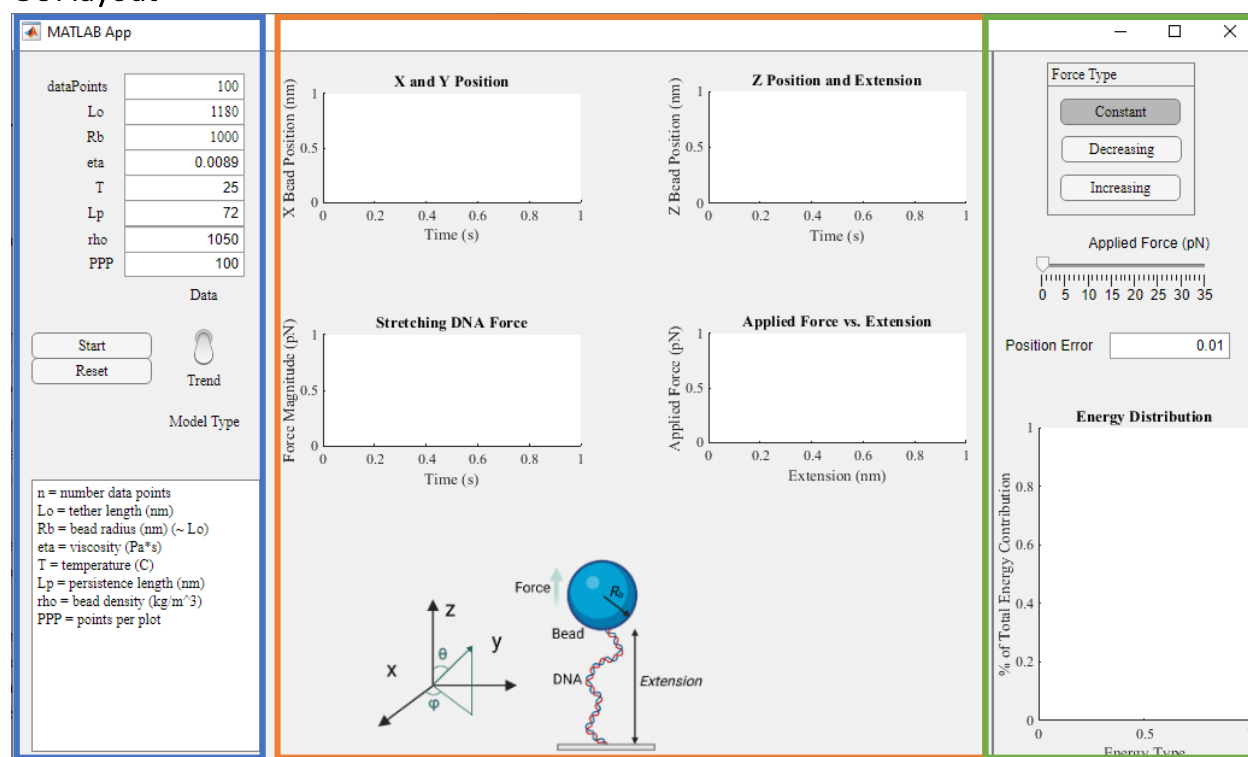


Figure 1: GUI overview and descriptions.

Figure 1 shows the GUI after startup. The section framed in blue (left) contains a list of variables which the user can modify; a description of each is found at the bottom of the section. The start button initiates the TPM simulation, whereas the reset button clears the workspace. The toggle switch allows the user to switch between two simulation types: 1) Data and 2) Trend, see section Model descriptions. The GUI does not have a stop button and the user will need to abort execution in the MATLAB command window by pressing "Ctrl+C". Depending on the state of the simulation, "Ctrl+C" needs to be pressed more than once to fully exit the simulation. Simulation data will not be saved if simulation is aborted.

The orange framed section (middle) displays the position and forces of the TPM simulation and a corresponding sketch. The force is currently only applied in z, but the code can be expanded to include a force application in x/y.

The green section (right) contains the force application mode: 1) constant, 2) decreasing, 3) increasing along with a slider to apply the force in z. While the constant button is pressed, the force is applied for the whole duration of the simulation at the value specified by the force slider. In contrast, the decreasing and increasing buttons create a linearly decreasing or increasing force during the duration of the simulation, with the maximum force being set by the slider and the minimum force being zero. The field position error relates to the constraint model (see section Model descriptions). The conservation of energy is displayed as a set of bar graphs that show potential and kinetic energy, respectively. This allows the user to quickly verify if the input parameters (such as viscosity) were physically reasonable and resulted in a dominance of potential over kinetic energy.

Model descriptions

The GUI uses two models for the simulation: 1) Data and 2) Trend as set by the toggle switch. The Data simulation is based on a physical description of tethered particle motion as described in the main of the publication (equations 1-16). The Trend model is a hybrid between the Data model and a constrained model based on a Markov process. The advantage of the Data model is that it obeys the probabilistic description of the TPM system, but can fail if large force jumps from one simulation point to the next occur. In some situations, the simulation is also slower than the hybrid model. In contrast, the Trend model switches between the probabilistic and the constrained simulation and as such the simulation output (position, forces, etc.) contains a larger errors in the force and position estimation, but runs more stable and, in general, faster than the Data model. Nevertheless, it perfectly captures any trends of the system, e.g. a reduction in fluctuations when the force is increased in a typical force-extension experiment. A randomly chosen parameter called “Position Error” can be adjusted to more closely match the constrained model to the probabilistic model and mainly influences the time scale. The default value of 0.01 works well for most TPM scenarios.

Starting the GUI

Unpack the zip file and open “TPMGUI.mlapp” as shown in Figure 2.

