A QMC Approach for High Dimensional Fokker-Planck Equations Modelling Polymeric Liquids

G. Venkiteswaran [a,*], M. Junk [b]

[a] *Graduiertenkolleg Mathematik und Praxis, Universität Kaiserslautern*
67653 Kaiserslautern, Germany

[b] *Fachbereich Mathematik, Universität des Saarlandes*
66041 Saarbrücken, Germany

Abstract

A classical model used in the study of dynamics of polymeric liquids is the bead-spring chain representation of polymer molecules. The chain typically consists of a large number of beads and thus the state space $\mathcal{V}$ of its configuration, which is essentially the position of all the constituent beads, turns out to be high dimensional. The distribution function governing the configuration of a bead-spring chain undergoing shear flow is a Fokker-Planck equation on $\mathcal{V}$. In this article, we present QMC methods for the approximate solution of the Fokker-Planck equation which are based on the time splitting technique to treat convection and diffusion separately. Convection is carried out by moving the particles along the characteristics and we apply the algorithms presented in [1] for diffusion. Altogether, we find that some of the QMC methods show reduced variance and thus slightly outperform standard MC.

Key words: QMC, diffusion equation, MC, Fokker-Planck equation

1 Introduction

In the present work we consider a Fokker-Planck equation which arises in connection with a mathematical model of dilute polymeric solutions. Such solutions show interesting non-Newtonian behavior like shear rate dependent

---

* Corresponding author.
   Email addresses: `gvenki@mathematik.uni-kl.de` (G. Venkiteswaran), `junk@num.uni-sb.de` (M. Junk).

viscosity, rod climbing etc., (see [2] for a comprehensive list). Our aim is to calculate the material functions namely, the viscosity and the normal stress difference which are responsible for the dynamic behavior of dilute polymeric solutions.

The polymer model we use is the bead-spring chain, also called the Rouse chain [3], in which sub-strings of the polymer molecules are represented by beads (see figure 1) and interactions are indicated by connecting springs (even though a more complicated interaction potential is employed). The configuration of such a chain is described by specifying all the connector vectors $\boldsymbol{Q}_i = \boldsymbol{r}_{i+1} - \boldsymbol{r}_i$, $i = 1, \ldots, n-1$, where $\boldsymbol{r}_\nu$, $\nu = 1, \ldots, n$ are the position vectors of the beads. Since each of the vectors $\boldsymbol{Q}_i$ has three components, the configuration space is $\mathbb{R}^{3(n-1)}$. For reasons of simplicity and clarity, we shall henceforth denote $3(n-1)$ by s. With $n = 20$, for example, the dimension of the configuration space is $s = 57$. The special case $n = 2$ is often referred to in the literature as dumbbell.

PSfrag replacements
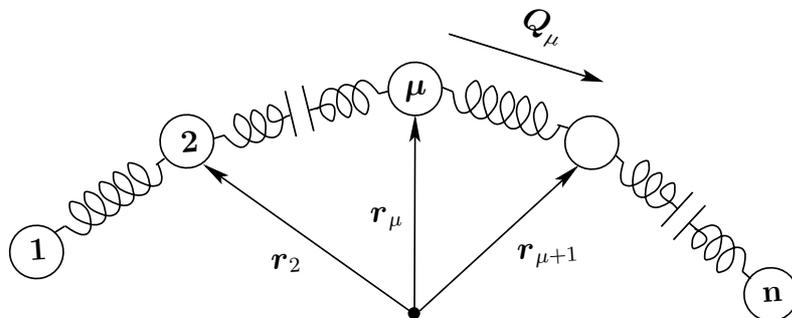
Fig. 1. Rouse model of a bead-spring chain.

The type of flow often used to characterize polymeric liquids is the shear flow whose linear velocity field is given by

$$v(\mathbf{x}) = \boldsymbol{\kappa}\,\mathbf{x}, \qquad \boldsymbol{\kappa} = \beta \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \tag{1}$$

where β is the constant shear rate. The solvent is considered to exert a drag force, and a random Brownian force on the chain, and the chain is considered to interact with itself through a potential which consists of two contributions, a quadratic attractive part that prevents the beads of the chain from going very far apart, and a Gaussian repulsive part called the excluded volume potential [4], that resists any pair of beads from coming very close to each other.

As the bead-spring system moves in the flow and there are random forces, the exact configuration of the system cannot be determined, but rather only a probabilistic estimate can be given. This probability, which describes the

2

likelihood of finding the chain near a particular configuration $\boldsymbol{Q}$ is given by $\psi(t, \boldsymbol{Q})d\boldsymbol{Q}$, where the density ψ is the so called configurational distribution function. In a dilute polymeric solution with a large number of non-interacting chains, the measure $\int_{\mathcal{C}} \psi(t, \boldsymbol{Q})d\boldsymbol{Q}$ specifies the percentage of chains having a configuration $\boldsymbol{Q}$ in a subset $\mathcal{C}$ of $\mathbb{R}^s$. According to [3], the distribution function is governed by a Fokker-Planck equation of the following non-dimensionalized form

$$\frac{\partial \psi}{\partial t} = -\sum_{j=1}^{n-1} \frac{\partial}{\partial \boldsymbol{Q}_j} \cdot \left(\boldsymbol{\kappa}\, \boldsymbol{Q}_j - \frac{1}{4} \sum_{k=1}^{n-1} A_{jk} \frac{\partial \phi}{\partial \boldsymbol{Q}_k} \right) \psi + \frac{1}{4} \sum_{j,\,k=1}^{n-1} A_{jk} \frac{\partial}{\partial \boldsymbol{Q}_j} \cdot \frac{\partial \psi}{\partial \boldsymbol{Q}_k} \quad (2)$$

where,

- $\partial/\partial \boldsymbol{Q}_j\cdot$ denotes divergence with respect to $\boldsymbol{Q}_j$ and $\partial\psi/\partial\boldsymbol{Q}_k$ the $\boldsymbol{Q}_k$ gradient.
- $\boldsymbol{\kappa}$ is the gradient of the solvent velocity field (see (1)).
- The $(n-1) \times (n-1)$ matrix $\boldsymbol{A}$ is the Rouse matrix, defined by,

$$A_{ij} = \begin{cases} 2 \text{ if } & |i-j| = 0 \\ -1 \text{ if } & |i-j| = 1 \\ 0 \text{ otherwise} \end{cases} \quad (3)$$

