Implementation and performance analysis of bridging Monte Carlo moves for off-lattice single chain polymers in globular states

Daniel Reith*,a, Peter Virnaua

^aInstitut für Physik, Staudingerweg 7, 55099 Mainz, Germany

Abstract

Bridging algorithms are global Monte Carlo moves which allow for an efficient sampling of single polymer chains. In this manuscript we discuss the adaptation of three bridging algorithms from lattice to continuum models, and give details on the corrections to the acceptance rules which are required to fulfill detailed balance. For the first time we are able to compare the efficiency of the moves by analyzing the occurrence of knots in globular states. For a flexible homopolymer chain of length N = 1000, independent configurations can be generated up to two orders of magnitude faster than with slithering snake moves.

Key words: off-lattice Monte Carlo simulations, globular polymers, knots

1. Introduction

The estimation of thermodynamic quantities for globular phases of single chain polymers remains a challenging task [1, 2, 3] because an effective sampling is hindered by self-entanglements, knots [4, 5, 6, 7, 8] and high density. Globular polymers display an interesting phase behavior [9, 10] and may serve as model systems to study aspects of biological macromolecules [11, 12, 13, 14]. Typically, approaches like the slithering snake algorithm [15, 16, 17, 18] or various types of chain growth algorithms like PERM [19] are applied to sample such configurations.

A new class of Monte Carlo moves which takes advantage of destroying and reconnecting bonds was first suggested independently by Mansfield [20] and Olaj and Lantschbauer [21] in 1982. Even though both papers focus on sampling (polydisperse) lattice melts, Ref. [20] also includes the so called backbite move which preserves monodispersity and can be applied to single chains as well. Two additional bridging moves for single polymer chains on the lattice were subsequently introduced by Deutsch in 1997 [22] and have since been combined with schemes to sample free energy landscapes [13, 9] like Wang-Landau sampling [23]. Nowadays, a rather evolved

^{*}Corresponding author

Email address: reithd@uni-mainz.de (Daniel Reith)

set of bridging methods exists for off-lattice atomistic polymer melts [24, 25, 26, 27, 28, 29], and one particular move was also ported to single chain simulations [30]. Apart from the obvious application of sampling static melt quantities, these moves are also applied to generate well-equilibrated starting configurations for subsequent Molecular Dynamics runs [31, 32].

Our paper focuses on the efficient generation of globular off-lattice polymers. To this extent we adapt the three single chain bridging moves of Mansfield and Deutsch [22] to continuum models. Technical aspects of our implementation which further improve efficiency are also discussed. In the second part we test the performance of the moves and compare it to a standard implementation of the slithering snake algorithm. Distinguishing independent configurations in globular states is a formidable task on its own as commonly used observables are typically unable to yield information about the topology of the chain. To this end, we determine whether or not a globular state is knotted and compare the times which are required to generate "topologically independent" configurations.

2. Description of the bridging moves and detailed balance

2.1. Backbite move

On a lattice, the number of neighbors of a particular monomer is always welldefined. In the continuum, we call two monomers neighbors if the distance *d* between them is within a certain range $d_{min} < d < d_{max}$. The basic idea of the backbite move is illustrated in figure 1.

First, we select an end monomer with equal probability. Neighbors of this monomer are identified and counted (n_{old_end}) , and one of these neighbors is chosen at random. (Fig. 1b). A new bond is created between the neighbor and the old end monomer. At the same time, the bond between the chosen neighbor and its successor (along the direction from the old end to the neighbor) is severed, turning the neighbor into the new terminus (Fig. 1c). Note that the selection probability for the reverse Monte Carlo move a_{yx} (from y to x) may be different from a_{xy} . Therefore, we also need to count the number of neighbors of the new end n_{new_end} (Fig. 1 d):

$$a_{xy} = \frac{1}{2} \cdot \frac{1}{n_{old_end}} \text{ and } a_{yx} = \frac{1}{2} \cdot \frac{1}{n_{new_end}}, \tag{1}$$

Moves are accepted with a modified Metropolis criterion:

$$r < \min\left[1, \frac{n_{old_end}}{n_{new_end}} \exp\left(-\beta\Delta V\right)\right],\tag{2}$$

where $r \in (0, 1)$ is a random number between 0 and 1. The prefactor $\frac{n_{oldend}}{n_{newand}}$ ensures that the asymmetric proposal probability is corrected. ΔV denotes the energy difference between the new state *y* and the old state *x*.