Name	Size	Item type
 getDragCoefXY.m	1 KB	MATLAB Code
 getDragCoefZ.m	1 KB	MATLAB Code
 GetNextTimeStep.m	1 KB	MATLAB Code
 MarkoSiggiaVectorized.m	2 KB	MATLAB Code
 TetherForce2.m	13 KB	MATLAB Code
 TPM sketch.png	438 KB	PNG File
 TPMGUI.mlapp	230 KB	MATLAB App

Figure 2: Opening the GUI.

MATLAB opens alongside the GUI and by default the active folder contains all subfunctions needed to run the GUI as shown in Figure 3.

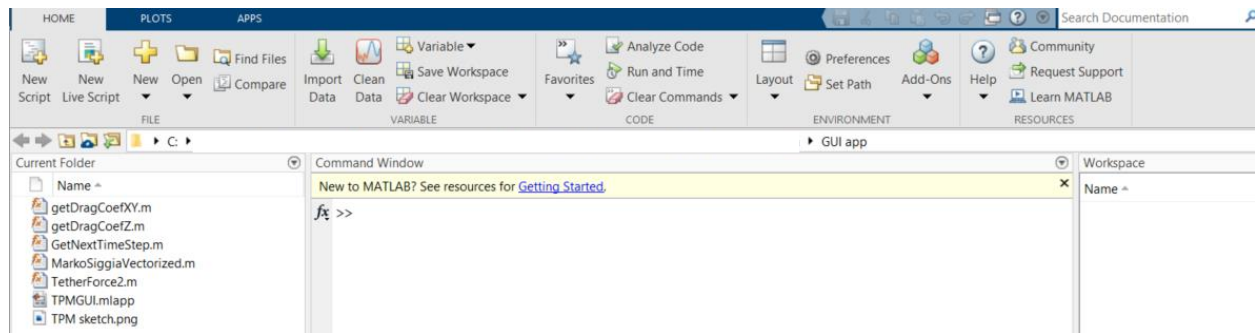


Figure 3: MATLAB window showing the current folder.

In parallel, the GUI opens up as a separate window as shown in Figure 1 and is ready to use.

Example 1: Simulation with a constant force turned on, off and modified during simulation

After starting the GUI, keep all parameters as default except “dataPoints”, which is set to 1000, “PPP” set to 10 and “Data” mode. This means that the simulation will contain 1000 data points and all plots will be updated every 10 computations. These settings will simulate less than one minute of data, where the user can manually apply forces to the TPM system.

1. Begin by pressing the “Start” button, watch as the graphs in the middle sections begin to display the simulation output.
2. Use the force slider on the right to apply a force and watch how the bead is being “pulled up” from the surface as indicated by a rise in the z position.
3. Move the force slider back to zero and watch how the bead “drops” back to the surface.
4. Repeat the force application for various forces until the end of the simulation.
5. At the end of the simulation, a pop-up window will appear that prompts the user to save the data as shown in Figure 4 (red arrow).
6. Enter a file name and press OK. The simulation data will be saved as a csv file in the same folder where the GUI is located.
7. After saving the data, the GUI updates the energy distribution bar plot. For a “realistic” simulation, the potential energy (PE) should be significantly larger than the kinetic energy (KE) as seen in Figure 5.
8. Press the “Reset” button and repeat the simulation with more data points, or vary the bead radius or tether length.

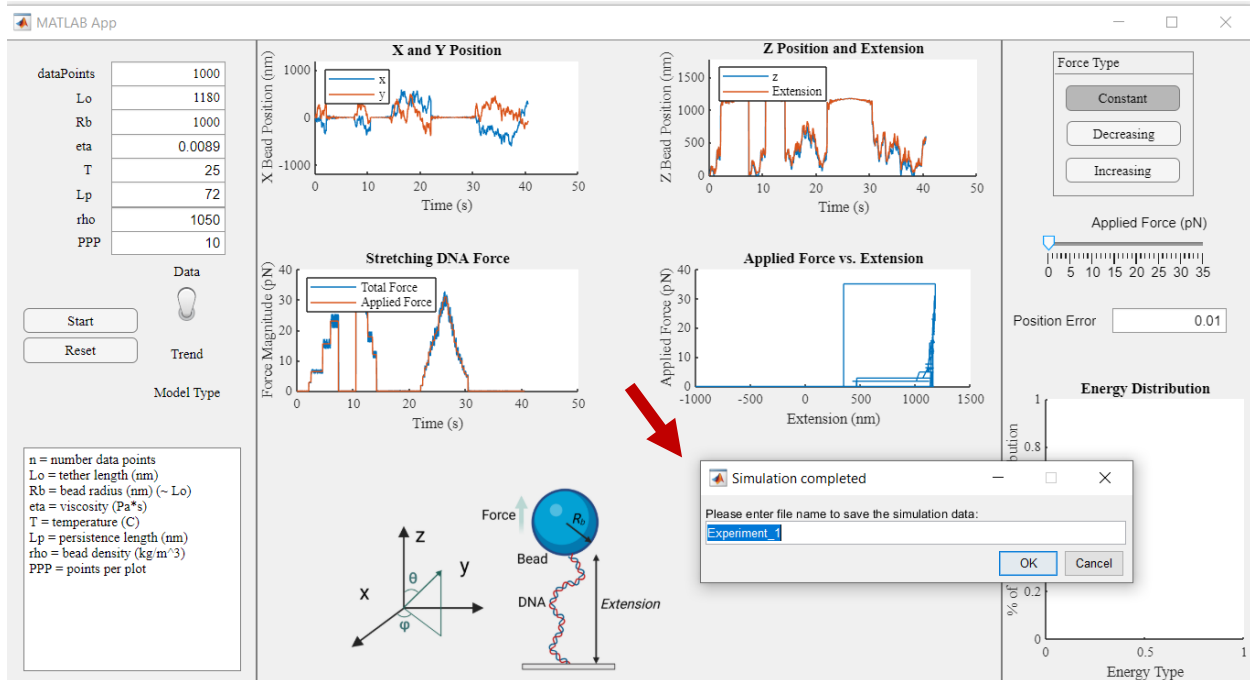


Figure 4: GUI after constant force simulation. After completion of the simulation, a pop-up window will appear (indicated by the red arrow) that prompts the user to specify a file name and save the simulation data.