- The quantity ϕ is the total potential energy of the bead-spring chain,

$$\phi(\boldsymbol{Q}) = \frac{1}{2} \sum_{i=1}^{n-1} \boldsymbol{Q}_i \cdot \boldsymbol{Q}_i + \frac{1}{2} \frac{z}{d^3} \sum_{\substack{\mu,\nu=1 \\ \mu \neq \nu}}^{n} \exp\left(-\frac{1}{2} \frac{r_{\mu\nu}^2(\boldsymbol{Q})}{d^2} \right). \quad (4)$$

Here, $r_{\mu\nu}$ is the magnitude of the vector $\boldsymbol{r}_{\mu\nu} = \boldsymbol{r}_\mu - \boldsymbol{r}_\nu$, connecting the pair of beads μ and ν. The parameter d controls the extent of the repulsive potential, and z describes its strength.

We shall later employ different initial conditions to determine the steady state solution of (2).

The stress tensor $\boldsymbol{\tau} = \boldsymbol{\tau}^s + \boldsymbol{\tau}^p$ characterizing the flow behavior of polymeric liquids consists of two contributions namely, one from the solvent $\boldsymbol{\tau}^s$, and the other from the polymer $\boldsymbol{\tau}^p$. The rheological properties of the polymer solution can be obtained by calculating the polymer contribution to the stress tensor, which is given by Kramers expression [3],

$$\boldsymbol{\tau}^p = -\sum_{j=1}^{n-1} \int_{\mathbb{R}^s} \boldsymbol{Q}_j \otimes \frac{\partial \phi}{\partial \boldsymbol{Q}_j} \psi_\infty d\boldsymbol{Q}. \quad (5)$$

where ψ_∞ is the stationary solution of (2). The two important rheological properties of a dilute polymer solution, undergoing simple shear flow, are the

3

viscosity η and the first normal-stress-difference coefficient Ψ given by

$$\eta = -\frac{\tau_{xy}^p}{\beta}, \qquad \Psi = -\frac{\tau_{xx}^p - \tau_{yy}^p}{\beta^2}. \qquad (6)$$

Numerically, the material functions η and Ψ are calculated at every time step till steady state is reached. A typical plot of viscosity and first normal stress difference coefficient is as shown in figure 2, taken from the dumbbell case $s = 3$ ($n = 2$). At this point, we wish to stress that the physically relevant
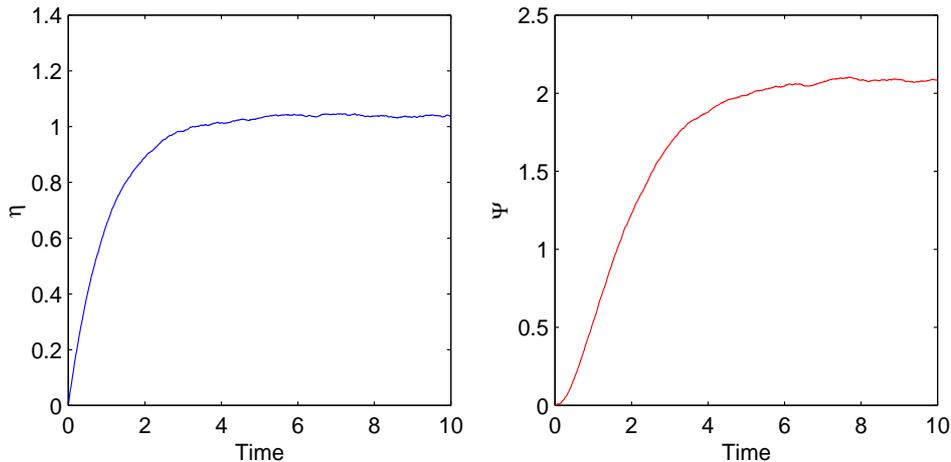
Fig. 2. Typical plot of viscosity (left) and first normal stress difference coefficient (right) versus time for the dumbbell case $s = 3$ ($n = 2$).

information which can be drawn from the computations are the stationary values. This is due to the fact that the Kramers expression (5) is only valid for the stationary case. Nevertheless, the time evolution of the integral functional (5) gives an idea of the dynamical behavior of the Fokker-Planck equation. For example, one can see a dimensional dependence: while in low dimensional cases the curves are monotonically increasing, in the higher dimensional case there is an initial spike in the values of viscosity and first normal stress difference coefficient as can be seen from figure 3. This just means that one needs to run the simulation considerably longer till the stationary values are obtained.

Note that the Cauchy problem for (2) has to be solved on $\mathbb{R}^{3(n-1)}$ and with $n = 20$, being a typical number of beads, this amounts to a Fokker-Planck problem on $\mathbb{R}^{57}$. The aim of this paper is to present fast and accurate methods that are applicable to such high-dimensional problems.

The paper is organized as follows. Section 2 outlines the various methods followed by section 3 on numerical results.

