COMPUTATIONAL SOFTWARE

Polymer Chain Generation for Coarse-Grained Models Using Radical-Like Polymerization

Morgane Mahaud[1], Zengqiang Zhai[1], Michel Perez[1], Olivier Lame[1], Claudio Fusco[1], Laurent Chazeau[1], Ali Makke[2], Grégory Marque[3] and Julien Morthomas[1,*]

[1] *Université de Lyon, INSA Lyon, MATEIS, UMR CNRS 5510, Villeurbanne, France.*
[2] *EPF, Lasmis laboratory, UTT-Troyes, France.*
[3] *EDF, Les Renardières, Moret-sur-Loing, France.*

Abstract. This paper presents major improvements in the efficiency of the so-called Radical-Like Polymerization (RLP) algorithm proposed in "Polymer chain generation for coarse-grained models using radical-like polymerization" [J. Chem. Phys. **128** (2008)]. Three enhancements are detailed in this paper: (1) the capture radius of a radical is enlarged to increase the probability of finding a neighboring monomer; (2) between each growth step, equilibration is now performed with increasing the relaxation time depending on the actual chain size; (3) the RLP algorithm is now fully parallelized and proposed as a "fix" within the "Lammps" molecular dynamics simulation suite.

Program Summary

Program title: Fix_rlp

Nature of problem: Implementing special fix file in parallelized LAMMPS code in order to create polymer system with the so-called Radical-Like Polymerization (RLP) algorithm proposed in [1] for coarse-grained models.

Software licence: GPL 2.0 GNU's GPL

Corresponding author. Email address: `julien.morthomas@insa-lyon.fr` (J. Morthomas)

CiCP scientific software URL: http://michel.perez.net.free.fr/fix_rlp.zip.

Distribution format: .zip

Programming language(s): C++/MPI

Computer platform: Should run on any architecture providing a C++ compiler

Operating system: Linux or any other OS with C++ compiler and MPI library.

Compilers: C++ compiler

RAM: Depends on system size and how many CPUs are used

External routines/libraries: LAMMPS (http://lammps.sandia.gov/),
    FFTW (http://www.fftw.org/)

Running time: Seconds to weeks, depending on system size, speed of CPU and how many CPUs
    are used.

Restrictions: The code is based on a former version of LAMMPS.

Additional Comments: With the Fix_rlp LAMMPS program, you will find in the .zip detailed
    explanations in the "fix_rlp.txt" file, a "README.txt" file for the installation and a complete
    example of homopolymer generation with RLP algorithm in the form of a Lammps input
    file.

1 Introduction

Coarse-grained molecular dynamics is a well-adapted tool for studying the mechanical
behavior of polymers [1, 2]. It is widely used for studying various mechanical prop-
erties (elastic constants [3, 4], strain hardening [5], failure [6, 7], *etc.*) for various poly-
mer structures (linear chains [8, 9], cross-linked chains [10], branched polymers [11, 12],
co-polymers [13, 14], gels [15]). To create these structures, several methods have been
developed. Equilibration of fully independent self-avoiding chains [16] is a rather sim-
ple method but requires equilibration times that are not tractable with long chains. The
classical "push-off" method generates self equilibrated chains by Monte-Carlo scheme
(random-walk). However, long equilibration times are required to relax overlapping par-
ticles [8]. This method has been improved by introducing soft potentials [17, 18]. The
equilibration stage can itself be accelerated *via* the double-bringing method [17, 19–21]
or by reducing the weak interaction cut-off at the beginning of this stage [22]. They are
compatible with most creation methods since they act only on equilibration, yet they
don't accelerate polymer creation stage. Another method to accelerate the equilibration
via a hybrid MD/continuum model was proposed in two versions by Kroger [23] and
by Senda *et al.* [24], but is limited to rather short chains. Other kinds of methods pro-
pose simultaneous chain growth and equilibration. Subramanian recently submitted an
original method where polymer chains are progressively extended by adding additional
beads between two existing structural units [12, 25]. Fast algorithms, based on indepen-
dent generation of chains and application of "slow push-off" potential (see *e.g.* [11]) lead

to rather fast equilibration times, but used soft potentials are quite far from physics of polymerization.

The Radical-Like Polymerization (RLP) algorithm originally proposed by Gao [26] and later developed by Perez *et al.* [27] establishes a physical framework (based on the radical polymerization), where chain are partially relaxed simultaneously while polymerization is in progress. This method has been used successfully for generating different polymer morphologies (linear [18, 28–30], branched [31], co-polymers [13, 32, 33]).

