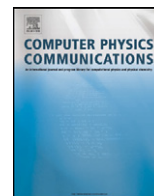




Contents lists available at ScienceDirect

Computer Physics Communications

www.elsevier.com/locate/cpc

REVL: A coarse-grained model for polymers

A.G. Bailey^{a,*}, C.P. Lowe^b, A.P. Sutton^a^a Department of Physics, Imperial College London, Exhibition Road, London SW7 2AZ, United Kingdom^b Van 't Hoff Institute for Molecular Sciences, Universiteit van Amsterdam, Nieuwe Achtergracht 188, 1018 WV Amsterdam, The Netherlands

ARTICLE INFO

Article history:

Received 15 September 2008
Received in revised form 27 January 2009
Accepted 28 January 2009
Available online xxxx

PACS:

61.41.+e
61.25.he
36.20.Ey

Keywords:

Molecular dynamics
Simulations
Constraints
Polymers

ABSTRACT

A new model for polymers is presented. REVL (Rigid, Excluded Volume, Langevin Dynamics) is similar to the coarse-grained, bead spring model for linear chains except that the inter-bead distance is rigidly constrained instead of using an inter-bead potential to encapsulate the connectivity. Static and dynamic results support that REVL accurately reproduces the single-chain behavior of real polymers known from experiment, theory, and published data from existing models. Additionally, a time step can be used that is at least comparable to simulations using a FENE potential without introducing any computational overhead for accessing longer time scale modes. REVL, and more simply the idea of using constraints in Cartesian coordinates for large simulations, was made computationally viable through the recent development of the algorithm MILC SHAKE. We expect it to improve established techniques and aid in the development of new models of import to large scale simulations that were not practicable before.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Using computational methods to approach questions in polymer science is now a familiar concept. Techniques such as Monte Carlo, molecular dynamics, electronic structure calculations, and continuum models have long been used to challenge theory and answer questions experiments cannot hope to probe. With the continuing advancement of technology, this third branch of research – along with theory and experiments – will have only an increasingly important role in scientific investigation.

Restricting ourselves to the area of polymer melts, molecular dynamics (MD) has proven to be a powerful technique for addressing fundamental questions of chain behavior and their viscoelastic properties resulting from topological constraints. Specifically, a seminal body of work by Kremer et al., addresses the connection between the microscopic description and macroscopic quantities and the validity of the Rouse and reptation models [1–4]. They were able to do so by foregoing inclusion of atomistic detail and coarse-graining each chain into a series of units, or beads, each representing a few monomers. The connectivity is encapsulated in an anharmonic potential, called the FENE (Finitely Extensible Non-linear Elastic) potential. A purely repulsive Lennard-Jones potential acts between all beads to enforce excluded volume. The result will

be referred to here as the bead spring model. Although perhaps the most conspicuous application was polymer melts, the basic model and variants are frequently applied to systems at lower densities. In particular, simulations of polymers with implicit solvent include many lipid and surfactant studies [5].

In these types of large scale studies, computational efficiency is of utmost importance. It is therefore desirable to increase the size of the time step as much as possible to quickly explore phase space and probe long time scale modes of interest. The integration step size in an atomistic simulation is constrained by the steepest potential. This generally corresponds to bonded interactions. One can get around this limitation by using constraints. The advantage is that the uninteresting, fast vibrational degrees of freedom are frozen out and a larger time step can be used.

On the other hand, if one is considering a coarse-grained model (as we are here) then these modes have already been integrated out [6]. This fact is exploited in the model proposed by Kremer and Grest, where the interaction between coarse-grained units is represented by a FENE potential. This potential is harmonic for small separations and the authors optimized the parameterization to maximize the time-step that they could use. However, for situations where the molecules under study are relatively inextensible then, even on a coarse-grained level, constraints could be the preferred option. One question that we address here is can this be done while reproducing correct physical behavior and without any significant computational overhead?

* Corresponding author.

E-mail address: aimee.bailey06@imperial.ac.uk (A.G. Bailey).

The best algorithms to calculate the constraint force are slowly convergent and computationally expensive [7,8]. This study using constraints has been made possible through the recent development of MILC SHAKE [9,10], an algorithm for calculating constraints that is capable of efficiently dealing with large systems, discussed in greater detail below.

We propose a new model to study polymers, best described as an interacting, freely-jointed chain coupled to a heat bath. For the sake of brevity, we will refer to it here as REVL (Rigid, Excluded Volume, Langevin Dynamics). It is the same as the bead spring model except that instead of using a FENE potential, the beads are rigidly constrained to be a constant separation. The improvements to the Rouse model – interaction and inertial effects – have thus been retained. Hydrodynamic effects are not taken into account. In this work, the first stage of the validation of the model, we check whether the static and dynamic properties of a single chain are accurately recovered.