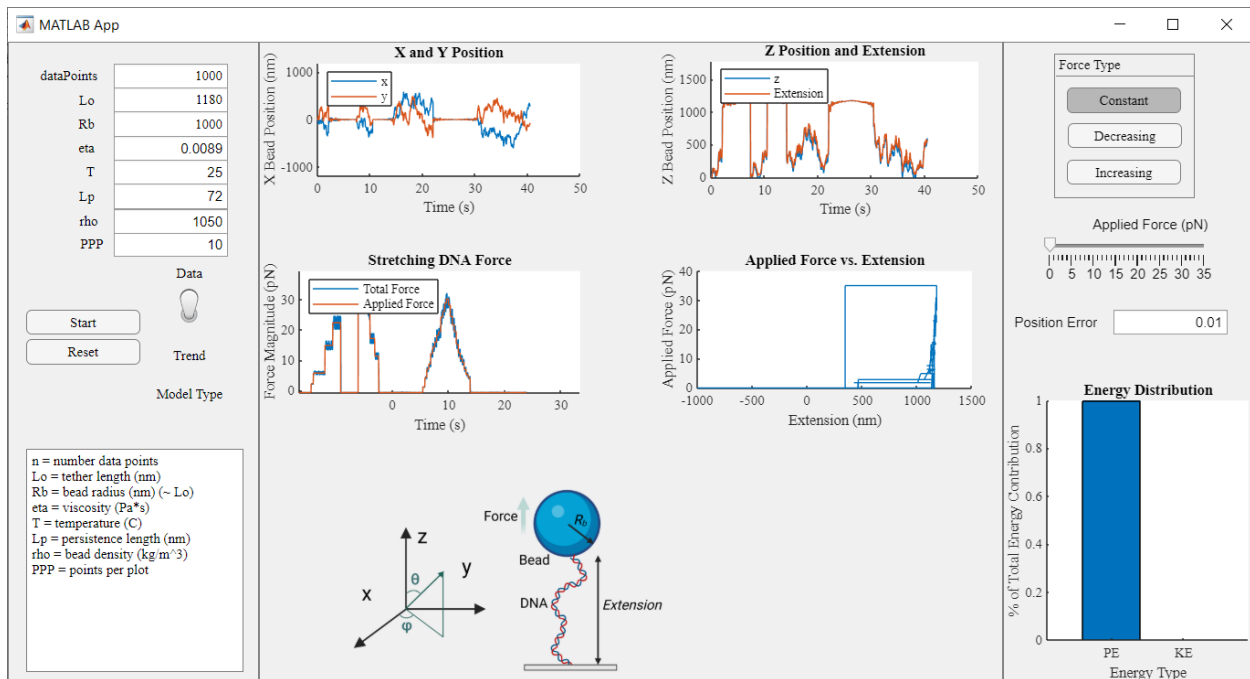


Figure 5: After saving the simulation data, the GUI will update the energy distribution on the right panel of the GUI. The potential energy (PE) should dominate the kinetic energy (KE) for a realistic TPM simulation.

Example 2: Simulation with linear force ramp in Data and trend model

This example illustrates the differences between the two simulation models. Keep all parameters set as in Example 1.

1. Toggle the switch to “Data” model.
2. Set the force slider to the maximum value (35 pN).
3. Press the button “Increasing”.
4. Press start and save the data if needed.
5. The simulation should look like as shown in Figure 7.
6. Press reset and toggle the switch to the “Trend” model.
7. Press start and save the data if needed.
8. The simulation should look like as shown in Figure 7.

The force extension trend looks similar for both model types, but the biggest difference is in the total estimated force as well as the x and y fluctuations.