4
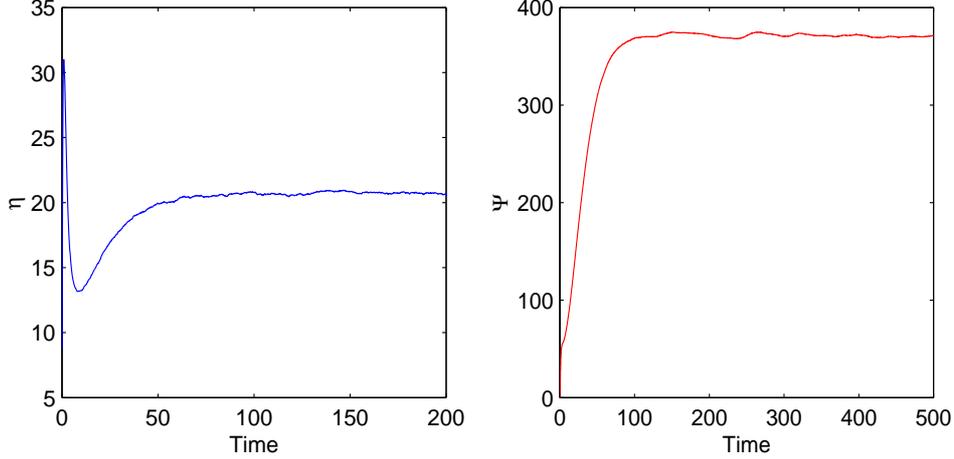
Fig. 3. Plot of viscosity (left) and first normal stress difference coefficient (right) versus time showing stationary situation in the case of 8 beads ($s = 21$).

2 The methods

It can be easily seen from (2) that the Fokker-Planck equation consists of an advective term as well as a diffusive term. In order to simulate such a process numerically, we shall split the convection part and the diffusion part so that the approximate solution of the original problem is written as a composition of the solution operators of the sub-problems. Time is discretized in steps of Δt. The splitting algorithm for (2) proceeds in several parts.

2.1 Change of coordinates

Since the Rouse matrix (3) is diagonalizable, the second order operator appearing in (2) can actually be transformed onto a Laplacian. The $(n-1) \times (n-1)$ orthogonal matrix Ω_{ij} which diagonalises A_{ij} is given by

$$\Omega_{ij} = \sqrt{\frac{2}{n}} \sin \frac{ij\pi}{n}, \quad i, j = 1, \ldots, n-1 \tag{7}$$

and satisfies the relation

$$\sum_{j=1}^{n-1} \sum_{k=1}^{n-1} \Omega_{ji} A_{jk} \Omega_{kl} = a_l \delta_{il} \tag{8}$$

where $a_i = 4\sin^2(i\pi/2n)$ are the eigenvalues of A_{ij}. If we denote the matrix $a_l \delta_{il}$ by D, then the last relation can be written as

$$\Omega^T A \Omega = D$$

5

Since D is a diagonal matrix, we have

$$\left(\Omega\sqrt{D}^{-1}\right)^T A\left(\Omega\sqrt{D}^{-1}\right) = I \tag{9}$$

Now, the transformation $Z = 2\Omega\sqrt{D}^{-1}$ actually transforms A into

$$\sum_{j=1}^{n-1}\sum_{k=1}^{n-1} Z_{ji}A_{jk}Z_{kl} = 4\,\delta_{il} \tag{10}$$

and also satisfies

$$\sum_{k=1}^{n-1} Z_{kj}Z_{ki} = 4\frac{\delta_{ji}}{a_j}$$

Introducing the new variables $\boldsymbol{Q}_j^*, j = 1, \ldots, n-1$, by

$$\boldsymbol{Q}_j^* = \sum_{k=1}^{n-1} Z_{jk}\boldsymbol{Q}_k \tag{11}$$

we obtain the transformed equation,

$$\frac{\partial\psi}{\partial t} = -\sum_{j=1}^{n-1}\frac{\partial}{\partial\boldsymbol{Q}_j^*}\cdot\left(\boldsymbol{\kappa}\boldsymbol{Q}_j^* - \frac{\partial\phi}{\partial\boldsymbol{Q}_j^*}\right)\psi + \sum_{j=1}^{n-1}\frac{\partial}{\partial\boldsymbol{Q}_j^*}\cdot\frac{\partial\psi}{\partial\boldsymbol{Q}_j^*}. \tag{12}$$

The form of the stress tensor however remains unchanged by this transformation.

For ease of notation, we shall supress the stars in the vectors $\boldsymbol{Q}_j^*$. Then, the structure of (12) leads us to consider initial value problems for advection equation,

$$\frac{\partial\psi}{\partial t} + \sum_{j=1}^{n-1}\frac{\partial}{\partial\boldsymbol{Q}_j}\cdot\left(\boldsymbol{\kappa}\boldsymbol{Q}_j - \frac{\partial\phi}{\partial\boldsymbol{Q}_j}\right)\psi = 0 \tag{13}$$

and the diffusion equation

$$\frac{\partial\psi}{\partial t} = \sum_{j=1}^{n-1}\frac{\partial}{\partial\boldsymbol{Q}_j}\cdot\frac{\partial\psi}{\partial\boldsymbol{Q}_j} \tag{14}$$

separately. In the splitting approach, the approximate solution of (12) at time Δt is obtained by first solving (13) for a time Δt, feeding the result into (14) and solving (14) again up to $t = \Delta t$.

Since (13) and (14) are posed on a very high dimensional space, we are using particle methods for the numerical approximation. The basic idea in this approach is to relate the unknown function ψ to the measure $\psi d\boldsymbol{Q}$ and to

6

approximate $\psi d\boldsymbol{Q}$ by a discrete measure,

$$\psi d\boldsymbol{Q} \sim \frac{1}{N} \sum_{i=1}^{N} \delta_{\boldsymbol{Q}^{(i)}}. \tag{15}$$

The next step is then to translate the dynamics described by (13) and (14) in the framework of the particles $\boldsymbol{Q}^{(i)}$ which fully determines our approximation. The details are given in the following subsections.