However the RLP algorithm was written on a in-house serial program, which hinders both its diffusion and efficiency. The aim of this paper is therefore to propose a parallelized and optimized version of the RLP algorithm in the form of a Lammps "fix" [34], which can readily be used by a larger community. This paper (i) recalls the basis of the Coarse Grained Molecular Dynamics (CGMD) model, (ii) details the improved RLP algorithm, (iii) documents its use within the Lammps framework and, (iv) demonstrates the efficiency gain brought by this new algorithm.

2   Coarse grained molecular dynamics model

The RLP algorithm is well suited for molecular dynamics models, in which atoms can move freely in space. It is used here on a CGMD model widely found in the literature [8]. In this model, units interact with each other through two potentials: one representing weak interactions and the other covalent bonds. A truncated Lennard-Jones potential is used for the weak interactions:

$$U_{\text{LJ}}(r) = \begin{cases} 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^{6} \right], & r \leq r_c^{LJ}, \\ 0, & r > r_c^{LJ}, \end{cases} \tag{2.1}$$

where $\epsilon = 1$ and $\sigma = 1$ are the Lennard-Jones parameters and $r_c^{LJ} = 2.5\sigma$ the cut-off radius. Covalent bonds are represented at the beginning of the RLP by an harmonic potential (Eq. (2.2)) and at the end by a finite extensible nonlinear elastic potential (FENE) (Eq. (2.3))

$$U_{harm}(r) = K(r - r_0)^2, \tag{2.2}$$

where $K = 100\epsilon/\sigma^2$ is the harmonic bond stiffness and $r_0 = 1.12\epsilon$ the bond length, chosen to be the same as the Lennard-Jones one

$$U_{\text{FENE}}(r) = -0.5kR_0^2 \ln \left[1 - \left(\frac{r}{R_0} \right)^2 \right] + 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^{6} \right] + \epsilon, \quad r < R_0, \tag{2.3}$$

where k is the FENE bond stiffness and R_0 the maximum extent of the bond. The second term is a Lennard Jones potential with a cutoff r_c^{FENE} with $\epsilon = 1$ and $\sigma = 1$. Here, classical values are used to avoid high frequency modes and chain crossing. $k = 30\epsilon/\sigma^2$, $R_0 = 1.5\sigma$ and $r_c^{FENE} = 2^{1/6}\sigma$. Units are expressed as a function of ϵ and σ. For a parameter X, results

are expressed as X^* with $x = x^*\sigma$, $T = T^*\epsilon/k_b$, $P = P^*\epsilon/\sigma^3$ and $t = t^*\sqrt{m/\epsilon\sigma^2} = t^*\tau$. The time step is $dt = 5 \times 10^{-3}\tau$. Calculations are made with the software Lammps of Sandia labs [34].

3   Radical like polymerization algorithm

In this section, we briefly recall the basis of the original RLP algorithm published in Ref. [27]. **New features of the algorithm are emphasized and written in bold characters**.

- Liquid preparation: Polymerization takes place in a Lennard-Jones liquid constituted of N_{monom} liquid beads. For the sake of polymerization efficiency, the total number of liquid bead exceed by around 25% the total number of beads that are meant to polymerize. For example, to prepare a melt of $M = 500$ chains of $N = 200$ units, *i.e.* 100000 units, the liquid is composed of $N_{monom} = 128000$ monomers.

- Nucleation: M radicals are chosen randomly among the units to create M chains. Each unit has the same probability $p = M/N_{monom}$ to become a radical.

- Propagation: (i) Each radical selects a monomer bead to bond within a neighboring sphere of radius R^*_{neigh}. In the original RLP algorithm, the capture radius R^*_{neigh} was set to 1.5: the maximum bond length for the FENE potential. **In order to improve the efficiency of the algorithm (*i.e.* to increase the number of selectable neighbors), the capture radius is now set to 2.5. However, as this change is not compatible with FENE potential, the created bond is of harmonic type (see Eq. (2.2)). The harmonic bond is later changed to a FENE bond in the finalization step.** This neighbor becomes radical and the previous radical becomes part of the chain. Then, (ii) the system is allowed to relax during $t_{bG} = 1.5\tau$ within NPT ensemble (Nose-Hoover thermostat and pressostat) at $T^* = 2$ and $P^* = 0.5$. **Another improvement consisted in increasing the relaxation time between each growth step as chains grow, with a power law: $t_{bG} = N(t)^{1.4} dt$, where $N(t)$ is the actual chain length during polymerization.** This adaptive relaxation time account the non-linear evolution of the reptation time with chain length [35].

- Termination: Stages (i) and (ii) of the Propagation stage are repeated until all chains reach their desired length N.

