Department of Mathematics and Statistics

Applications of Monte Carlo Methods in Studying Polymer Dynamics

by

Changqiong Wang

Thesis submitted for the degree of
Doctor of Philosophy

June 2018

Declaration of Authorship

I confirm that this is my own work and the use of all material from other sources has been properly and fully acknowledged.

SIGNATURE:

DATE:

Abstract

Polymer melts generally demonstrate complicated dynamic and stress relaxation behaviours, which are usually difficult to be described by theoretical models and typically involve many parameters. The thesis specifically centers on developing novel models to study entangled polymer melts with the help of Monte Carlo methods. We commence by providing a brief description of the problems and prior knowledge for polymer and Bayesian statistics in Chapter 1.

In Chapter 2, we develop a multi-bead coarse-grained model based on single-particle dynamics at the level of the individual molecule. Next, we show how to embed Kalman filter into Markov chain Monte Carlo (MCMC) paradigm to draw inferences on the model parameters. Then the estimates of parameters can be used to reproduce the dynamics of the center of mass of single chains in molecular dynamics (MD) simulations. We explore the performance of coarse-grained models for linear chains with different hidden beads and find that the multi-bead model is preferable to have Rouse-like structure rather than asymmetric star structure.

The next part of the thesis investigates two different models dealing with nonlinear systems. The first model is the extension of the multi-bead model for dealing with nonlinear interactions and the second model is described by the generalized Langevin equation with memory kernel. The MCMC method fails for the first model due to the fact that the particle MCMC using for parameter estimation ends up with

noisy likelihood and is unable to explore the parameter space sufficiently. Laplace transform and numerical approximation are applied to get the estimates of unknown parameters for the second model. Comparing the linear multi-bead model and the model with memory kernel, we find that the former is more promising to describe the dynamics of entanglement polymers.

In Chapter 4, we introduce Monte Carlo methods in combination with the slip-spring model, which was originally developed for describing dynamics of entangled polymers, for detecting entanglements in the polymer melts obtained from MD simulations. The total number, the effective lengths and the locations of the anchor points of the slip-springs will be well decided by Bayesian statistical methods. The Bayesian alternative can also compute the posterior distribution of different models and provide uncertainty analysis on the estimation of model parameters.

Finally, in Chapter 5, we provide concluding remarks and discuss the limitations of our methodologies, and point out possible future research directions.

Acknowledgements

I would like to express most heartfelt gratitude to my supervisors Prof. Alexei Likhtman, Dr. Richard Everitt, Dr. Zuowei Wang and Dr. Patrick Ilg. Their continuous guidance, immense knowledge and great enthusiasm and motivation in science have helped me overcome enormous difficulties encountered during my PhD studies and research.

My sincere gratitude also goes to all my colleagues at the Department of Mathematics and Statistics. I am grateful to all my friends for their great help and support that made me have a wonderful and enjoyable time during my stay at Reading.

Lastly and most importantly, I would like to express my sincere gratitude to my parents for their remarkable efforts in my upbringing and education. I am also grateful to my boyfriend, Cheng Ye, who has been supporting me unconditionally during these years. We have frequently discussed problems, exchanged opinions and shared experiences together, and this thesis could not have been completed without his great help.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Overview

Polymers are large molecules composed of many elementary units called monomers. The monomers are connected by covalent bonds. The number of covalent bonds that constitute the polymer is referred to as the degree of polymerization, denoted as N. N is typically quite large, ranging from 10^2 to 10^6 and some polymers of length up to 10^{10} can be found in the natural world. There are many types of polymers according to their branching architectures, such as linear, star, ring, comb, dendrimers and Cayley trees. Examples of polymer chains of different architectures are shown in Fig. 1.1.

In general, common chemical materials with the same molecular weight but constituted of different units usually show dramatically different behaviours. However, polymer materials with different monomer types but with the same global topology often exhibit similar behaviours. Some of the polymer chemical characteristics, including the molecular structures, the degree of polymerization and chemical composition are fixed after the polymerization or after being synthesized. It is interesting to note that their conformation determined by the relative positions of their monomers continuously change under thermal fluctuativity. The conformation of a linear chain comprised of N bonds can be described by a set of $N+1$ position vectors $\{\mathbf{R}_i\}_{i=0}^{N}$. Alternatively, its conformation can be represented by a set of bond vectors $\{\mathbf{r}_i\}_{i=1}^{N}$,

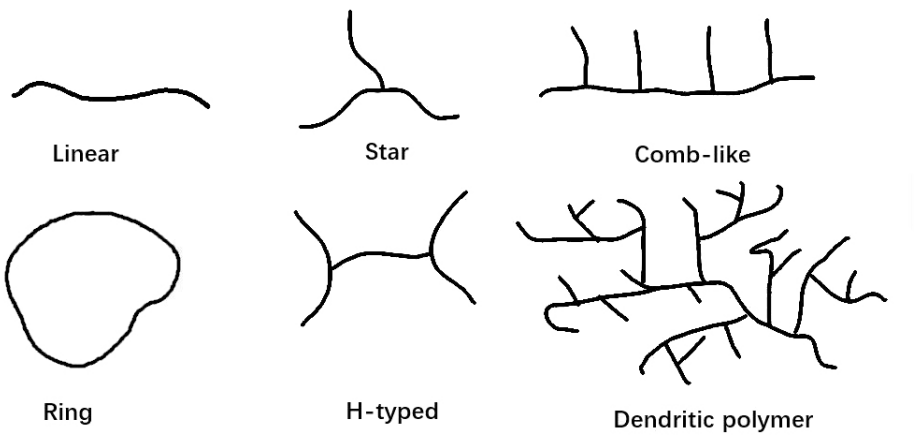


Figure 1.1: Different type of polymer architectures

where

$$\mathbf{r}_i = \mathbf{R}_i - \mathbf{R}_{i-1}, \qquad i = 1, 2, ..., N. \tag{1.1}$$

The end-to-end vector of the chain is defined as

$$\mathbf{R}_e = \mathbf{R}_N - \mathbf{R}_0 = \sum_{i=1}^{N} \mathbf{r}_i. \tag{1.2}$$

The mean square of the end-to-end vector is given by

$$\begin{aligned} \langle \mathbf{R}_e^2 \rangle &= \left\langle \left(\sum_{i=1}^{N} \mathbf{r}_i \right) \cdot \left(\sum_{j=1}^{N} \mathbf{r}_j \right) \right\rangle \\ &= \sum_{i=1}^{N} \sum_{j=1}^{N} \langle \mathbf{r}_i \cdot \mathbf{r}_j \rangle, \end{aligned} \tag{1.3}$$

where $\langle \mathbf{R}_e^2 \rangle$ is often considered as a measure of the size of a linear chain. This quantity is not well defined for polymers with other architectures, since sometimes there are either too many ends, e.g., star and comb polymers, or no ends at all, e.g., ring polymers. An alternative way to characterize the size of a polymer with any topology

is to evaluate the radius of gyration,

$$\mathbf{R}_g^2 = \frac{1}{N+1}\sum_{i=0}^{N}(\mathbf{R}_i - \mathbf{R}_{cm})^2\,, \tag{1.4}$$

where $\mathbf{R}_{cm}$ is the center of mass of the polymer defined as

$$\mathbf{R}_{cm} = \frac{1}{N+1}\sum_{i=0}^{N}\mathbf{R}_i. \tag{1.5}$$

A polymer chain can take a large number of conformations due to thermal fluctuations and intra- and inter-chain interactions. It is more practical to study the statistic properties of polymer chains.

Let us start with one of the simplest models: the freely jointed model. Consider a linear chain composed of $N+1$ beads $\{\mathbf{R}_i\}_{i=0}^{N}$, the length of each bond $\mathbf{r}_i$ is b ($b>0$) and the coordinates of the bonds are independent of each other, which implies

$$\langle \mathbf{r}_i\rangle = \mathbf{0}, \quad \langle \mathbf{r}_i\cdot\mathbf{r}_j\rangle = \begin{cases} b^2, & i=j \\ 0, & i\neq j. \end{cases}$$

Therefore, the mean square distance of end-to-end vector of a freely jointed chain, $\langle \mathbf{R}_e^2\rangle$, can be written as

$$\begin{aligned}\langle \mathbf{R}_e^2\rangle &= \sum_{i=1}^{N}\langle \mathbf{r}_i^2\rangle + \sum_{i\neq j}\langle \mathbf{r}_i\cdot\mathbf{r}_j\rangle \\ &= Nb^2. \end{aligned}\tag{1.6}$$

Now we focus on more complex models. Assume the orientation of each bond is independent while the length of each bond is allowed to fluctuate with $\langle \mathbf{r}_i\rangle = \mathbf{0}$ and $\langle \mathbf{r}_i^2\rangle = b^2$ where b is known as the Kuhn length. If N is large enough, it can be proved that the distribution of $\langle \mathbf{R}_e^2\rangle$ with N bonds is Gaussian,

$$P(\mathbf{R}_e, N) = \left(\frac{3}{2\pi N b^2}\right)^{3/2}\exp\left\{-\frac{3\mathbf{R}_e^2}{2Nb^2}\right\}, \tag{1.7}$$

which gives

$$\langle \mathbf{R}_e \rangle = \mathbf{0} \text{ and } \langle \mathbf{R}_e^2 \rangle = Nb^2.$$

Further details about the chain statistic properties can be found in Doi and Edwards [1] and Rubinstein and Colby [2]. Note that in a real polymer, the consecutive bonds cannot be completely independently orientated due to the chain connectivity and steric hindrance. On the other hand, if we take a coarse-graining approach by treating a set of consecutive monomers, say n monomers, as a new segment, the new bond vectors which connect the center of mass of consecutive coarse-graining segments would exhibit random behaviour when n is large enough.

1.2 The Gaussian Chain

Doi and Edwards [1] showed that, in the study of statistical properties of chain end-to-end vector $\mathbf{R}_e$, the local configuration of sufficiently long chains only depends on the effective bond length b. Therefore, to analyse the statistical properties of a polymer chain, we start with Gaussian chains.

Consider a linear chain with N bonds, $\{\mathbf{r}_i\}_{i=1}^N$, where the length of $\mathbf{r}_i$ is distributed according to Gaussian distribution:

$$\phi(\mathbf{r}) = \left(\frac{3}{2\pi b^2}\right)^{3/2} \exp\left(-\frac{3\mathbf{r}^2}{2b^2}\right). \tag{1.8}$$

In such case, we have

$$\langle \mathbf{r} \rangle = \mathbf{0}, \quad \langle \mathbf{r}^2 \rangle = b^2. \tag{1.9}$$

The joint probability density of a chain with configuration $\{\mathbf{r}_i\}_{i=1}^N$ is

$$\Phi(\{\mathbf{r}_i\}) = \prod_{i=1}^{N} \left[\frac{3}{2\pi b^2}\right]^{3/2} \exp\left(-\frac{3\mathbf{r}_i^2}{2b^2}\right) \tag{1.10}$$

$$= \left(\frac{3}{2\pi b^2}\right)^{3N/2} \exp\left(-\sum_{i=1}^{N} \frac{3(\mathbf{R}_i - \mathbf{R}_{i-1})^2}{2b^2}\right). \tag{1.11}$$

Such a chain with joint probability density described as $\Phi(\{\mathbf{r}_i\})$ is called the Gaussian chain. Though the Gaussian chain does not provide us with details of the local structure of the chain, it tells us the global properties of a chain. Compared to freely jointed model mentioned before, the mathematical calculation involving a Gaussian chain is much easier to achieve.