2. The model

REVL consists of representing a polymer chain as a series of connected beads. Each ‘bead’ represents a statistical unit, generally many times the persistence length of the molecule, and its coordinates indicate where the mass and forces are concentrated. Each one follows a Langevin equation,

$$\mathbf{F}_i = m_i \ddot{\mathbf{r}}_i = -\nabla U_i - \gamma \dot{\mathbf{r}}_i + \mathbf{d}_i. \quad (1)$$

Here, the total force is \mathbf{F}_i , the mass is m_i , and the position is \mathbf{r}_i , where i represents the bead index. Differentiation with respect to time is denoted by $[\dot{\quad}]$. The function U is the conservative potential, discussed below. The last two terms encapsulate the effect of the environment. The penultimate is a viscous drag contribution with a strength dictated by the friction coefficient γ . The final term is the stochastic force in a random direction, represented by a Gaussian white noise term \mathbf{d} . The parameter γ is related to the amplitude of \mathbf{d} by the fluctuation dissipation theorem, which in three dimensions is

$$\langle \mathbf{d}(t) \cdot \mathbf{d}(t') \rangle = 6\gamma k_B T \delta(t - t'). \quad (2)$$

The constant k_B is Boltzmann’s constant, T is the temperature, and δ is the Dirac delta [11]. Using the Einstein relation, the diffusion coefficient is $D = k_B T / \gamma N$ [12].

U in Eq. (1) has two contributions. The first is a repulsive, shifted Lennard-Jones (L-J) potential, U^R .

$$U_{ij}^R = 4\epsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 + \frac{1}{4} \right], \quad r_{ij} \leq \sigma 2^{1/6}. \quad (3)$$

U^R is zero when r_{ij} is greater than $2^{1/6}\sigma$. The constants ϵ and σ are the L-J energy and length scales, respectively. This force acts between all bead pairs, serving to enforce excluded volume.

The second contribution to U enforces the connectivity of the chain. Instead of introducing an inter-bead potential, as is done in the bead spring model, we force the separation between neighboring beads to be fixed at some distance l by using constraints in Cartesian coordinates. To do so, a zero term is added to the potential, following Lagrange’s method of undetermined coefficients, that takes the form

$$U_i^C = \sum_{i \rightarrow p} \lambda_{ip} \sigma_{ip}, \quad (4)$$

where the summation is over each bead p that is connected to bead i . The λ terms represent the undetermined Lagrange multipliers that are calculated numerically. Each constraint σ_{ip} is zero when the constraint is satisfied, so that U_i^C is then zero. (Please see Appendix A.)

The model is in essence equivalent to the bead spring model except the manner in which the connectivity of the material is represented: an anharmonic spring versus constraints. The form of the repulsive potential and the parameters in the results section were chosen to facilitate direct comparisons between this work and published results from single molecule studies using the bead spring model.

3. MILC SHAKE

The choice to use constraints in a Cartesian representation requires further explanation. There are two approaches one could consider to hold neighboring sites at a fixed position: formulating the equations in generalized coordinates and incorporating the length constraint implicitly in calculations, or working solely in Cartesian coordinates and applying Lagrange multipliers for the constraints.

There exists a body of work formulated in generalized coordinates to model rigid polymer systems undergoing Brownian dynamics, from which one can draw many parallels to the model presented here [13,14]. Two downsides to this approach can be immediately pointed out. First, it is mathematically complicated. Second, transforming between real space and coordinate space is necessary for the calculation of real-space forces such as interactions, which is tedious and expensive. It is not a practical means to model a freely-jointed chain.

The other route is to use a purely Cartesian representation. Until now, the undetermined Lagrange multipliers in Eq. (4), and hence the constraint forces, would have to be calculated using the algorithms SHAKE and RATTLE [7,8]. Improvements have been made to these algorithms [15,16], but by and large the calculations are a large proportion of the CPU time. In the development of a model that aspires to simulate large systems over long time scales, computational efficiency is a paramount concern.

One can sidestep these issues by using a purely Cartesian representation and the new algorithm MILC SHAKE (Matrix Inverted Linearized Constraints), introduced above [9]. MILC SHAKE is a numerical recipe to calculate constraint forces in Cartesian coordinates that exploits a commonly encountered architectural feature: ‘bonds’ that are connected sequentially, which we will refer to here as a linear architecture. This geometry is ubiquitous in problems addressed by computational physicists. Applications include, but are not limited to, inextensible biological filaments, alkanes using the united atom model, and, in the case studied here, models of polymers. With a linear architecture, the non-linear set of equations for the Lagrange multipliers can be solved efficiently using simplified Newton iteration [17]. The algorithm exploits the fact that the linearized equations form a tridiagonal set that can be solved in order n operations. MILC SHAKE is summarized in Appendix A.

Numerical results are presented in Ref. [9] for the case of a simple linear chain tested in a dynamic simulation of an elastic filament. The algorithm can easily be extended to the case of the ring – the Jacobian is a cyclic tridiagonal matrix for a ringed system, meaning there are two additional elements in the upper right-hand and lower left-hand corners. To use MILC SHAKE, a cyclic tridiagonal matrix solver must then be applied to solve for the constraint forces [18]. Fig. 1 shows the CPU time per constraint required to calculate the constraint force, as a function of the number of constraints in the system. Data points are averages of over 1000 time steps. The starting error in the constraints is approximately 0.1%. A ring (squares) is a factor more expensive than a linear chain (circles); however, the performance of MILC SHAKE is orders of magnitude better than that of SHAKE for both system morphologies for the largest polymer chains and a significant factor better for the smallest ones investigated.