The backbite move was first suggested by Mansfield for a lattice model [20]. Note that for a lattice model, the number of neighbors need not be counted necessarily. One can simply choose a potential site and reject the move if there is no monomer present. Successful implementations in the continuum were reported recently in Refs. [10, 33].

2.2. Internal long range move of type II

In Ref. [22] Deutsch proposes two additional moves for single chains in globular states, which were originally used to search for ground-states in the HP-model [34]. Both Monte Carlo moves cut and "rewire" the polymer internally, and have to our knowledge not been ported to a continuum model so far. First, we recall Deutsch's definition of parallel and anti-parallel bonds: Two opposing strands along the chain made up of monomers *i*, *i* + 1 and *j*, *k* are named parallel if k - j = +1 and anti-parallel if k - j = -1 (see figs. 2, 3). In the continuum monomers which are about to be rewired need to be located within $d_{min} < d < d_{max}$. In this notation new bonds are always created between *i* and *j* and *i* + 1 and *k*, which need to be neighbors. When bonds are anti-parallel (and not parallel as suggested in the original publication), we attempt a move of type I. Likewise, if bonds are parallel we try a type II move. For pedagogical reasons we will first discuss moves of type II (Fig. 2) because they are easier to implement.

First we select one bond b_1 made up of monomers (i, i + 1) at random (with probability 1/(N-1)) and choose one of n_{b1} neighboring parallel bonds b_2 . If no such bonds exist, the move is rejected. Bonds b_1 between i and i + 1 and between b_2 between j and k are cut and reconnected to b_3 between i and j and b_4 between i + 1 and k. This leads to following selection probabilities from state x to y:

$$a_{xy} = \frac{1}{N-1} \left(\frac{1}{n_{b_1}} + \frac{1}{n_{b_2}} \right).$$
(3)

Note that the same configuration could have been chosen by first selecting b_2 instead of b_1 , hence we get the additional summand. For the reverse move, we obtain

$$a_{yx} = \frac{1}{N-1} \left(\frac{1}{n_{b_3}} + \frac{1}{n_{b_4}} \right).$$
(4)

A move is accepted if

$$r < \min\left[1, \frac{n_{b_1}n_{b_2}(n_{b_3} + n_{b_4})}{n_{b_3}n_{b_4}(n_{b_1} + n_{b_2})}\exp\left(-\beta\Delta V\right)\right].$$
(5)

As before, ΔV denotes the energy difference between the new state *y* and the old state *x*. Note that the selection probabilities are symmetric in the original publication of Deutsch [22]. One simply checks for neighboring bonds on the lattice and rejects the move if no adequate bond is present.

2.3. Internal long range move of type I

Figure 3 provides details on the long range move of type I. First we select one bond (i, i+1) at random (with probability 1/(N-1)) and choose one of the neighboring antiparallel bonds b_2 (with probability $1/n_{b1}$). Bonds b_1 between i and i + 1 and between b_2 between j and k are cut and reconnected to b_3 between i and j and b_4 between i + 1and k such that the chain is split into a linear and a circular part. From the circular part which consists of n_z bonds we choose a random bond (b_5 in fig. 3 with probability $1/N_z$) and check if neighboring bonds belong to the linear part. One of these n'_{b5} bonds is chosen at random (b_6) and the chain is "rewired" to form the anti-parallel bonds b_7 and b_8 . Sometimes it is possible to reconnect bonds b_5 and b_6 in different ways (as indicated by dotted lines in fig. 2). In this case each potential connection contributes to n'_{b_5} . If there is no such bond b_6 , the move is rejected. Note that in all steps, the number of neighboring bonds need to be determined. This leads to following selection probabilities from state *x* to *y*:

$$a_{xy} = \frac{1}{N-1} \cdot \frac{1}{n_z} \cdot \frac{1}{n'_{b_5}} \left(\frac{1}{n_{b_2}} + \frac{1}{n_{b_1}} \right)$$
(6)

