

## 1 Introduction

In this short article, we discuss the program used in the exercises for the dissipative particle dynamics simulations.

## 2 Model and Dissipative Particle Dynamics

The system under study is a two-dimensional (2D) linear polymer chain in an explicit solvent. The polymer chain consists of  $N$  beads (monomers) connected together by harmonic bonds.

We study a system described by  $N_{\text{tot}}$  particles. The mass of particle  $i$  is denoted by  $m_i$ , and its position and velocity by  $\vec{r}_i$  and  $\vec{v}_i$ , respectively. Interparticle interactions are characterized by the pairwise conservative, dissipative, and random forces exerted on particle “ $i$ ” by particle “ $j$ ”, respectively, and are given by

$$\vec{F}_{ij}^C = \alpha_{ij} \omega^C(r_{ij}) \vec{e}_{ij}, \quad (1)$$

$$\vec{F}_{ij}^D = -\gamma \omega^D(r_{ij}) (\vec{v}_{ij} \cdot \vec{e}_{ij}) \vec{e}_{ij}, \quad (2)$$

$$\vec{F}_{ij}^R = \sigma \omega^R(r_{ij}) \xi_{ij} \vec{e}_{ij}. \quad (3)$$

Here  $\vec{r}_{ij} \equiv \vec{r}_i - \vec{r}_j$ ,  $r_{ij} \equiv |\vec{r}_{ij}|$ ,  $\vec{e}_{ij} \equiv \vec{r}_{ij}/r_{ij}$ , and  $\vec{v}_{ij} \equiv \vec{v}_i - \vec{v}_j$ . The  $\xi_{ij}$  are symmetric random variables with zero mean and unit variance, and are independent for different pairs of particles and different times.

The pairwise conservative force introduced in Eq. (1) is written in terms of a weight function  $\omega^C(r_{ij})$ , which is chosen to be of the form

$$\omega^C(r_{ij}) = \begin{cases} 1 - r_{ij}/r_c & , \text{ for } r_{ij} \leq r_c \\ 0 & , \text{ for } r_{ij} > r_c \end{cases} \quad (4)$$

where  $r_c$  is a cut-off distance. The strength of the conservative force  $\alpha_{ij}$  has three different values depending on the pair of particles (water–water, water–monomer, monomer–monomer).

Since we are dealing with a polymer chain, the conservative DPD–term  $\omega^C(r_{ij})$  is complemented by a harmonic potential between adjacent beads. Here we describe the polymer as a chain consisting of harmonic springs of natural length zero. Letting  $k$  be the spring constant, the energy of the chain is

$$U = \frac{1}{2} k \sum_{i=2}^N (r_{i,i-1})^2,$$

where  $r_{i,i-1} = |\vec{r}_i - \vec{r}_{i-1}|$ .

The weight functions  $\omega^D(r_{ij})$  and  $\omega^R(r_{ij})$  of the dissipative and random forces are coupled since the thermal heat generated by the random force must be balanced by dissipation. The precise relationship between these two forces is determined by the fluctuation-dissipation theorem, and can be expressed as follows:

$$\omega^D(r_{ij}) = [\omega^R(r_{ij})]^2 \quad (5)$$

and

$$\sigma^2 = 2\gamma k_B T^* . \quad (6)$$

Here  $T^*$  is the canonical temperature of the system. A common choice (which we follow in this work) is to set  $\omega^R(r_{ij}) = \omega^C(r_{ij})$ .

To evolve the system in time, one integrates Newton's equations of motion which are now given by the set of stochastic differential equations

$$d\vec{r}_i = \vec{v}_i dt , \quad (7)$$

$$d\vec{v}_i = \frac{1}{m_i} \left( \vec{F}_i^C dt + \vec{F}_i^D dt + \vec{F}_i^R \sqrt{dt} \right) . \quad (8)$$

Here  $\vec{F}_i^R = \sum_{j \neq i} \vec{F}_{ij}^R$  is the total random force acting on particle “ $i$ ”, and  $\vec{F}_i^C$  and  $\vec{F}_i^D$  are defined similarly. The velocity increment due to the random force in Eq. (8) is written in a form which can be given a precise meaning by identifying it as the infinitesimal increment of a Wiener process. In practice, finite time increments are used in the simulations, and the equations of motion [Eqs. (7) and (8)] have to be solved by some integration procedure. To this end, we use the so-called DPD–VV scheme. (For details, see the reference [J. Chem. Phys. **116**, 3967 (2002)]).

The length scale in the simulations is defined by  $r_c = 1$  and the time scale is given in units of  $r_c \sqrt{m/k_B T^*}$ . The energy scale is defined by setting the desired thermal energy to unity via  $k_B T^* = 1$ .