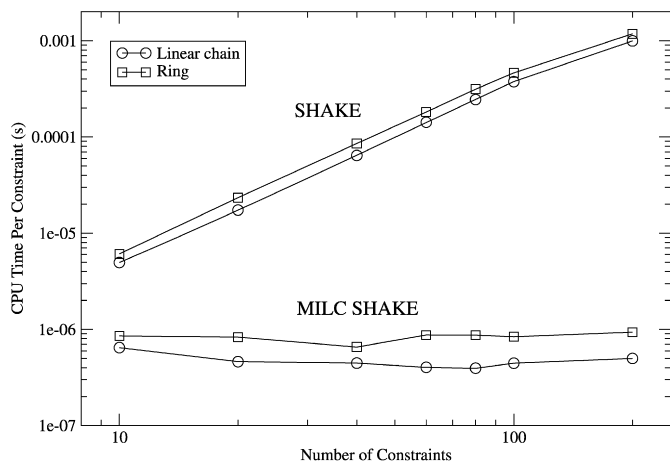


Fig. 1. CPU time per constraint that is required to calculate a constraint force as a function of the number of system constraints.

As mentioned in the introduction, one motivation for using constraints is generally to maximize the total simulation time compared to the CPU wall clock time. The goal is negated if a significant portion of the CPU time has to be devoted to actually calculating the constraint forces. The difference in wall clock time per time step for the two models reduces to a comparison between the cost of calculating a constraint force versus an inter-site force, since all non-bonding contributions to the force on a bead are the same in both models. Consider that for a constraint force using MILC SHAKE, the wall clock time is plotted in Fig. 1 (approximately a microsecond). For comparison, we calculated the wall clock time of a FENE potential force to be approximately 9 μ s. Using our in-house implementation, the wall clock time of REVLD is smaller than that of the bead spring model by almost an order of magnitude for an equivalent time-step. We leave our evaluation general, however, because bonded interactions can amount to a relatively small fraction of the CPU time compared to non-bonded forces for high density simulations. Therefore, any savings will depend greatly on the system of interest and the implementation. It suffices to say that based on our analysis REVLD will conclusively not increase the wall clock time.

Until now, the MILC SHAKE algorithm has been described, but only tested on the specialized test case of an elastic filament. It was speculated that the method is efficient for polymers but not shown. Here we address these issues by applying it to a more general class of problem. MILC SHAKE is incorporated into the REVLD model, which we use to investigate static and dynamic properties of both linear and ringed architectures.

4. Results

The data discussed here are intended to confirm known results from statistical polymer theory that our model must obey to be valid. Furthermore, results are presented in a manner so that they can be compared directly to results using the FENE potential published in Ref. [1]. Static properties of the single polymer chains are first confirmed, followed by dynamic properties. We initially focus on the ideal case, in which the repulsive potential, U^R , is turned off. This we will refer to as a random walk (RW) chain. The self-avoiding walk (SAW) is recovered when U^R is turned on and excluded volume is taken into account. This is the full REVLD model, and will be referred to as such in the results.

4.1. Simulation methods and parameters

The equation of motion, Eq. (1), was integrated using the method described by van Gunsteren and Berendsen in Ref. [19].

For all simulations Boltzmann's constant, the mass of a polymer bead, the inter-bead separation, the L-J length scale σ , and the L-J energy scale ϵ were set to unity. The temperature for all calculations was 1.2ϵ . All length scales are measured in units of σ . The friction coefficient, γ , is either set to 0.5 or 1.5. We investigated various chain lengths, in the range of 50–200 beads.

For all simulations the time step was within the range $0.012\tau - 0.040\tau$, where τ is the L-J time scale defined by $\sigma/(m\epsilon)^{1/2}$. The largest step sizes were used for the RW model, while the smallest (0.012τ) was necessary with the excluded volume interaction. Comparing to the single-chain studies reported in Ref. [4], the magnitude of dt is up to five times the step size for the RW and at least a factor of two and a half larger for simulations with the full REVLD model. The size of the time step was maximized by confirming that the simulation was stable in an NVE ensemble prior to applying the thermostat. This is the same procedure as that used by Grest et al. in Ref. [4]. We should point out, however, that in later work on melts Kremer and Grest used a significantly longer time-step. The time-step reported in [4] is actually quite a conservative estimate. Using the velocity Verlet algorithm we find that in fact for the isolated excluded volume chain a time step of 0.012τ is in fact possible. That is, our algorithm allows the same time-step when excluded volume interaction is included as the model using the optimized FENE potential.