The Gaussian chain represents a chain with consecutive beads connected by harmonic springs. The potential energy of the whole chain is given by

$$U(\{\mathbf{R}_i\} = \frac{3k_BT}{2b^2}\sum_{i=1}^{N}(\mathbf{R}_i - \mathbf{R}_{i-1})^2, \tag{1.12}$$

where k_B is the Boltzmann constant and T is temperature.

Let us consider the distribution of the vector $\mathbf{R}_0 - \mathbf{R}_n$ between monomer 0 and monomer n. The distribution of $\mathbf{R}_n - \mathbf{R}_0$ can be written as

$$P(\mathbf{R}_n - \mathbf{R}_0) = \int d\mathbf{R}_1 ... \int d\mathbf{R}_{n-1} \exp\left[-\frac{3k_BT}{2b^2}\left(\sum_{i=1}^{n}(\mathbf{R}_i - \mathbf{R}_{i-1})^2\right)\right]. \tag{1.13}$$

The integration in Eq. (1.13) can be performed in a stepwise manner,

$$\begin{aligned} P(\mathbf{R}_2 - \mathbf{R}_0) &= \int d\mathbf{R}_1 \exp\left[-\frac{3k_BT}{2b^2}\left((\mathbf{R}_1 - \mathbf{R}_0)^2 + (\mathbf{R}_2 - \mathbf{R}_1)^2\right)\right] \\ &= \left(\frac{2\pi b^2}{3k_BT}\right)^{\frac{d}{2}} \frac{1}{2^{d/2}} \exp\left[-\frac{3k_BT}{4b^2}(\mathbf{R}_2 - \mathbf{R}_0)^2\right], \end{aligned}$$

where d is the dimension of the position vector $\mathbf{R}_0$. Similarly, to complete the inductive step, we obtain

$$\begin{aligned} &\int d\mathbf{R}_{n-1} \exp\left[-\frac{3}{2b^2}\left(\frac{1}{n-1}(\mathbf{R}_{n-1} - \mathbf{R}_0)^2 + (\mathbf{R}_n - \mathbf{R}_{n-1})^2\right)\right] \\ &\qquad = \left(\frac{2\pi b^2}{3k_BT}\right)^{\frac{d}{2}} \frac{1}{(\frac{n}{n-1})^{d/2}} \exp\left[-\frac{3k_BT}{2b^2}\cdot\frac{1}{n}(\mathbf{R}_n - \mathbf{R}_0)^2\right]. \end{aligned}$$

Hence, $\mathbf{R}_n - \mathbf{R}_0$ given by induction as

$$\begin{aligned} P(\mathbf{R}_n - \mathbf{R}_0) &= \int d\mathbf{R}_1 .. \int d\mathbf{R}_{n-1} \exp\left[-\frac{3k_BT}{2b^2}\left(\sum_{i=1}^{n}(\mathbf{R}_i - \mathbf{R}_{i-1})^2\right)\right] \\ &= \left(\frac{2\pi b^2}{3k_BT}\right)^{\frac{d(n-1)}{2}} \frac{1}{n^{d/2}} \exp\left[-\frac{3k_BT}{2nb^2}(\mathbf{R}_n - \mathbf{R}_0)^2\right] \\ &= \left(\frac{2\pi b^2}{3k_BT}\right)^{\frac{dn}{2}} \left(\frac{3k_BT}{2\pi nb^2}\right)^{d/2} \exp\left[-\frac{3k_BT}{2nb^2}(\mathbf{R}_n - \mathbf{R}_0)^2\right]. \end{aligned} \tag{1.14}$$

The expression in Eq. (1.14) tells us that, in a Gaussian chain, the distribution of a vector between any two monomers is also Gaussian.

1.3 Brownian Motion

Brownian motion was first observed by a botanist named Robert Brown in 1827 when he was looking at the motion of pollen grains in water. It describes the random motion of particles suspended in a fluid where the molecules of the fluid hit the particles from different directions with different velocities. Brownian motion is governed by Newton's second law

$$\mathbf{F} = m\mathbf{a}, \tag{1.15}$$

where $\mathbf{a} = \frac{d^2\mathbf{R}}{dt^2}$ is the acceleration of the particle and m is the mass of the particle. Considering the dynamics of polymer chains, the random force $\mathbf{F}$ acting on a monomer comes from frequent collisions with surrounding monomers. Due to the chaotic character of the collisions, it is natural to assume that $\mathbf{F}$ is Gaussian with zero mean and is delta-correlated in time,

$$\langle \mathbf{F} \rangle = \mathbf{0} \text{ and } \langle \mathbf{F}(t), \mathbf{F}(t') \rangle = \sigma^2 \delta(t, t'), \tag{1.16}$$

where σ^2 is the variance of the random force.

Assume we focus on one of the monomers of a polymer chain moving through the liquid with constant velocity $\mathbf{v}$, the surrounding liquid will resist its motion with

a friction force, which is given by

$$\mathbf{f} = \xi \mathbf{v}, \tag{1.17}$$

where ξ is friction coefficient. If the solvent is treated as a continuum medium, the motion of the monomers along the chain is governed by the Langevin equation of motion [1],

$$m\frac{d^2\mathbf{R}_i}{dt^2} = -\frac{U(\mathbf{R}_0, \cdots, \mathbf{R}_N)}{\partial \mathbf{R}_i} - \xi\frac{d\mathbf{R}_i}{dt} + \mathbf{f}_i(t), \quad i = 0, \cdots, N, \tag{1.18}$$

where $U(\mathbf{R}_0, \cdots, \mathbf{R}_N)$ is the total potential of the chain. The random force $\mathbf{f}_i(t)$ is assumed to be delta-correlated in time and independent for different monomers as given in Eq. (1.16). The Fluctuation-dissipation theorem [3] establishes the following relationship between the random force $\mathbf{f}_i(t)$ and the friction coefficient ξ,

$$\langle \mathbf{f}_i(t) \cdot \mathbf{f}_j(t') \rangle = 2k_B T \xi \delta_{ij} \delta(t - t'). \tag{1.19}$$

In the overdamped case, the inertia of beads is usually negligible, leading to a reduced form of Eq. (1.18)

$$\xi\frac{d\mathbf{R}_i}{dt} = -\frac{U(\mathbf{R}_0, \cdots, \mathbf{R}_N)}{\partial \mathbf{R}_i} + \mathbf{f}_i(t), \quad i = 0, \cdots, N. \tag{1.20}$$

1.4 The Rouse Models

Extensive studies have been carried out on the properties of linear polymer chains. The most successful model for studying the dynamics of short chains is the Rouse model [4]. A Rouse chain is composed of a set of massless beads connected by harmonic springs. Fig. 1.2 is the sketch of a Rouse chain. The intra-chain potential $U(\{\mathbf{R}_i\}_{i=0}^N)$ shown in Eq. (1.20) is given by

$$U(\{\mathbf{R}_i\}_{i=0}^N) = \frac{1}{2}k\sum_{i=1}^{N}(\mathbf{R}_i - \mathbf{R}_{i-1})^2, \tag{1.21}$$

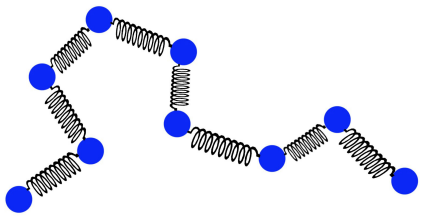


Figure 1.2: Rouse chain model

where $k = \frac{3k_BT}{b^2}$, as shown in Eq. (1.12), is the spring constant which guarantees the mean square of bond length to be b^2.

In such case, the equations of motion for individual monomers along the Rouse chain can be rewritten as:

$$\begin{aligned}
\xi\frac{d\mathbf{R}_0}{dt} &= -\frac{3k_BT}{b^2}(\mathbf{R}_1 - \mathbf{R}_0) + \mathbf{f}_0(t), \\
\xi\frac{d\mathbf{R}_i}{dt} &= -\frac{3k_BT}{b^2}(\mathbf{R}_{i-1} + \mathbf{R}_{i+1} - 2\mathbf{R}_i) + \mathbf{f}_i(t), \quad 1 \le i \le N-1, \\
\xi\frac{d\mathbf{R}_N}{dt} &= -\frac{3k_BT}{b^2}(\mathbf{R}_{N-1} - \mathbf{R}_N) + \mathbf{f}_N(t).
\end{aligned} \tag{1.22}$$

Using following matrix notations

$$\begin{aligned}
\mathbf{R} &= (\mathbf{R}_0, \mathbf{R}_1, \cdots, \mathbf{R}_N)^T, \\
\mathbf{f}(t) &= (\mathbf{f}_0(t), \mathbf{f}_1(t), \cdots, \mathbf{f}_N(t))^T.
\end{aligned}$$

Eq. (1.22) can be rewritten as

$$\xi\frac{d\mathbf{R}}{dt} = -\frac{3k_BT}{b^2}\boldsymbol{A}\mathbf{R} + \mathbf{f}(t), \tag{1.23}$$

where the connectivity matrix of the Rouse chain is a diagonal block matrix

$$\mathbf{A} = \begin{pmatrix}
1 & -1 & 0 & 0 & \cdots & & \\
-1 & 2 & -1 & 0 & \cdots & & \mathbf{0} \\
0 & -1 & 2 & -1 & \cdots & & \\
 & & & \cdots & & & \\
 & & & \cdots & 2 & -1 & 0 \\
 & \mathbf{0} & & \cdots & -1 & 2 & -1 \\
 & & & \cdots & 0 & -1 & 1
\end{pmatrix}.$$

Applying orthogonal transformation $\mathbf{\Psi}$ on $\mathbf{R}$, we have $\mathbf{X} = \mathbf{\Psi R}$, Eq. (1.22) turns out to be diagonal with new variable $\{\boldsymbol{X_p}\}_{p=0}^{N}$,

$$\xi_p \frac{\partial \mathbf{X}_p}{\partial t} = -k_p \mathbf{X}_p + \mathbf{f}_p(t), \tag{1.24}$$

$$\mathbf{X}_p = \frac{1}{N+1} \sum_{i=0}^{N} \mathbf{R}_i \cos\left(\frac{\pi p(i+1/2)}{N+1}\right), \tag{1.25}$$

where $\mathbf{X}_p$ is known as normal mode, and the random term $\mathbf{f}_p(t)$ is characterized by the Fluctuation-dissipation theorem shown in Eq. (1.19),

$$\langle \mathbf{f}_p(t) \mathbf{f}_q(t') \rangle = 2k_b T \xi_p \delta_{pq} \delta(t - t'). \tag{1.26}$$

It is interesting to note that the first component with $p = 0$ is the center of mass of the chain, which indicates that the motion of center of mass of a Rouse chain is Brownian motion characterized by the following formula,

$$\begin{aligned} k_0 &= 0, \quad \xi_0 = (N+1)\xi, \\ \mathbf{X}_0 &= \frac{1}{N+1} \sum_{i=0}^{N} \mathbf{R}_i = \mathbf{R}_{cm}. \end{aligned}$$

For mode numbers p larger that 0, we have

$$\begin{aligned} \xi_p &= 2(N+1)\xi, \\ k_p &= \frac{24 k_B T (N+1)}{b^2} \sin^2\left(\frac{\pi p}{2(N+1)}\right), \end{aligned}$$

which provides the following relaxation time τ_p of normal mode $\mathbf{X}_p$,

$$\tau_p = \frac{\xi_p}{k_p} = \frac{\xi b^2}{12 k_B T} \sin^{-2}\left(\frac{\pi p}{2(N+1)}\right) \tag{1.27}$$

Hence, the longest relaxation time, known as the Rouse time τ_R, is given by

$$\tau_R = \tau_1 = \frac{\xi b^2}{12 k_B T} \sin^{-2}\left(\frac{\pi}{2(N+1)}\right) \tag{1.28}$$

If the length of the chain N is large enough, $\sin^{-2}\left(\frac{\pi}{2(N+1)}\right)$ can be approximated with $\frac{4(N+1)^2}{\pi^2}$, then we obtain

$$\tau_R \approx \frac{\xi b^2 (N+1)^2}{3\pi^2 k_B T}. \tag{1.29}$$

Rouse time is an important concept in the study of polymer dynamics. It characterizes the time scale for a polymer to diffuse a distance of its own size. On the time scales that shorter than the Rouse time, the polymer will show viscoelastic features; while on the time scales that larger than the Rouse time, the motion of the chain can be viewed as a single-particle diffusion process [2].