### 3 Program

The parameters (not to be changed) which fix the maximum values of the system are the following:

Variable	Description
Npmax	Maximum number of particles in a system
maxnab	Maximum number of particles in the Verlet table
maxt_vacf	Maximum number of points in the velocity autocorrelation function
Nclen	Maximum number of monomers in a polymer chain

Main variables describing the system are the following:

Variable	Description
dLx	System size in $x$ -direction
dLy	System size in $y$ -direction
rho0	Total particle number density of all particles in a system
len_chain	Number of monomers (beads) in a polymer chain
d_bond	Force constant for the harmonic spring between adjacent monomers in a chain
pm	Particle mass (equal for all particles in a system)
alpha(1,1)	Strength of the conservative force (water–water)
alpha(1,2)	Strength of the conservative force (water–monomer)
alpha(2,2)	Strength of the conservative force (monomer–monomer)
sigma	Strength of the random force
iseed	Seed for the pseudorandom number generator
dt	time step for integrating the equations of motion
isteps	Number of time steps to equilibrate the system
maxstp	Number of time steps carried out after equilibration
incstp	Describes how often we calculate the physical quantities i.e., quantities are calculated every “incstp”th time step
issstp	Describes how often we take snapshots of the polymer chain

## 4 Recommendations

Since the time is limited, we recommend studying rather simple and small systems that can be simulated in a few minutes of cpu-time. To this end, we strongly suggest the following values of parameters to start with:

dLx	=	16
dLy	=	16
rho0	=	3.0
len_chain	=	16
d_bond	=	10
pm	=	1
alpha(1,1)	=	10
alpha(1,2)	=	3
alpha(2,2)	=	10
sigma	=	3
dt	=	0.02
isteps	=	1000
maxstp	=	4000
incstp	=	10
issstp	=	1000

The cpu-time consumed by the program depends on the parameters (above) as well as the cpu used. A couple of test simulations will describe whether the system size and the time scale should be reduced to get decent results in a few minutes.

If you wish to scale the values of these variables, we suggest that you keep all the variables fixed except for the following ones:

dLx	=	16
dLy	=	16
rho0	=	3.0
len_chain	=	16
iseed	=	1
dt	=	0.02
isteps	=	1000
maxstp	=	4000
incstp	=	10
issstp	=	1000

The initial seed, in particular, should be changed for every simulation.

Currently, the program limits the length of the polymer (len\_chain) to be less than or equal to 20. If you wish to study larger polymers, you need to change the parameter

parameter( Nclen = 20 )

to match the maximum value of the length of the chain.

## 5 Physical quantities studied by the model

The DPD program provides one with data of the following physical observables:

File	Column	Description
data.dat	1	Number of time steps done
data.dat	2	Time in DPD units
data.dat	3	Momentum of the center-of-mass of the system
data.dat	4	Temperature of the system
rg2.dat	1	Number of time steps done
rg2.dat	2	Radius of gyration squared $R_g^2$ (instantaneous values)
rg2.dat	3	End-to-end distance of the chain (instantaneous values)
vcf.dat	1	Time in DPD units
vcf.dat	2	Center-of-mass velocity autocorrelation function of the polymer $\phi(t)$
vcf.dat	3	Integrated $\phi(t)$ , i.e., $\mathcal{D}(t) = \int_0^t dt' \phi(t')$

## 6 Graphics

To generate pictures or graphics of the data, you can either use local public-domain software or establish a connection to your home institute. Due to copyrights and other practical reasons, programs such as Matlab etc. are not available on local computers.

The easiest way to obtain reasonable pictures of the data is to use *gnuplot*. Then, to visualize the time evolution of the temperature of the system, type

```
> gnuplot
> plot "data.dat" using 2:4
```

The files “fort.???” contain snapshots of the polymer chain at different times (fort.101, fort.102, etc.), which can be visualized by typing

```
> gnuplot
> plot "fort.101" using 1:2 with linespoints
```

Further information on *gnuplot* is available at  
<http://www.duke.edu/~hpgavin/gnuplot.html>

## 7 Examples of results

As examples of results for the physical quantities discussed above, Figures 1 – 2 demonstrate some of the main features for the temperature and the radius of gyration. Figure 3 shows the scaling of  $R_g^2$  with an exponent  $\nu \approx 0.79$ . The deviation from the precise value of 0.75 is due to finite size effects.