Graphs of dynamic properties were averaged over 3000 initial configurations. The static properties were calculated by averaging over the same number of uncorrelated states. In this paper, results are in agreement with theory if the statistical error of the data is within 3% of the average, where the 'statistical error' is the standard deviation. The amount of elapsed simulation time required for the molecule to be uncorrelated with its initial configuration was taken conservatively to be twice the longest relaxation time τ_N , also called the Rouse time. This was substantiated by looking at the correlation function of the end-to-end vector.

4.2. Statics

The ratio of the mean square radius of gyration to the end-to-end distance of a random walk is $1/6$ [12]. For all of the chain sizes investigated here, we have found this to be the case within acceptable statistical error. For the ideal case, it is also known that the ratio of the mean square radius of gyration of a ring to that of a linear chain of equal number of beads is 0.5 [20]. Our simulation results for a system of 50 beads are consistent with this value.

With the inclusion of the repulsive potential leading to the full REVLD model, the chains follow a self-avoiding walk. The ratio of the mean square radius of gyration to the end-to-end distance for the SAW is, from theory and our simulations, $1/6$. The ratio of the mean square radius of gyration of a ring to a linear chain is $\langle R_g^2 \rangle_{ring} / \langle R^2 \rangle_{linear} \approx 0.57$ with excluded volume [20]. Simulations of a ring and linear chain of 50 beads are consistent with this value.

The structure factor for a RW and SAW chain for $N = 200$ is plotted in Fig. 2. The two sets of data within the region of importance, specifically where $2\pi / \langle R^2 \rangle^{1/2} \ll q \ll 2\pi / \sigma$, should have different slopes. In this region, from theory $S(q) \propto q^{-1/\nu}$, where $\nu \approx 0.58$ for a SAW and $\nu = 0.5$ for a RW. Guidelines for the theoretical scaling are provided, and good agreement is observed for the RW. The dips in the curves near $\log(q) = -0.88$ and -0.65 for the SAW and RW, respectively, are a result of the finite size of the molecule. However, these features reside outside the area of interest.

The radial distribution function shows some structure at small distances. Notably, for the SAW the RDF is zero until $r/\sigma = 1$, where there is a peak. This distance corresponds to the length scale of the purely repulsive L-J potential applied to enforce ex-

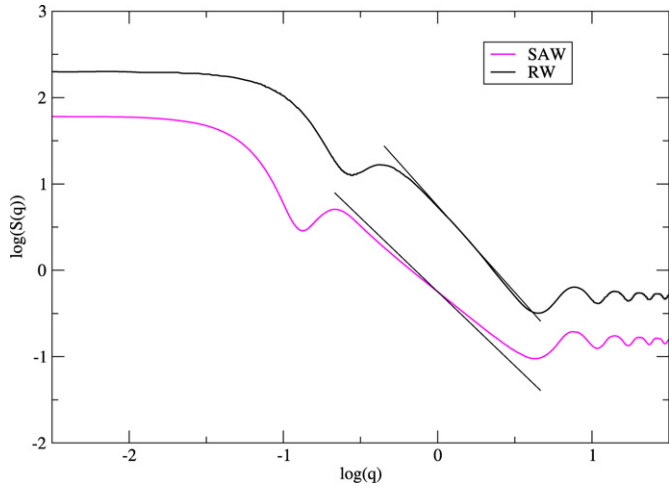


Fig. 2. Structure factor of a linear chain using the RW and SAW models with $N = 200$. Guidelines for the theoretical slope are included.

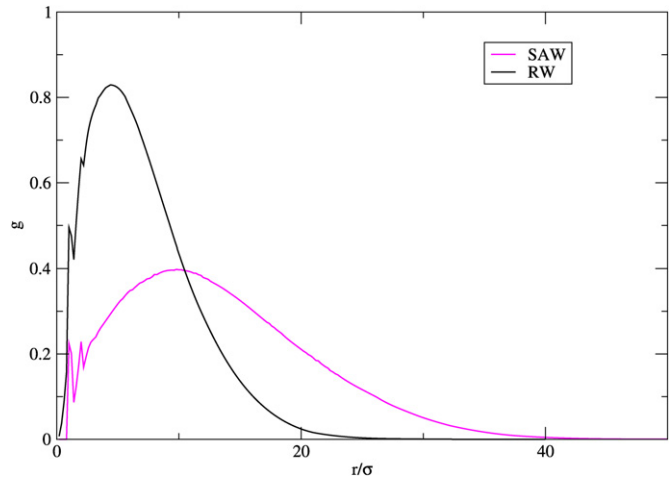


Fig. 3. Radial distribution function of a linear chain using the RW and SAW models with $N = 200$.

cluded volume. There is also a small peak indicating a second nearest-neighbor distance. The RDF of the RW shows some structure for first and second nearest neighbors, but the plots are indicative of the data one would expect for an amorphous structure. See Fig. 3.