- Finalization: Remaining monomers are removed. **Then, in order to transform all harmonic bonds into FENE bonds, the system is equilibrated $5 \times 10^2 \tau$ within NPT ensemble (Nose-Hoover thermostat and pressostat) at $T^* = 2$ and $P^* = 0.5$, quenched at 0K and the positions of all atoms are set to minimize the energy (conjugate gradient).** The system is finally equilibrated during $t_{eq}\tau$ within NPT ensemble (Nose-Hoover thermostat and pressostat) at $T^* = 2$ and $P^* = 0.5$. The equilibration duration depends on the chain length, as showed in Table 1.

4   Software installation and numerical examples

4.1   Integration of the Fix_rlp program in LAMMPS

In order to use this patch code, you need to copy the two files "fix_rlp.cpp" and "fix_rlp.h" to the \src folder of LAMMPS and then compile them as you compile LAMMPS (see detailed instructions in the "README.txt" file). You will get an executable file (here lammps.exe) and you can run the job LAMMPS script (here test.txt), like this:

```
mpirun −np 8 ./lammps.exe < test.txt
```

The "fix rlp" command is based on the "fix bond/create" from Lammps. It presents the following form (see detailed in the "fix_rlp.txt" file):

```
fix ID groupID rlp Nevery Iontype radtype intype Rneigh bondtype lmax keyword values
```

- ID = function instance identifier

- groupID = identifier of the group on which the function apply

- Nevery = attempt bond creation every this many steps

- Iontype = type of the units which do not belong to a chain

- radtype = type of the radical units and chains ends at the end of run

- intype = type of the units which belonged to a chain and are not radicals

- Rneigh = two atoms separated by less than R_{neigh} can bond (distance units)

- bondtype = type of created bonds

- lmax = maximal size of a molecule, radicals included

The optional keywords and their values are:

- maxbound bmax. bmax is the maximal number of bonds of type bondtype an atom can have

- prob proba seed. Create a bond with probability proba using seed for random number generation.

Below, an example of homopolymer generation with RLP algorithm, a LAMMPS input script file, starting with a initial structure file "Polymer2000.dat" that you can find in the .zip file:

```
#### STEP 1 : Creation of monomer bath
#————————————————— Simulation box
dimension 3
units lj
boundary p p p
atom_style bond

#loading of the atom box created aside
read_data Polymer2000.dat
mass * 1.0

pair_style lj/cut 2.5
pair_coeff * * 1.0 1.0 2.5

# ————————————————— Output
thermo_style custom step temp etotal press
thermo 100

#————————————————— Melting
velocity all create 2 123456789 dist gaussian
fix fix_melt all npt temp 2.0 2.0 100 iso 0.5 0.5 100
run 100000

print "monomer bath created"

#### STEP 2 : Creation of radicals
# ————————————————— Radicals activation
label loop_radical
variable M equal 50
variable k loop 1 ${M}
variable krad equal ${k}*35
set atom ${krad} type 2
set atom ${krad} mol ${k}
next k
jump test.txt loop_radical

print "radicals created"


#### STEP 3 : Radical−Like Polymerization
#————————————————— Variables setting
#number of timestep between two growing steps
```

```
variable runstep equal 300
#maximal number of growing steps
variable numberofcycles equal 50
#maximal lenght of a polymer
variable N equal 10


#———————————————————— Bond setting
special_bonds fene extra 5
bond_style harmonic
bond_coeff 1 100 1.12


print "RLP environnement created"


#———————————————————— RLP
fix fixrlp all rlp ${runstep} 1 2 3 2 1 ${N} prob 0.5 12345
variable runrlp equal ${runstep}*${numberofcycles}
run ${runrlp}


print "RLP finished"


#### STEP 4 : delete loners
group delete_monomer type 1
delete_atoms   group delete_monomer
unfix fixrlp
run 100000


print "loners erased"


#### STEP 5 : Harmonic to Fene
min_style cg
minimize  1e−4 1e−4 10000 10000
bond_style fene
bond_coeff 1 30 1.5 1 1
run 5000000
```

4.2   Characterization of generated melts

The Flory characteristic ratio $C_n(n)$, which is an important measure of the conformation and rigidity of individual chains, is defined as

$$C_n(n) = \frac{\langle r^2(n) \rangle}{n \langle l^2 \rangle}, \tag{4.1}$$

where l is the bond length. $\langle r^2(n) \rangle$ is the mean square internal distance, defined as

$$\langle r^2(n) \rangle = \sum_{i=1}^{M} \frac{\sum_{j=1}^{N^i-n} (r_j^i - r_{j+n}^i)^2 / (N^i - n)}{M}, \tag{4.2}$$

where r_j^i is the position of jth unit of the ith chain and N^i the ith chain length, and $\langle l^2 \rangle = \langle r^2(1) \rangle$.