2.2   Approximation of the initial value

The initialization step of every particle simulation is the particle approximation of the initial value. For our Fokker-Planck equation (12), we shall later employ several initial conditions:

- (SN) $\psi_0(\boldsymbol{Q}) = (2\pi)^{-s/2} \exp(-|\boldsymbol{Q}|^2/2)$

- (DD) $\psi_0(\boldsymbol{Q}) = \delta_{\mathbf{0}}$

- (EQ) $\psi_0(\boldsymbol{Q}) = N_{eq} \exp(-\phi)$, with ϕ as in (4) (refer [3]).

The abbreviations SN, DD and EQ stand for standard normal, Dirac delta and equilibrium distribution respectively.

The approximation of (DD) is the most simple where we just initialize N particles with the configuration vector zero, that is, all beads are in the same position. In this case, repulsion will be dominant in the initial phase of the evolution. For the case (SN), we employ the inversion method (see [1] or [5]) which is applicable in this case since the cumulative distribution function is available and is easily invertible. However, this is not the case for the most physical condition (EQ) which describes the equilibrium state of the polymeric liquid without a shear flow. In this situation we resort to the acceptance-rejection technique [5]. Accordingly, the required dominating function is taken as

$$N_{eq} \exp\left(-|\boldsymbol{Q}|^2/2\right) \geq \psi_0(\boldsymbol{Q}) \tag{16}$$

where N_{eq} is estimated by a numerical evaluation. It may be noted that this method is generally slow in the sense that one needs to produce a large number of points in order to qualify a few of them.

2.3   Convection

The result of the initial sampling step is a set of particle positions approximating the initial value. In order to carry out convection, we transport each

of the particles along the characteristics. If we set

$$C = \kappa Q_j - \frac{\partial \phi}{\partial Q_j} \qquad (17)$$

then (13) can be written as

$$\frac{\partial \psi}{\partial t} + \text{div}\,(C\psi) = 0 \qquad (18)$$

We now move the particles along the characteristics of (18) given by the solution of $\dot{Q} = C(Q)$. Using explicit Euler discretization leads us to

$$Q_{k+1}^{(i)} = Q_k^{(i)} + \Delta t\, C(Q_k), \quad i = 1, \ldots, n. \qquad (19)$$

For a similar approach, we refer to [6].

It may be noted that the evaluation of C requires the evaluation of $\partial \phi / \partial Q_j$ and from expression (4), it is clear that this involves high computational cost for a system with a large number of beads. We shall later see that the transport part constitutes a major portion of the computational time.

2.4 Diffusion

In this subsection, we mainly recapitulate the various algorithms presented in [1] for the simulation of diffusion. In [1], at each time step, the particle positions were first sorted and later incremented in a quasi-random way. The sorting introduced a permutation τ and the quasi-random incremental step gave rise to another permutation σ. Accordingly the one step movement of the particles is given by

$$Q_{k+1}^{(i)} = Q_k^{(i)} + \sqrt{2\Delta t}\, H^{-1}(Y_{\sigma^{-1}(\tau(i))}), \quad i = 1, \ldots, N. \qquad (20)$$

where $H(z) = (H(z_1), \ldots, H(z_s))$ with H given by

$$H(x) = \frac{1}{2}\left(1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right) \quad \text{where} \quad \text{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt \qquad (21)$$

and $\frac{1}{N}\sum_{i=1}^{N} \delta_{Y_i}$ is a measure approximation of the indicator function of the s-dimensional unit cube.

8

In the previous subsections, we have seen how to interpret the dynamics of equation (13) and (14) in the framework of particles. In every time step, at the end of the convection step, the particles reach an intermediate stage $\boldsymbol{Q}^{(i)}_{n'+1/3}, i = 1, \ldots, N$, which is then followed by diffusion which completes one cycle (sorting of the particles may give rise to a second intermediate stage $\boldsymbol{Q}^{(i)}_{n'+2/3}$).

Our notation for the different type of algorithms will be as in [1] with the only difference that also the convection step is included. The algorithm with $\sigma = \tau = id$ and particle approximations based on independent (pseudo) random numbers is called MC. The name QMC$(r, 0)$ with $1 \leq r \leq s$ stands for the algorithm with r-dimensional mixing and no sorting. Similarly, QMC$(0, r)$ indicates the algorithm with no mixing but r-dimensional sorting and QMC(l, r) represents the algorithm where both mixing and sorting is performed in l and r of the s dimensions respectively. For example, QMC(s, s) is the algorithm proposed in [7] where both sorting and mixing is done in all the coordinates. QMC$(0, 1)$ would be an algorithm where only sorting is performed along one coordinate and QMC$(1, 1)$ involves sorting and mixing along a single coordinate.

To summarize the results of [1], it is observed that only QMC$(0, 1)$ and QMC$(1, 1)$ are the algorithms which have faster convergence than MC and are applicable in high dimensional cases. In what follows, we shall mainly use the QMC$(1, 1)$ algorithm.

The complete algorithm for the Fokker-Planck equation can be summarized as follows.

$$\underline{\text{ALGORITHM QMC}(l, r)}$$

(1) **Input** n, $1 \leq m \in \mathbb{Z}$, Δt and T

(2) **Set** $s = 3(n-1)$ and choose b, the least prime $\geq s + \max(l, r)$; $N = b^m$

(3) **Approximate the initial value**, $\psi_0 d\boldsymbol{Q} \approx \dfrac{1}{N} \sum_{i=1}^{N} \delta_{\boldsymbol{Q}_0^{(i)}}$

(4) **Initialize** $n' \leftarrow 0$

(5) WHILE $(n'\Delta t < T)$
{

    (a) **Transport** all the N particles along the integral curves of $\boldsymbol{C}$
        FOR $i = 1, N$
$$\boldsymbol{Q}_{n'+1/3}^{(i)} = \boldsymbol{Q}_{n'}^{(i)} + \Delta t \, \boldsymbol{C}^{(i)}(\boldsymbol{Q}_{n'})$$
        END