1.5 The Coarse-Grained Model

The dynamics of a polymer chain in melt or solutions cover a wide range of time and length scales, typically from picoseconds to days and from nanometers to millimeters, or even longer. It usually requires different techniques to study the dynamics at different length and time scales. If we are interested only in the mesoscopic to macroscopic behaviours of the molecular system, detailed chemical structures and fast fluctuations at very short time scales can be neglected, and only their average values are essential. It is possible to develop coarse-grained models that involve only the mesoscopic variables by averaging over the unessential atomic details. Molecular

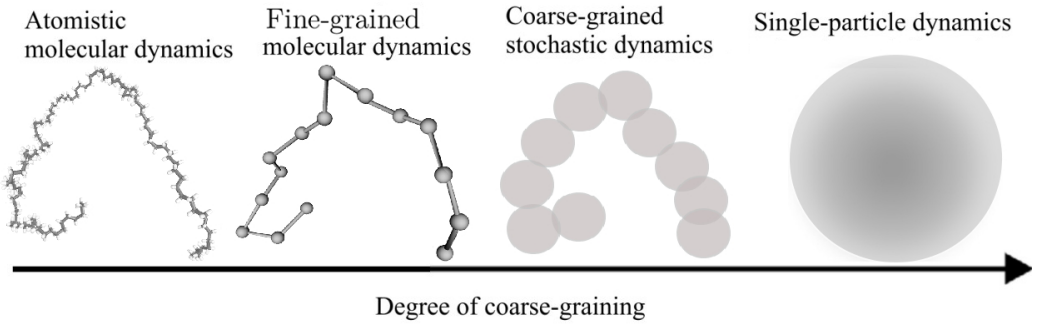


Figure 1.3: Sketches of different levels of coarse-graining for a polymer chain

dynamics (MD) simulation is a powerful tool in studying the dynamics of polymers which can be simulated based on different levels of coarse-graining. Fig. 1.3 shows the schematic representation of different levels of coarse-graining for a polymer chain. The most detailed model takes into account atomistic interactions such that specified chemical properties are well represented. Grouping up several atoms into one bead, we obtain the fine-grained coarse-graining model. A higher level of coarse-graining can be achieved by lumping more atoms into a larger bead. Representing a whole polymer chain with a single particle, one gets the most coarse-graining model. MD simulations at the atomic level are very slow and usually take incredibly long time to reach equilibrium states. Due to the computational power limitation, it is necessary to develop coarse-graining models to study polymer dynamics at large time and length scales.

In this thesis, we are interested in the most coarse-graining model where the whole polymer chain is represented by a single particle — the chain center of mass. We will study single-particle dynamics using trajectories extracted from fine-grained MD simulations.

1.6 Entanglements in Polymer Melts

The Rouse model is a cornerstone of polymer dynamics which can analytically express almost all the observables of interest. However, it is not suitable for describing the dynamics of long polymers. It is well established that the dynamics of polymers are heavily dependent on the length and structure of the chains [1] [5]. The melts and concentrated solutions of long polymer chains exhibit extremely slow dynamics due to topological constraints caused by the uncrossibility of the chains, called entanglements. Entanglements will restrict the lateral motion of the chains and lead more complicated rheological properties. The dynamics and rheology of long polymers are of great interest, especially for practical applications, such as improving the proces-

sivity of polymer melts, the strength of adhesives and cracking resistance of fibres. Fig. 1.4 shows a snapshot of the primitive paths of a long polymer chain and its neighbouring chains.

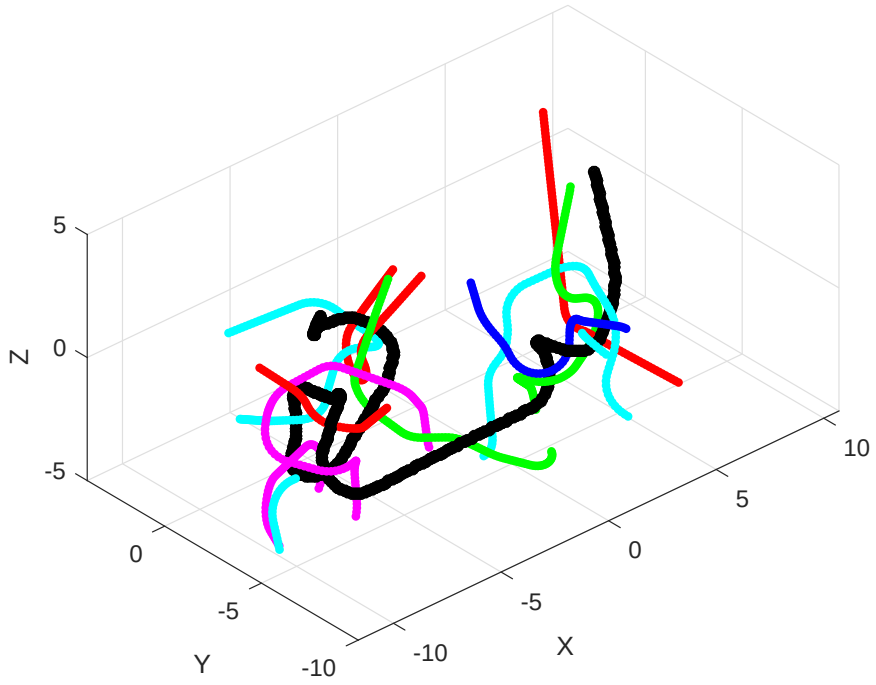


Figure 1.4: A long polymer chain (black) and its neighbouring chains

Extensive efforts have been devoted to defining the microscopic pictures of entanglements. Several numerical tools have been reported for detecting entanglements at microscopic scales, including primitive path analysis (PPA) method developed by Everaers et al. [6] [7] [8] [9], the tube axis method by Likhtman [10], isoconfiguration ensemble [11] and the contact map method by Likhtman [12] [13]. Our aim is to introduce novel approaches based on Bayesian statistics and Monte Carlo methods to detect the existence of entanglements and infer the number of entanglements along the chain backbone. We will discuss these methods with further details in Chapter 4.

1.7 Bayesian Statistics and Monte Carlo Methods

Bayesian statistics, named after Thomas Bayes, refers a class of methods widely used in the field of probability and statistics. Bayesian statistics is based on the belief that the uncertainty of unknowns should be quantified by probabilities. One of the most important opinions of Bayesian statistics is that "probability is orderly opinion, and that inference from data is nothing other than the revision of such opinion in the light of relevant new information" [14]. This message delivers a key idea that Bayesian statistics involves updating our knowledge based on the new up-to-date information.

Different from the maximum likelihood estimate (MLE) methods, which gives a single point estimate with confidence interval estimate, Bayesian statistics provides the full posterior probability of the unknown parameter, which means that Bayesian statistics is able to use probability to quantify the uncertainty of the parameters.

1.7.1 Bayesian Statistics

Bayesian framework is based on a simple rule of probability, which is known as Bayesian formula:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}, \tag{1.30}$$

where A and B are events of interest with probability $P(A)$ and $P(B) > 0$. The conditional probability $P(A|B)$ indicates the probability of observing event A when B is known, and vice versa.

Bayes' theorem interprets the above formula as follows: event A represents a hypothesis and event B is the observed evidence; the Bayesian rule establishes the relationship between the prior probability of the hypothesis $P(A)$ and the posterior probability of the hypothesis $P(A|B)$ after evidence B is given. Probabilities $P(B|A)$ and $P(B)$ are called likelihood and evidence, respectively. Using such interpretation,

the Bayes' theorem can be interpreted as

$$\text{Posterior probability } = \frac{\text{Likelihood}}{\text{Evidence}} \times \text{Prior probability.} \tag{1.31}$$

Let us now make it more specific. Assume we observed a set of data $Y = \{y_i\}_{i=1}^{L}$, where L is the number of observation and y_i is drawn from a probability distribution $p(\cdot|\boldsymbol{\theta})$ with unknown parameter vector $\boldsymbol{\theta}$. The probability of observing y_i is given by

$$y_i \sim p(y_i|\boldsymbol{\theta}), \quad i = 1, \cdots, L. \tag{1.32}$$

Parameter $\boldsymbol{\theta}$ is unknown but we have the hypothesis that $\boldsymbol{\theta}$ is distributed from probability $p(\cdot|\alpha)$ where α is called hyper-parameter. In such case, Eq. (1.31) can be written as

$$p(\boldsymbol{\theta}|Y, \alpha) = \frac{p(Y|\boldsymbol{\theta}, \alpha)}{p(Y|\alpha)} \times p(\boldsymbol{\theta}|\alpha), \tag{1.33}$$

where the evidence term $p(Y|\alpha)$ is known as the marginal likelihood, which is the probability of observing data Y marginalized over all possible $\boldsymbol{\theta}$,

$$p(Y|\alpha) = \int_{\boldsymbol{\theta}} p(Y|\boldsymbol{\theta}, \alpha) p(\boldsymbol{\theta}|\alpha) d\boldsymbol{\theta}. \tag{1.34}$$

We will discuss the marginal likelihood in more details later in Chapter 4.

1.7.2 Prior

Prior information is important in Bayesian statistics. It indicates the uncertainty about the quantity of interest before we perform any analysis of the observation. Prior knowledge is usually obtained from previous knowledge, such as from historical data and experiments. A strong prior, which is also called informative prior, provides specific information about the quantity. In such cases, it makes the posterior mainly dominated by the prior and be less affected by the data. The stronger the prior is, the more data would be needed for decreasing the impact of the prior.

However, sometimes, we expect a non-informative prior that has less influence on the analysis. To set up a weak prior belief, we usually adopt a prior that is rather flat over the parameter space, e.g., uniform distribution. The good news is that the influence of the prior will decrease as the number of observations increases. Note that this is the case even when we use a very informative prior. The large data set will gradually correct the belief and eventually give us the correct posterior even if the prior available is significantly wrong!

1.7.3 Monte Carlo Integrations

Bayesian inference involves calculating integrals which are usually analytically intractable. An efficient way to solve this problem is to perform numerical approximation. Monte Carlo (MC) methods are a class of methods generating repeated random sampling to obtain the numerical approximation of the intractable integration. It was invented by Stanislaw Ulam in the late 1940s. Combined with Metropolis algorithm [15], which is the original version of Markov Chain Monte Carlo (MCMC) methods, it is developed slowly until Hastings [16] extended it to a more general case in 1970. However, the methods can be applied to various areas only in theory because of the lack of powerful computation machines. The situation began to changes in 1984 when Geman proposed a particle MCMC (known as the Gibbs Sampler) for image processing. It was not until 1990 that the paper of Gelfand and Smith [17] raised the awareness of MC methods to the statistical community. After that, MC methods gained broad applicability and became one of the most important tools for computer simulations.