In order to confirm that the melt is properly equilibrated, the characteristic ratio $C_n(n)$ is measured and compared to a target function which is an average from conformations of brute-force equilibrated melt from $5 \times 10^5 \tau$ to $10^6 \tau$ for the system of 200 chains of 500 units. $C_n(n)$ of four samples of different chain lengths are represented on Fig. 1 after stabilization and compared to the target function. We observe that all systems are well equilibrated with this procedure. We can notice a sizable deformation of the characteristic ratio of the chain of $N=500$ for $n>100$, which is due to poor statistic for large n. Note that the final equilibration duration is reduced compared to the reference of Auhl *et al.*, *e.g.* $t_{eq} = 2.6 \times 10^6 \tau$ for $N=350$ versus $t_{eq} = 10^5 \tau$ for $N=500$ in our case, see Table 1.

The topology of the melt was also characterized by calculating the average contour length, $\langle L_{pp} \rangle$, the entanglement length, N_e, and the average number of interior kinks, $\langle Z \rangle$ using the Z1 code [36]. The entanglement length N_e was estimated by the classical S-coil formula, $N_e = (N-1) \langle r^2(N) \rangle / \langle L_{pp} \rangle^2$. In Table 2, the values of $\langle L_{pp} \rangle$ and $\sqrt{\langle L_{pp}^2 \rangle}$, $\langle Z \rangle$ and N_e are reported for $N=500$ after $t_{eq} = 10^5 \tau$ and compared to the brute force
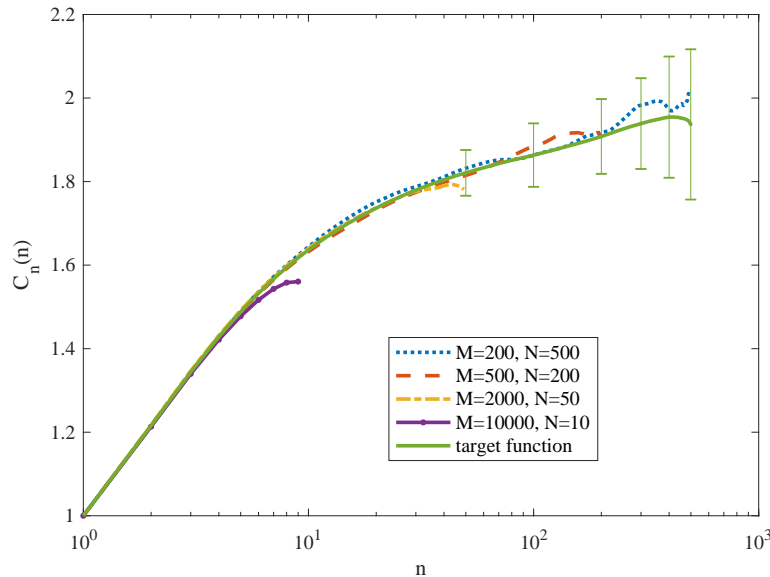

Figure 1: The characteristic ratio $C_n(n)$ for different chain lengths obtained with the updated RLP method and compared to the brute-force equilibrated melt (target function). The error bars of the target function have been added for $n=50$, 100, 200, 300, 400 and 499.

Table 1: Final equilibration stage duration for different chain lengths.

Chain length N	10	50	200	500
t_{eq} (units of τ)	$< 2 \times 10^3$	2.5×10^3	2.5×10^4	10^5

Table 2: Chain topology: average contour length, $\langle L_{pp} \rangle$ and $\sqrt{\langle L_{pp}^2 \rangle}$, entanglement length, N_e, and average number of interior kinks, $\langle Z \rangle$ for $N = 500$ after $t_{eq} = 10^5 \tau$ and for the brute force equilibrated melt.

	$\langle L_{pp} \rangle$	$\sqrt{\langle L_{pp}^2 \rangle}$	N_e	$\langle Z \rangle$
brute-force equilibrated melt	69.45	71.00	93.99	8.38
$t_{eq} = 10^5 \tau$ for $N = 500$	68.65	70.13	95.33	8.36

equilibrated melt. The agreement is satisfactory for all investigated variables, proving the correct equilibration of the $N = 500$ system for $t_{eq} = 10^5 \tau$.