Note that the same configuration could have been chosen by first selecting b_2 instead of b_1 , hence we obtain the additional summand. For the reverse move, we get

$$a_{xy} = \frac{1}{N-1} \cdot \frac{1}{n_z} \cdot \frac{1}{n'_{b_4}} \left(\frac{1}{n_{b_8}} + \frac{1}{n_{b_7}} \right)$$
(7)

This leads to the following Metropolis criterion:

$$r < \min\left[1, \frac{n_{b_1}n_{b_2}n'_{b_5}(n_{b_7} + n_{b_8})}{n_{b_7}n_{b_8}n'_{b_4}(n_{b_1} + n_{b_2})}\exp\left(-\beta\Delta V\right)\right].$$
(8)

As before, ΔV denotes the energy difference between the new state *y* and the old state *x*.

2.4. Implementation

In our program monomers of the chain are stored in sequence in a double-linked list. This allows for an efficient detection of parallel and anti-parallel bonds for internal moves of type II and I. Even more importantly, we set up neighbor tables in which the neighbors of each monomer are stored. Unfortunately, this neighbor table needs to be updated each time local moves, which are essential for the ergodicity of the algorithms, are attempted. In order to minimize these efforts, we always perform several local displacements for all particles before we return to a sequence of bridging moves.

As indicated in previous sections, the internal move of type I is by far the most complicated to implement because we need to take care of the chain separation in both linear and circular parts. The easiest move is implement is the backbite move followed by the internal bridging move of type II.

3. Model and performance analysis

In the following we compare the performance of the moveset with an implementation of the slithering-snake algorithm [17]: We choose with probability 0.5 one end of the chain and attempt to attach the end monomer to a random position at the opposite terminus without changing the bond length. Our model system consists of a simple flexible Lennard-Jones + FENE homopolymer [35, 4, 36] with cut and shifted Lennard-Jones interactions between all monomers

$$V_{LJ} = \begin{cases} 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 + \frac{127}{16384} \right] & \text{, if } r < 2\sqrt[6]{2} \\ 0 & \text{, otherwise} \end{cases}$$
(9)

and FENE interactions between adjacent beads:

$$V_{FENE} = -33.75\epsilon \ln\left[1 - \left(\frac{r}{1.5\sigma}\right)^2\right].$$
 (10)

Note that for a homopolymer as described in Eqs. 9 and 10, the energy difference in the Metropolis criterion of the bridging moves only consists of changes in the bond energy as the position of the particles are only altered by local moves. All simulations took place at $T = 1.66 \epsilon/k_B$, which corresponds to $0.5T_{\Theta}$ (for a chain of infinite length [37].) Therefore, for N = 200 to N = 1000 only globular states are observed.

In an attempt to gauge the efficiency of the three bridging moves, we determined whether or not a particular globule is unknotted by calculating the Alexander polynomial [38]. Note that mathematically knots are only well-defined in closed curves. However, if we connect the ends of an open chain in a well-defined manner we can still obtain corresponding information about the "knottedness" of the globule. Ref. [38] provides details on this topic as well as on our implementation of the Alexander polynomial. The particular closure used in this work is described in Ref. [14]. To quantify the number of independent configurations generated by the bridging algorithms we measured the correlation times between unknotted configurations [39]:

$$A(k) = \frac{\langle U_i U_{i+k} \rangle - \langle U_i \rangle^2}{\langle U_i^2 \rangle - \langle U_i \rangle^2}$$
(11)

where

$$U_i = \begin{cases} 0, & \text{if the } i\text{th configuration contains a knot} \\ 1, & \text{if the } i\text{th configuration is unknotted.} \end{cases}$$
(12)

An "independent configuration" is defined by the decrease of the auto-correlation function to e^{-1} . Note that our approach is particularly well-suited for the task of comparing bridging moves as it focuses on changes in topology and self-entanglements. In our current implementation neither bridging moves nor slithering snake moves are ergodic because they do not alter bond lengths. Therefore, it becomes necessary to combine the algorithms with local Monte Carlo moves: In our case, each simulation run spent 26 % of the time with local displacements.