4.3. Dynamics

Three quantities describe the dynamics of the system. The first two, listed in Eq. (5), represent the diffusion of single monomers, and the last is the diffusion of the center of mass of the molecule.

$$\begin{aligned}
 g_1 &= \langle [|\mathbf{r}_i(t_0 + t) - \mathbf{r}_i(t_0)|]^2 \rangle, \\
 g_2 &= \langle [|\mathbf{r}_i(t_0 + t) - \mathbf{R}_{c.m.}(t_0 + t) - (\mathbf{r}_i(t_0) - \mathbf{R}_{c.m.}(t_0))|^2] \rangle, \\
 g_3 &= \langle [|\mathbf{R}_{c.m.}(t) - \mathbf{R}_{c.m.}(t_0)|]^2 \rangle.
 \end{aligned}
 \tag{5}$$

In our notation, $\mathbf{R}_{c.m.}$ is the coordinate for the center of mass, and t_0 is the starting time. From theory, the first quantity, g_1 , should scale with $t^{1/2}$ for times much less than τ_N . Far beyond τ_N , g_1 should scale linearly with t . The second quantity, g_2 , should scale with $t^{1/2}$ and t^0 before and after τ_N , respectively. The diffusion of the center of mass, g_3 , should scale linearly with t beyond the shortest relaxation time, τ_0 .

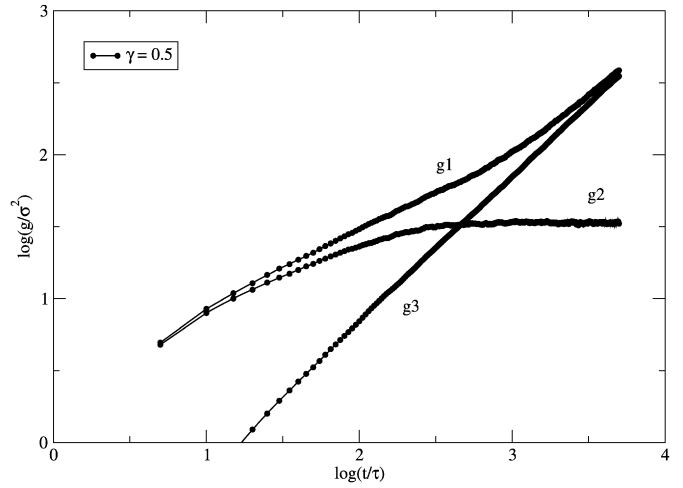


Fig. 4. g_1 , g_2 , and g_3 of a RW linear chain with $N = 200$ and $\gamma = 0.5$.

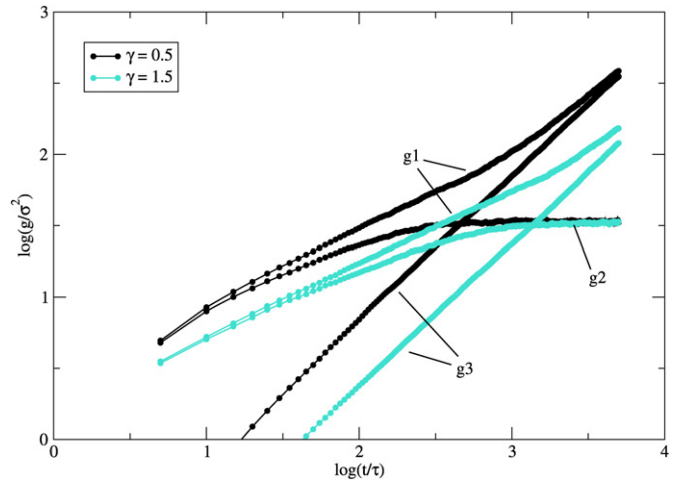


Fig. 5. g_1 , g_2 , and g_3 for a RW linear chain with constant length ($N = 200$) and variable friction ($\gamma = 0.5$ and 1.5).

Any viable model for a polymer melt would necessarily recover the scaling behavior of g_1 , g_2 , and g_3 . The correct dynamics for a random walk using our model was confirmed and is displayed in Fig. 4. Results from a simulation of a linear chain of 200 beads with a friction coefficient of $\gamma = 0.5$ are plotted. In this report, both g_1 and g_2 are averaged over ten interior monomers. Outer monomers were found to be more mobile, as was also found in previous studies [1].

Increasing γ results in a smaller diffusion coefficient according to the Einstein relation; therefore, the graph of g_3 is shifted down. This effect is shown in Fig. 5. For any choice of friction coefficient, given all other parameters are constant, g_2 will plateau at a single value for $t \gg \tau_N$, as observed. The single monomer diffusion at $t \ll \tau_N$ is also suppressed, shown by a downward shift in that region of g_1 . The molecular diffusion coefficients for the two γ 's investigated are calculated from the graphs to be the expected values of $D = 1.2 \times 10^{-2}$ ($\gamma = 0.5$) and $D = 4.0 \times 10^{-3}$ ($\gamma = 1.5$), within 2%.

Fig. 6 compares g_2 and g_3 for both the RW and SAW for $\gamma = 0.5$. The results are not unexpected. Both have an equivalent center-of-mass diffusion coefficient. The mean distance between monomers and the center of mass, indicated by the y-value of the g_2 plateau, is greater for the SAW than the RW, indicating the chains are swollen.