    (b) Construct the permutation τ (by sorting if $r > 0$). Set up
$$\boldsymbol{Q}_{n'+2/3}^{(i)} = \boldsymbol{Q}_{n'+1/3}^{(\tau^{-1}(i))}$$

    (c) **Diffusion**
        Construct the permutation σ (by mixing if $l > 0$) using $\boldsymbol{\lambda}_k$
        from a $(0, m, s + l)$-net $(\boldsymbol{\lambda}_k, \boldsymbol{Y}_k)$, $k = 1, \ldots, N$

        FOR $k = 1, N$
$$\boldsymbol{Q}_{n'+1}^{(\sigma(k))} = \boldsymbol{Q}_{n'+2/3}^{(\sigma(k))} + \sqrt{2\Delta t} \, \boldsymbol{H}^{-1}(\boldsymbol{Y}_k)$$
        END

    (d) $n' \leftarrow n' + 1$
}

The applicability of QMC(l, r) is restricted by memory limitations (refer section 2.4 of [1]) and it is henceforth clear that only very small values of l and r allow us to work in high dimensions and this is especially true for the polymer model described earlier.

3   Numerical simulation of Fokker-Planck equation

In this section we summarize the results of various numerical simulations that have been carried out using different approaches. We mainly compare the computations based on straightforward MC algorithm (MC without variance reduction etc.,) with the algorithms outlined in the previous section. All the computations are done on a AMD Athlon 1400 MHz machine with 1.5GB memory running Debian Linux 3.0. The CPU time we shall refer to is as measured on this machine. The complete implementation is done in ANSI C language.

In our computations we take as $(0, m, s + l)$-net in base b the Faure sequence [8]. The fast Faure generator implementation is due to Eric Thiémard [9] based on the idea presented in [10], which requires only $O(ms)$ time compared to $O(m^2s)$ proposed in [11,8,12]. For the Monte Carlo simulation, we use the Unix inbuilt random number generator function *drand*48. For sorting, we use the *quicksort* algorithm proposed by Hoare, [13].

We are interested in the steady state solution of (12) subject to a suitable initial condition in order to evaluate the polymer contribution to the stress tensor given by (5) and calculate the viscosity η and first normal stress difference coefficient Ψ given by (6).

3.1 Fixation of parameters

First we study the dependence of the steady state values of η and Ψ on the initial conditions. We consider the three different initial conditions outlined in subsection 2.2. We recall that the abbreviations SN, DD and EQ stand for standard normal, Dirac delta and equilibrium distribution respectively. From
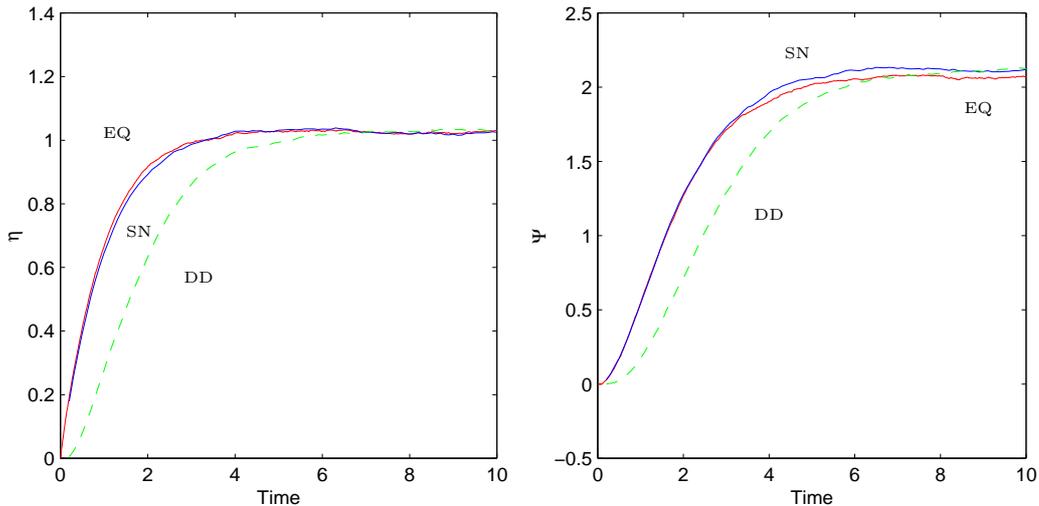
Fig. 4. Non-dimensional viscosity (left) and first normal stress difference coefficient (right) for different choices of initial condition for the case of dumbbell $s = 3$ ($n = 2$) with $z = 0.1$, $\beta = 1.0$ and $d = 0.5$.

figures 4 and 5, it is clear that, as expected, the convergence is not affected by the choice of initial conditions. But a careful examination shows that steady state is attained ahead of time in the case of choice (SN) and (EQ) compared to (DD). The only disadvantage of choice (EQ) is that, many particles should be produced before a required number is selected. In the case of dumbbells only 10% of the particles are accepted on an average. So we conclude that (SN) is the optimal choice both accuracy wise and computational cost wise.