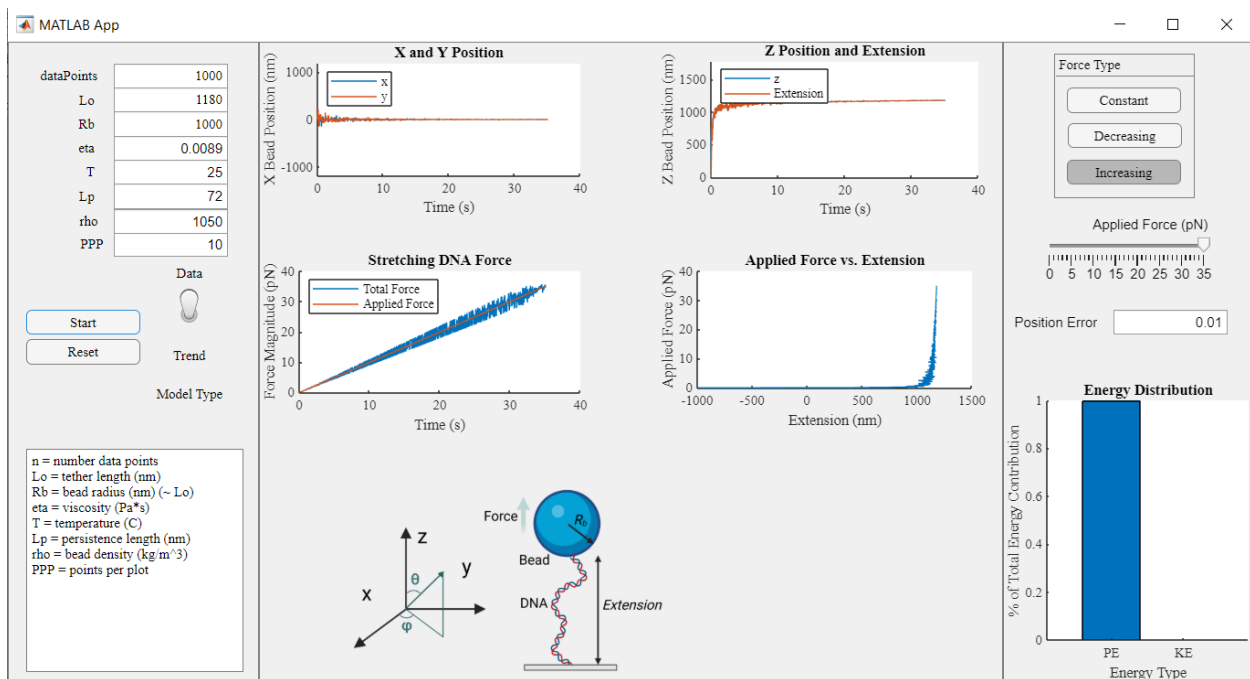


Figure 6: GUI after running a linearly increasing force ramp using the Data model.

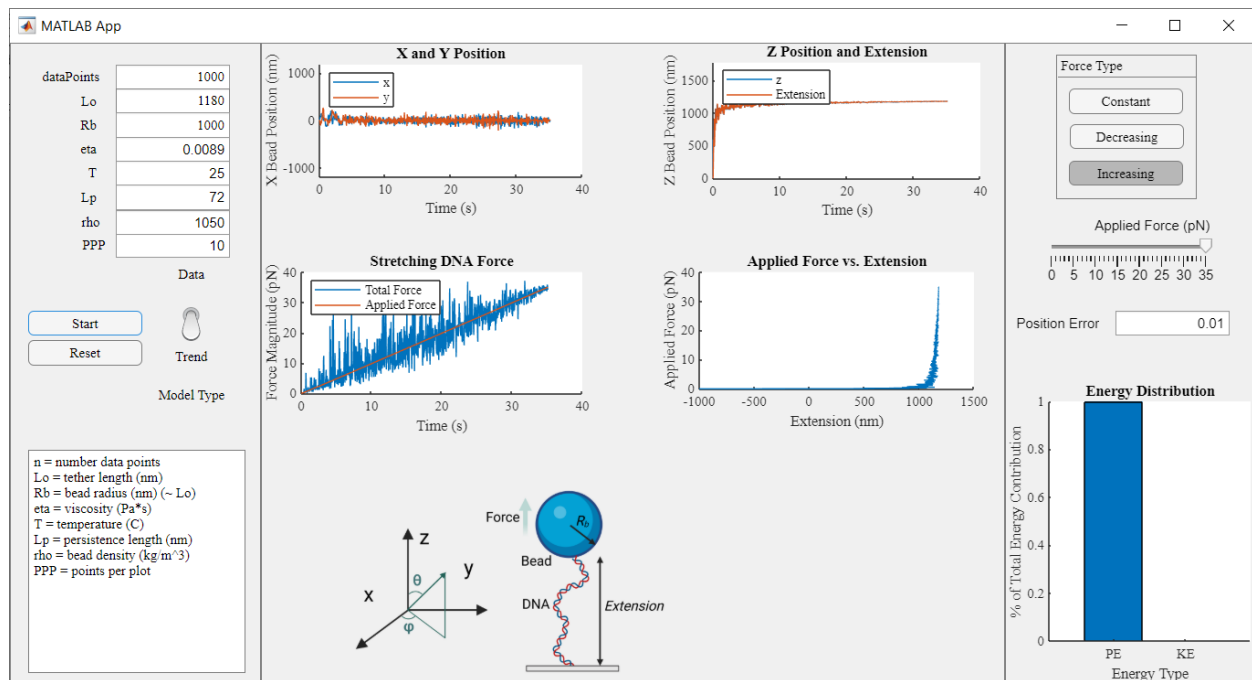


Figure 7: GUI after running a linearly increasing force ramp using the Trend model.

Common Errors

Persistence length and applied force

The model is sensitive to the input parameters and some internal functions yield fatal errors. For example, if the persistence length (L_p) is set too small, then the simulation will run into an error solving for the force as shown in Figure 8 and Figure 9 if the applied force is too large.

1. In either Data or Trend mode, set L_p to 20 (nm).
2. Set the force button to constant.
3. Set the force slider to 0.
4. Start the simulation.
5. Increase the force to 5 pN, simulation should continue.
6. Increase the force to 15 pN, simulation should continue.
7. Increase the force to 25 pN, simulation should stop and error message in Figure 9 appear.

