In Silico Single-Molecule Manipulation of DNA with Rigid Body Dynamics
Pascal Carrivain, Maria Barbi, Jean-Marc Victor

To cite this version:

Pascal Carrivain, Maria Barbi, Jean-Marc Victor. In Silico Single-Molecule Manipulation of DNA with Rigid Body Dynamics. PLoS Computational Biology, Public Library of Science, 2014, 10 (2), 10.1371/journal.pcbi.1003456. hal-01342589

HAL Id: hal-01342589
https://hal.sorbonne-universite.fr/hal-01342589
Submitted on 6 Jul 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License
**In Silico** Single-Molecule Manipulation of DNA with Rigid Body Dynamics

Pascal Carrivain$^{1,2}$, Maria Barbi$^1$, Jean-Marc Victor$^1$

$^1$ Laboratoire de Physique Théorique de la Matière Condensée, CNRS UMR 7600, Université Pierre et Marie Curie, Paris, France, $^2$ Institut de Génétique Humaine (IGH), CNRS UPR 1142, Montpellier, France

**Abstract**

We develop a new powerful method to reproduce *in silico* single-molecule manipulation experiments. We demonstrate that flexible polymers such as DNA can be simulated using rigid body dynamics thanks to an original implementation of Langevin dynamics in an open source library called Open Dynamics Engine. We moreover implement a global thermostat which accelerates the simulation sampling by two orders of magnitude. We reproduce force-extension as well as rotation-extension curves of reference experimental studies. Finally, we extend the model to simulations where the control parameter is no longer the torsional strain but instead the torque, and predict the expected behavior for this case which is particularly challenging theoretically and experimentally.


**Methods article.**

This is a *PLOS Computational Biology* Methods article.

**Introduction**

The mechanical and topological properties of DNA and protein-DNA assemblies are of primary importance in many biological processes, including transcription, replication, chromatin organization and remodeling. Since techniques have become available enabling the manipulation of single-molecules [1,2], a large amount of experimental data have been accumulated on the mechanical response of DNA and protein-DNA assemblies under stretching forces and twisting torsions, in particular from optical and magnetic tweezers experiments [2–5]. In magnetic tweezers experiments, a DNA molecule is grafted at one end to a coverslip and at the other end to a magnetic bead. The bead is trapped in the magnetic field of a pair of magnets that may be translated, thus exerting a varying force on the bead. Moreover the pair of magnets may be rotated at a certain number of turns, thus constraining the linking number of the DNA molecule. After the stretching force and the number of turns are applied to the bead, the only physical variable that can be directly measured is the DNA extension, i.e. the distance between its two ends. Therefore the interpretation of the experimental results requires an important modeling effort, particularly in the more complex cases where DNA is associated with proteins, as for instance in chromatin assemblies [6,7]. Although theoretical approaches may be successful in some cases [8–10], simulations are often crucial tests of the proposed model validity, when they are not the unique possible way of dealing with the system complexity.

In this spirit, we aim to develop an efficient tool to manipulate single-molecules *in silico* reproducing optical and magnetic tweezers experiments. This task is challenging since the DNA model should have precise specifications to reproduce the behavior of DNA accurately. We need to: (i) model a polymer, i.e. an articulated chain; (ii) reproduce the effective diameter of DNA (depending on electrostatic conditions) and, when proteins are present, have the possibility to model their shape and steric hindrance; (iii) deal with collisions, especially in order to reproduce DNA supercoiled structures (plectonemes) and steric effects in DNA-protein assemblies; (iv) reproduce DNA twisting and bending elasticities; (v) include statistical mechanics features to account for temperature and thermal motion. Beside these essential points, we also wish to simulate the system dynamics, which may be important in some cases, e.g. when hysteresis is observed under magnetic tweezers [7] or for *in vivo* chromosome dynamics experiments in the cell nucleus [11,12].

This ambitious list of specifications is beyond the reach of traditional simulation approaches where particles interact through 2-body potentials (as in Molecular Dynamics or Monte Carlo simulations [13,14] with a given force field). The need to deal with frozen degrees of freedom in coarse grained modeling may be addressed through holonomic constraints, as in the SHAKE algorithm [15,16], where an iterative approach is adopted. However, collision detection and steric hindrance may only be accounted for in this scheme by introducing additional steps. More recently, non iterative algorithms have been developed [17], that subsequently led to the development of new powerful tools, called “physics engines”. These have been designed by the engineering and robotics communities to reproduce the dynamic behaviour of
Author Summary

Video game techniques are designed to simulate rigid body dynamics of macroscopic bodies, e.g., characters or vehicles, in a realistic manner. However they are not able to deal with temperature effects, hence they are not able to deal with molecules. In order to extend these powerful techniques to molecular modeling, we implement here Langenbek Dynamics in an open source library called Open Dynmics Engine. Moreover we add a “global thermostat” to this Langenbek Dynamics, which accelerates the simulation sampling by two orders of magnitude. With these radically new simulation techniques, we prove that we can accurately reproduce single-molecule manipulation experiments in silico, in particular force-extension as well as rotation-extension curves of reference experimental studies. The method developed here represents an unparalleled tool for the study of more complex single molecule manipulation experiments, notably when DNA interacts with proteins. Furthermore the simulation technique that we propose here has all the functionalities required to tackle the nuclear organization of chromosomes at every length scale, from DNA to whole nuclei.

Simulating DNA with Physics Engines

Articulated systems of rigid bodies. Physics engines are acquiring an increasing importance, notably in the fields of computer graphics and video games, where they are now widely used to simulate rigid body motion under realistic conditions and in real-time. Open Dynamics Engine (ODE) is one of the most popular rigid-body dynamics open source library for robotics simulation applications [18]. As other physics engines, ODE simulates the kinematics of articulated systems by using permanent joints that impose holonomic constraints, instead of bond potentials. The same method is used to manage collisions: when overlapping between bodies is detected, a temporary joint is locally created that reproduces the action of the contact forces, without the need for explicit permanent 2-body interaction potentials (see section “Materials and Methods” for details on how ODE manages joints and collisions).

These extremely efficient simulators haven’t, up to now, been used in statistical mechanics. Although well adapted to mechanical simulations, physics engines lack coupling to a thermal bath. The main novelty of our approach is the implementation of Langevin-Euler equation in the ODE software. Moreover we improve the simulation efficiency of this Langevin dynamics by extending the “global thermostat” algorithm designed by Busi and Parinello in 2008 [19] to physics engines. This algorithm allows much faster yet unbiased sampling of the phase-space. As a first step toward simulating DNA-protein assemblies, we focus here on bare DNA and show how to perform in silico single molecule manipulation of DNA.

Materials and Methods

Introduction to physics engines

In rigid body dynamics simulations run with ODE, the state of a system consisting of N rigid bodies is described by the positions \( \mathbf{r}_i \) of their centres of mass, a quaternion representation of their orientations \( \mathbf{q}_i \), and their linear and angular velocities \( \mathbf{v}_i \) and \( \omega_i \) respectively. These velocities are collected in the column vector \( \mathbf{v} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N, \omega_1, \omega_2, \ldots, \omega_N]^T \). We use the superscript \( T \) to denote the transpose of a vector or a matrix everywhere in this article. The vector \( \mathbf{L} = \mathbf{Mv} \) then collects all linear and angular momenta, where \( \mathbf{M} = [\mathbf{m}_1, \mathbf{I}_1 \ldots \mathbf{m}_N, \mathbf{I}_N] \) is a 6N × 6N block diagonal matrix whose elements are the mass matrices \( \mathbf{m}_i \) and inertia matrices \( \mathbf{I}_i \) of the N bodies (with \( 1 \) the 3 × 3 identity matrix). The Newtonian dynamics equation therefore reads \( \mathbf{L} = \mathbf{F} \) where the generalized force \( \mathbf{F} \) is a vector collecting forces and torques applied to the system. These forces and torques may be external, due for example to gravity or magnetic fields, or internal, as a consequence of the mechanical constraints between the rigid bodies that make up the system.