4.3 Effect of capture radius

To obtain an estimation of the improvement due to the capture radius set in the propagation stage, capture radii R_{neigh}^* of 1.5, 2 and 2.5 have been tested. $R_{neigh}^* = 1.5$ corresponds to the cutoff value of the FENE bond, which is the value used in the original RLP paper [27]. $R_{neigh}^* = 2.5$ corresponds to the cutoff radius of the Lennard-Jones pair potential. A melt is prepared as described above. The chain length N is set to 200 and the number of chains M to 500. All simulations have been made on the same 32 processors. They include the nucleation, propagation and termination stages. Simulation times are presented in Table 3.

Table 3: Effect of capture radius on simulation duration of RLP process (nucleation-propagation-termination). Simulations are performed on 32 processors.

R_{neigh}^*	1.5	2	2.5
$t(s)$	477	386	359

As expected, increasing R_{neigh}^* leads to a simulation time drop, due to an improvement in the propagation stage efficiency. It can be noticed that this improvement is more important between 1.5 and 2 than between 2 and 2.5.

The RLP completion ratio (total polymerized units over NM) is represented as a function of time on Fig. 2. In a first stage, polymerization ratio is proportional with time (constant rate polymerization reaction). In a second stage, polymerization ratio reaches a plateau, the last few units to polymerize are the longest to capture. Whereas capture radius has a limited effect during the first stage, is has a strong impact on the second stage, facilitating the capture of new monomers when they start to be "rare" in the melt. Note
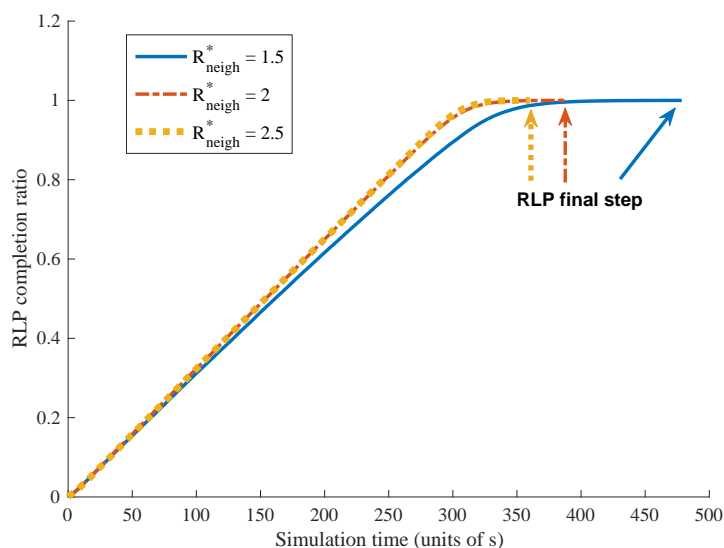
Figure 2: RLP process completion ratio as a function of capture radius R^*_{neigh} = 1.5, 2 and 2.5. A larger capture radius decrease the total generation time.

that going to higher values than 2.5 would imply an increase of the Lennard-Jones cut-off radius, used by Lammps to calculate the neighbor list, which would lead to unnecessary additional computational time.

4.4   Effect of increasing relaxation time during polymerization

In order to optimize global simulation time for complete polymerization and equilibration, a non-constant, and increasing, relaxation time between each growth step (t_{bG}) has been introduced. In order to test this improvement, two simulations are compared for a system of $M = 500$ chains of $N = 200$ monomers: (1) a polymerization with constant $t_{bG} = 1.5\tau$ relaxation time between each growth step (corresponding a total of $4.5 \times 10^2 \tau$) followed by an equilibration of $10^4 \tau$ (like the original RLP algorithm); (2) a polymerization with increasing relaxation time between each growth step $t_{bG} = N(t)^{1.4} dt$, where $N(t)$ is the actual size of the polymer chain during polymerization (corresponding a total of $1.5 \times 10^3 \tau$ followed by an equilibration of $8.5 \times 10^3 \tau$).

In Fig. 3, the characteristic ratio $C_n(n)$ of the two simulations are compared with the target function. It can be clearly seen that increasing the relaxation time t_{bG} during polymerization leads to a better equilibration compared to a constant one, *for the same total simulation time* including polymerization and equilibration.