11
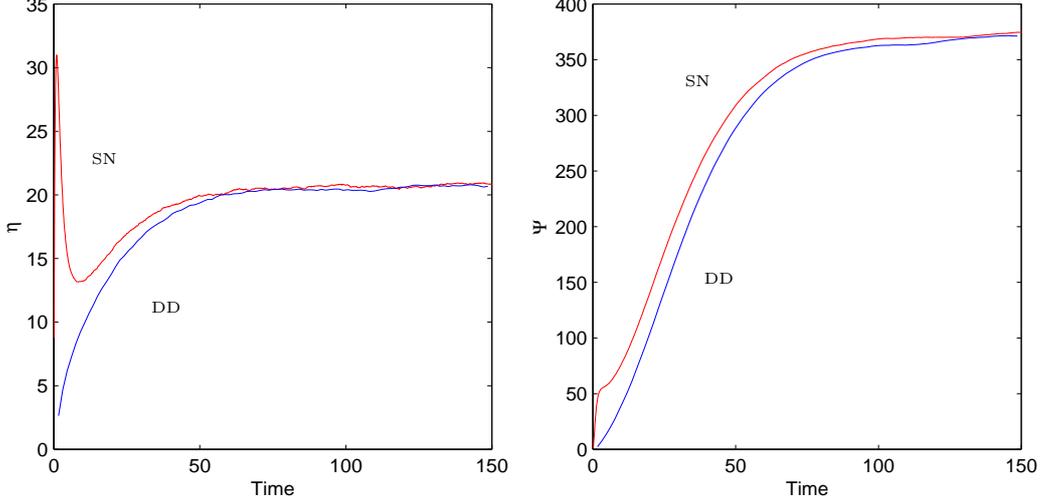
Fig. 5. Non-dimensional viscosity (left) and first normal stress difference coefficient (right) for choices (SN) and (DD) of initial condition for the case $s = 21$ ($n = 8$) with $z = 0.1$, $\beta = 1.0$ and $d = 0.5$.

Since we are interested in the steady state values of the viscosity and the first normal stress difference coefficient, we would now study the influence of the time step on the stationary value of QMC$(1, 1)$. We take as test case the chain with 5 beads ($s = 12$) and 13^4 (28561) particles. Three different time steps, namely 0.01, 0.1 and 1.0 are considered. From figure 6, it is evident that the difference in the values obtained using $\Delta t = 0.1$ and $\Delta t = 0.01$ is less than 1%. Also the case $\Delta t = 0.01$ takes ten times more time compared to $\Delta t = 0.1$. In view of these facts, we conclude that it is judicious to consider $\Delta t = 0.1$ for the simulations.
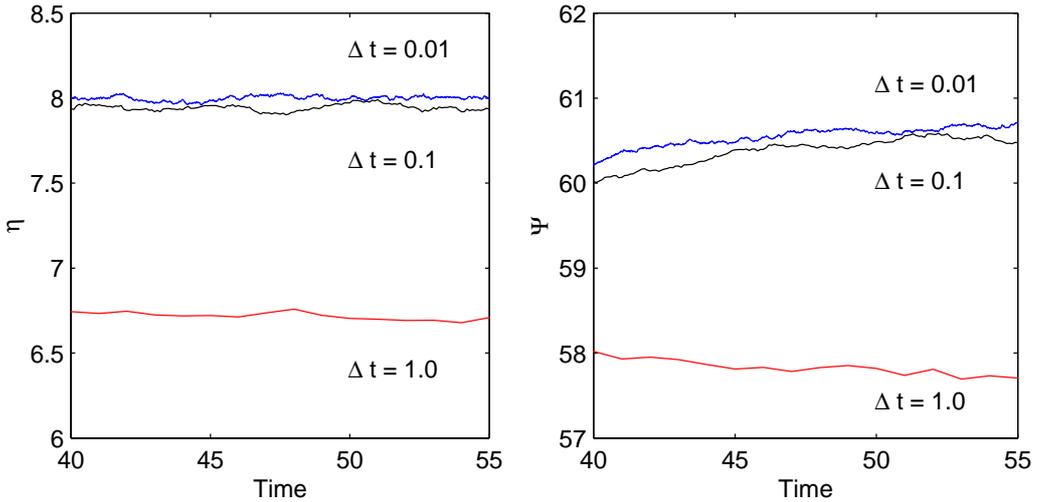
Fig. 6. Time accuracy of the scheme for calculating viscosity (left) and first normal stress difference coefficient (right) with $z = 0.1$, $\beta = 1.0$ and $d = 1.0$.

12

Having studied the time dependence with a fixed number of particles, we now study the dependence of the viscosity and first normal stress difference coefficient on the sample size. We consider the dumbbell case and march with a time step of 0.1 till time $T = 10.0$ with 5^5, 5^6 and 5^7 particles. It can be seen
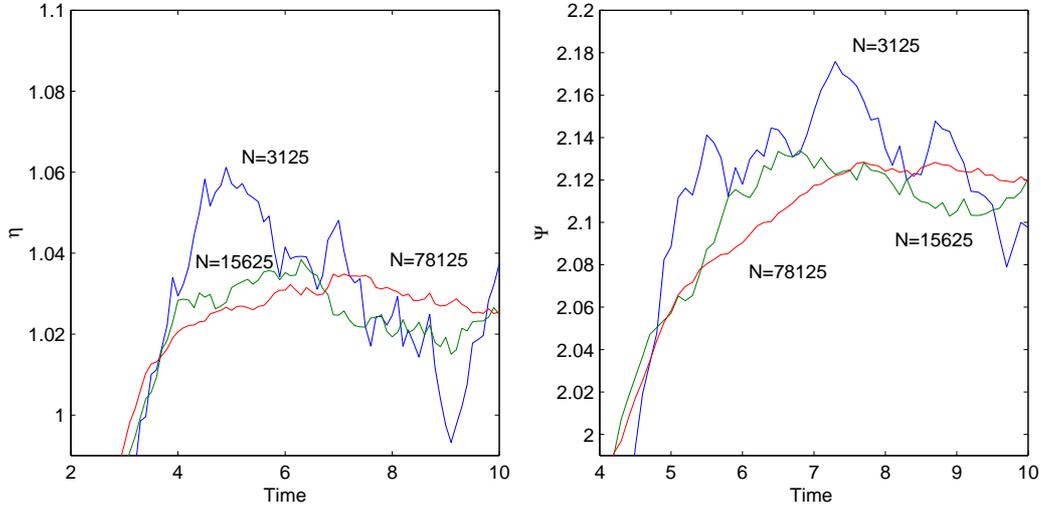
Fig. 7. Dependence of viscosity (left) and first normal stress difference coefficient on the sample size for the case $s = 3$ ($n = 2$) with $z = 0.1$, $\beta = 1.0$ and $d = 0.5$.

from figure 7 that as the particle number increases, the oscillations decline in magnitude and steady state is attained ahead in time.