Fig. 4 shows the percentage of unknotted configurations for chain lengths ranging from N = 200 up to N = 1000: For N = 200 roughly 90 % of all configurations are unknotted, for N = 1000 only 20 % are unknotted [4]. Figure 5a shows the number of generated configurations per minute divided by our estimate for the geometric correlation time as a function of chain length. All configurations were created on a single core of a Core2Quad Q6700 processor running at 3.33 Ghz. Figure 5b shows the same data normalized by the results for the slithering snake moves. Note that the time-limiting step in this analysis is not the generation of globular states but the subsequent knot analysis which took place on a supercomputer. For chains larger than N = 400 all bridging moves become more efficient than slithering snake moves. The internal bridging move of type II is the most efficient followed by the backbite move and the internal move of type I. If we consider that the latter is by far the most difficult to code among the three, we do not recommend its implementation. Interestingly, the time to generate an "independent configuration" appears to be almost independent of chain length for type II and backbite moves, whereas it increases rapidly for slithering snake moves as expected. Finally, it is worth noting that combinations of moves may even yield better results. A combination of internal II and backbite moves is about two orders of magnitude faster for N = 1000 than slithering snake moves.

4. Conclusion

The aim of this paper is threefold. First, we have implemented three efficient lattice bridging moves for single flexible globular homopolymer chains in a continuum model, and discussed difficulties arising from the implementation. In an attempt to measure performance we determined correlation times between unknotted globular states. To our knowledge this is the first time in which topology is considered to gauge the efficiency of global Monte Carlo moves. From this analysis, we conclude that all bridging moves become more efficient than slithering snake moves for chains larger than N = 400 monomers. Among the three moves, the internal bridging move of type II is the most successful. The most complicated move (internal I) performs worst and we do not recommend its implementation. For N = 1000 a combination of backbiting and internal bridging moves of type II is two orders of magnitude faster than our implementation of the slithering snake move.

The exact speed–up factors should, however, be taken with a grain of salt and rather serve as a guideline. Long-range correlations, which are only relaxed by local moves, were e.g. not considered, and results also depend a lot on the model under consideration. If chain stiffness is included [40], efficiency will drop, and results will also deteriorate if the implementation does not include look-up tables. Nonetheless, we believe to have demonstrated that bridging algorithms are indeed the method of choice for the simulation of long globular polymers in the continuum.

Acknowledgments

We would like to particularly thank T. Wüst whose comments helped us tremendously to implement and further improve the bridging moves. We would also like to thank K. Binder, W. Paul and M. Taylor for fruitful discussions on the topic. This work received financial support from the Deutsche Forschungsgemeinschaft (SFB 625-A17) and the MWFZ Mainz. Computing time on JUMP and JUROPA supercomputers at the NIC Jülich and on the linux cluster at the ZDV Mainz are gratefully acknowledged, too.

References

- [1] van Rensburg, E. J. J., J. Phys. A: Math. Theor. 42 (2009) 323001 (97pp).
- [2] Binder, K. and Paul, W., Macromolecules 41 (2008) 4537.
- [3] Sokal, A. D., Nucl. Phys. B Proc.Suppl. 47 (1996) 172.