Most notably, in articulated systems, as is the case of polymers, rigid bodies are connected by mechanical joints. A joint is a relationship that is enforced between two bodies so that they can have only certain positions and orientations relative to each other, and ODE provides different types of joints according to the kind of articulation that has to be implemented, e.g. ball-and-socket, hinge, slider or universal.

Mathematically a joint imposes some holonomic constraint between both connected bodies. Such a constraint is an equation that reads \( \delta = 0 \) where \( \delta \) is the distance between both joint bearings, e.g. the center of the ball of one body and the the center of the socket of the other one. The constrained distance \( \delta \) is purely geometrical, depending only on the relative position and orientation of both jointed bodies. The position and orientation of each of the N bodies the articulated system is composed of depend on time \( t \). Therefore the constraints \( \delta = 0 \) of the articulated system can be derived with respect to time to get the kinematic constraints in the form \( \mathbf{J}\mathbf{V} = 0 \) where we introduce the jacobian matrix of constraints \( \mathbf{J} \) (see subsection “Exact solution for \( \lambda \)” for a detailed example). This velocity-based description is used in ODE as in most game/physics engines.

So, mechanical joints exert reaction forces and torques on the joint bearings. These internal mechanical constraints can be collected into a generalized constraint force \( \mathbf{F}_c \) which, by virtue of the principle of virtual work \( \mathbf{F}_c^T \mathbf{V} = 0 \), reads \( \mathbf{F}_c = \mathbf{J}^T \mathbf{\lambda} \) where \( \mathbf{\lambda} \) is a vector of Lagrange multipliers that precisely accounts for the reaction forces and torques coming from the joint bearings [16]. The Newtonian dynamics equation therefore reads \( \mathbf{L} = \mathbf{F}_c + \mathbf{F}_i \) where \( \mathbf{F}_c \) and \( \mathbf{F}_i \) stand for the external and internal contributions to the generalized force respectively. As the constraint force reads \( \mathbf{F}_c = \mathbf{J}^T \mathbf{\lambda} \), the Newtonian dynamics equation becomes an equation for \( \lambda \) in the form: \( \mathbf{L} = \mathbf{F}_i + \mathbf{J}^T \mathbf{\lambda} \).

Solving this equation for \( \lambda \) should moreover satisfy the holonomic constraints \( \delta = 0 \) at every timestep \( t \). However the discretization used in the numerical calculation results in errors on \( \delta \) so that \( \delta(t) \) is generally not equal to 0. Then, in order to have \( \delta(t + \Delta t) = 0 \) at the next timestep, the kinematic constraint \( \mathbf{J}\mathbf{V} = 0 \) should be adapted accordingly. Indeed \( \delta(t + \Delta t) = \delta(t) + \mathbf{J}(t)\mathbf{V}(t + \Delta t)\Delta t \) according to the Euler semi-implicit integration scheme which is used in velocity-based algorithms. Hence \( \mathbf{J}(t)\mathbf{V}(t + \Delta t) = - \frac{\delta(t)}{\Delta t} \). But then this implies that the kinematic constraint is not equal to zero at time \( t + \Delta t \), i.e. \( \mathbf{J}(t + \Delta t)\mathbf{V}(t + \Delta t) \neq 0 \), so that the joint bearings will continue to move apart afterwards. In order to keep both \( \delta \) and \( \mathbf{J}\mathbf{V} \) close to zero at every timestep, ODE introduces an error reduction parameter \( k_{\text{err}} \) in the kinematic equation \( \mathbf{J}(t)\mathbf{V}(t + \Delta t) = - k_{\text{err}} \frac{\delta(t)}{\Delta t} \) [18]. This parameter has to be adjusted to some optimal value between 0 (no correction at all) and 1 (complete correction of \( \delta \) in one timestep). However setting \( k_{\text{err}} = 1 \) is not recommended since, as said above, this would imply that the joint bearings will continue to move apart afterwards with maximal velocities. ODE recommends values between 0.1 and 0.8.
In addition to $k_{	ext{er}}$, ODE introduces a second ingredient to soften the rigid constraints by allowing the violation of the constraint equation by an amount proportional to the restoring force $\lambda$. More explicitly, a “constraint force mixing” diagonal matrix $K_{\text{cm}} = k_{\text{cm}} \lambda$ is defined, such that $\mathbf{J}(t)\mathbf{V}(t + \Delta t) = -\frac{k_{\text{er}}}{\Delta t} \mathbf{g}(t) - k_{\text{cm}} \mathbf{\lambda}(t)$ (implicit integration) [18]. This is equivalent to introducing a spring-damper system (spring constant of $\frac{k_{\text{er}}}{\Delta t} \mathbf{g}(t)$ and damping constant of $\frac{1 - k_{\text{er}}}{k_{\text{cm}}}$ with implicit integrator between the joint bearings; this can be understood as analogous to a bead-spring model. Nevertheless there is a major difference between this effective spring and a regular spring: the term $-\frac{k_{\text{er}}}{\Delta t} \mathbf{g}(t)$ constrains the velocity whereas a regular spring constrains the acceleration. As a result, no energy is stored in this effective spring, at odds with a regular spring which stores an averaged energy $k_B T$ (see below subsection “Preliminary tests of validity and performance of the global thermostat”) along with the histograms of energy in Figure 1.

In particular, ODE uses a powerful software called libiscd [20] to detect collisions between two convex shapes. Whenever overlapping is detected between two rigid bodies, ODE attaches a temporary joint between them called a “contact joint”. Defining vector $\mathbf{c}_1$ (resp. $\mathbf{c}_2$) that connects the center of mass of body 1 (resp. 2) to the contact point and denoting $\mathbf{n}$ the common normal to both bodies at the contact point (directed from 2 to 1), the kinematic constraint imposed by the contact joint would read $\mathbf{n}^T (\mathbf{v}_1 + \mathbf{v}_2 \times \mathbf{c}_2) - 2 \mathbf{n}^T (\mathbf{v}_2 + \mathbf{v}_2 \times \mathbf{c}_2) = 0$ in the perfect case when the holonomic constraint imposed by the contact joint reads exactly $\mathbf{\delta} = 0$. However, in practice $\mathbf{\delta}$ is not equal to 0 because of discretization errors, hence the kinematic constraint imposed by the contact joint actually reads:

$$\mathbf{J}(t)\mathbf{V}(t + \Delta t) = -\frac{k_{\text{er}}}{\Delta t} \mathbf{n}^T (\mathbf{r}_1 + \mathbf{c}_1 - \mathbf{r}_2 - \mathbf{c}_2)(t) - k_{\text{cm}} \mathbf{n}^T \mathbf{\lambda}(t)(t)$$

(1)

with

$$\mathbf{J} = \begin{pmatrix} \mathbf{n}^T & -\mathbf{n}^T \mathbf{c}_1^0 & \mathbf{n}^T \mathbf{c}_2^0 \end{pmatrix}$$

(2)

$$\mathbf{\lambda}^T \mathbf{n} \geq 0$$

(3)

The right hand side of Eq(1) deals with the already existing overlapping of the two bodies in contact at time $t$ when collision is first detected, or with their residual overlapping while the contact joint exists.