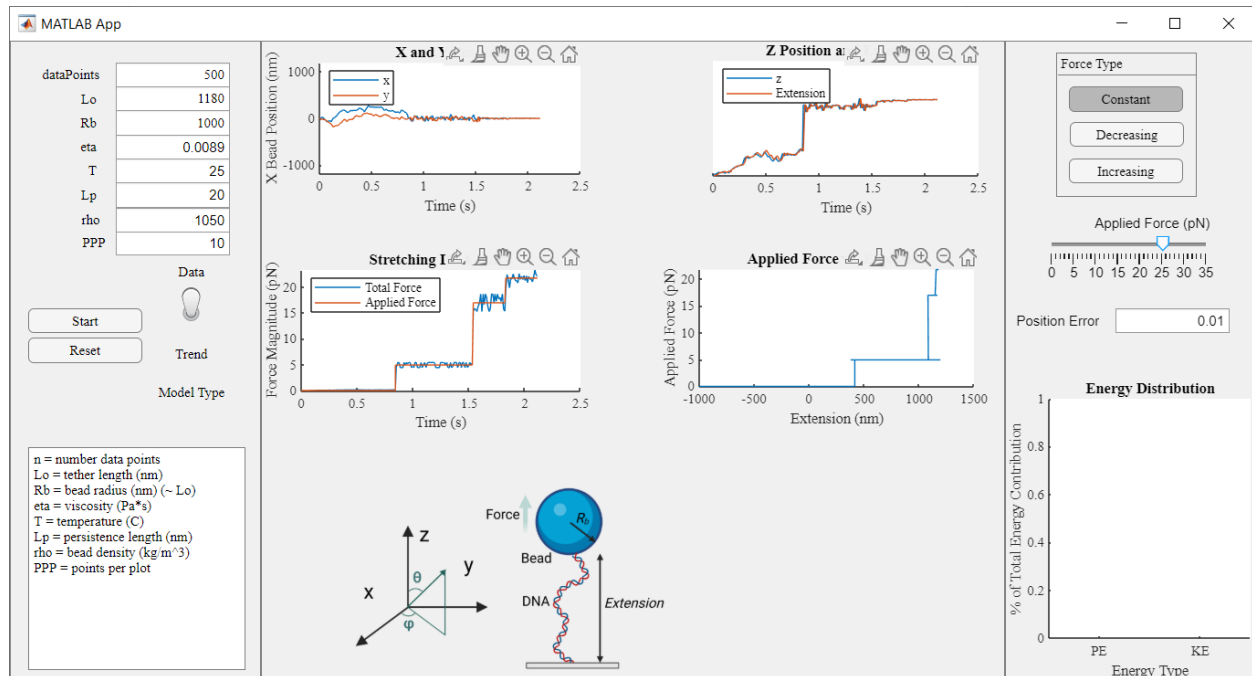


Figure 8: Example of error. A low persistence length results in errors solving for the force.

```
Error using fzero
Initial function value must be finite and real.

Error in MarkoSiggiaVectorized (line 24)
tempForcez = fzero(tempFuncz,1e-14);

Error in TetherForce2>RandWalkSim (line 255)
    DNAForceX = MarkoSiggiaVectorized(kbT, Lp, Lo, extensionX,extensionY,extensionZ);

Error in TetherForce2 (line 14)
[Xsim,Ysim,Zsim, FullExtension, AngleSimThetaDegree, AngleSimPhiDegree, DNAForce, time] = RandWalkSim(DNAForceX,DNAForceY,DNAForceZ,FullExtension,AngleSimThetaDegree,AngleSimPhiDegree,time);

Error in TPMGUI/StartButtonPushed (line 172)
    [t,df,x,y,z,extension,dfx,dfy,dfz,theta,phi]=TetherForce2(forceArray(1+i-1,:),timeStep,FullExtension,AngleSimThetaDegree,AngleSimPhiDegree,time);

Error in appdesigner.internal.service.AppManagementService/executeCallback (line 138)
    callback(appOrUserComponent, event);

Error in matlab.apps.AppBase>@(source,event)executeCallback(appdesigner.internal.service.AppManagementService/executeCallback)
    newCallback = @(source, event)executeCallback(appdesigner.internal.service.AppManagementService/executeCallback);

Error using matlab.ui.control.internal.controller.ComponentController/executeUserCallback
Error while evaluating Button PrivateButtonPushedFcn.

fx >> |
```

Figure 9: Error codes when the persistence length is set too small. The MATLAB function “fzero” returns an error and stops the simulation.

Saving data

After a successful simulation, the user is asked to save the data. If “Cancel” is pressed, then the error shown in Figure 10. Note that this error can be mitigated by checking if the variable “answer” is empty or not. If a file already exists with the name that the user had specified, then the original file will be overwritten with the most recent simulation data.

```
Error in TPMGUI/StartButtonPushed (line 318)
    str=append(answer{1},'.csv');

Error in matlab.apps.AppBase>@(source,event)executeCallback(appdesigner.internal.service.AppManagementService/executeCallback)
    newCallback = @(source, event)executeCallback(appdesigner.internal.service.AppManagementService/executeCallback);

Error while evaluating Button PrivateButtonPushedFcn.

fx >>
```

Figure 10: Error after pressing “Cancel” at the save data pop-up window.

Case of kinetic energy dominating

As mentioned in the publication, the simulation is valid, if the potential energy (PE) dominates. Typically, kinetic energy dominates, if low values for the viscosity are set as shown in Figure 11.