MC methods are used to approximate integrals that are not analytically solvable. To illustrate this, let $\pi(\cdot)$ denote a probability measure which is defined on a measurable space (E, ϵ). There is an integrable function $f(\cdot)$ of interest and our goal

is to estimate the integral with respect to $\pi(\cdot)$:

$$E_\pi[f(X)] = \int_E f(x)\pi(dx) \tag{1.35}$$

Performing MC methods to approximate this integral, we draw independent samples $\{x_i\}_{i=1}^K$ from probability distribution $\pi(\cdot)$, which yields

$$\hat{E}_\pi[f(X)] = \frac{1}{K}\sum_{i=1}^{K} f(x_i). \tag{1.36}$$

According to Strong Law of Large Numbers, we know that the estimator $\hat{E}_\pi[f(X)] \to E_\pi[f(X)]$ almost surely as $K \to \infty$, as long as the expectation is finite.

When performing MC computational methods to approximate intractable integrals, an essential criterion for evaluating the effectiveness of a method is the variance of the estimator.

Suppose we have samples $\{x_i\}_{i=1}^K$ draw from density π of interest, if the variance of $f(X)$ with respect to π is finite, say σ_f^2, the Central Limit Theorem tells us that the estimator $\hat{E}_\pi[f(X)]$ is asymptotically normal,

$$\lim_{K\to\infty} \sqrt{K}(\hat{E}_\pi[f(X)] - E_\pi[f(X)]) \to \mathcal{N}(0, \sigma_f^2). \tag{1.37}$$

Hence, the variance of the estimator is given by σ_f^2/K.

1.7.4 Markov Chain Monte Carlo Methods

Eq. (1.36) illustrates how to simulate independent and identically distributed (i.i.d.) random variables from $\pi(\cdot)$ to approximate integral $\pi(f)$. However, when performing Bayesian inference, the posterior probability is not always analytical and sometimes cannot be sampled from directly. An efficient method that helps us get rid of this problem is to set up a Markov chain that has the posterior as its stationary distribution. In this section, we will give a brief introduction to MCMC methods. A comprehensive description of the methods, techniques together with further references, can be found

in Robert and Casella [18].

A Markov chain is a sequence of variables $\{x_i\}_{i=1}^K$ with Markov property:

$$p(x_i|x_{1:i-1}) = p(x_i|x_{i-1}), \quad i = 2, \cdots, K, \tag{1.38}$$

where $x_i \in E$ denotes the state at time i. The joint distribution of a Markov Chain is fully determined by the transition kernel $p(x_i|x_{i-1})$. Density π is considered to be the invariant distribution of transition $p(x_i|x_{i-1})$ if it satisfies:

$$\pi(x_i) = \int_E \pi(x_{i-1})p(x_i|x_{i-1})dx_{i-1}, \tag{1.39}$$

which can be written as $\pi = \pi p$.

For a Markov chain with invariant distribution $\pi(\cdot)$, an important property is reversible. A Markov chain is reversible if there exists relationship between a probability distribution $\pi(\cdot)$ and transition kernel $p(x_{i-1}|x_i)$

$$\pi(x_i)p(x_{i-1}|x_i) = \pi(x_{i-1})p(x_i|x_{i-1}), \quad \text{for any } x_i, x_{i-1} \in E. \tag{1.40}$$

The property of reversibility is also known as detailed balance. Eq. (1.41) shows that probability distribution $\pi(\cdot)$ is a Markov chain's invariant distribution if the chain is reversible with respect to $\pi(\cdot)$.

$$\begin{aligned} \int_E \pi(x_{i-1})p(x_i|x_{i-1})dx_{i-1} &= \int_E \pi(x_i)p(x_{i-1}|x_i)dx_{i-1} \\ &= \pi(x_i)\int_E p(x_{i-1}|x_i)dx_{i-1} \\ &= \pi(x_i). \end{aligned} \tag{1.41}$$

We can see from Eq. (1.39) and (1.41) that the key to constructing a Markov chain with target $\pi(\cdot)$ is to find a proper transition kernel $p(x_i|x_{i-1})$. Metropolis-Hastings (M-H) algorithm is a sufficient way to achieve it.

1.7.5 Metropolis-Hastings Sampler

In order to obtain sequence $\{x_i\}_{i=1}^{K}$ with invariant distribution $\pi(\cdot)$, we apply the MCMC Metropolis-Hastings sampling in the following procedure.

Algorithm 1 MCMC Metropolis-Hastings Sampler

1: Set initial guess of the parameters x_1
2: **For** $i = 2$ **to** K **do**
3: Draw x^* from a proposal density $q\Big(x^*|x_i\Big)$
4: Calculate probability $\pi(x^*)$
5: Evaluate the acceptance probability of the move $\gamma\Big(x_i, x^*\Big)$ with

$$\gamma\Big(x_i, x^*\Big) = \texttt{min}\left\{1, \frac{\pi(x^*)}{\pi(x_i)}\frac{q\Big(x_i|x^*\Big)}{q\Big(x^*|x_i\Big)}\right\}$$

6: Generate $u \sim U_{(0,1]}$
7: **If** $u \leq \gamma\Big(x_i, x^*\Big)$, **Then**
8: Set $x_{i+1} = x^*$
9: **Else**
10: Set $x_{i+1} = x_i$
11: **End If**
12: **End For**
13: **Return** $\{x_i\}, i = 1, ..., K$

The efficiency of the algorithm depends on the proposal we choose. A poorly designed proposal would lead to low acceptance rate and end with a poor mixing of Markov chain. More details can be found in section 2.4.

1.7.6 Importance Sampling

An alternative way to sample from posterior is Importance Sampling (IS) which is capable of simulating from the target distribution by making use of proposal distribution. This method is useful especially when it is hard to sample from the target. Instead of drawing samples from the prior, IS draws points from an importance

function $q(\cdot)$. Thus Eq. (1.33) can be rewritten as

$$p(\boldsymbol{\theta}|Y,\alpha) = \frac{p(Y|\boldsymbol{\theta},\alpha)p(\boldsymbol{\theta}|\alpha)}{p(Y|\alpha)q(\boldsymbol{\theta})}q(\boldsymbol{\theta}). \tag{1.42}$$

Drawing i.i.d. samples $\{\boldsymbol{\theta}_i\}_{i=1}^{N_p}$ from the proposal $q(\cdot)$, the weight w_i of each sample is given by

$$w_i = \frac{p(Y|\boldsymbol{\theta}_i,\alpha)p(\boldsymbol{\theta}_i|\alpha)}{q(\boldsymbol{\theta}_i)}, \quad i = 1,\cdots,N_p, \tag{1.43}$$

where N_p is sample size.

The weighted points obtained from IS sampler provide an approximation to the posterior distribution

$$p(\boldsymbol{\theta}|Y,\alpha) \sim \sum_{i=1}^{N_p} w_i \delta_{\boldsymbol{\theta}_i}. \tag{1.44}$$

Importance Sampling is of wide applicability on the inference of normalized constant which refers to the marginal likelihood $p(Y|\alpha)$. After plugging in proposal density $q(\cdot)$, the marginal likelihood $p(Y|\alpha)$ in Eq. (1.34) is given by

$$p(Y|\alpha) = \int_{\boldsymbol{\theta}} \frac{p(Y|\boldsymbol{\theta},\alpha)p(\boldsymbol{\theta}|\alpha)}{q(\boldsymbol{\theta})}q(\boldsymbol{\theta})d\boldsymbol{\theta}. \tag{1.45}$$

Therefore, the average of the weight is an unbiased estimate of the marginal likelihood

$$\hat{p}(Y|\alpha) = \frac{1}{N_p}\sum_{i=1}^{N_p} \frac{p(Y|\boldsymbol{\theta}_i,\alpha)p(\boldsymbol{\theta}_i|\alpha)}{q(\boldsymbol{\theta}_i)} = \frac{1}{N_p}\sum_{i=1}^{N_p} w_i. \tag{1.46}$$

For the goal of approximating posterior distribution, IS is very useful for dealing with low-dimensional problems. However, it is not preferable over MCMC for complicated problems. It is important to note that MCMC does not directly give an estimate of the marginal likelihood. However, IS can provide an unbiased estimate of the marginal likelihood. The choice of the proposal is essential to implement IS since the variance of the estimator is governed by the distance of the proposal and the posterior distribution [19]. Therefore, in practice, it is challenging to find an appropriate proposal, especially for high dimensional problems. It can be seen from Eq. (1.45) that an ideal choice of

importance proposal would be the target,

$$q(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|Y, \alpha) \propto p(Y|\boldsymbol{\theta}, \alpha)p(\boldsymbol{\theta}|\alpha). \tag{1.47}$$

In such case, the weight is constant which equals $\frac{1}{N_p}$. However, the target distributions are usually too complicated to be sampled from directly. Alternatively, we can choose proper proposals that are close to the target distribution. The effective sample size (ESS) is a useful criterion for judging the distance between the proposal and the target

$$ESS = \frac{N_p}{1 + \text{var}(\tilde{w}_i)}, \tag{1.48}$$

where $\tilde{w}_i$ is the normalized weight given by

$$\tilde{w}_i = \frac{w_i}{\sum_{j=1}^{N_p} w_j}, \quad i = 1, \cdots, N_p. \tag{1.49}$$

Neal [20] introduced an adaptive tempering technique, known as Annealed Importance Sampling (AIS), to approximate the intractable target distribution $\pi(\cdot)$. AIS makes lower variance estimators than IS by introducing intermediate distributions that bridge between the proposal and the target. The intermediate distribution $\pi_t(\cdot)$ is defined as following:

$$\pi_t(\boldsymbol{\theta}|Y) = p(\boldsymbol{\theta})^{1-t} p(\boldsymbol{\theta}|Y)^t, \quad 0 \leq t \leq 1.$$

A set of distributions $\{\pi_t(\boldsymbol{\theta}|Y)\}$ act as the intermediate distributions to build a smooth path between the prior and the posterior distribution. To facilitate such annealing scheme, we then introduce Markov transition kernel to jump from one intermediate distribution to the next. AIS gives a lower variance estimate than IS of both the posterior and the marginal likelihood. We will look at this in section 4.6.

1.7.7 Pseudo Marginal

Andrieu and Roberts [21] formalised a method known as Grouped Independence Metropolis-Hastings (GIMH), which was originally introduced by Mark Beaumont [22]. The basic idea of GIMH is described as following:

Suppose we have a set of observed data Y, which is generated by a model associated with parameters vector $\boldsymbol{\theta}$ and latent variable vector X. The target distribution of interest is $\pi(\boldsymbol{\theta}|Y)$, which is the posterior distribution of $\boldsymbol{\theta}$ for given data Y. A standard approach to obtaining the approximation of intractable target $\pi(\boldsymbol{\theta}|Y)$ is to set up an MCMC simulation (see Alg. 1) on the joint parameter space $(\boldsymbol{\theta}, X)$. However, such an algorithm usually turns out to be inefficient. Since the latent variable X is usually defined on high dimensional space which results in low acceptance rate. A straightforward way to address this problem is to apply an approach that can update the variables sequentially. However, such approach can be quite slow if $\boldsymbol{\theta}$ and X are highly dependent in the posterior.