3.2   Comparison of methods

Now that we have fixed the initial condition, we have all the ingredients to compare MC with QMC methods. Figure 8 (left) shows the viscosity obtained in a single run, using MC and the figure in the middle shows the same using $QMC(1, 1)$. It is clear that the $QMC(1, 1)$ trajectory has less oscillations compared to the MC trajectory, meaning that one has to average MC over several runs to obtain a similar result. The average values obtained from the three separate runs of MC is shown in figure 8 (right).

Note that we have restricted the η scale to show the behavior more clearly in the stationary part of the curve. A similar behavior is observed also in the case of first normal stress difference coefficient. For a single run with 13^4 (28561) particles, the Monte Carlo method took 3.7 seconds whereas $QMC(1, 1)$ took about 4 seconds on an average for a single time step. Since we need to average MC values over several trajectories, to get a result whose variance is comparable to that of $QMC(1, 1)$, we conclude that $QMC(1, 1)$ works better both in accuracy and computational time.

13
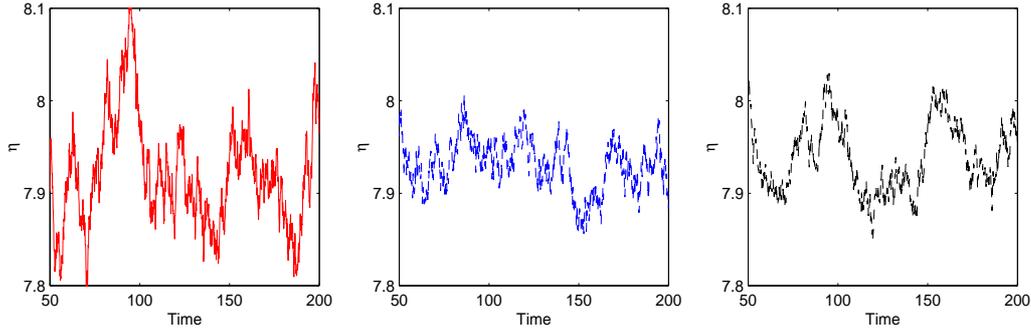
Fig. 8. Viscosity trajectory with a single run of MC (left), QMC(1, 1) (middle) and the average of three separate MC simulations (right) for the case $s = 12$ ($n = 5$) using 13^4 (28561) particles with $z = 0.1$, $\beta = 1.0$ and $d = 1.0$.

Table 1 summarizes the mean and standard deviation of the viscosity and first normal stress difference coefficient values obtained using the various QMC algorithms and its comparison with MC. The simulation is run till $T = 200$ with a step size of 0.1 and with 13^4 (28561) particles and the averaging is done only in the stationary part of the trajectory. Algorithm QMC(3, 2) is not directly comparable with the others for it works with a different particle number. We took 13^4 (28561) particles for algorithms MC, QMC(1, 0), QMC(0, 1) and QMC(1, 1) and 17^4 (83521) particles for QMC(3, 2) algorithm. It can be inferred once again (compare with [1]) from the results of QMC(1, 0) simulation, that sorting is absolutely essential for convergence.

Method	Mean(η)	Std. deviation(η)	Mean(Ψ)	Std. deviation(Ψ)
MC	7.920106	0.0581	60.190969	0.4191
QMC(1, 0)	14.151654	0.4493	107.421801	5.2260
QMC(0, 1)	8.078955	0.0276	61.716518	0.1626
QMC(1, 1)	7.930789	0.0274	60.371526	0.2182
QMC(3, 2)	7.932184	0.0195	60.348455	0.1414

Table 1
Mean and standard deviation comparison of the different QMC algorithms with MC for the case $s = 12$ ($n = 5$) with 13^4 (28561) particles. For the computation only the stationary part of the trajectory ($t \geq 50$, as shown in figure 8) is considered. Only for the QMC(3, 2) algorithm 17^4 (83521) particles are considered with $z = 0.1$, $\beta = 1.0$ and $d = 1.0$.

At this stage, we have the following situation: MC and QMC both work for high dimensions, the former can be implemented in a straightforward manner whereas the latter requires sorting the particle positions at each time step. The advantage with QMC is that the results have less noise compared to MC, but the extra processes may take up additional time. But what we observe (figure 9) is that this does not contribute significantly as transport dominates the total

14
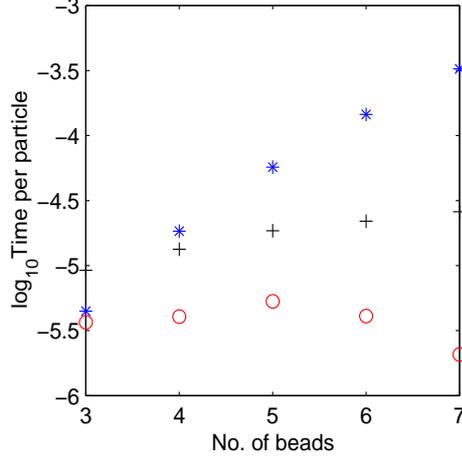
Fig. 9. Time taken for transport (stars), sorting (circles) and diffusion (+) per time step for 3, 4, 5 and 6 beads.