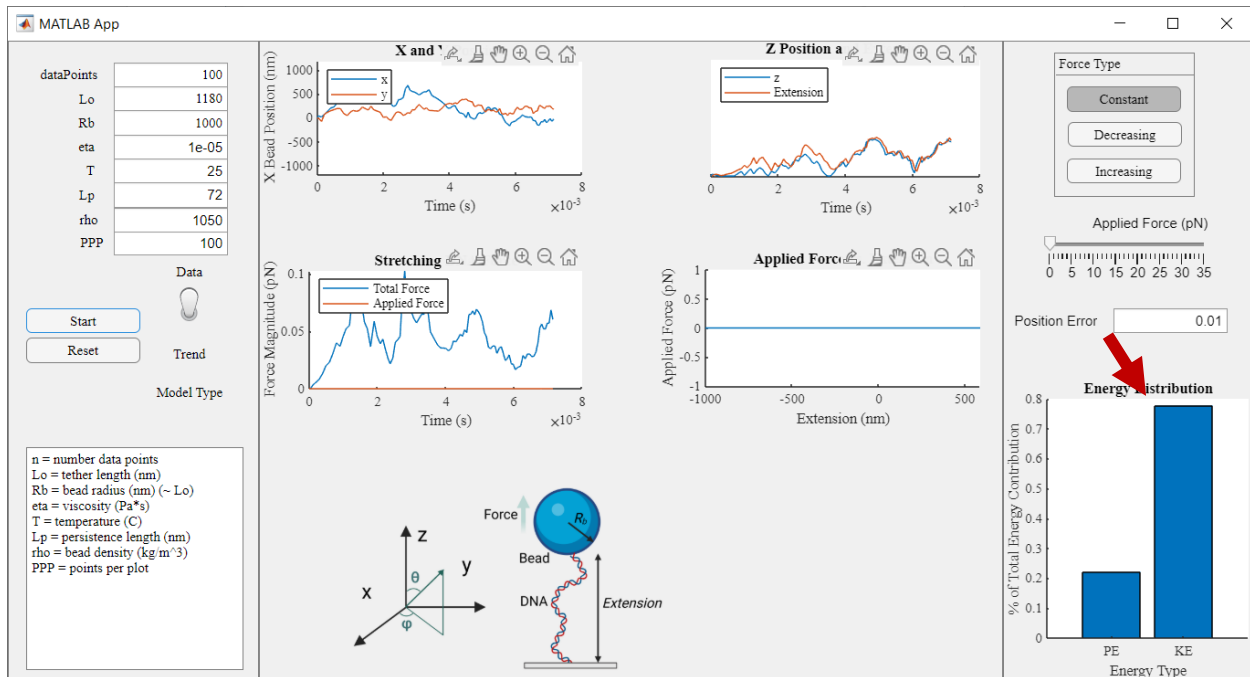


Figure 11: Example of kinetic energy (KE) dominating. The viscosity (η) was set to $10^{-5} \text{ Pa}\cdot\text{s}$.

Sample lesson plan for Tethered Particle Motion

History of TPM

Tethered Particle Motion (TPM) is a powerful method used in biophysics to study the properties of individual molecules (mainly DNA and DNA-binding proteins). It began in the 1980s when scientists wanted to understand how DNA and proteins behaved at a single-molecule level. At first, TPM experiments involved attaching a microscopic bead to a molecule of interest (such as DNA) and then tethering the bead to a surface using various methods (1). By observing the motion of the bead under a microscope, researchers could indirectly study the behavior of the attached molecule. This technique provided unprecedented insights into the dynamics and mechanical properties of biomolecules.

Over the years, TPM evolved with advancements in technology and equipment. High-resolution microscopy, external force detection and application, and sophisticated data analysis methods were introduced, enabling more accurate measurements and deeper understanding of molecular interactions. TPM has revolutionized biophysical research, leading to breakthroughs in fields like DNA mechanics, protein folding, and enzymatic activity. Its applications expanded beyond biophysics to include drug development, nanotechnology, and materials science, and the concept of TPM is the foundation of many cutting-edge research tools.

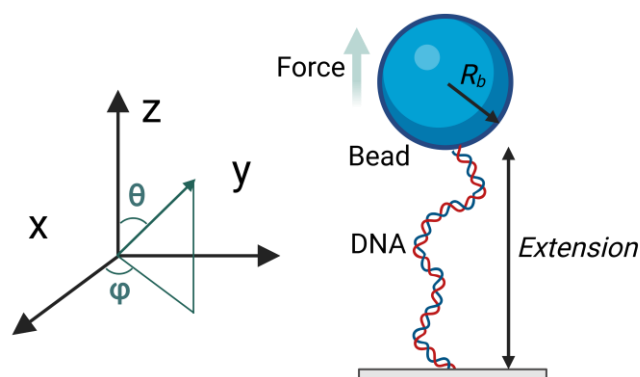


Figure 1: Schematic setup of a TPM experiment, where a micron-sized bead is attached to a piece of double-stranded DNA.

Learning objective

Typically, TPM experiments require sophisticated experimental setup, preparation and data analysis to obtain basic information such as bead position and applied external forces acting on the bead, and are barely covered in a class room setting. The goal of this learning module is to familiarize the student with the basics of tethered particle motion by means of conducting a “TPM experiment” in silico. Further, the students are exposed to different ways to simulate TPM data, with the intention to deepen their understanding of computational modelling and identifying strengths and limitations of any modelling approach.

Example 1: Relationship between the mean excursion (root-mean-square fluctuation, RMS) and the tether length

The mean excursion (RMS) of a tethered bead depends on the size of the bead as well as the tether length (2, 3). However, this relationship is not linear. Without further information about the system, how would you

- Study the relationship between RMS and tether length?
- Develop a simple model to correlate RMS and tether length?

Solution:

Using the TPM simulation tool, run simulations without applied force using the default values of the GUI, keeping all parameters constant, except the tether length. See the Case study and Figure 5 in the main manuscript for details about which tether lengths were used for the example.

The analysis script supplied with the lesson plan walks through the step-wise computations. After running the simulations, importing the CSV data files and plotting the RMS values versus tether length, one should obtain a curve that looks similar to the one shown in Figure 2.