4.5   Parallelisation efficiency

The original RLP algorithm was written as a in-house serial MD code. In order to improve (i) its ease of use and (ii) its computational efficiency, it has been adapted as a "fix" to the
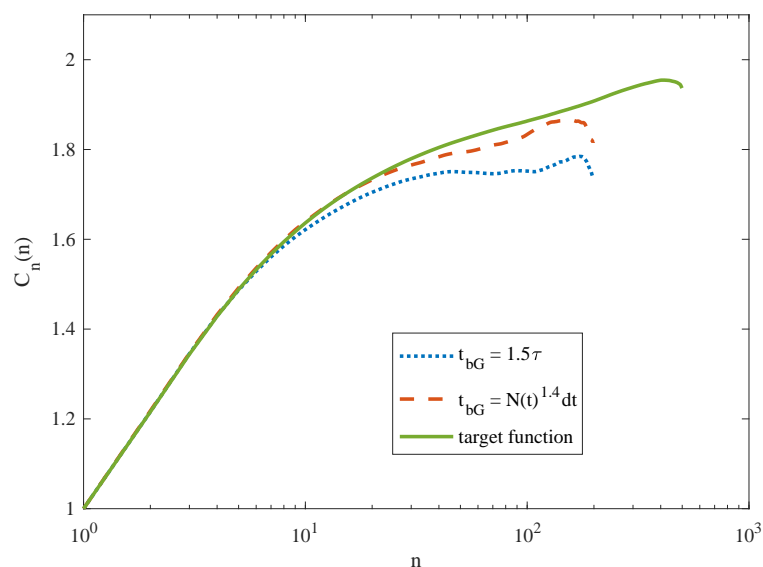
Figure 3: Effect of increasing relaxation time during polymerization: comparison between polymerization with constant ($t_{bG}=1.5\tau$) and increasing relaxation time ($t_{bG}=N(t)^{1.4}dt$ where $dt=5\times10^{-3}\tau$) during polymerization. The upper curve is the brute-force equilibrated melt (target function). The increasing relaxation time clearly leads to a better equilibration.

Lammps MD suite. One of the best feature of Lammps is its parallelization efficiency, which is almost linear. The parallelization efficiency of the RLP process implemented as a Lammps fix is tested on a sample of 500 chains of 200 units on 1, 2, 4, 8, 16, 32, 64 and 124 processors. A liquid containing $N_{monom}=128000$ monomers is created and the nucleation, propagation and termination phases are simulated with $R^*_{neigh}=2.5$. Results are shown Fig. 4. They are compared to the ideal time improvement where the simulation time is divided by a factor 2 when the number of processors is doubled. We can notice the simulation time is close to the optimal time up to 32 processors, proving thus that the RLP algorithm is efficiently parallelized.

5 Conclusion

The radical-like polymerization algorithm described in the original article of Perez *et al.* [27] has been extensively improved. Propagation step efficiency has been improved by enlarging the capture radius of radicals up to $R^*_{neigh}=2.5$. RLP algorithm is now available as a Lammps "fix" that can be readily used by the whole Lammps community. The parallelization efficiency of RLP algorithm is almost linear, leaving the parallelization capability of Lammps intact. In this paper, only linear homopolymers are designed. Nonetheless the RLP algorithm can be used for poly-dispersed chains, branched polymers, or even block co-polymers.
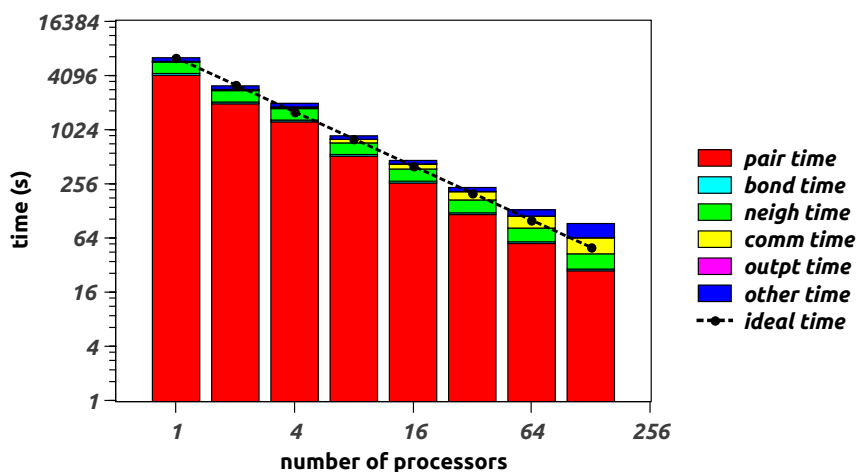
Figure 4: Benchmark of initialization and propagation stages. Simulation time as a function of the processors number. The dotted line represent an ideally parallelized algorithm.

Acknowledgments

References

[1] J. Padding, W. Briels, Systematic coarse-graining of the dynamics of entangled polymer melts: the road from chemistry to rheology, Journal of Physics: Condensed Matter 23 (23) (2011) 233101.