By inserting the constraint force $\mathbf{F}_c = \mathbf{J}^T \mathbf{\lambda}$ into the equation of motion $\dot{\mathbf{\lambda}} + \mathbf{F}$ and taking the first-order discretisation of this equation, one can easily get the following expression to be solved for $\mathbf{\lambda}$:

$$\begin{pmatrix} \mathbf{J} \mathbf{M}^{-1} \mathbf{J}^T + \frac{k_{\text{cm}}}{\Delta t} \mathbf{I} \end{pmatrix} \mathbf{\lambda} =$$

$$-\frac{k_{\text{er}}}{\Delta t} \mathbf{\delta} - \mathbf{J} \left( \frac{\mathbf{V}}{\Delta t} + \mathbf{M}^{-1} (\mathbf{F}_c + \mathbf{M} \mathbf{V}) \right)$$

(4)

This equation is of the form $\mathbf{A} \mathbf{\lambda} = \mathbf{B}$. Importantly, the addition of the term $\frac{k_{\text{cm}}}{\Delta t}$ to each diagonal term $\mathbf{a}$ of the matrix $\mathbf{J} \mathbf{M}^{-1} \mathbf{J}^T$ provides a symmetric positive definite matrix $\mathbf{A}$, thus greatly increasing the solution accuracy of Eq.(4). From this equation, the vector $\mathbf{\lambda}$ of Lagrange multipliers, hence the constraint force $\mathbf{F}_c$, can be determined. Then the motion solver (semi-implicit Euler integrator) gives the new positions and orientations of the articulated bodies at time $t + \Delta t$. It is advantageous to choose the Exponential Map parametrization [21] for the quaternion integration.
Simulating DNA with Physics Engines

Solving constraints

In general Eq.(4) has to be solved numerically and ODE has two algorithms to do so, one based on the Successive-Over-Relaxation (SOR) method [22] and the other based on the Linear Complementary Problem (LCP) [23]. LCP time complexity is of order m and space complexity (memory) of order m where m is the number of constraint rows [18]; whereas SOR time complexity is of order mNSOR where NSOR is the number of successive-over-relaxation and space complexity of order m [18]. Both algorithms have equivalent performances when m = \sqrt{NSOR}. But in general LCP is more accurate, although much more time consuming, than SOR. We compared these two algorithms for a chain of length N = 300 without noticing significant differences in the errors on \( \delta \) (error on the colocalization of joint bearings). In order to save computational time, we preferentially run the SOR method with a value of \( \omega_{\text{SOR}} = 1.7 \) for the relaxation factor and \( N_{\text{SOR}} = 100 \). These values are different from the default values in ODE and work well for a linear chain of rigid bodies connected with ball-and-socket joints. But in some cases, the SOR method does not converge and we then switch to the LCP method, which always converges. However in the case where there are no collisions between the rigid bodies the articulated system is composed of, we were able to derive an exact solution for \( \lambda \) (see next subsection). Therefore we solve Eq.(4) according to the following scheme:

1. if there are no collisions at time \( t \), we use the exact solution for \( \lambda \),
2. if some collisions are detected, then we run the SOR method and check for the accuracy of the solution. More specifically the solution is accepted if \( \| \lambda + \frac{k_{\text{res}}}{k_{\text{fim}}} \delta + \frac{1}{k_{\text{fim}}} [\mathcal{J} \mathcal{V}(t + \Delta t)] \| / \| \lambda \| < 10^{-4} \) then the simulation step is restarted with the LCP method.
3. if \( \| \lambda + \frac{k_{\text{res}}}{k_{\text{fim}}} \delta + \frac{1}{k_{\text{fim}}} [\mathcal{J} \mathcal{V}(t + \Delta t)] \| / \| \lambda \| > 10^{-4} \) then the simulation step is restarted with the LCP method.

Exact solution for \( \lambda \) when there are no collisions

For the sake of clarity, let us first consider the example of four rigid cylinders of length \( l \) each connected with ball-and-socket joints at the extremities \( r_i \pm \frac{l}{2} t_i \), with the first one anchored to some fixed point, taken as the origin of the coordinates. The vector \( t_i \) is the tangent to the cylinder \( i \). The jacobian matrix \( \mathcal{J} \) associated with this system is tridiagonal when there are no collisions, in which case it reads:

\[
\mathcal{J} = \begin{pmatrix}
-1 & \frac{l}{2} t_i^0 & 0 & 0 & 0 & 0 \\
1 - \frac{l}{2} t_i^0 & -1 - \frac{l}{2} t_i^0 & 0 & 0 & 0 & 0 \\
0 & 1 - \frac{l}{2} t_i^0 & -1 - \frac{l}{2} t_i^0 & 0 & 0 & 0 \\
0 & 0 & 1 - \frac{l}{2} t_i^0 & -1 - \frac{l}{2} t_i^0 & 0 & 0 \\
0 & 0 & 0 & 1 - \frac{l}{2} t_i^0 & -1 - \frac{l}{2} t_i^0 \\
0 & 0 & 0 & 0 & 1 - \frac{l}{2} t_i^0 & -1 - \frac{l}{2} t_i^0
\end{pmatrix}
\]  

For each cylinder, the antisymmetric matrix \( t^0 \) is associated with the cross product \( t^0 \) and has the property \( t^0 T = -t^0 \):

\[
t^0 = \begin{pmatrix}
0 & -t_z & t_y \\
t_z & 0 & -t_x \\
-t_y & t_x & 0
\end{pmatrix}
\]

The transpose Jacobian matrix \( \mathcal{J}^T \) and mass matrix \( \mathcal{M} \) are given by:

\[
\mathcal{J}^T = \begin{pmatrix}
-1 & 1 & 0 & 0 \\
\frac{l}{2} t_i^0 & \frac{l}{2} t_i^0 & 0 & 0 \\
0 & -1 & 1 & 0 \\
\frac{l}{2} t_i^0 & \frac{l}{2} t_i^0 & 0 & 0 \\
0 & 0 & -1 & 1 \\
0 & 0 & \frac{l}{2} t_i^0 & \frac{l}{2} t_i^0 \\
0 & 0 & 0 & -1 \\
0 & 0 & 0 & \frac{l}{2} t_i^0
\end{pmatrix}
\]

\[
\mathcal{M} = \begin{pmatrix}
m_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & I_1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & m_2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & I_2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & m_3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & I_3 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & m_4 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & I_4
\end{pmatrix}
\]

Then we get:

\[
\mathcal{M}^{-1} \mathcal{J}^T = \begin{pmatrix}
-m_1^{-1} & m_1^{-1} & 0 & 0 \\
\frac{l}{2} I_1^{-1} t_1^0 & \frac{l}{2} I_1^{-1} t_1^0 & 0 & 0 \\
0 & -m_2^{-1} & m_2^{-1} & 0 \\
\frac{l}{2} I_2^{-1} t_2^0 & \frac{l}{2} I_2^{-1} t_2^0 & 0 & 0 \\
0 & 0 & -m_3^{-1} & m_3^{-1} \\
0 & \frac{l}{2} I_3^{-1} t_3^0 & \frac{l}{2} I_3^{-1} t_3^0 & 0 \\
0 & 0 & 0 & -m_4^{-1} \\
0 & 0 & 0 & \frac{l}{2} I_4^{-1} t_4^0
\end{pmatrix}
\]

and we deduce the final result for the symmetric matrix \( \mathcal{H} = \mathcal{M}^{-1} \mathcal{J}^T \):