- [4] Virnau, P., Kantor, Y., and Kardar, M., J. Am. Chem. Soc. 127 (2005) 15102.
- [5] Grosberg, A. and Nechaev, S., Adv. Polym. Sci. 106 (1993) 1.
- [6] Mansfield, M., Macromolecules 27 (1994) 5924.
- [7] Grosberg, A., Poly. Sci. Ser. A 51 (2009) 70.
- [8] Orlandini, E., Stella, A. L., and Vanderzande, C., Phys. Biol. 6 (2009).
- [9] Rampf, F., Paul, W., and Binder, K., Europhys. Lett. 70 (2005) 628.
- [10] Taylor, M. P., Paul, W., and Binder, K., Phys. Rev. E 79 (2009) 050801.
- [11] Grosberg, A., Nechaev, S., and Shakhnovich, E., J. Phys. France 49 (1988) 2095.
- [12] Lua, R. and Grosberg, A., PLoS Comput. Biol. 2 (2006) e45.
- [13] Wüst, T. and Landau, D. P., Phys. Rev. Lett. 102 (2009) 178101.
- [14] Virnau, P., Mirny, L. A., and Kardar, M., PLoS Comput. Biol. 2 (2006) e122.
- [15] Kron, A., Vysokomolekulyarnye Soedineniya 7 (1965) 1228.
- [16] Kron, A., Ptitsyn, O., Skvortsov, A., and Federov, A., Molek. Biol 1 (1967) 576.
- [17] Wall, F. and Mandel, F., J. Chem. Phys. 63 (1975) 4592.
- [18] Mandel, F., J. Chem. Phys. 70 (1979) 3984.
- [19] Grassberger, P., Phys. Rev. E 56 (1997) 3682.
- [20] Mansfield, M., J. Chem. Phys. 77 (1982) 1554.
- [21] Olaj, O. and Lantschbauer, W., Makromol. Chem. Rapid Commun. 3 (1982) 847e58.
- [22] Deutsch, J., J. Chem. Phys. 106 (1997) 8849.
- [23] Wang, F. and Landau, D. P., Phys. Rev. Lett. 86 (2001) 2050.
- [24] Pant, P. and Theodorou, D., Macromolecules 28 (1995) 7224.
- [25] Uhlherr, A., Macromolecules **33** (2000) 1351.
- [26] Wick, C. and Siepmann, J., Macromolecules 33 (2000) 7207.
- [27] Uhlherr, A., Mavrantzas, V., Doxastakis, M., and Theodorou, D., Macromolecules 34 (2001) 8554.
- [28] Theodorou, D. N., Variable-connectivity monte carlo algorithms for the atomistic simulation of long-chain polymer systems, in *Bridging Time Scales: Molecular Simulations for the Next Decade*, edited by Nielaba, P., Mareschal, M., and Ciccotti, G., pages 67–127, Springer, 2002.

- [29] Mavrantzas, V., Monte carlo simulation of chain molecules, in *Handbook of Materials Modeling*, edited by Yip, S., pages 2583–2597, Springer, 2005.
- [30] Karayiannis, N. C., Giannousaki, A. E., Mavrantzas, V. G., and Theodorou, D. N., J. Chem. Phys. 117 (2002) 5465.
- [31] Auhl, R., Everaers, R., Grest, G. S., Kremer, K., and Plimpton, S. J., J. Chem. Phys. **119** (2003) 12718.
- [32] Meyer, H. et al., Eur. Phys. J. E 26 (2008) 25.
- [33] Schnabel, S., Vogel, T., Bachmann, M., and Janke, W., Chem. Phys. Lett. 476 (2009) 201.
- [34] Lau, K. and Dill, K., Macromolecules 22 (1989) 3986.
- [35] Kremer, K. and Grest, G., J. Chem. Phys. 92 (1990) 5057.
- [36] Virnau, P., Müller, M., MacDowell, L. G., and Binder, K., J. Chem. Phys. 121 (2004) 2169.
- [37] Binder, K., Müller, M., Virnau, P., and MacDowell, L. G., Polymer plus solvent systems: Phase diagrams, interface free energies, and nucleation, in *Adv. Polym. Sci.*, volume 173, pages 1–104, Springer, Berlin, 2005.
- [38] Virnau, P., Detection and visualization of physical knots in macromolecules, in *Computer simulation studies in Condensed Matter Physics*, edited by Landau, D., Lewis, S., and Schüttler, H., Elsevier, 2009 (inpress).
- [39] Landau, D. and Binder, K., A guide to Monte Carlo simulations in statistical physics, Cambridge University Press, Cambridge [u.a.], 3. ed. edition, 2009.
- [40] Cifra, P., Benkova, Z., and Bleha, T., J. Phys. Chem. B 112 (2008) 1367.