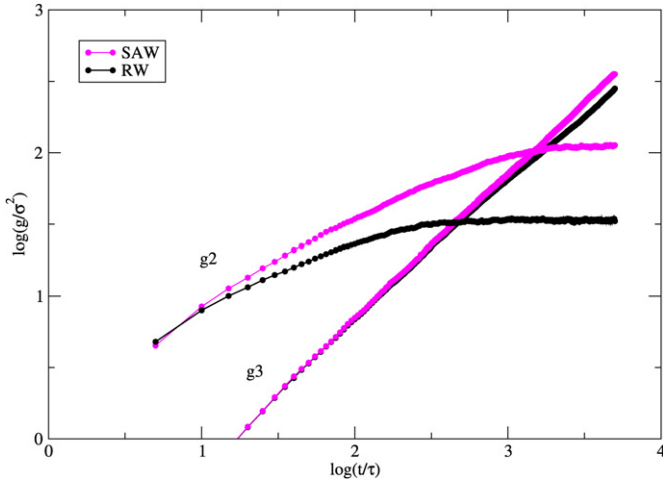


Fig. 6. g_2 and g_3 for a linear chain using the RW and SAW models with $N = 200$ and $\gamma = 0.5$.

5. Discussion and conclusions

In this article we have demonstrated that with the efficient MILC SHAKE algorithm, correct polymer modeling, as proposed by Kremer et al., is possible using time steps at least as long as those possible when the bonding interaction between beads is a FENE potential optimized to allow the maximum time-step. For cases where physical considerations require a significantly stiffer spring interaction, a longer time-step should be possible using constraints. This is the subject of current research. Here we restrict ourselves to comparing with the optimal FENE case. The MILC SHAKE algorithm replaces the evaluation of the FENE interaction and it is sufficiently efficient that it introduces no computational overhead in terms of execution speed. These advantages are significant for single stiff chain studies with implicit solvent. Lipid and surfactant models, many of which use the bead spring model, are a large and important sector of this class of systems. Application of the model to polymer melts is the focus of ongoing research.

All calculations in this paper were performed on a single processor, but parallelization of MILC SHAKE (and hence REVL) is certainly possible. There are multiple ways to proceed. Techniques to parallelize all non-constraint forces included in REVL are by now standard, so let us focus on the parallelization of the constraint calculation. One option is to calculate all forces except the constraints in parallel, leaving the constraint calculation to be handled in series. The speedup is limited by the fraction of serial calculations. If one were using SHAKE iteration to calculate the constraints, this could be up to 10% of the total execution time according to Ref. [21]. However, MILC SHAKE requires a substantially less CPU time, as is evidenced by Fig. 1, making this approach a straightforward and possibly viable option. Regardless, parallelization of MILC SHAKE is feasible using methodology described in Ref. [21] to iteratively solve a coupled set of linear equations. Whether a parallel implementation would outperform models that use a bonding potential (which are trivially parallelized) is however questionable.

We should stress that REVL would not be a feasible coarse-grained model for polymers without the development of MILC SHAKE for calculating constraint forces in a Cartesian representation. In Ref. [9] it was tested for a very simple system with limited applicability. That represents a best case scenario. Here we show that the method can still have applications for much more general problems, those with more realistic Hamiltonians combined with a stochastic thermostat. Consequently, this work illustrates how MILC SHAKE can be used to contribute to existing simula-

tion methods. Additionally, we expect it to aid in the development of new models for large scale simulations using constraints that were previously infeasible.

Acknowledgements

A.G. Bailey thanks the Thouron Award and the National Science Foundation Graduate Research Fellowship Program for support.

Appendix A. MILC SHAKE algorithm

Let us start with a single molecule. The position of bead i at time t is denoted by $\mathbf{r}_i(t)$. The aim is to rigidly constrain bonded pairs of beads to be a specified distance apart. The constraints can be written in the form of n equations, where n is the number of bonds (constraints). Defining $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ as the bond vector and l_{ij} as the bond length, they are

$$\sigma_{ij}(\{\mathbf{r}(t)\}) = \mathbf{r}_{ij}^2(t) - l_{ij}^2 = 0. \quad (\text{A.1})$$

Eq. (A.1) only holds when the magnitude of the bond vector equals the specified bond length. Using Lagrange's method of undetermined multipliers [22], we can enforce the constraint equations while integrating Newton's laws. After the introduction of a zero potential term, the equation of motion in Cartesian coordinates is

$$m_i \frac{d^2 \mathbf{r}_i(t)}{dt^2} = - \frac{\partial}{\partial \mathbf{r}_i} \left[U(\{\mathbf{r}(t)\}) + \sum_p \lambda_{ip} \sigma_{ip}(\{\mathbf{r}(t)\}) \right], \quad (\text{A.2})$$

where the summation is over all beads, indexed by p , connected to bead i . $U(\{\mathbf{r}(t)\})$ is the potential energy of the system, and m_i is the mass of bead i . The λ_{ip} terms represent the undetermined Lagrange multipliers between the two indexed beads.