\[
\mathcal{H} = \begin{pmatrix}
\frac{1}{m_1} - T_1^0 & -\frac{1}{m_1} - T_1^0 & 0 & 0 \\
-\frac{1}{m_1} - T_1^0 & \frac{1}{m_1} - T_1^0 + \frac{1}{m_2} - T_2^0 & -\frac{1}{m_1} - T_1^0 & 0 \\
0 & -\frac{1}{m_2} - T_2^0 & \frac{1}{m_2} - T_2^0 + \frac{1}{m_3} - T_3^0 & -\frac{1}{m_2} - T_2^0 \\
0 & 0 & -\frac{1}{m_3} - T_3^0 & \frac{1}{m_3} - T_3^0 + \frac{1}{m_4} - T_4^0 \\
0 & 0 & 0 & \frac{1}{m_4} - T_4^0
\end{pmatrix}
\]

where we define the matrix \( T_i^0 = \frac{l^2}{4} I_i t_i^0 t_i^0 \). We then write \( T_i^0 \) in the associated principal axis body frame as \( T_i^{0*} \):
The vector $\lambda$ collects the Lagrange multipliers associated to each joint respectively $\lambda = \lambda_j^{T}_{\sum_{j=1}^{10}}$. The equation $H_\lambda = X$ gives us a system of coupled equations on $\lambda$. Note that ODE solves in one time all the constraints of this articulated system. This is not the case with the SHAKE algorithm where an unconstrained step is first performed, before correcting the positions and orientations iteratively to get the constraints satisfied eventually. The term $J^{-1}F_c$ from Eq. (4) is given by:

$$ \left( \begin{array}{cccccc}
-1 & -\frac{f_1^0}{m_1} & 0 & 0 & 0 & 0 \\
1 & -\frac{f_1^0}{m_1} & 1 & -\frac{f_1^0}{m_1} & 0 & 0 \\
0 & 0 & 1 & -\frac{f_1^0}{m_1} & 0 & 0 \\
0 & 0 & 0 & 1 & -\frac{f_1^0}{m_1} & -1 - \frac{f_1^0}{m_1} \\
0 & 0 & 0 & 0 & 1 & -\frac{f_1^0}{m_1} \\
0 & 0 & 0 & 0 & 0 & 1
\end{array} \right) \left( \begin{array}{c}
\frac{f_{j1}}{m_1} + \frac{f_{j2}}{m_1} \\
\frac{f_{j1}}{m_1} \\
\frac{f_{j1}}{m_1} + \frac{f_{j2}}{m_1} \\
\frac{f_{j1}}{m_1} + \frac{f_{j2}}{m_1} \\
\frac{f_{j1}}{m_1} + \frac{f_{j2}}{m_1} \\
\frac{f_{j1}}{m_1}
\end{array} \right) \right. $$

The method explained here is the exact solution of the problem $H_\lambda = X$ where no collisions are present in the system. With this exact resolution the simulation is faster than the SOR algorithm ($N_{SOR} = 100$ and $\omega_{SOR} = 1.7$) with a gain of 4.

Simulating DNA molecules under magnetic tweezers

To model a DNA molecule, we build a linear chain of rigid cylinders connected with ball-and-socket joints with the first one anchored to the ground. We denote $H$ the matrix $J^{-1}F_c$ with the following properties:

$$ H_{ij} = \frac{1}{m_i} - T_i^0 + \frac{k_{ij}}{\Delta t} $$

$$ H_{ii} = \frac{1}{m_i} - T_i^0 + \frac{k_{ii}}{\Delta t} $$

$$ H_{j+1} = -\frac{1}{m_i} T_i^0 $$

$$ H_{ii+1} = H_{ii+1} $$

$$ H_{ij} = 0 \quad |j-i| > 1 $$

We can now generalise the previous example to the case of a linear chain of $N$ rigid cylinders connected with ball-and-socket joints with the first one anchored to the ground. We denote $H$ the matrix $J^{-1}F_c$ with the following properties:

$$ H_{ij} = \frac{1}{m_i} - T_i^0 + \frac{k_{ij}}{\Delta t} $$

$$ H_{ii} = \frac{1}{m_i} - T_i^0 + \frac{k_{ii}}{\Delta t} $$

$$ H_{j+1} = -\frac{1}{m_i} T_i^0 $$

$$ H_{ii+1} = H_{ii+1} $$

$$ H_{ij} = 0 \quad |j-i| > 1 $$

Using the following decomposition $LDELT$ for the matrix $H$ where $L$ is a block lower matrix with block identity matrix on the diagonal and where $D$ is the block diagonal matrix it is easy to show that $L_{ij} = 0$ for $|j-i| > 1$. From these we get the following equations:

$$ D_1 = H_{ii1} $$

$$ L_{ii+1} = D_1^{-1} H_{ii+1} $$

$$ D_i = H_{ii} - L_{ii+1} D_i L_{ii+1}^T $$

In order to solve the linear system of equations $H_\lambda = X$ we define $\lambda' = DL^T \lambda$ and solve the problem $L \lambda' = X$ in an iterative way:

$$ \lambda_i = X_1 $$

$$ \lambda_i = X_i - L_{ii-1} \lambda_{i-1} $$

and we get the final solution for $\lambda$ by solving the problem $\lambda' = DL^T \lambda$.

The method explained here is the exact solution of the problem $H_\lambda = X$ where no collisions are present in the system. With this exact resolution the simulation is faster than the SOR algorithm ($N_{SOR} = 100$ and $\omega_{SOR} = 1.7$) with a gain of 4.

Simulating DNA molecules under magnetic tweezers

To model a DNA molecule, we build a linear chain of rigid cylinders connected with ball-and-socket joints with the first one anchored to the ground. We denote $H$ the matrix $J^{-1}F_c$ with the following properties:

$$ H_{ij} = \frac{1}{m_i} - T_i^0 + \frac{k_{ij}}{\Delta t} $$

$$ H_{ii} = \frac{1}{m_i} - T_i^0 + \frac{k_{ii}}{\Delta t} $$

$$ H_{j+1} = -\frac{1}{m_i} T_i^0 $$

$$ H_{ii+1} = H_{ii+1} $$

$$ H_{ij} = 0 \quad |j-i| > 1 $$

Using the following decomposition $LDELT$ for the matrix $H$ where $L$ is a block lower matrix with block identity matrix on the diagonal and where $D$ is the block diagonal matrix it is easy to show that $L_{ij} = 0$ for $|j-i| > 1$. From these we get the following equations:

$$ D_1 = H_{ii1} $$

$$ L_{ii+1} = D_1^{-1} H_{ii+1} $$

$$ D_i = H_{ii} - L_{ii+1} D_i L_{ii+1}^T $$

In order to solve the linear system of equations $H_\lambda = X$ we define $\lambda' = DL^T \lambda$ and solve the problem $L \lambda' = X$ in an iterative way:

$$ \lambda_i = X_1 $$

$$ \lambda_i = X_i - L_{ii-1} \lambda_{i-1} $$

and we get the final solution for $\lambda$ by solving the problem $\lambda' = DL^T \lambda$.