[2] K. Binder, Monte Carlo and Molecular Dynamics Simulations Polymer, Oxford University Press, Inc., 1995.
URL http://dl.acm.org/citation.cfm?id=526532

[3] M. Vladkov, J.-L. Barrat, Linear and nonlinear viscoelasticity of a model unentangled polymer melt: molecular dynamics and rouse modes analysis, Macromolecular theory and simulations 15 (3) (2006) 252–262.

[4] M. Kroger, W. Loose, S. Hess, Rheology and structural changes of polymer melts via nonequilibrium molecular dynamics, Journal of Rheology (1978-present) 37 (6) (1993) 1057–1079.
URL http://scitation.aip.org/content/sor/journal/jor2/37/6/10.1122/1.550409

[5] R. S. Hoy, M. O. Robbins, Strain hardening of polymer glasses: Effect of entanglement density, temperature, and rate, Journal of Polymer Science Part B: Polymer Physics 44 (24) (2006) 3487–3500.

[6] S. W. Sides, G. S. Grest, M. J. Stevens, Large-scale simulation of adhesion dynamics for end-grafted polymers, Macromolecules 35 (2) (2002) 566–573.

[7] J. Rottler, M. O. Robbins, Growth, microstructure, and failure of crazes in glassy polymers, Physical Review E 68 (1) (2003) 011801.

[8] K. Kremer, G. S. Grest, Dynamics of entangled linear polymer melts: A molecular-dynamics simulation, The Journal of Chemical Physics 92 (8) (1990) 5057–5086.

[9] K. Farah, A. C. Fogarty, M. C. Böhm, F. Müller-Plathe, Temperature dependence of coarse-grained potentials for liquid hexane, Physical Chemistry Chemical Physics 13 (7) (2011) 2894–2902.

[10] Y. R. Sliozberg, T. L. Chantawansri, Computational study of imperfect networks using a coarse-grained model, The Journal of chemical physics 139 (19) (2013) 194904.

[11] Y. R. Sliozberg, J. W. Andzelm, Fast protocol for equilibration of entangled and branched polymer chains, Chemical Physics Letters 523 (2012) 139–143.

[12] G. Subramanian, An Iterative Method for Producing Equilibrated Symmetric Three-Arm Star Polymer Melts in Molecular Dynamics, Macromolecular Theory and Simulations 20 (1) (2011) 46–53.
URL `http://onlinelibrary.wiley.com.sci-hub.io/doi/10.1002/mats.201000062/full`

[13] A. Makke, O. Lame, M. Perez, J.-L. Barrat, Influence of tie and loop molecules on the mechanical properties of lamellar block copolymers, Macromolecules 45 (20) (2012) 8445–8452.
URL `http://pubs.acs.org/doi/abs/10.1021/ma301286y`

[14] J. R. Brown, Y. Seo, T. A. D. Maula, L. M. Hall, Fluids density functional theory and initializing molecular dynamics simulations of block copolymers, The Journal of chemical physics 144 (12) (2016) 124904.

[15] Y. R. Sliozberg, T. L. Chantawansri, J. L. Lenhart, J. W. Andzelm, Structural and mechanical properties of advanced polymer gels with rigid side-chains using coarse-grained molecular dynamics, Polymer 55 (20) (2014) 5266–5275.
URL `http://www.sciencedirect.com/science/article/pii/S0032386114007666`

[16] T. Vettorel, K. Kremer, Development of entanglements in a fully disentangled polymer melt, Macromolecular Theory and Simulations 19 (1) (2010) 44–56.
URL `http://onlinelibrary.wiley.com/doi/10.1002/mats.200900065/full`

[17] R. Auhl, R. Everaers, G. S. Grest, K. Kremer, S. J. Plimpton, Equilibration of long chain polymer melts in computer simulations, The Journal of chemical physics 119 (24) (2003) 12718–12728.

[18] L. A. Moreira, G. Zhang, F. Müller, T. Stuehn, K. Kremer, Direct equilibration and characterization of polymer melts for computer simulations, Macromolecular Theory and Simulations 24 (5) (2015) 419–431.
URL `http://onlinelibrary.wiley.com/doi/10.1002/mats.201500013/full`

[19] N. C. Karayiannis, V. G. Mavrantzas, D. N. Theodorou, A novel Monte Carlo scheme for the rapid equilibration of atomistic model polymer systems of precisely defined molecular architecture, Physical review letters 88 (10) (2002) 105503.
URL `http://journals.aps.org/prl/abstract/10.1103/PhysRevLett.88.105503`