We can write the position after a time step Δt as the sum of the unconstrained positions after applying all forces except constraints ($\{\tilde{\mathbf{r}}\}$) plus the correction due to the constraint forces ($\{\mathbf{F}^C\}$). To order Δt^2 , the updated positions are

$$\mathbf{r}_i(t + \Delta t) = \tilde{\mathbf{r}}_i(t + \Delta t) + \frac{\Delta t^2}{2m_i} \mathbf{F}_i^C(t). \quad (\text{A.3})$$

These forces act along the bond vectors. Following from Eq. (A.2), a constraint force takes the form

$$\mathbf{F}_i^C(t) = \frac{1}{\Delta t^2} \sum_p \lambda_{ip} \frac{\partial \sigma_{ip}(\{\mathbf{r}(t)\})}{\partial \mathbf{r}_i} = \frac{2}{\Delta t^2} \sum_p \lambda_{ip} \mathbf{r}_{ip}(t). \quad (\text{A.4})$$

The factors of Δt are absorbed in the multiplier for convenience. For brevity, the position-dependence of the force has not been written explicitly. Now the updated positions are

$$\mathbf{r}_i(t + \Delta t) = \tilde{\mathbf{r}}_i(t + \Delta t) + \frac{1}{m_i} \sum_p \lambda_{ip} \mathbf{r}_{ip}(t). \quad (\text{A.5})$$

Substituting Eq. (A.5) into Eq. (A.1), we arrive at the system of n equations that we need to solve:

$$\sigma_{ij} = 0 = \left[\tilde{\mathbf{r}}_{ij}(t + \Delta t) + \frac{1}{m_i} \sum_p \lambda_{ip} \mathbf{r}_{ip}(t) - \frac{1}{m_j} \sum_q \lambda_{jq} \mathbf{r}_{jq}(t) \right]^2 - l_{ij}^2. \quad (\text{A.6})$$

The indices p and q cycle through all beads bonded to beads i and j , respectively.

MILC SHAKE is an efficient algorithm for solving this set of equations, developed by Bailey et al. [9]. This method applies to simple linear and ring architectures.

Algorithm 1. Pseudo code of MILC SHAKE, given a predefined tolerance τ_0 .

```

FOR all molecules
  Calculate  $\mathbf{J}$ ,  $\delta^0$ ,  $\tilde{\sigma}^0$ 
  LOOP
    Solve  $\mathbf{J}\lambda^k = \tilde{\sigma}^k$ 
    Calculate the current positions,  $\{\mathbf{r}^k\}$ , using  $\lambda^k$ 
    Calculate  $\delta^k(\{\mathbf{r}^k\})$ 
    Add  $\delta^k$  to  $\tilde{\sigma}^k$ 
    FOR all constraints
      Calculate  $\tau_{ij}$  using  $\{\mathbf{r}^k\}$ 
    END FOR
    IF  $\max(\tau_{ij}) \leq \tau_0$ 
      SAVE current positions,  $\{\mathbf{r}^k\}$ 
      EXIT LOOP
    END IF
    Increment index  $k$ 
  END LOOP
END FOR

```

To solve Eq. (A.6), we use a Newton-like method. The procedure involves first calculating the $n \times n$ Jacobian. Restricting our attention to a linear architecture, it is

$$\begin{aligned} \mathbf{J}_{i,i-1} &= \frac{-2}{m_i} \tilde{\mathbf{r}}_{i,i+1}(t + \Delta t) \cdot \mathbf{r}_{i-1,i}(t), \\ \mathbf{J}_{i,i} &= \frac{2}{\mu_{i,i+1}} \tilde{\mathbf{r}}_{i,i+1}(t + \Delta t) \cdot \mathbf{r}_{i,i+1}(t), \\ \mathbf{J}_{i,i+1} &= \frac{-2}{m_{i+1}} \tilde{\mathbf{r}}_{i,i+1}(t + \Delta t) \cdot \mathbf{r}_{i+1,i+2}(t), \end{aligned} \quad (\text{A.7})$$

where μ_{ij} is the reduced mass of two beads i and j defined as $\mu_{ij} = m_i m_j / (m_i + m_j)$. The first and last equations, representing the end bonds, only have a diagonal and one off-diagonal component. Explicitly, they are

$$\begin{aligned} \mathbf{J}_{1,1} &= \frac{2}{\mu_{1,2}} \tilde{\mathbf{r}}_{1,2}(t + \Delta t) \cdot \mathbf{r}_{1,2}(t), \\ \mathbf{J}_{1,2} &= \frac{-2}{m_2} \tilde{\mathbf{r}}_{1,2}(t + \Delta t) \cdot \mathbf{r}_{2,3}(t), \\ \mathbf{J}_{n,n-1} &= \frac{-2}{m_n} \tilde{\mathbf{r}}_{n,n+1}(t + \Delta t) \cdot \mathbf{r}_{n-1,n}(t), \\ \mathbf{J}_{n,n} &= \frac{2}{\mu_{n,n+1}} \tilde{\mathbf{r}}_{n,n+1}(t + \Delta t) \cdot \mathbf{r}_{n,n+1}(t). \end{aligned} \quad (\text{A.8})$$

All other matrix elements are zero. The Jacobian is then used to solve the system of linear equations

$$\tilde{\sigma}(\{\tilde{\mathbf{r}}(t + \Delta t)\}) = \mathbf{J}\lambda, \quad (\text{A.9})$$

where $\tilde{\sigma}$ is the vector of n constraint residues.