The method explained here is the exact solution of the problem $H_\lambda = X$ where no collisions are present in the system. With this exact resolution the simulation is faster than the SOR algorithm ($N_{SOR} = 100$ and $\omega_{SOR} = 1.7$) with a gain of 4.
connect the cylinders to each other by ball-and-socket joints. The radius of the cylinders is set according to the salt buffer concentration of the experimental data we compare with. Indeed, since the DNA molecule is highly negatively charged, DNA-DNA electrostatic repulsion affects the double helix response in single molecule experiments [9,24–27]. This effect can be easily and implicitly included in simulations and theoretical models by introducing an effective DNA radius \( r = r_0 + r_{cd} \) where \( r_0 \) is the crystallographic radius of the DNA double helix and \( r_{cd} \) accounts for the DNA-DNA electrostatic repulsion [28–31]. It turns out that \( r_{cd} \) may be set equal to the Debye length \( \lambda_d \) of the salt buffer solution. As \( \lambda_d = 10/\sqrt{c} \) with \( c \) the salt concentration given in mmol and \( \lambda_d \) in nm, we set the effective radius to \( r = 2 \) nm in 100 mmol monovalent salt buffer for comparison with the reference experimental data of Moscoui et al [32]. Alternatively we use an effective radius \( r = 4 \) nm to fit the experimental data obtained in 10 mmol monovalent salt buffer by Smith et al [1].

We performed all our in silico single molecule experiments with a DNA molecule of contour length \( L = 1 \) µm. The corresponding number of DNA cylinders in the chain is therefore \( N = 300 \). The DNA molecule is anchored, at one end, to a planar surface (mimicking the microscope coverslip), and at the other end, to a rotatable bead (mimicking the magnetic bead). We set the bead radius to \( L/2\pi \) in order to prevent the DNA from looping around it. At both ends of the DNA chain, the rigid cylinders are tangent to their attachment surface.

The final problem that remains to be addressed is how to obtain a correct definition of the bending and twisting deformation of DNA. We have solved this problem by a special choice of the connecting joints and by introducing appropriate restoring torques reacting to the bending and twisting deformations. This has been done based on the bending and twisting energies that are defined according to the usual expressions \( \beta E_b = g_b (1 - \cos \theta) \) and \( \beta E_t = g_t \phi^2/2 \) respectively. The rigidity constants \( g_b \) and \( g_t \) are related to the bending and twisting persistence lengths \( p \) and \( t \) respectively, through the following equations:

\[
\mathcal{L}(g_b) = \langle \cos \theta \rangle 
\]

\[
\langle \cos \theta \rangle = \frac{2p - l}{2p + l} 
\]

\[ g_t = t/l \]

where \( \mathcal{L} \) is the Langevin function (see supplementary Text S1). The bending angle \( \theta \in [0, \pi] \) and twisting angle \( \phi \in [-\pi, \pi] \) are related to the standard Euler transformation \( ZXZ \) and are given by

\[
\cos \theta = t_3^1 t_2 
\]

\[
(1 + t_3^1 t_2) \cos \phi = m_2^T m_1 + \vec{m}_2^T \vec{m}_1 
\]

\[
(1 + t_3^1 t_2) \sin \phi = m_3^T m_1 - \vec{m}_3^T \vec{m}_1 
\]

where \( t_i \) is the tangent vector of cylinder \( i \), \( m_i \) a vector normal to \( t_i \), and \( \vec{m}_i = t_i \otimes m_i \). These three vectors are the principal axis of cylinder \( i \).

We finally get the following expression for the global restoring torque between two connected DNA segments (see supplementary Text S1 for the complete derivation of this equation):

\[
\beta \Gamma_{b+t} = g_b t_3^1 \otimes t_2 + \frac{g_t \phi}{1 + \cos \theta} (t_1 + t_2),
\]

We recall that, for DNA, estimates of the bending persistence length give \( p = 50 \) nm for \( 10 - 500 \) mmol salt buffer (see Refs. [1,33–35]); whereas estimates of the twisting persistence length give \( t = 95 \) nm for \( 10 - 100 \) mmol salt buffer [9,32]. According to the size \( l = 3.34 \) nm of the unit cylinder we find \( g_b = 15.5 \) and \( g_t = 28.4 \).

**Langevin dynamics and global thermostat implementation**

Although well adapted to mechanical simulations, ODE lacks coupling to a thermal bath. As physics engines impose to deal with dynamics equations including inertial terms, in particular for computing constraint forces (collected in \( \mathcal{F}_c \)), we need to turn to some implementation of stochastic isothermal molecular dynamics.
in order to thermalize the system: isothermal to simulate the system at constant temperature, stochastic to ensure ergodicity. The corresponding algorithms are all related to Langevin dynamics and can be cast into local and global thermostats. In local thermostats, such as standard Langevin dynamics, a correction force including both a frictional term and a stochastic term is exerted on each particle to drive the system to the canonical distribution at a prescribed temperature. Global schemes of Langevin dynamics are designed to minimize the perturbation introduced by the thermostat on the Hamiltonian trajectory [so called “disturbance” as defined originally in [36]], hence on the dynamical properties, such as autocorrelation functions, and related quantities, such as diffusion coefficients. In these globally applied thermostats the stochastic term of the correction force acting on each particle is proportional to the momentum of that particle. Two main global algorithms have been designed so far: (a) Stochastic Velocity Rescaling methods, most notably the “global thermostat” introduced by Bussi and Parrinello [19], (b) the Nose-Hoover Langevin thermostat [37]. Here we first show how to implement Langevin-Euler equation in the ODE software. Moreover we show that the global thermostat introduced by Bussi and Parrinello is so remarkably adapted to this implementation that it improves quite significantly the sampling efficiency with respect to local Langevin dynamics (by two orders of magnitude in typical situations), while preserving the time-dependent properties such as autocorrelation functions. The sampling efficiency is defined as usual as the number of independent configurations generated during the time necessary to reach thermal equilibrium.

To begin with, we add to the “mechanical” forces $\mathbf{F}$ an additional, thermal contribution $\mathbf{G}=-\Sigma \mathbf{L}+\Xi \mathbf{W}$ containing a frictional term $-\Sigma \mathbf{L}$ and a random force vector $\Xi \mathbf{W}$. $\Sigma$ is the matrix of the coupling frequencies to the thermostat, $\Xi$ the matrix of white noise amplitudes and $\mathbf{W}$ a generalized vector of normalized and independent Wiener processes. $\Xi$ and $\Sigma$ are related through the fluctuation-dissipation theorem, which reads here

$$\beta \Sigma_{ii} = 2M_i \Sigma_{ii}^*$$  \hspace{1cm} (31)$$

where $\beta=1/k_B T$ with $T$ the temperature of the thermal bath and where the superscript $*$ denotes that the matrices $\Xi$ and $\Sigma$ are chosen to be diagonal in the principal axis body frame (where the matrix $\mathbf{M}$ is diagonal by definition). For simplicity, we choose to fix all the $\Sigma_{ii}$ to a common frequency $\Sigma^*$. Note that $(\Sigma^*)^{-1}$ is the relaxation time of the thermostat, i.e. the autocorrelation time of the kinetic energy (see supplementary Figure S1).

We then improve the sampling efficiency of this Langevin dynamics by extending the “global thermostat” algorithm designed by Bussi and Parinello in 2008 [10] to physics engines. This algorithm allows faster yet correct sampling of the phase space in the canonical ensemble. However, it is designed for the translational degrees of freedom only. In order to apply it to an articulated rigid body system, we therefore have to extend it to the rotational degrees of freedom and adapt it to the ODE software.