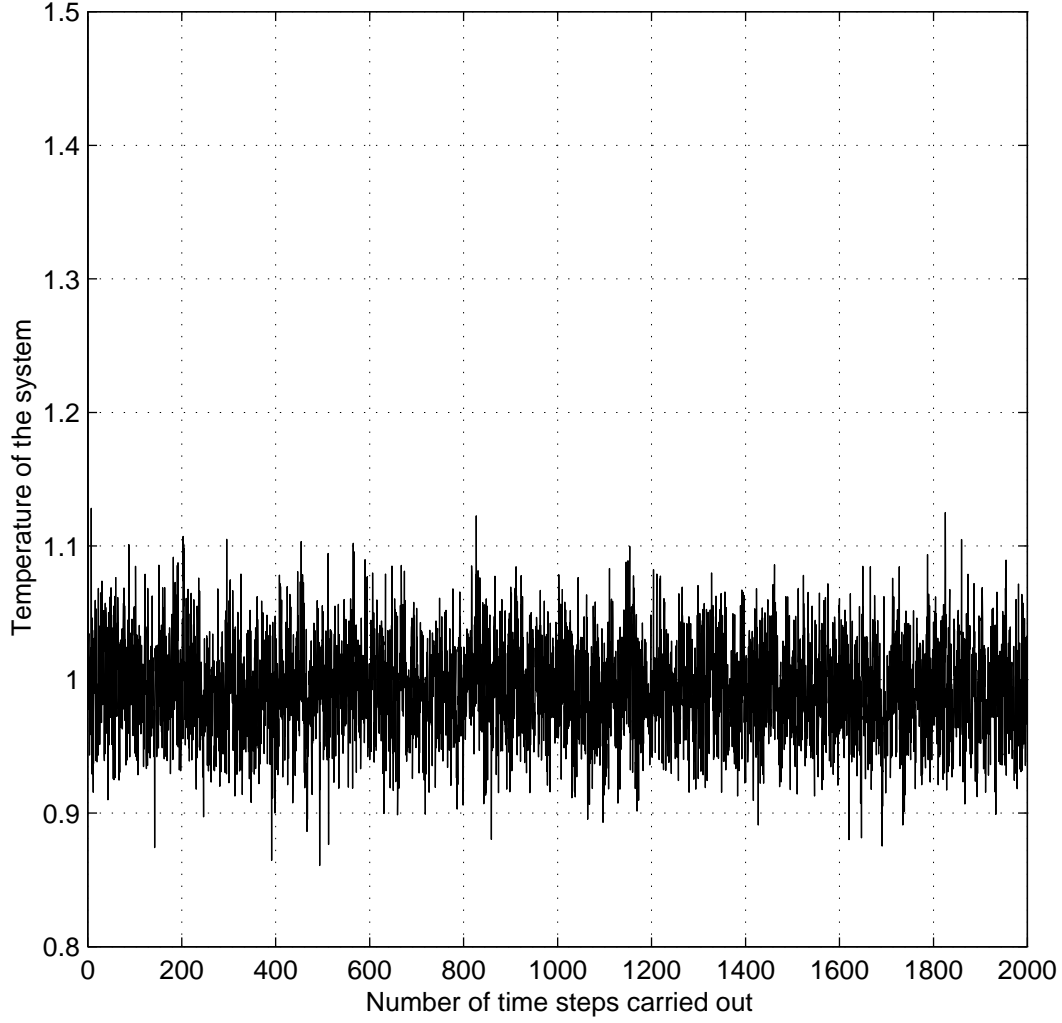


Figure 1: Temperature of the DPD system (example).