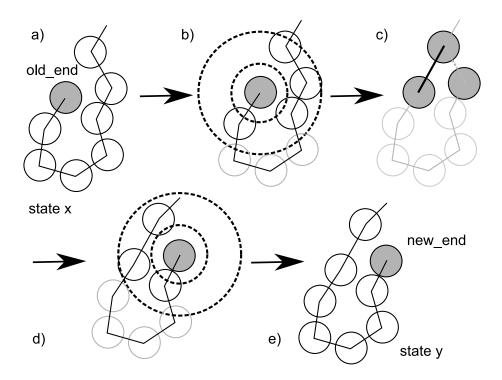


Figure 1: Off-lattice version of the backbite move [20].

a) An end monomer is chosen with equal probability.
b) Neighbors (within distance d_{min} < d < d_{max}) of this monomer are identified and counted. One of these neighbors is selected at random.

c) A new bond is created between the neighbor and the old end monomer. The bond between the chosen neighbor and its successor (along the direction from the old end to the neighbor) is cut.

d) The number of neighbors of the new end needs to be counted to fulfill detailed balance.

e) The chosen neighbor becomes the new end and the new configuration is accepted with a modified Metropolis criterion.

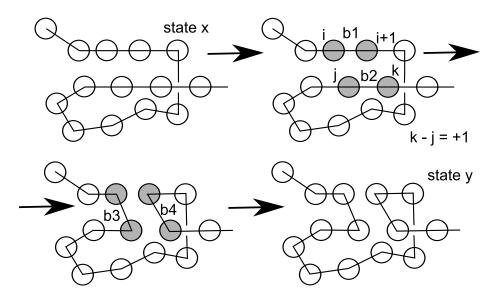


Figure 2: Internal long range move of type II (parallel bonds: k-j=+1) For details see main text.

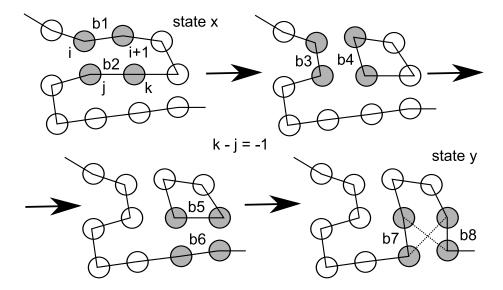


Figure 3: Internal long range move of type I (anti-parallel bonds: k-j=-1): The chain is split into a linear and a circular part and reconnected. For details see main text.

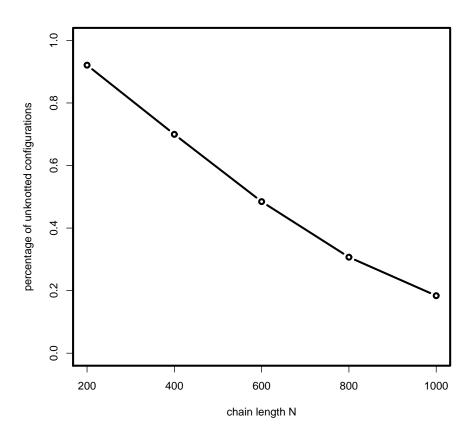


Figure 4: Percentage of unknotted configurations for single bead-spring polymer chains of size N=200 to N=1000 in the globular phase ($T = 1.66 \epsilon/k_B$)

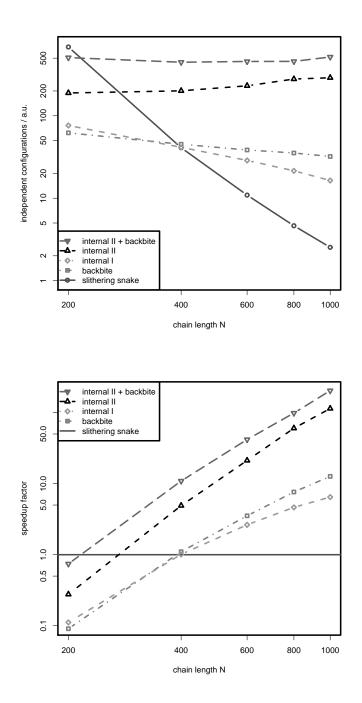


Figure 5: Performance analysis for single bead-spring polymer chains of size N=200 to N=1000 in the globular phase ($T = 1.66 \epsilon/k_B$)

a. Number of "independent configurations" per minute as defined in the main text.

b. Speed-up factor of bridging moves in comparison with slithering snake moves. For this particular model all rebridging moves become more efficient than slithering snake moves for chains larger than 400 monomers. For clarity we have used logarithmic scale on both axis.