To this aim, we replace the traditional Langevin-Euler correction

| Table 1. Main set of parameters used for the DNA model and for the numerical simulations. |
|-----------------|----------------|---------------------------------|-----------------|
| Entity          | Parameter      | Typical value                  | Definition       |
| DNA parameters  | $r$            | 2 nm                           | DNA cylinder effective radius in 100 mmol salt |
|                 | $4$ nm         |                                | DNA cylinder effective radius in 10 mmol salt |
|                 | $l$            | 3.34 nm                        | DNA cylinders length (10 basepairs) |
|                 | $N$            | 300                            | number of DNA cylinders |
|                 | $t$            | 28.4                           | twisting rigidity constant (corresp. to bending persistence length : $t = 95$ nm) |
|                 | $p$            | 15.5                           | bending rigidity constant (corresp. to a contour length $L = 1 \mu m$) |
|                 | $m_0$          | $1.62 \times 10^{-22}$ kg      | mass of each DNA cylinder (10 basepairs) |
| Time units      | $t_0$          | $1.7660 \times 10^{-16}$ s     | natural system time unit $t_0 = l/\sqrt{m_0/k_B T}$ |
| Simulation parameters | $\Delta t$   | $0.000592$ s                   | simulation time step |
|                 | $\Sigma^*$     | $10 \times 1^{-1}$ $t_0$       | thermostat coupling frequency |
|                 | $k_{\text{cpu}}$ | 0.8                            | ODE error reduction parameter |
|                 | $k_{\text{sol}}$ | $10^{-3} t_0/m_0$              | ODE constraint force mixing parameter (hard) |
|                 | $\alpha_{\text{SOR}}$ | 1.7                            | SOR relaxation factor |
|                 | $N_{\text{SOR}}$ | 100                            | number of successive-over-relaxations |

doi:10.1371/journal.pcbi.1003456.t001
force (local thermostat) \(g\) by a corresponding global version \(\tilde{g}\), which reads

\[
\tilde{g} = \sum_i \left[ \frac{\langle E \rangle}{E} \left(1 - \frac{1}{2\beta \langle E \rangle} \right) - 1 \right] L + \sqrt{\frac{\sum_i \beta E}{\beta E}} W L \tag{32}
\]

Eq. (32) shows that \(g\) is proportional to \(L\), so that the stochastic force and torque globally associated with the thermostat is in the same direction as \(L\). Hence, a free particle, i.e. a particle not connected to any other particle, will move on a straight line between two collisions. Note that, nevertheless, the particle will undergo Brownian motion along this straight trajectory. The global version of the Langevin dynamics minimizes the disturbance induced by the thermostat on the Hamiltonian trajectory (equal to \(g^T M^{-1} \tilde{g}\) according to its definition in Ref. [19], but extended here to the rotational degrees of freedom), nevertheless retaining the same thermalization speed as usual Langevin dynamics (see supplementary Figure S1).

When used in the framework of a velocity-based algorithm such as ODE, the global thermostat presents a remarkable advantage. This is because, in this case the global Langevin contribution \(\tilde{g}\) is decoupled from the constraint forces, in the sense that it cancels out in the equations for \(F_r\). More precisely, with our definition of \(\tilde{g}\) (see Eq. (32)), the contribution \(J M^{-1} \tilde{g}\) to the term \(J M^{-1} F\) in Eq. (4) is always zero. In other words, \(\tilde{g}\) not only minimizes the disturbance of the Hamiltonian trajectory \(g^T M^{-1} \tilde{g}\), but also does not disturb the generalized constraint force \(F_c\). Both effects cooperate to achieve a dramatic acceleration of the simulation sampling, that is, in the case of our model, approximately 100 times faster than with the local thermostat (see below “Preliminary tests of validity and performance of the global thermostat” and Figure 2). Importantly this acceleration is compatible with the correct computation of dynamical properties, such as autocorrelation functions.

**Parameter settings of our implementation of ODE**

In all DNA simulations presented in this article, we choose the length of the cylinders as the unit length \(l_0 = l\), the mass of the cylinders as the unit mass \(m_0 = m\) and the unit of thermal agitation \(k_B T\) as the unit of energy \(E_0 = k_B T\), from which we deduce the unit of time \(t_0 = l_0 \sqrt{m_0 / E_0}\). The complete set of parameters of our simulations is given in Table 1. We also choose to deal collisions with a restitution coefficient equal to 1 without surface friction. Hence, when two rigid bodies collide, the constraint force \(l\) imposed by the contact joint that temporarily connects them is directed along their common normal \(n\) at the contact point. We finally choose an error reduction parameter \(k_{err} = 0.8\) and a constraint force mixing parameter \(k_{c/m} = 10^{-9} l_0 / m_0\).

**Results/Discussion**

**Preliminary tests of validity and performance of the global thermostat**

We start to validate our methodology by simulating a DNA molecule without any constraint applied on the bead (neither stretching nor twisting). To this aim, we first check the equipartition theorem. When the system is at thermal equilibrium, its temperature is related to the kinetic energy through the equation

\[
2 \langle E \rangle = \langle L^T M^{-1} L \rangle = (6N - n_c) k_B T \tag{33}
\]

where \(n_c\) is the number of non-redundant holonomic constraints. This relation is standard since \(6N - n_c\) is just the number of degrees of freedom (dof) of the system. Moreover the distribution of the kinetic energy of the system at thermal equilibrium follows a Boltzmann law and therefore reads:

\[
P(E) = \frac{(\beta E)^{n_d/2} \exp(-\beta E)}{\Gamma(n_d)} \tag{34}
\]

with \(n_d = \text{dof} / 2 - 1\). We checked this relation for a DNA molecule of length \(L = 1 \mu\text{m}\) coupled to the two different Langevin thermostats, local and global respectively. The resulting histograms are shown in Figure 1, confirming that: (i) the kinetic energy is correctly sampled at thermal equilibrium with both thermostats, (ii) there is indeed no energy stored in the joints, although these have been softened by effective springs (see above the error reduction parameter \(k_{err}\) in subsection “Introduction to physics engines”).

We then quantified the simulation sampling efficiency by means of the autocorrelation function \(C_{RR}(t) = \langle R(t) \cdot R(0) \rangle / \langle L^2 P \rangle\) of the end-to-end distance \(R\) of a DNA molecule of length \(L = 1 \mu\text{m}\) coupled to the two different Langevin thermostats, local and global respectively. Here the average is performed over the time \(t\) and \(\tau\) denotes the lag-time of the autocorrelation function. A demonstration of the performance of the global thermostat in terms of relaxation rapidity is given in Figure 2. Fitting the exponential

---

**Figure 3. Comparison to force-extension curves.** Red circles: dimensionless stretching force \(fp\) as a function of the mean relative extension \(\epsilon = \langle z_p \rangle / L\). Here the radius of the cylinders is \(r = 4 \text{ nm}\) corresponding to the 10 mmol monovalent salt buffer used in [1]. The other parameters of the simulation are given in Table 1. For comparison, the black solid line reproduces the analytical Worm-Like-Chain force-extension approximation formula [35]. Black triangles correspond to a numerical fit of the exact Worm-Like-Chain model [34] with the same persistence length \(p = 50 \text{ nm}\). Blue crosses: we also show simulations in the limit case \(t = 2p\), with no torsional rigidity \((g = 0)\) and no collisions, and compare it to the theoretical force-extension curve of a Freely-Jointed-Chain (FJC, blue solid line). The statistical error bars on the simulation points are all smaller than the symbol size.

[doi:10.1371/journal.pcbi.1003456.g003]
decrease of both relaxation curves shows that with use of the global thermostat we reach the saturation value at $S \sim 10^4$ whereas this value is reached at $S \sim 10^6$ in the case of the local thermostat, thus resulting in an acceleration factor of about $100$ for this system composed of $N \sim 300$ articulated rigid bodies. Note that the dimensionless lag-time $S \tau$ is equal to $(S \tau_0)(\Delta t/\tau_0)^\alpha = 0.006 n_{\text{step}}$ [see Table 1], with $n_{\text{step}} = \tau/\Delta t$ the corresponding number of time steps. Then a typical run using the global thermostat is of the order of tens of millions of time steps, whereas it is of the order of billions of time steps with usual Langevin dynamics. A striking illustration of the sampling acceleration provided by the global thermostat is also given in supplementary Video S1.