computational time. Though the time taken for diffusion is quite significant in the case of 3 beads, it is overtaken by transport in higher dimensions (4, 5, 6 and 7 beads). In fact the transport part seriously affects the computational time if we go to real high dimensions as depicted in figure 10.
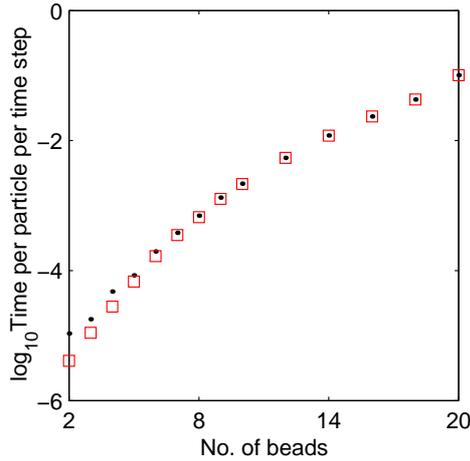
Fig. 10. Average time taken by a particle for a single run of QMC$(0, 1)$ (dots) and the corresponding MC algorithm (squares) for various dimensions. Note that the time for initial sampling is not taken into account.

3.3 Benchmark comparison

We finally present results for the benchmark case, i.e, the dumbbell case, to compare with the results presented in [14]. Various combinations of z, β and d are taken and the non-dimensional viscosity and first normal stress difference coefficient are calculated.

15

The method presented in [14] proceeds as follows. The values of η and Ψ are computed for various values of the time discretization Δt with an MC algorithm, and finally the results are extrapolated to zero time step to find the steady state values. In QMC$(1,1)$ algorithm, we took 5^8 (390625) particles and $\Delta t = 0.1$.

Parameters	η [14]	η QMC$(1,1)$	Absolute error
$z = 0.1$, $\beta = 1$, $d = 0.5$	1.03113	1.03132	1.89 x 10^{-4}
$z = 0.1$, $\beta = 4$, $d = 0.5$	1.00876	1.01128	2.52 x 10^{-3}
$z = 0.1$, $\beta = 1$, $d = 1.0$	1.01085	1.00960	1.25 x 10^{-3}
$z = 0.1$, $\beta = 4$, $d = 1.0$	1.00396	1.00305	9.09 x 10^{-4}

Table 2
Comparison of viscosity values obtained from QMC$(1,1)$ and the MC algorithm in [14] for different values of the parameters z, β and d

Parameters	Ψ [14]	Ψ QMC$(1,1)$	Absolute error
$z = 0.1$, $\beta = 1$, $d = 0.5$	2.06980	2.12424	5.44 x 10^{-2}
$z = 0.1$, $\beta = 4$, $d = 0.5$	2.01768	2.07676	5.90 x 10^{-2}
$z = 0.1$, $\beta = 1$, $d = 1.0$	2.03040	2.07980	4.94 x 10^{-2}
$z = 0.1$, $\beta = 4$, $d = 1.0$	2.00824	2.05938	5.11 x 10^{-2}

Table 3
Comparison of first normal stress difference coefficient values obtained from QMC$(1,1)$ and the MC algorithm in [14] for different values of the parameters z, β and d

Tables 2 and 3 are to be taken as reference only since the exact value is not known for the sake of comparison.

4 Conclusion

It can in general be observed from the numerical experiments, that QMC results show reduced variance in comparison to the corresponding MC ones. As a consequence, the MC results have to be averaged over several runs (or equivalently, considerably more particles have to be used) in order to obtain the same variance as the QMC result. Since the overhead in the QMC diffusion algorithm is masked by the advection step of the Fokker-Planck dynamics, both the methods have comparable speed and repetition of MC runs immediately leads to a corresponding factor in computational time. Thus, for fixed

accuracy, through certain QMC algorithms, we achieve improvement in computational time over the simple MC method.

Finally, we want to stress that the purpose of this paper is the comparison of the different QMC codes with the simplest MC algorithm. There are several variance reduction techniques associated with MC methods and the same may be carried over to QMC. Currently such investigations are being carried out.

References

[1] G. Venkiteswaran, M. Junk, QMC algorithms for diffusion equations in high dimensions, Mathematics and Computers in Simulation .

[2] R. Byron Bird, C. F. Curtiss, R. C. Armstrong, O. Hassager, Dynamics of Polymeric liquids, Vol. 1, Wiley Interscience, 1987.

[3] R. Byron Bird, C. F. Curtiss, R. C. Armstrong, O. Hassager, Dynamics of Polymeric liquids, Vol. 2, Wiley Interscience, 1987.

[4] H. C. Öttinger, Stochastic Processes in Polymeric Fluids, Springer, 1996.

[5] L. Devroye, Non-uniform random variate generation, Springer, New York, 1986.

[6] P. A. Raviart, An analysis of particle methods, in: F. Brezzi (Ed.), Numerical Methods in Fluid Dynamics, Springer-Verlag, Berlin, 1985, pp. 243–324.

[7] C. Lécot, F. E. Khettabi, Quasi-Monte Carlo simulation of diffusion, Journal of Complexity 15 (1999) 342–359.

[8] H. Faure, Discrépance de suites associées à un systéme de numération (en dimension s), Acta Arith. 42 (1982) 337–351.

[9] E. Thiémard, Economic generation of low-discrepancy sequences with a b-ary gray code, EPFL-DMA-ROSO, RO981201 (1998).

[10] S. Tezuka, Uniform random numbers: Theory and practice, Kluwer Academic Publishers, Boston, 1995.

[11] H. Niederreiter, Random number generation and quasi-Monte Carlo methods, Vol. 6, Society for Industrial and Applied Mathematics, 1992.

[12] I. M. Sobol, On the distribution of points in a cube and the approximate evaluation of integrals, USSR Comput. Math. Math. Phys .

[13] R. Sedgewick, Algorithms in C, Addison-Wesley, Reading, Massachusetts, 1990.

[14] J. Ravi Prakash, H. C. Öttinger, Viscometric functions for a dilute solution of polymers in a good solvent, Macromolecules 32 (6) (1998) 2028–2043.