The pseudo-marginal is a useful trick that helps us get rid of this. The basic idea is to approximate the 'ideal' algorithm that targets the posterior

$$\pi(\boldsymbol{\theta}|Y) \propto p(\boldsymbol{\theta})\, p(Y|\boldsymbol{\theta})$$

with an unbiased approximation. Andrieu and Roberts [21] have established that if an unbiased approximation is used to approximate the posterior $\pi(\boldsymbol{\theta}|Y)$ when evaluating the acceptance probability, one will end up with the right target distribution. A sufficient way to obtain such an unbiased approximation of the target density is to make use of IS strategy. By introducing integer $N_p \geq 1$ and importance proposal $q_{\boldsymbol{\theta}}(\cdot)$, the target density can be approximated by

$$\hat{\pi}(\boldsymbol{\theta}|Y) = \frac{1}{N_p} \sum_{i=1}^{N_p} \frac{\pi(\boldsymbol{\theta}, X_i|Y)}{q_{\boldsymbol{\theta}}(X_i)}, \quad \text{with } X_i \sim q_{\boldsymbol{\theta}}(\cdot). \tag{1.50}$$

Since the hidden variable X is usually high dimensional, sampling from IS might

not be the most efficient way to update X. However, it does help us overcome the challenge of exploring the joint space where X and $\boldsymbol{\theta}$ are highly dependent.

Andrieu and Vihola [23] state that the pseudo-marginal MCMC mixes well only in the case that we have a low variance estimator of the posterior. A lower variance estimator would improve the mixing of the chain. It is important to note that if X is high dimensional, it would be difficult to obtain such a low estimator by plugging IS. If X is a time series, we can use a particle filter (PF) to get around this.

1.8 Thesis Outline

The remainder of the thesis is structured as follows. In Chapter 2, we apply the Monte Carlo methods to study super coarse-grained models for unentangled polymer dynamics. Trajectories of the centre of mass of polymer chains extracted from molecular dynamics simulations are used as observations of the coarse-grained models. MCMC method is implemented to perform parameter estimates and uncertainty assessments. We will show that MCMC algorithm works well for our coarse-grained models. By introducing the adaptive scheme, the MCMC samples have lower correlation time compared with the standard MCMC paradigm. We investigate a set of models with different complexity to reproduce the dynamics of the center of mass of single chains in MD simulations and find that 3-bead model is adequate to represent the time-dependent diffusion coefficient at medium and large time scales.

In Chapter 3, we introduce particle MCMC (PMCMC) to study nonlinear and non-Gaussian system. We illustrate how the observation error and the number of particles impact the efficiency of particle MCMC. A super coarse-grained model characterized by Generalized Langevin Equation is introduced to study the chain center of mass diffusion.

We use the slip-spring model combined with Bayesian statistics to study polymer entanglements in Chapter 4. Basic concepts of slip-spring model are adapted to

describe the effects of entanglements, in which a set of slip-springs are introduced to the target polymer for detecting the entanglements along its backbone. To estimate the number of entanglements, Bayesian statistical methods, including RJMCMC, AIS RJMCMC and SMC are implemented for model comparison and parameter estimates. Our approach can detect the number of entanglements, their locations and strengths of the entanglement effects for both concatenated polymer rings and central segment of linear polymers.

Finally, in Chapter 5, we provide concluding remarks and discuss the limitations of our methodologies, and point out possible future research directions.

Chapter 2

Application of Markov Chain Monte Carlo Methods on Studying Coarse-Grained Models

In this chapter, we aim to develop a novel coarse-grained model to simulate entangled polymer melts. We are going to look at single-particle dynamics on the level of individual molecules. Unlike other coarse-grained approaches which consider both inter- and intra-chain interactions, single-particle tracking focuses on only the chain center of mass diffusion. In this way, we can significantly reduce the number of free parameters in the coarse-grained model.

We first derive a simple stochastic model to describe the dynamics of the centers of mass of the polymer chains. Next, we apply MCMC methods to draw inferences on the unknown model parameters for given center of mass trajectories extracted from MD simulations. Kalman filter is introduced to evaluate the likelihood function. To explore the parameter space, Metropolis-Hastings sampler is used to propose new samples for constructing MCMC paradigm. Adaptive techniques is introduced to tune the Metropolis-Hastings sampler to improve the efficiency of the MCMC sampler.

2.1 Introduction

Polymer melts and solutions show complicated dynamic behaviour in response to deformation. The equations describing this phenomenon are usually nonlinear and include many parameters, which renders them solvable only in either a numerical or analytical manner by using drastic approximations. This is our first look at Bayesian statistics and Monte Carlo methods in polymer applications. We start with a relatively simple model that focuses on the dynamics of the center of mass of liner polymers.

Two of the most important dynamic properties used to characterize the properties of a dynamic system are mean squared displacement (MSD) and velocity autocorrelation function (VAF). The mean squared displacement of the chain center of mass is defined as

$$g_3(t) = \langle |\mathbf{R}_{cm}(t) - \mathbf{R}_{cm}(0)|^2 \rangle, \tag{2.1}$$

where $\langle \cdot \rangle$ denotes averaging over all the atoms and the center of mass at time t is given by Eq. (1.5). The displacement of center of mass $\mathbf{R}_{cm}(t) - \mathbf{R}_{cm}(0)$ in time can be described as an integral of its velocity $\mathbf{v}(t)$:

$$\mathbf{R}_{cm}(t) - \mathbf{R}_{cm}(0) = \int_0^t \mathbf{v}(u)du. \tag{2.2}$$

The square of this displacement is

$$\left(\mathbf{R}_{cm}(t) - \mathbf{R}_{cm}(0)\right)^2 = \int_0^t \int_0^t \mathbf{v}(u)\mathbf{v}(u')dudu'. \tag{2.3}$$

Defining an auxiliary variable $u' = u+s$, we obtain the following formula by integrating over u and taking the ensemble average,

$$g_3(t) = 2\int_0^t (t-s)a(s)ds, \tag{2.4}$$

where $a(s)$ is the velocity autocorrelation function defined as $a(s) = \langle \mathbf{v}(0)\mathbf{v}(s) \rangle$. Re-

arranging Eq. (2.4) yields,

$$g_3(t) = 2t \int_0^t a(s)ds - 2 \int_0^t sa(s)ds. \tag{2.5}$$

The MSD of the center of mass describing the chain diffusivity consists of two parts. The first term on the right includes the time t explicitly. The velocity autocorrelation function $a(t)$ decays to zero when t is large which leads to the convergence of the integral $\int_0^t a(s)ds$ as $t \to \infty$. Since the second term also integrates to a fixed value for large t, we have

$$D = \lim_{t\to\infty} \frac{1}{3} \int_0^t a(s)ds, \qquad C = \lim_{t\to\infty} \int_0^t sa(s)ds, \tag{2.6}$$

where D is the diffusion coefficient. The limiting behaviour of diffusion at large time scales thereby can be written as

$$g_3(t) \approx 6Dt. \tag{2.7}$$

The chain center of mass diffusion coefficient is given by

$$D = \lim_{t\to\infty} \frac{g_3(t)}{6t}. \tag{2.8}$$

It is important to note that the time dependence of the VAF $a(t)$ cannot be ignored for small t. We can see from Eq. (2.5) that the MSD must depend on VAF at short time scales. At large time scales, the motion of the molecules becomes simple Brownian motion where the velocity autocorrelation functions have decayed to zero, i.e., the molecules have 'forgotten' the speed and direction they began with at $t = 0$.

2.2 Single-Particle Dynamics

Let us consider the simplest model that describes the motion of a single-particle using Ornstein–Uhlenbeck process [24]. An Ornstein–Uhlenbeck process, x_t, satisfies

the following stochastic differential equation:

$$dx_t = a(x^0 - x_t)\,dt + \sigma\,dW_t$$

where $a > 0$, x^0, and $\sigma > 0$ are parameters and W_t denotes the Wiener process with mean and variance given by

$$\langle W_t \rangle = W_0,\ \mathrm{Var}(W_t) = t.$$

The drift term depends on the current state of the process and the parameter x^0 in the drift term is known as the long-term mean which represents the equilibrium level of the process. If the current state of the process is less than the long-term mean x^0, the drift will be positive; if the current value of the process is greater than x_0, the drift will be negative. Suppose $\mathbf{R}_t$ indicates the position of a single particle at time t, the equation of motion of the particle is given as

$$d\mathbf{R}_t = -a\mathbf{R}_t dt + \sqrt{b} d\mathbf{W}_t, \tag{2.9}$$

where the long-term mean is considered as $\mathbf{R}^0 = \mathbf{0}$ and drift term is proportional to the state $\mathbf{R}_t$ itself, b determines the dynamic noise strength, and $\mathbf{W}_t$ is Wiener process.

We aim at evaluating the posterior distribution of the model parameter $\boldsymbol{\theta} = \{a, b\}$ for given observed data. The position of the diffusing particle can be modelled by discretizing Eq. (2.9) using Euler approximation with time step Δ_t,

$$\mathbf{R}_n - \mathbf{R}_{n-1} = -a\mathbf{R}_{n-1}\Delta_t + \sqrt{b\Delta_t}\boldsymbol{\eta}_n,$$

where $\boldsymbol{\eta}_n$ is a set of independent random variables drawn from the standard normal distribution, $\boldsymbol{\eta}_n \sim N(\mathbf{0}, \boldsymbol{E})$, $\boldsymbol{E}$ denotes identity matrix. Rearranging the above equation, we have

$$\mathbf{R}_n = (1 - a\Delta_t)\mathbf{R}_{n-1} + \sqrt{b\Delta_t}\boldsymbol{\eta}_n, \tag{2.10}$$

from which $\mathbf{R}_n$ can be considered as a first-order autoregressive (AR) process, where the current state only depends on the previous state and an extra noise term. One has to be careful in the choice of Δ_t which is essential for the stability of the Euler approximation and the second order stationarity of the process. In order to prevent the amplification of the errors in the iteration process, it requires $|1 - a\Delta_t| \leq 1$ which implies $\Delta_t \leq \frac{2}{a}$.

The transition density of the process with given $\{a, b\}$ can be written as

$$p(\mathbf{R}_n|\mathbf{R}_{n-1}, a, b) = \frac{1}{\sqrt{2\pi b\Delta_t}}\exp\left(-\frac{(\mathbf{R}_n - (1 - a\Delta_t)\mathbf{R}_{n-1})^2}{2b\Delta_t}\right). \tag{2.11}$$

Therefore, the distribution of $\mathbf{R}_n$ with given $\mathbf{R}_{n-1}$ is Gaussian

$$\mathbf{R}_n \sim \mathcal{N}\left((1 - a\Delta t)\mathbf{R}_{n-1}, b\Delta_t\right).$$

Assume we have a time series of particle positions $Y = \{y_n\}_{n=1}^L$, we will evaluate the likelihood function in order to use Bayes' formula shown in Eq. (1.33) to obtain posterior distribution of the model parameters a and b. The likelihood function is computed from the joint probability density of $p(y_{1:L}|a, b)$. Note that the time series is Markov, the likelihood function will be expressed as follows,

$$\begin{aligned} p(Y|a, b) &= p(y_{2:L}|y_1, a, b)p(y_1|a, b) \\ &= p(y_1|a, b)\prod_{n=2}^{L} p(y_n|y_{n-1}, a, b) \\ &= p(y_1|a, b)(2\pi b\Delta_t)^{-L/2}\exp\left(-\sum_{n=2}^{L}\frac{(y_n - \hat{y}_n|y_{n-1})^2}{2b\Delta_t}\right), \end{aligned}$$

where $\hat{y}_n|y_{n-1}$ is the expectation of y_n conditioning on y_{n-1}, a and b,

$$\hat{y}_n|y_{n-1} = E\left[y_n|y_{n-1}, a, b\right]$$

Recall that we have demonstrated how to evaluate posterior density in Eq. (1.33).

The posterior density $\pi(a, b|Y)$ is given by

$$\pi(a, b|Y) \sim p(Y|a, b)P(a, b). \tag{2.12}$$

where $P(a, b)$ denotes the joint prior of (a, b). In some cases, one can choose conjugate priors to make the posterior tractable, see section 4.4. If the posterior $\pi(a, b|Y)$ is tractable, one can sample (a, b) from the target directly. Otherwise, to construct MCMC based on M-H Sampler (Alg. 1) is a useful method to obtain sample from the intractable target.