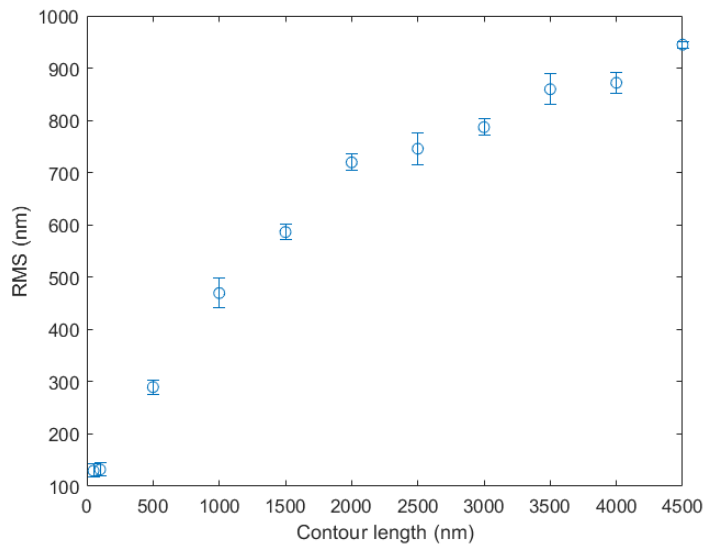


Figure 2: RMS vs tether length example similar to Fig. 5 of the main manuscript

Looking at the relationship between these two variables, we see that some sort of power series fit may describe the data well.

A simple power fit could be written as $RMS = A * L_o^B$, where A and B are the fit parameters and L_o is the tether contour length. A power series can easily be fit in MATLAB by linearization and performing a linear regression. The linearized equation can be written as $\log(RMS) = \log(A) + B * \log(L_o)$. Solving, this equation, we obtain $A=16.75$ and $B=0.48$ as parameters. The fit can be verified by plotting the developed power series model along the original data as shown in Figure 3. One may round up B to $B \sim 0.5$, which means that the RMS is proportional to the square-root of the tether length. For more information these topic, see (4).

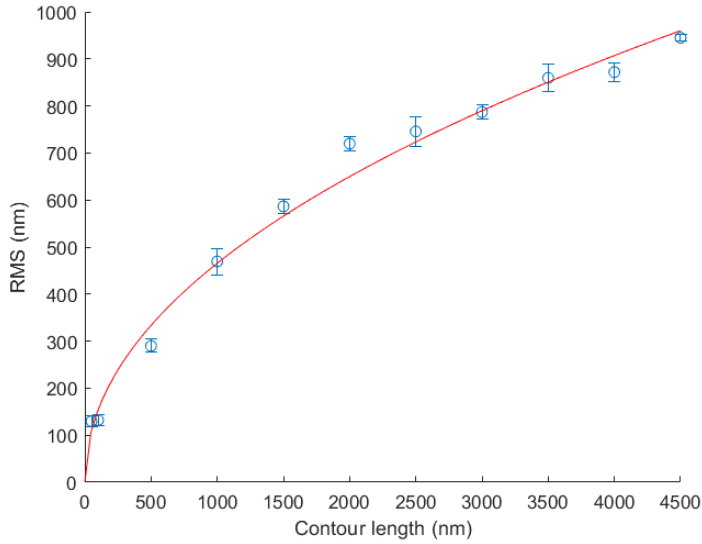


Figure 3: Power law fit (red line) to the simulated data (blue circles)

Example 2: Fitting of Worm-like chain model (WLC) to simulated force-extension data

Double-stranded DNA is a fascinating biopolymer and its mechanical flexibility plays a key role in many cellular functions, such as DNA replication and transcription. In nonenveloped DNA viruses, the packing into virus-like particles (VLPs) can be better understood by measuring the mechanical properties of DNA upon binding with structural proteins (5). The mechanical properties of DNA are typically assessed by creating force-extension curves, where the ends of DNA are systematically being pulled away from each other. One parameter, which often changes upon protein-binding to DNA is the persistence length. The persistence length can be estimated using various models (6), one of which is the Marko-Siggia Wormlike chain (WLC) model:

$$F = \left(\frac{k_B * T}{L_p} \right) \left[\frac{1}{4 * (1 - x/L_0)^2} - \frac{1}{4} + \frac{x}{L_0} \right]$$

Here, F is the external applied force, k_B is the Boltzmann constant, T the temperature, L_p the persistence length and L_0 the contour length of DNA.

The goal of this exercise is to estimate the persistence length of DNA using the above model. Note, that the developed simulation used a modified version of the above model, which considers the stretch modulus of DNA.

Solution:

Using the TPM simulation tool, set the parameters as shown in Figure 4. Next, press the “Increasing” button for the force type and move the force slider to approximately 25 pN. Keeping the model type as “Data”, run the simulation. Repeat the process for a persistence length of 90 nm.