An iterative procedure is required to converge to the correct values of the vector λ . We use a simplified Newton iteration, the Chord Method [23], to do so. On the first iteration, Eq. (A.9) is solved “as is” for a first approximation to the solution. The n -dimensional vector of instantaneous residues at iteration k , defined

as δ^k , is then added to $\tilde{\sigma}^{k-1}$ to form the sum of the constraint residues leading up to the current iteration, $\tilde{\sigma}^k$. The δ^k term approximates the non-linear contribution and tends to zero in subsequent iterations, as required for convergence. The iterative procedure is therefore

$$\mathbf{J}\lambda^k = \tilde{\sigma}^k = \tilde{\sigma}^{k-1} + \delta^k \quad (\text{A.10})$$

$$\delta_{i,i+1}^k = \mathbf{r}_{i,i+1}^k(t + \Delta t)^2 - l_{i,i+1}^2, \quad (\text{A.11})$$

where the set of vectors, $\mathbf{r}_{i,i+1}^k(t + \Delta t)$, are the bond vectors calculated at iteration k using the current values of the constraint forces to update the bead positions. The Jacobian is calculated only once in this procedure. The procedure is summarized in pseudo code in Algorithm 1.

The cost of a simplified Newton iteration is generally order n^3 , corresponding to the cost of solving Eq. (A.10) every iteration. However, the Jacobian of this set of equations – when they are ordered sequentially, matching their relative position along the contour of the chain – is of a very special form: it is tridiagonal. Inverting a tridiagonal matrix can be done easily and efficiently in order n operations [18]. This being the case, the cost of the algorithm is order n .

References

- [1] G.S. Grest, K. Kremer, Phys. Rev. A Rapid Comm. 33 (1986) 3628.
- [2] K. Kremer, G.S. Grest, I. Carmesin, Phys. Rev. Lett. 61 (1988) 566.
- [3] K. Kremer, G.S. Grest, J. Chem. Phys. 92 (1990) 5057.
- [4] K. Kremer, G.S. Grest, in: K. Binder (Ed.), Monte Carlo and Molecular Dynamics Simulations in Polymer Science, Oxford University Press, New York, 1995, pp. 194–271.
- [5] I.R. Cooke, K. Kremer, M. Deserno, Phys. Rev. E 72 (2005) 011506.
- [6] D. Frenkel, B. Smit, Understanding Molecular Simulation, Academic Press, London, UK, 1996.
- [7] J.-P. Ryckaert, G. Ciccotti, H.J.C. Berendsen, J. Comput. Phys. 23 (1977) 327.
- [8] H. Andersen, J. Comput. Phys. 52 (1983) 24.
- [9] A.G. Bailey, C.P. Lowe, A.P. Sutton, J. Comput. Phys. 227 (2008) 8949.
- [10] If the web page no longer exists at the time of attempted access, please contact the authors for code, <http://www.cmth.ph.ic.ac.uk/people/aimee.bailey/milcshake.html>.
- [11] N.G. Van Kampen, Stochastic Processes in Physics and Chemistry, Elsevier, Oxford, 2007.
- [12] P.G. de Gennes, Scaling Concepts in Polymer Physics, Cornell University Press, Ithaca, NY, 1979.
- [13] G.H. Nyland, P. Skjetne, A. Mikkelsen, A. Elgsaeter, J. Chem. Phys. 105 (1996) 1198.
- [14] D.C. Morse, Adv. Chem. Phys. 128 (1994) 65.
- [15] P. Gonnet, J. Comput. Phys. 220 (2007) 740.
- [16] V. Krautler, W.F. van Gunsteren, P.H. Hunenberger, J. Comput. Chem. 22 (2001) 501.
- [17] P.E. Gill, W. Murray, M.H. Wright, Numerical Linear Algebra and Optimization, Addison-Wesley, Redwood City, CA, 1991.
- [18] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes in FORTRAN: The Art of Scientific Computing, second ed., Cambridge University Press, New York, NY, 1992.
- [19] W.F. van Gunsteren, H.J.C. Berendsen, Mol. Phys. 45 (1982) 637.
- [20] J.J. Prentis, J. Chem. Phys. 76 (1982) 1574.
- [21] Y. Weinbach, R. Elber, J. Comput. Phys. 209 (2005) 193.
- [22] H. Goldstein, Classical Mechanics, second ed., Addison-Wesley, Reading, MA, 1980.
- [23] P.E. Gill, W. Murray, M.H. Wright, Numerical Linear Algebra and Optimization, Addison-Wesley, Redwood City, CA, 1991.