This simple model considers the motion of a polymer as a single-particle diffusion and ignores the memory effects and interactions with surrounding polymers. To capture more details of the dynamics of the polymers, we are going to look at a more complicated model by assuming that a set of hidden variables are constantly interacting with the target observable.

2.3 Single-Particle Dynamics with Hidden Variables

We represent a polymer chain with a single particle where the coordinate of the particle is the chain center of mass. The effects of the surrounding media and neighbouring chains could be described using a set of invisible particles. Suppose we have $N + 1$ particles with coordinates $\{\boldsymbol{R}_i\}_{i=0}^{N}$, the first particle with coordinate $\boldsymbol{R}_0$ is observable representing the target chain center of mass. The rest of the particles are latent variables and connected with $\boldsymbol{R}_0$ by harmonic springs. We assume that the probability of future state of the system depends only on the current state and not on the sequence of states that preceded it. Such models are known as Hidden Markov Models (HMMs) in which the sequence of states that 'generate' the observable are hidden from the observer.

HMMs are widely used in speech recognition systems, computational molecular

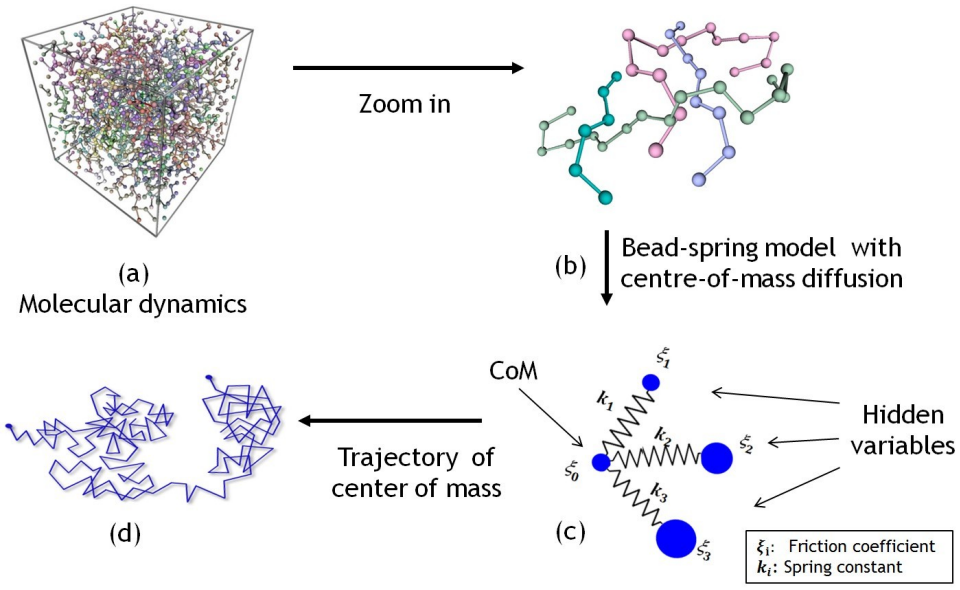


Figure 2.1: (a) Molecular dynamics simulation-box snapshot; (b) a snapshot of linear polymer chain (green) and its neighbouring chains in one of the configurations; (c) Coarse-grained model; (d) Trajectory of the chain center of mass

biology and other areas of artificial intelligence and pattern recognition. The equation of motion of the $(N+1)$-particle system is given by

$$\begin{cases} \xi_i d\boldsymbol{R}_i = k_i(\boldsymbol{R}_0 - \boldsymbol{R}_i)dt + \sqrt{2\xi_i}d\boldsymbol{W}_i, \\ \xi_0 d\boldsymbol{R}_0 = \sum_{i=1}^{N} k_i(\boldsymbol{R}_i - \boldsymbol{R}_0)dt + \sqrt{2\xi_0}d\boldsymbol{W}_0, \end{cases} \tag{2.13}$$

where k_i is the spring constant, ξ_i is the friction coefficient and $\boldsymbol{W}_i$ is Wiener process. Rearranging the above equations in vector form yields

$$d\mathbf{R} = -\mathbf{A}\mathbf{R}dt + \mathbf{B}d\mathbf{W}, \tag{2.14}$$

where

$$\mathbf{A} = \begin{pmatrix} \sum_{i=1}^{N} k_i/\xi_0 & -k_1/\xi_0 & \cdots & \cdots & -k_N/\xi_0 \\ -k_1/\xi_1 & k_1/\xi_1 & & & \\ \vdots & & \ddots & & 0 \\ \vdots & 0 & & \ddots & \\ -k_N/\xi_N & & & & k_N/\xi_N \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} \sqrt{2/\xi_0} & & & & \\ & \sqrt{2/\xi_1} & & 0 & \\ & & \ddots & & \\ & 0 & & \ddots & \\ & & & & \sqrt{2/\xi_N} \end{pmatrix}.$$

Let us start from the simplest model with $\mathbf{R} = (\boldsymbol{R}_0, \boldsymbol{R}_1)^T$, where the connectivity matrix is of the form

$$\mathbf{A} = \begin{pmatrix} k_1/\xi_0 & -k_1/\xi_0 \\ -k_1/\xi_1 & k_1/\xi_1 \end{pmatrix}.$$

We can apply eigen decomposition to matrix $\mathbf{A}$ as what we have done in section 1.4. The diagonal matrix $\mathbf{\Lambda}$ consists of eigenvalues of $\mathbf{A}$,

$$\mathbf{\Lambda} = \mathbf{Q}^{-1}\mathbf{A}\mathbf{Q}, \tag{2.15}$$

where the i^{th} column of $\mathbf{Q}$ is the eigenvector corresponding to the i^{th} eigenvalue Λ_{ii}. Matrices $\mathbf{\Lambda}$, $\mathbf{Q}$ and $\mathbf{Q}^{-1}$ are given by

$$\mathbf{\Lambda} = \begin{pmatrix} 0 & 0 \\ 0 & \frac{k_1(\xi_0+\xi_1)}{\xi_0\xi_1} \end{pmatrix}, \mathbf{Q} = \begin{pmatrix} 1 & -\frac{\xi_1}{\xi_0+\xi_1} \\ 1 & \frac{\xi_0}{\xi_0+\xi_1} \end{pmatrix}, \mathbf{Q}^{-1} = \begin{pmatrix} \frac{\xi_0}{\xi_0+\xi_1} & \frac{\xi_1}{\xi_0+\xi_1} \\ -1 & 1 \end{pmatrix}.$$

Applying the linear operator $\mathbf{Q}^{-1}$ to $\mathbf{R}$, we obtain $\mathbf{X} = \mathbf{Q}^{-1}\mathbf{R}$ where $\mathbf{X}$ are the normal coordinates of the model,

$$\boldsymbol{X}_0 = \frac{\xi_0\boldsymbol{R}_0 + \xi_1\boldsymbol{R}_1}{\xi_0+\xi_1}, \quad \boldsymbol{X}_1 = \boldsymbol{R}_1 - \boldsymbol{R}_0.$$

Substitute $\mathbf{R} = \mathbf{Q}\mathbf{X}$ and $\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{-1}$ into Eq. (2.14) yields

$$d\boldsymbol{X}_0 = \sqrt{2/\zeta_0}d\boldsymbol{W}_0', \tag{2.16}$$

$$d\boldsymbol{X}_1 = -\frac{1}{\tau_1}\boldsymbol{X}_1 dt + \sqrt{2/\zeta_1}d\boldsymbol{W}_1', \tag{2.17}$$

where $\zeta_0 = \xi_0 + \xi_1$, $\zeta_1 = \frac{\xi_0\xi_1}{\xi_0+\xi_1}$, $\tau_1 = \frac{\zeta_1}{k_1}$ and $\boldsymbol{W}_i'$, $i = 1, 2$, is Wiener process. We can see from Eq. (2.16) that the first mode corresponds to the center-of-mass motion which is governed by Wiener process with diffusion coefficient $D_0 = 1/\zeta_0$. Eq. (2.17) indicates that $\boldsymbol{X}_1$ is characterized by bond vector which can be expressed as Ornstein-Uhlenbeck process with relaxation time τ_1 and diffusion coefficient $D_1 = 1/\zeta_1$. It is

straightforward to write down the solutions of these two stochastic processes

$$\boldsymbol{X}_0(t) = \boldsymbol{X}_0(0) + \sqrt{2D_0}\int_0^t d\boldsymbol{W}_0' = \boldsymbol{X}_0(0) + \sqrt{2D_0}\boldsymbol{W}_0'(t), \tag{2.18}$$

$$\boldsymbol{X}_1(t) = \boldsymbol{X}_1(0)\exp(-t/\tau_1) + \sqrt{2D_1}\int_0^t \exp\left(-\frac{t-t'}{\tau_1}\right) d\boldsymbol{W}_1'. \tag{2.19}$$

The mean-square displacements of the modes are given by

$$\langle(\boldsymbol{X}_0(t) - \boldsymbol{X}_0(0))^2\rangle = 2D_0 t, \tag{2.20}$$

$$\langle(\boldsymbol{X}_1(t) - \boldsymbol{X}_1(0))^2\rangle = 2D_1\tau_1\left(1 - e^{-t/\tau_1}\right). \tag{2.21}$$

Note that $\mathbf{X}$ and $\mathbf{R}$ are related by a linear operator $\mathbf{Q}^{-1}$, $\boldsymbol{R}_0(t)$ and $\boldsymbol{R}_1(t)$ thereby can be expressed in terms of $\boldsymbol{X}_0(t)$ and $\boldsymbol{X}_1(t)$,

$$\boldsymbol{R}_0(t) = \boldsymbol{X}_0(t) - \frac{\xi_1}{\xi_0+\xi_1}\boldsymbol{X}_1(t), \quad \boldsymbol{R}_1(t) = \boldsymbol{X}_0(t) + \frac{\xi_0}{\xi_0+\xi_1}\boldsymbol{X}_1(t)$$

The MSD of $\boldsymbol{R}_0$ is given by

$$\left\langle(\boldsymbol{R}_0(t) - \boldsymbol{R}_0(0))^2\right\rangle = 2D_0 t + 2D_1\tau_2\left(\frac{\xi_1}{\xi_0+\xi_1}\right)^2\left(1 - e^{-t/\tau_1}\right). \tag{2.22}$$

For the cases of $N > 1$, applying diagonal decomposition $\mathbf{A} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{-1}$, substituting $\mathbf{R}{=}\mathbf{Q}\mathbf{X}$ into Eq. (2.14) and rearranging the equation, we obtain the diagonal stochastic differential equation

$$d\mathbf{X} = -\boldsymbol{\Lambda}\mathbf{X}dt + \mathbf{D}d\mathbf{W}, \tag{2.23}$$

where $\mathbf{D}{=}\mathrm{diag}(\sqrt{2d_0}, \sqrt{2d_1}, ..., \sqrt{d_N})$, $d_i = \frac{1}{2}\sum_{j=0}^{N}\mathbf{B}'(i,j)^2$, $\mathbf{B}' = \mathbf{Q}^{-1}\mathbf{B}$. Note that the first eigenvalue is $\lambda_0 = 0$, and the corresponding eigenvector is $(1,1,1,...,1)^T$, which yields

$$\left\langle(\boldsymbol{X}_0(t) - \boldsymbol{X}_0(0))^2\right\rangle = \frac{2}{\sum_{i=0}^{N}\xi_i}t. \tag{2.24}$$