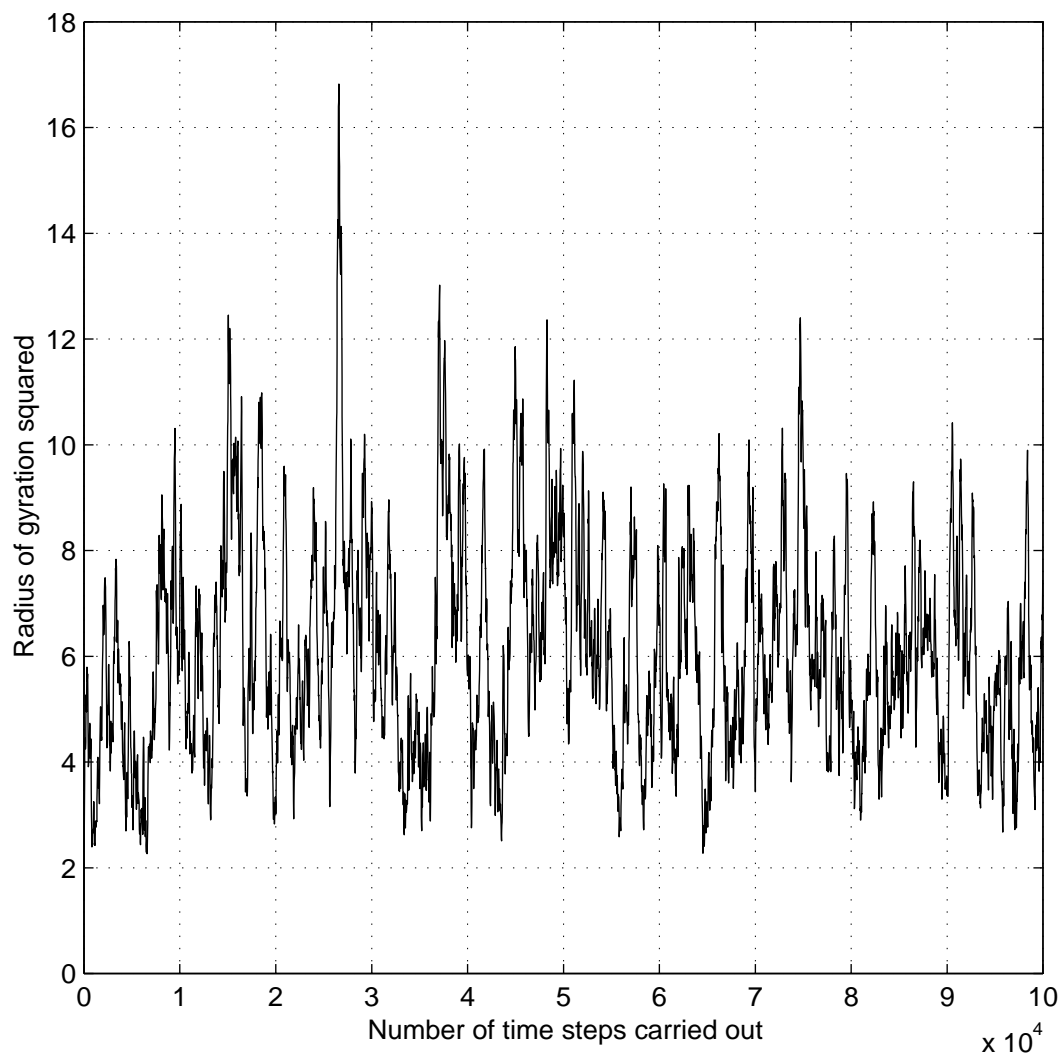


Figure 2: Radius of gyration squared for a polymer chain by DPD (example).

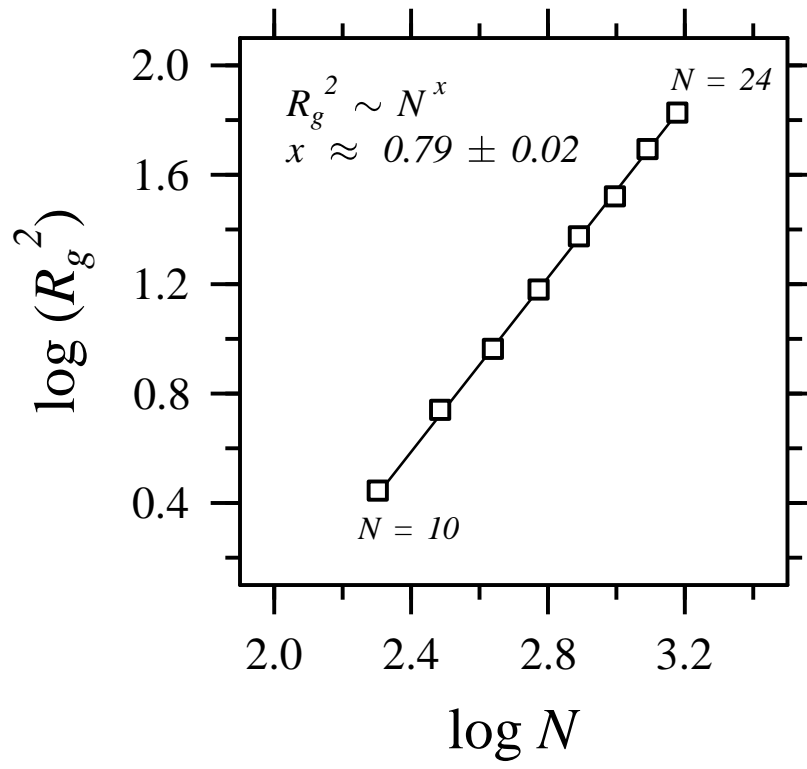


Figure 3: Example of scaling for the radius of gyration by DPD.