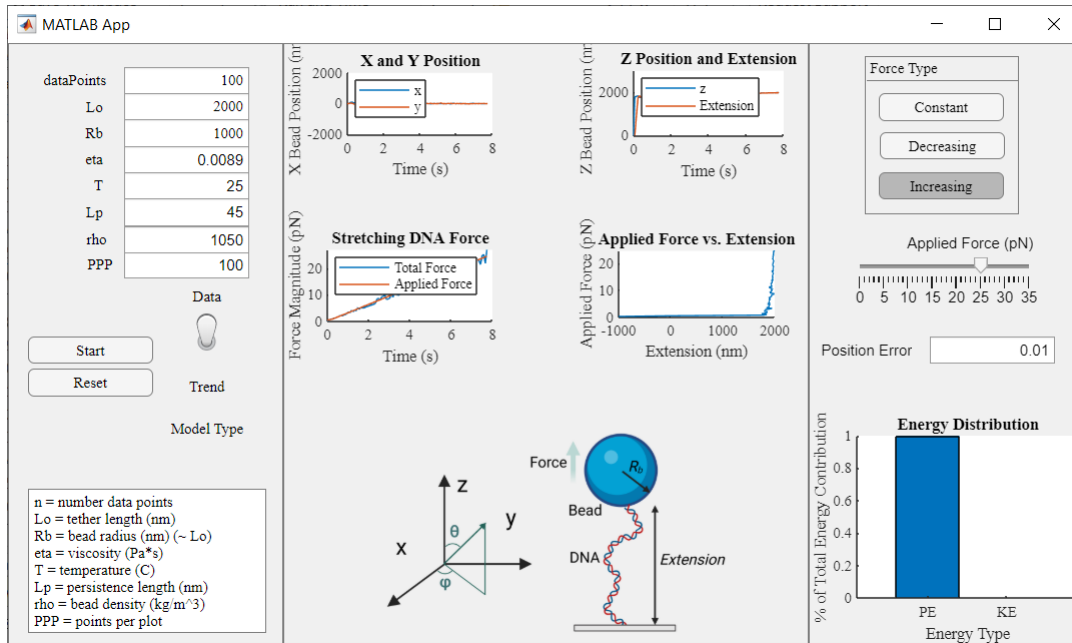


Figure 4: GUI settings for creating a force extension curve.

The analysis script supplied with the lesson plan walks through the details of the computation. Import the data to MATLAB and plot the extension (3-D position of DNA in space) versus force for both runs as shown in Figure 5. Notice that both curves are relatively similar, despite a 2x difference in persistence length in the simulation.

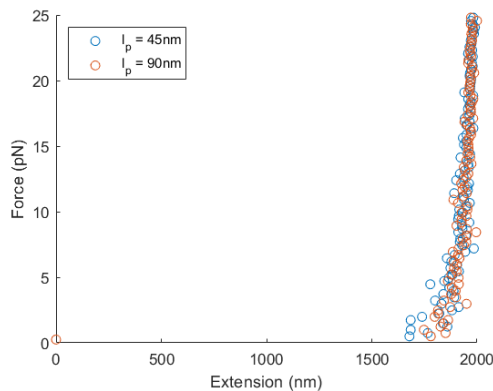


Figure 5: Force-extension curves for two simulation runs with different persistence length L_p .

Next, fit the WLC model to the data as shown above. The fit parameters are the persistence length, L_p , and the contour length, L_c and are summarized in Table 1. Note that despite a difference in persistence length in the simulation, both yielded approximately the same values for the fit parameters.

Table 1: Fit parameters of Marko-Siggia WLC fit

Experiment	Estimated L_p (nm)	Estimated L_c (nm)	R^2
$L_p=45\text{nm}$	14.81	2100	0.77
$L_p=90\text{nm}$	14.72	2104	0.72

We can visualize the goodness of fit by plotting the force-extension data alongside the model with the estimated parameters as shown in Figure 6. One potential reason for the discrepancies between the estimated and simulated persistence and contour length, respectively, is most likely the model that has been used to fit the data. As mentioned above, the simulation takes into account the enthalpic stretching of DNA in the form of a stretch modulus (K_o), which is not considered in the WLC model described above. As a bonus question, students could be asked to implement a curve fitting approach using the modified Marko-Siggia model (6) and the original Odijk WLC model (7) and compare the obtained fit parameters with the fit from the WLC model presented here.

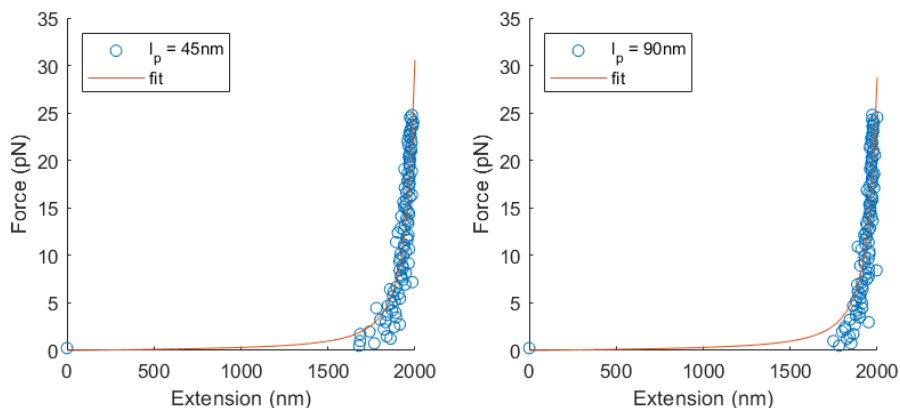


Figure 6: WLC fits for the simulation run at $L_p=45\text{nm}$ (left) and $L_p=90\text{nm}$ (right).

References

1. D. A. Schafer, J. Gelles, M. P. Sheetz, R. Landick, Transcription by single molecules of RNA polymerase observed by light microscopy. *Nature* **352**, 444–448 (1991).
2. N. Pouget, *et al.*, Single-particle tracking for DNA tether length monitoring. *Nucleic Acids Res.* **32** (2004).
3. P. C. Nelson, *et al.*, Tethered particle motion as a diagnostic of DNA tether length. *J. Phys. Chem. B* **110**, 17260–17267 (2006).
4. S. H. W. K. O. D. W. L. S. D. S. Craig John Benham, *Mathematics of DNA Structure, Function and Interactions (The IMA Volumes in Mathematics and its Applications)*, C. J. Benham, *et al.*, Eds. (Springer Berlin Heidelberg, 2009).
5. M. G. M. Van Rosmalen, *et al.*, Revealing in real-time a multistep assembly mechanism for SV40 virus-like particles. *Sci. Adv.* **6**, 1–8 (2020).
6. M. D. Wang, H. Yin, R. Landick, J. Gelles, S. M. Block, Stretching DNA with optical tweezers. *Biophys. J.* **72**, 1335–1346 (1997).
7. T. Odijk, Stiff Chains and Filaments under Tension. *Macromolecules* **28**, 7016–7018 (1995).