For the mode numbers i, we obtain solution of MSD of mode $\boldsymbol{X}_i(t)$ with relaxation time $\tau_i = 1/\Lambda_{ii}$

$$\left\langle (\boldsymbol{X}_i(t) - \boldsymbol{X}_i(0))^2 \right\rangle = 2d_i\tau_i \left(1 - e^{-t/\tau_i}\right), i = 1, ..., N. \tag{2.25}$$

Recall that **R=QX**, coordinate $\mathbf{R}$ can be written in terms of $\mathbf{X}$,

$$\boldsymbol{R}_i = \sum_{j=0}^{N} Q_{ij}\boldsymbol{X}_j.$$

Therefore, the MSD of $\boldsymbol{R}_i$ is given by

$$\begin{aligned}
&\left\langle (\boldsymbol{R}_i(t) - \boldsymbol{R}_i(0))^2 \right\rangle \\
&= \sum_{j=0}^{N} Q_{ij}^2 \left\langle (\boldsymbol{X}_j(t) - \boldsymbol{X}_j(0))^2 \right\rangle + \sum_{j\neq k} Q_{ij}Q_{ik} \left\langle (\boldsymbol{X}_j(t) - \boldsymbol{X}_j(0)) (\boldsymbol{X}_k(t) - \boldsymbol{X}_k(0) \right\rangle \\
&= \frac{2}{\sum_{j=0}^{N}\xi_j} t + \sum_{j=1}^{N} 2Q_{ij}^2 d_j\tau_j \left(1 - e^{-t/\tau_j}\right).
\end{aligned} \tag{2.26}$$

The second summation $\sum_{j\neq k} Q_{ij}Q_{ik} \left\langle (\boldsymbol{X}_j(t) - \boldsymbol{X}_j(0)) (\boldsymbol{X}_k(t) - \boldsymbol{X}_k(0) \right\rangle$ vanishes since the normal modes $\boldsymbol{X}_j$ and $\boldsymbol{X}_k$ $(j \neq k)$ are independent of each other, which yields,

$$\langle (\boldsymbol{X}_j(t) - \langle \boldsymbol{X}_j(t) \rangle)(\boldsymbol{X}_k(t) - \langle \boldsymbol{X}_k(t) \rangle) \rangle = 0, \quad j \neq k. \tag{2.27}$$

Once the model parameters are available, Eq. (2.26) can be used to compute the time-dependent diffusion coefficient $D(t)$ in Eq. (2.8).

Let us consider using Bayesian inference to draw inference on model parameters $\left(\{k_i\}_{i=1}^N, \{\xi_i\}_{i=0}^N\right)$. Applying Euler approximation to Eq. (2.14) and rearranging it, we have

$$\mathbf{R}_n = \boldsymbol{M}\mathbf{R}_{n-1} + \boldsymbol{\eta}_n, \quad n = 1, ..., N, \tag{2.28}$$

where $\boldsymbol{M} = \boldsymbol{E} - \boldsymbol{A}\Delta_t$ is the transition matrix, $\boldsymbol{E}$ denotes the identity matrix and $\boldsymbol{\eta}_n$ is a Gaussian vector with zero mean and covariance matrix $\boldsymbol{S} = \boldsymbol{B}\boldsymbol{B}^T\Delta_t$.

Note that we observe only one of the particle coordinates. Without loss of generality, we assume that there exists observation error. The observation at the n^{th} time

step is given by

$$y_n = \boldsymbol{C}\mathbf{R}_n + \zeta_n, \tag{2.29}$$

where $\boldsymbol{C} = (\ 1 \quad 0 \quad \cdots \quad 0\ )$ is the observation operator and the observation error is thought of being Gaussian $\zeta_n \sim N(0, S_0)$ and time-independent

$$\langle \zeta_n \zeta_m \rangle = 0, \text{ for } n \neq m.$$

Kalman filter is useful to recursively estimate the state of the system and the uncertainty of the estimate [25] [26]. The forecast of state $\boldsymbol{R}_n^f$, uncertainty $\boldsymbol{P}_n^f$ and forecast of the observation y_n^f are as follows,

$$\begin{aligned} &\boldsymbol{R}_n^f = \boldsymbol{M}\boldsymbol{R}_{n-1}^a, \quad \boldsymbol{P}_n^f = \boldsymbol{M}\boldsymbol{P}_{n-1}^a\boldsymbol{M}^T + \boldsymbol{S}, \\ &y_n^f = \boldsymbol{C}\boldsymbol{R}_n^f, \end{aligned}$$

where $\boldsymbol{R}_{n-1}^a$ is the updated estimate of state and $\boldsymbol{P}_{n-1}^a$ is the updated uncertainty of the estimate at $n-1$. Using the difference between the observation y_n and the forecast of measurement y_n^f, we have

$$\boldsymbol{R}_n^a = \boldsymbol{R}_n^f + \boldsymbol{K}_n(y_n - y_n^f),$$

$$\boldsymbol{P}_n^a = \boldsymbol{P}_n^f - \boldsymbol{K}_n\boldsymbol{C}\boldsymbol{P}_n^f,$$

where $y_n - y_n^f$ indicates the discrepancy between the predicted measurement and the actual measurement. Matrix $\boldsymbol{K}_n$ is referred to as *gain*. The choice of $\boldsymbol{K}_n$ would be the one that minimizes the posterior error covariance which gives us the most reliable estimation. To derive such an optimal $\boldsymbol{K}_n$, a widely used approach is to calculate the expectation of the covariance estimate and take derivative of the trace with respect to $\boldsymbol{K}_n$. The optimal gain ensures that the matrix derivative equals zero, which yields

$$\begin{aligned}\boldsymbol{K}_n &= \boldsymbol{P}_n^f \boldsymbol{C}^T (\boldsymbol{C}\boldsymbol{P}_n^f \boldsymbol{C}^T + S_0)^{-1} \\ &= \frac{\boldsymbol{P}_n^f \boldsymbol{C}^T}{\boldsymbol{C}\boldsymbol{P}_n^f \boldsymbol{C}^T + S_0}.\end{aligned} \tag{2.30}$$

Further details about the Kalman gain can be found in [26] [27].

Assume the prior distribution of $\mathbf{R}_1$ is Gaussian $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the estimates of the state and its uncertainty are given by

$$\begin{aligned} y_1^f &\sim \mathcal{N}(\boldsymbol{C}\boldsymbol{\mu}, \boldsymbol{C}\boldsymbol{\Sigma}\boldsymbol{C}^T), \\ \boldsymbol{R}_1^f &= \boldsymbol{\mu}, \quad \boldsymbol{P}_1^f = \boldsymbol{\Sigma}. \end{aligned}$$

Note that the time series is Markov with given first observable y_1. Therefore the likelihood function for parameter set $\boldsymbol{\theta}$ with given observations $\{y_n\}_{n=1}^L$ can be computed using the following formula

$$P(y_{2:L}|\boldsymbol{\theta}, y_1) = \prod_{n=2}^{L} P(y_n|y_{n-1}, \boldsymbol{\theta}). \tag{2.31}$$

The conditional probability $P(y_n|y_{n-1}, \boldsymbol{\theta})$ is given by

$$P(y_n|y_{n-1}, \boldsymbol{\theta}) = \frac{1}{\sqrt{2\pi|Q_n|}} \exp\left[-\frac{1}{2}(y_n - \boldsymbol{C}\boldsymbol{M}\mathbf{R}_{n-1}^f)^T Q_n^{-1} (y_n - \boldsymbol{C}\boldsymbol{M}\mathbf{R}_{n-1}^f)\right], \tag{2.32}$$

where the innovation covariance Q_n is given by $Q_n = \boldsymbol{C}\boldsymbol{P}_n^f\boldsymbol{C}^T + S_0$. Substituting Eq. (2.32) into Eq. (2.31) and taking the logarithm of both sides yields

$$\begin{aligned}\log\ P(y_{1:L}|\boldsymbol{\theta}) &= \log p(y_1) + \sum_{n=2}^{L} \log P(y_n|y_{n-1}, \boldsymbol{\theta}) \\ &= \log p(y_1) - \frac{1}{2}\sum_{n=2}^{L}\Big[\log(2\pi) + \log|Q_n| + (y_n - \boldsymbol{C}\boldsymbol{M}\mathbf{R}_{n-1}^f)^T Q_n^{-1}(y_n - \boldsymbol{C}\boldsymbol{M}\mathbf{R}_{n-1}^f)\Big)\Big].\end{aligned} \tag{2.33}$$

Given parameter set $\boldsymbol{\theta}$, Alg. 2 demonstrates how to implement Kalman filter to evaluate the likelihood. We will show how to plug Kalman filter into MCMC in the following section.

Algorithm 2 Kalman Filter

Initialize $\mathbf{R}_1^f, \boldsymbol{P}_1^f$ and y_1^f
for $n = 1 : L$ **do**
  Calculate innovation covariance: $Q_n = \boldsymbol{C}\boldsymbol{P}_n^f\boldsymbol{C}^T + S_0, \quad e_n = y_n - \boldsymbol{C}\boldsymbol{M}\mathbf{R}_n^f$
  Evaluate Kalman gain: $\boldsymbol{K}_n = \boldsymbol{P}_n^f\boldsymbol{C}^T Q_n^{-1}$
  Update state: $\mathbf{R}_n^a = \mathbf{R}_n^f + \boldsymbol{K}_n e_n$
  Update estimate covariance: $\boldsymbol{P}_n^a = \boldsymbol{P}_n^f - \boldsymbol{K}_n\boldsymbol{C}\boldsymbol{P}_n^f$
  Evaluate likelihood: $\log P(y_n|y_{n-1}, \boldsymbol{\theta}) = -\frac{1}{2}\Big(\log(2\pi) + \log|Q_n| + e_n^T Q_n^{-1} e_n\Big)$
  Predict state estimate: $\mathbf{R}_{n+1}^f = \boldsymbol{M}\mathbf{R}_n^a$
  Predict estimate covariance: $\boldsymbol{P}_{n+1}^f = \boldsymbol{M}\boldsymbol{P}_n^a\boldsymbol{M}^T + \boldsymbol{S}$
end for
Return log-likelihood of the parameters:
  $\log\left(y_{1:L}|\boldsymbol{\theta}\right) = \log p(y_1) + \sum_{n=2}^{L} \log P(y_n|y_{n-1}, \boldsymbol{\theta})$

2.4 Application of Markov Chain Monte Carlo Methods

In the previous section, we have discussed diffusion process where the particle dynamics can be viewed as an autoregressive stochastic process governing path of the particles. We also demonstrated how to apply time series analysis methods directly on the coordinate of particles to calculate the likelihood function. In this section, we will apply Markov Chain Monte Carlo (MCMC) methods to estimate model parameters and the uncertainty of the estimates.

MCMC provides a powerful tool for exploring the posteriors which are intractable and difficult to sample from directly. By plugging M-H sampler (see section 1.7.5) in MCMC algorithm, we can draw a sequence of samples distributed from the target distribution. M-H sampler is a simple and broadly applicable method that allows the acceptance rate of a move depend only on the ratio of the target probability. This acceptance criterion ensures that the chain is always homogeneous and reversible, irrespective of the chosen proposal density $q(\cdot)$. Fig. 2.2 shows a path exploring the two-dimensional parameter space. The blue squares represent the accepted moves while the red marks are the moves that are rejected.