We also compute the tangent-tangent correlation function $\langle t_i^T t_j \rangle$ along the polymer with both local and global thermostats. No significant deviations were found between both thermostats. Results obtained with the global thermostat are plotted in supplementary Figure S2 along with the corresponding theoretical curves.

A simple calculation shows that the tangent-tangent correlation function decreases as $\exp(-|i-j|/\rho)$, from which one can calculate the average bending $\langle \cos \theta \rangle$. With the DNA persistence length $p = 50 \text{ nm}$, this quantity amounts to 0.9355, to be compared to the result from a fit of the simulation curves, giving $\langle \cos \theta \rangle_{\text{sim}} = 0.9359$. The same comparison can be done for the twist angle (with $t = 95 \text{ nm}$), for which the simulation average $\langle \phi \rangle_{\text{sim}} = 0.0352$ matches the theoretical value 0.0332. These comparisons show that our simulation results are in very good agreement with the analytical formulae, thus validating (i) our implementation of the bending and twisting rigidities and (ii) the correct sampling of the DNA conformation space by means of the global Langevin thermostat.

Comparison with the experimental DNA stretching response

We then simulate reference force-extension curves, both theoretical and experimental. We thus perform simulations at given stretching force $f_e$ along the z-axis (normal to the DNA anchor surface), and without torsional constraints on the magnetic bead. In order to fit the experimental data obtained by Smith et al [1] in 10 monovalent salt buffer, we set here the DNA radius (i.e. the radius of the unit cylinders) to $r = 4 \text{ nm}$. The resulting force-extension curve is given in Figure 3 where we plot (red circles) the dimensionless stretching force $b f_p$ as a function of the dimensionless mean relative extension $e = \langle z_0 \rangle/L$. Here $\langle z_0 \rangle$ denotes the mean DNA extension, i.e. the mean distance between the bead and the anchor surface, at zero torsional constraint. For comparison, we also plot (black solid line) the analytical Worm-Like-Chain (WLC) interpolation fitting curve proposed by Bouchiat et al [35] as well as the numerical solution of the WLC model (black triangles) obtained by Marko and Siggia with the same persistence length $p = 50 \text{ nm}$ [34]. The simulation reproduces pretty well the WLC behavior, thus validating our implementation of the DNA bending rigidity. Note that, at low
forces, the extension saturates at a value greater than zero because of the impenetrable ground and magnetic bead that both confine the DNA molecule. This effect is more pronounced than in the experimental curve [1] because the ratio of the bead radius to the DNA length is higher in our simulations.

In Figure 3, we also show the results obtained in the limit case \( l \approx 2p \), for which \( g_b = 0 \) (see Eqs.(24–25)), and when there are no collisions. In this case we expect to observe a Freely-Jointed-Chain response (with \( N \approx 10 \) segments of 300 bp each). The analytical force-extension relation for FJC is given by the well-known expression as a Langevin function \( L(2\beta p) \) and it is also reproduced in Figure 3. Again, the simulation results are in very good agreement with the theoretical formula.

Comparison with the experimental DNA torsional response

More interestingly, magnetic tweezers also allow the application of a torsional strain on a single DNA molecule at constant stretching force. This torsional strain is equal to the number of turns of the magnetic bead around the z-axis due to the rotation of the magnets. The number of turns of the bead is also equal to \( \Delta Lk \), the variation of the linking number of the DNA double helix with respect to the intrinsic twist of the DNA double helix \( Lk_0 = L/h_0 \) with \( h_0 = 3.57 \text{ nm} \) the pitch of the DNA. And we define as usual the DNA relative overtwist as \( \sigma = \Delta Lk/Lk_0 \).

Simulations at constant strain (fixed overtwist). We simulate the rotation-extension behaviour of our DNA model by imposing the bead rotation and compare to experimental data from [32], where the mean relative extension \( \varepsilon = \langle z_T \rangle / L \) is given for different stretching forces as a function of the fixed relative overtwist \( \sigma \) (Figure 4). An example of the simulated dynamics at a stretching force \( f \approx 0.74 \text{ pN} \) and \( \Delta Lk = 15 \) is shown in supplementary Video S2. Again excellent agreement is observed between the experimental bell-shape curves (also called “hat curves”) and the simulated curves. We went further to check the validity of the DNA radius which is set here to \( r \approx 2 \text{ nm} \) according to the 100 monovalent salt buffer used in these experiments. To this aim we refer to a series of papers by Neukirch and co–workers [9,24,25] where they showed that the linear part of the “hat curves” can be expressed as a function of the supercoiling radius \( \rho \), the superhelical angle \( \theta \) and the ratio \( 2p/\tan \theta \) of the twisting and bending persistence lengths, as

\[
\langle z_T \rangle = \frac{1}{\rho} \left( \frac{1}{\cos 2\theta} - 1 \right) - \frac{4\pi \rho}{H \sin 2\theta} \sigma.
\]

where \( \rho \) and \( \theta \) depend on the force applied on the bead.

The comparison between the experimental estimations of the three parameters \( \rho, \theta, \) and \( 4\pi \rho/(H \sin 2\theta) \) deduced from [32] and the corresponding results from our simulations are given as...
functions of the applied force in the three supplementary figures S3, S4, S5. The good agreement observed validates the value applied to the effective DNA radius $r_\sim=2n \text{ m}$ in 100 mmol monovalent salt buffer.

Simulations at constant stress (fixed torque). In the characteristic rotation-extension “hat curves” the DNA extension decreases linearly as a function of $s$ once it reaches a critical point where the resulting applied torque crosses a critical buckling value $\Gamma_c$. This linear decrease corresponds to the formation of plectonemes: any additional turn beyond the buckling transition is absorbed into the growing plectoneme without changes in the torque [10]. As standard magnetic tweezers cannot measure torques, the buckling torque can be only indirectly deduced by integrating the change of the molecule extension with respect to the applied force [32,38]. However, new setups have recently been proposed that enforce a torque and allow its measurement: one of them uses an angular optical trap [39–41], another one a magnetic nanorod coupled to a magnetic bead [42] and a third one a soft magnetic tweezer [43]. These innovative experiments confirm that the torque stays constant during the plectoneme formation, and allow the investigation of the dependence of $\Gamma_c$ on the applied stretching force.

Our technique also allows us to simulate the DNA response when both the stretching force $f$ and the twisting torque $\Gamma$ are imposed, while the number of turns of the magnetic bead $L_k$ is free to evolve. In this case we compute the average overtwist $\langle \sigma \rangle$ for fixed values of the twisting torque $\Gamma$. Figure 5 shows $\Gamma(\langle \sigma \rangle)$ for three different values of the external force. A clear transition from “pure extended” DNA (left part of the curves) to “pure plectonemic” DNA (right part) is observed at (almost) constant critical torque $\Gamma_c$. In the pure extended state, simulations and experimental results [32] are in very good agreement. In this regime $\Gamma(\langle \sigma \rangle)$ is linear and the corresponding slope agrees with theoretical predictions [10]. The estimations of the critical torque obtained in [32] by integrating the change in the molecule extension are reported as horizontal lines in Figure 5. Our simulation correctly reproduces these values, including the dependence of $\Gamma_c$ on the stretching force $f$ throughout the whole range of salt concentrations explored (data not shown).