[20] N. C. Karayiannis, A. E. Giannousaki, V. G. Mavrantzas, An advanced Monte Carlo method for the equilibration of model long-chain branched polymers with a well-defined molecular architecture: Detailed atomistic simulation of an H-shaped polyethylene melt, The Journal of chemical physics 118 (6) (2003) 2451–2454.
URL `http://scitation.aip.org/content/aip/journal/jcp/118/6/10.1063/1.1543580`

[21] L. D. Peristeras, I. G. Economou, D. N. Theodorou, Structure and volumetric properties of linear and triarm star polyethylenes from atomistic Monte Carlo simulation using new internal rearrangement moves, Macromolecules 38 (2) (2005) 386–397.
URL `http://pubs.acs.org/doi/abs/10.1021/ma048364p`

[22] Y. R. Sliozberg, M. Kröger, T. L. Chantawansri, Fast equilibration protocol for million atom

systems of highly entangled linear polyethylene chains, The Journal of Chemical Physics 144 (15) (2016) 154901.

[23] M. Kroger, Efficient hybrid algorithm for the dynamic creation of wormlike chains in solutions, brushes, melts and glasses, Computer physics communications 118 (2) (1999) 278–298.
URL `http://www.sciencedirect.com/science/article/pii/S001046559800160X`

[24] Y. Senda, M. Fujio, S. Shimamura, J. Blomqvist, R. M. Nieminen, Fast convergence to equilibrium for long-chain polymer melts using a MD/continuum hybrid method, The Journal of chemical physics 137 (15) (2012) 154115.
URL `http://scitation.aip.org/content/aip/journal/jcp/137/15/10.1063/1.4759036`

[25] G. Subramanian, A topology preserving method for generating equilibrated polymer melts in computer simulations, The Journal of chemical physics 133 (16) (2010) 164902.
URL `http://scitation.aip.org/content/aip/journal/jcp/133/16/10.1063/1.3493329`

[26] J. Gao, An efficient method of generating dense polymer model melts by computer simulation, The Journal of chemical physics 102 (2) (1995) 1074–1077.
URL `http://scitation.aip.org/content/aip/journal/jcp/102/2/10.1063/1.469455`

[27] M. Perez, O. Lame, F. Leonforte, J.-L. Barrat, Polymer chain generation for coarse-grained models using radical-like polymerization, The Journal of chemical physics 128 (23) (2008) 234904.

[28] A. Makke, M. Perez, O. Lame, J.-L. Barrat, Mechanical testing of glassy and rubbery polymers in numerical simulations: Role of boundary conditions in tensile stress experiments, The Journal of chemical physics 131 (1) (2009) 014904.

[29] A. Makke, M. Perez, J. Rottler, O. Lame, J.-L. Barrat, Predictors of Cavitation in Glassy Polymers under Tensile Strain: A Coarse-Grained Molecular Dynamics Investigation, Macromolecular Theory and Simulations 20 (9) (2011) 826–836.

[30] K. Farah, H. A. Karimi-Varzaneh, F. Müller-Plathe, M. C. Böhm, Reactive molecular dynamics with material-specific coarse-grained potentials: growth of polystyrene chains from styrene monomers, The Journal of Physical Chemistry B 114 (43) (2010) 13656–13666.

[31] Z. Zhang, L. Wang, Z. Wang, X. He, Y. Chen, F. Müller-Plathe, M. C. Böhm, A coarse-grained molecular dynamics–reactive monte carlo approach to simulate hyperbranched polycondensation, RSC Advances 4 (100) (2014) 56625–56636.

[32] A. Makke, M. Perez, O. Lame, J.-L. Barrat, Nanoscale buckling deformation in layered copolymer materials, Proceedings of the National Academy of Sciences 109 (3) (2012) 680–685.
URL `http://www.pnas.org/content/109/3/680.short`

[33] F. Léonforte, Evolution of entanglements during the response to a uniaxial deformation of lamellar triblock copolymers and polymer glasses, Physical Review E 82 (4) (2010) 041802.

[34] S. Plimpton, Fast Parallel Algorithms for Short-Range Molecular Dynamics, Journal of Computational Physics 117 (1) (1995) 1 – 19. doi:http://dx.doi.org/10.1006/jcph.1995.1039.
URL `http://www.sciencedirect.com/science/article/pii/S002199918571039X`

[35] P.-G. De Gennes, Scaling concepts in polymer physics, Cornell university press, 1979.

[36] M. Kroger, Shortest multiple disconnected path for the analysis of entanglement in two- and three-dimensional polymeric systems, Computer Physics Communications, 168 (2005) 209.