Note that we sporadically obtained two plectonemes in the same DNA molecule during some simulation runs. Moreover we also observed that plectonemes diffuse along the DNA molecule when the global thermostat is started up (see supplementary Video S1). Interestingly, both features - multiplectonemes and plectoneme diffusion - have been recently observed experimentally [44] and theoretically explained [45]. Note that nevertheless these observations have been obtained with a 21-kb DNA molecule, hence with a much longer molecule than in our simulations (3-kb). This may explain why we did not observe the “multiplectoneme phase” described in [45].

The increase of the average overtwist $\langle \sigma \rangle$ observed at the critical torque in Figure 5 corresponds to the formation of
plectonemes, until the entire DNA molecule is supercoiled. We explored this critical regime by recording, at given \( f \) and \( \Gamma \), the time evolution of both the molecule length and overtwist. When the critical torque is approached, a clear buckling instability appears, with DNA fluctuating between the pure extended state, characterized by a small supercoiling and a large extension, and a pure plectonemic state, with opposite characteristics [10] (Figure 6). Note the anticorrelation between supercoiling and extension.

Beyond the buckling torque, DNA is in the pure plectonemic state, and an increasing torque will tighten the molecule supercoiling. Interestingly, while experimental results [32] do not provide measurements of the average overtwist \( \langle \sigma \rangle \) in this pure plectonemic state, our simulations allow to explore this regime. An example of the simulated dynamics at a stretching force \( f = 0.74 \) pN and twisting torque \( \Gamma = 15 \) pN\( \cdot \)nm is shown in supplementary Video S3. We observe that our simulation results deviate from Marko’s theoretical prediction according to which \( \Gamma_1(\sigma) = k_B T \kappa_{\text{plecto}} (2\pi/h_0) \langle \sigma \rangle \), with \( \kappa_{\text{plecto}} \) the twist persistence length of the plectoneme and \( h_0 \) the pitch of the DNA double helix [10]. Instead we get an affine law which may be cast in a similar form: \( \Gamma(\sigma) = k_B T (2\pi/h_0) T (\sigma - \sigma_0) \) with \( T = 49.60 \pm 0.77 \) nm and \( \sigma_0 = 0.0610 \pm 0.001 \). The presence of the reference value \( \sigma_0 \) seems rather natural since this is the overtwist above which the system enters the pure plectonemic state. The meaning of the \( T \) length is an open question, but we notice that its value is close to the DNA bending persistence length \( p \).

Concluding remarks

The method developed here can conveniently describe all physiological situations involving DNA positive supercoiling, which are of main importance for DNA transcription or replication. One limitation of the modeling developed so far is that extensive modifications of the double helix structure are not accounted for, e.g. base pair opening that occurs when negative supercoiling is applied (DNA denaturation), or the S-DNA transition observed at extremely high force, or the P-DNA transition under very positive torque. Nevertheless this methodology gives an unparalleled opportunity to study more complex biological systems, such as protein-DNA complexes: in particular, we are currently addressing the modeling of magnetic tweezers response of chromatin fibres [7], by associating our in silico DNA model with a rigid body representation of the histone octamer. More recently, in vivo experiments also enable the measurement of the dynamics of chromosome loci in the cell nucleus [11,12,46].

Supporting Information

Figure S1 Autocorrelation function of the kinetic energy \( C_{\text{EE}}(\Sigma^\tau \Sigma^\tau) \) (symbols) of a polymer of size \( L = 1 \) μm and persistence length \( p = 50 \) nm and for three different cylinder lengths: \( l = 3.34 \) (circles), \( l = 6.68 \) (triangles) and \( l = 10.02 \) μm (crosses). We compare the theoretical exponential model \( \exp(-|y-iL|/p) \) for the tangent-tangent correlation to our simulation results (blue lines).

(TIF)

Figure S2 Average tangent-tangent correlation \( \langle f^t_i f^t_i \rangle \) (symbols) for a polymer of size \( L = 1 \) μm and persistence length \( p = 50 \) nm and for three different cylinder lengths: \( l = 3.34 \) (circles), \( l = 6.68 \) (triangles) and \( l = 10.02 \) μm (crosses). We compare the theoretical exponential model \( \exp(-|y-iL|/p) \) for the tangent-tangent correlation to our simulation results (blue lines).

(TIFF)

Figure S3 Supercoiling radius \( \rho \) estimation from simulation results (black circles) and from experimental results [32] (red triangles).

(TIFF)

Figure S4 Helical angle estimation \( \theta \) from simulation results (black circles) and from experimental results [32] (red triangles).

(TIFF)

Figure S5 Slope estimation \( 4\pi p/(\mu H \sin 2\theta) \) from simulation results (black circles) and from experimental results [32] (red triangles).

(TIFF)

Text S1 In this appendix we address the problem of how to obtain a correct and computationally efficient definition of the bending and twisting rigidities for a polymer in general and a DNA molecule in particular.

(PDF)

Video S1 Illustration of the sampling efficiency of global versus local thermostat. This 1 min 23 s video records the simulated dynamics of a DNA molecule manipulated by magnetic tweezers after thermal equilibrium has been reached. The stretching force is \( f = 0.74 \) pN and the number of turns of the bead is fixed to \( \Delta LK = 15 \). From 0 to 20 s the simulation is run with Langevin local thermostat. Then the global thermostat is started up at 21 s and run until 41 s, strikingly accelerating the dynamics of the DNA molecule. Moreover the plectoneme starts to diffuse along the DNA molecule. Then the local thermostat is run again from 42 s to 1 min 02 s, resulting in a dramatic slowing down of the dynamics. And finally the global thermostat is started up again at 1 min 03 s and run until the end of the video at 1 min 23 s. The DNA molecule is composed of \( N = 300 \) rigid cylinders of radius \( r = 2 \) nm and \( l = 3.34 \) nm. The other parameters of the simulation are given in Table 1. This video is also available at http://vimeo.com/51918121.

(AVI)

Video S2 Simulated dynamics of a DNA molecule manipulated by magnetic tweezers with a stretching force \( f = 0.74 \) pN and fixed number of turns of the bead \( \Delta L K = 15 \). This dynamics is performed with the global thermostat after thermal equilibrium has been reached. Note the diffusion of the plectoneme all along the DNA molecule. The DNA molecule is composed of \( N = 300 \) rigid cylinders of radius \( r = 2 \) nm and \( l = 3.34 \) nm. The other parameters of the simulation are given in Table 1. This video is also available at http://vimeo.com/51918378.

(AVI)

Video S3 Simulated dynamics of a DNA molecule manipulated by magnetic tweezers with a stretching force \( f = 0.74 \) pN and fixed twisting torque \( \Gamma = 15 \) pN\( \cdot \)nm. Global thermostat is used all over the simulation. At initial time the number of turns of the bead is zero and the torque starts to be applied to the bead (the rotation of the bead can be followed thanks to the small orange tag on the sphere). Because the applied torque (\( \Gamma = 15 \) pN\( \cdot \)nm) is well above the buckling torque (\( \Gamma_b \approx 10 \) pN\( \cdot \)nm), the DNA molecule is in the pure plectonemic state at the end of the simulation. The DNA molecule is composed of \( N = 300 \) rigid cylinders of radius \( r = 2 \) nm and \( l = 3.34 \) nm. The other parameters of the simulation are given in Table 1. This video is also available at http://vimeo.com/51918151.

(AVI)

Acknowledgments

We acknowledge Julien Mozziconacci and Hua Wong for having introduced us to physics engines in general and ODE in particular, and Fabien Palisson for fruitful discussions about the statistical mechanics.
foundations of the global thermostat. We are grateful to Rhoda Hawkins for careful reading of the manuscript.

References


Author Contributions

Analyzed the data: PC MB JMV. Wrote the paper: PC MB JMV.