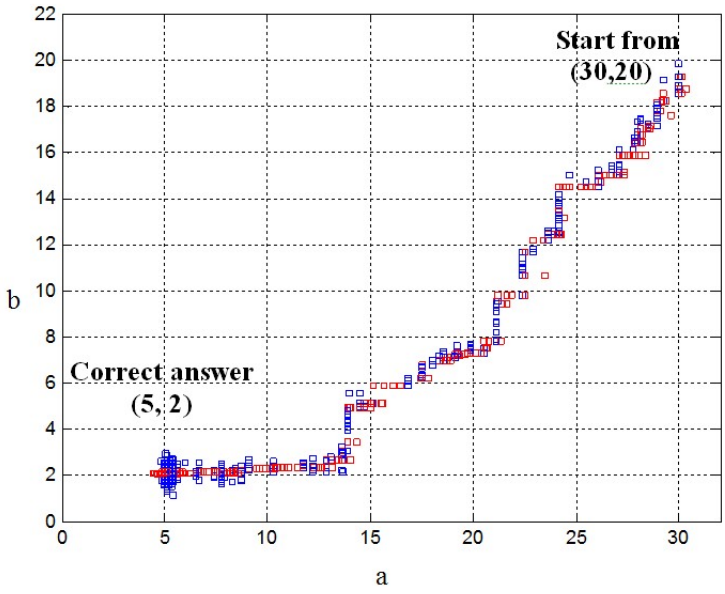


Figure 2.2: Sampling from a two-dimensional distribution using the Metropolis-Hastings algorithm.

The efficiency of the sampler depends on the choice of proposal. The proposal density $q(\cdot)$ is usually selected from a class of symmetric distributions which satisfy $q(\boldsymbol{\theta}|\boldsymbol{\theta}^*) = q(\boldsymbol{\theta}^*|\boldsymbol{\theta})$. By doing this, the acceptance probability $\gamma(\boldsymbol{\theta}, \boldsymbol{\theta}^*)$ in Alg. 1 depends only on the ratio of the posterior density

$$\gamma(\boldsymbol{\theta}, \boldsymbol{\theta}^*) = \min\left\{1, \frac{\pi(Y|\boldsymbol{\theta}^*)\pi(\boldsymbol{\theta}^*)}{\pi(Y|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}\right\}. \tag{2.34}$$

Obviously, the efficiency of the sampler is determined by the acceptance rate and the M-H sampler will fail if the acceptance rate is low. Since algorithms with low acceptance rate will always reject the proposal moves and end up with highly dependent samples, a straightforward way of getting rid of this issue is to make sufficiently small moves at each step. However, 'too high' acceptance rate will lead to poor mixing as well. As a results, the MCMC will be unlikely to move away from a particular state which leads the chain getting stuck in local maxima.

There is no general construction rule for choosing a suitable proposal distribution $q(\cdot)$. An important criterion that evaluates the efficiency of the sampler is the

autocorrelation function (ACF),

$$\mathrm{ACF}(\tau) = \frac{E[(\boldsymbol{\theta}_t - \hat{\boldsymbol{\theta}})(\boldsymbol{\theta}_{t+\tau} - \hat{\boldsymbol{\theta}})]}{\mathrm{var}(\boldsymbol{\theta})}, \tag{2.35}$$

where $\mathrm{E}(\boldsymbol{\theta})$ and $\mathrm{var}(\boldsymbol{\theta})$ denote the mean and variance of samples $\{\boldsymbol{\theta}_t\}$. For a discrete sequence of samples $\{\boldsymbol{\theta}_n\}_{n=1}^{K}$, the discrete ACF can be written as

$$\mathrm{ACF}(j) = \frac{\sum_{n=j+1}^{K} \boldsymbol{\theta}_n^{'} \boldsymbol{\theta}_j^{'}}{\sum_{n=j+1}^{K} \boldsymbol{\theta}_n^{'2}} \tag{2.36}$$

where $\boldsymbol{\theta}_j^{'}$ is given by

$$\boldsymbol{\theta}_j^{'} = \boldsymbol{\theta}_j - \frac{1}{K} \sum_{i=1}^{K} \boldsymbol{\theta}_i.$$

ACF is a common diagnostic tool for determining the independence of sequential samples. Successive samples in the Markov chain are not independent. Higher autocorrelation function at large lags (e.g., lag=10 or 20) means that less information about the target distribution is provided by each iteration. Therefore, larger sample size is needed for fully exploring the whole parameter space. ACF is useful for determining the number of samples to represent proper posterior density and to guarantee convergence of sampling process.

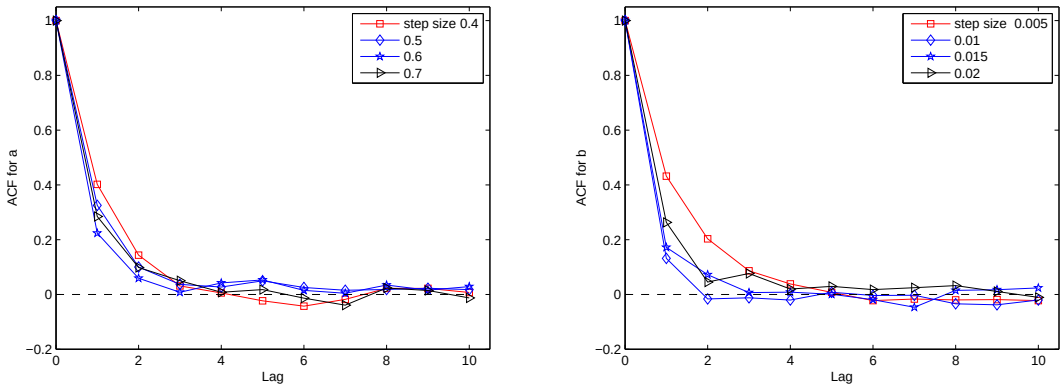
A practical method for generating suitable samples is to tune the proposal distribution adaptively. By doing this, one can obtain appropriate samples with the minimum autocorrelation criterion. Roberts and Rosenthal [28] discussed optimal scaling for various M-H algorithms. A 'good' proposal distribution should make the acceptance rate as close as possible to the optimal value which minimises the autocorrelation criterion. Roberts and Rosenthal also stated that the optimal acceptance rate for a one-dimensional proposal is 0.44 for standard Gaussian target densities and presented various theoretical results for the high-dimensional problems [28].

We will sequentially update the individual components of parameters which reduces a multi-dimensional issue to a one-dimensional problem. Note that this dimension

reduction method works only when the posterior dependence between the different components is weak. We monitor the acceptance rate of each component and adapt the scale of proposal distribution $q(\cdot)$ such that the approximate acceptance rate is close to the optimal value. In our case, we sample $\boldsymbol{\theta}^*$ from Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ which centres at current state $\mu = \boldsymbol{\theta}_i$. The scale of the proposal distribution σ_i^2 is tuned every 20 MCMC iterations. The variance σ_i^2 can be tuned according to the acceptance rate r_{i-1} and previous proposal scale σ_{i-1}^2,

$$\sigma^2 = \begin{cases} 1.5\sigma_{i-1}^2, & r_{i-1} > 55\%, \\ 0.75\sigma_{i-1}^2, & r_{i-1} < 35\%, \\ \sigma_{i-1}^2, & \text{Otherwise.} \end{cases}$$

where the acceptance rate r_{i-1} is evaluated over the past 20 MCMC iterations. Here we are talking about monitoring the acceptance rate and adapting the scale of the proposals for individual components.

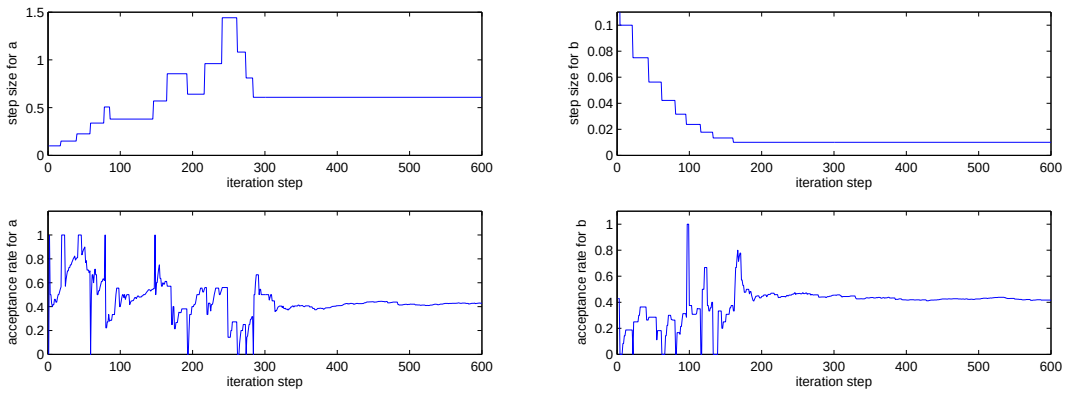


(a) (b)

Figure 2.3: The ACF of MCMC samples using different proposal scales

To demonstrate the efficiency of adaptive MCMC scheme, we first look at a toy model. A stochastic process governed by Eq. (2.9) is investigated. The sequence of observations $\{x_n\}_{n=1}^{1000}$ are generated from a latent model with drift coefficient $a = 10$, and noise term $b = 2$. The time increment is $\Delta_t = 0.03$. Fig. 2.3 shows the

autocorrelation drops off slower when the widths of proposal distribution are either too narrow or too wide compared with the 'optimal' scales. In a high-dimensional example [28] , such issues could be more significant.

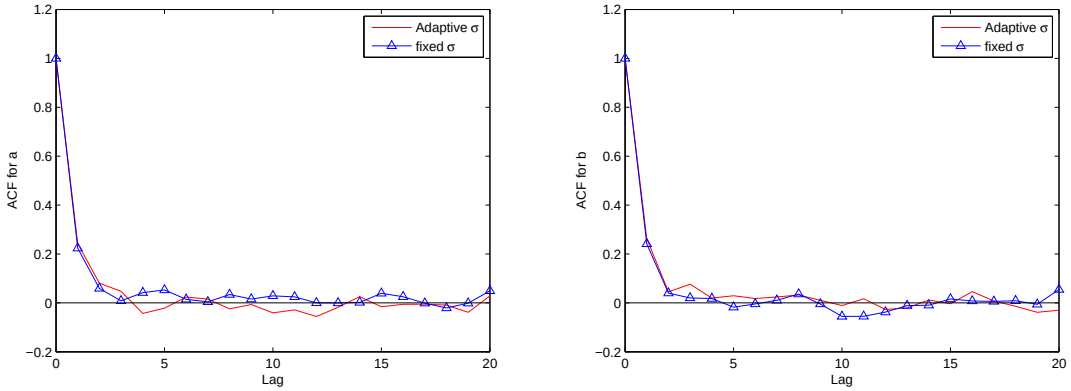


(a) (b)

Figure 2.4: Acceptance rate and corresponding variances of proposal distributions for a toy model

Fig. 2.4 shows the traces of the acceptance rate and the corresponding scales of proposals for parameters a and b, respectively. We calculate the acceptance rate r and update the scales of proposal density for each parameter every 20 steps. As we can see from Fig. 2.4(a), in the adaptive MCMC M-H sampling, high acceptance rate r induces an increase in the variance of proposal distribution, referred to as step size here. Large step size leads to a drastic decline in the acceptance rate. Low acceptance rate leads to a reduction in the step size which in turn boosts the acceptance rate. The proposal scale is adaptively tuned in such manner until it reaches the 'optimal' level. After 300 MCMC iterations, the acceptance rates for both a and b converge to approximately 0.41. To save computational efforts and ensure the reversibility of the MCMC sampler, the adaptive scheme will be terminated after the MCMC reaches equilibrium. With the help of adaptive scheme, we can get rid of local maximum and explore parameter space efficiently with a reasonable acceptance rate.

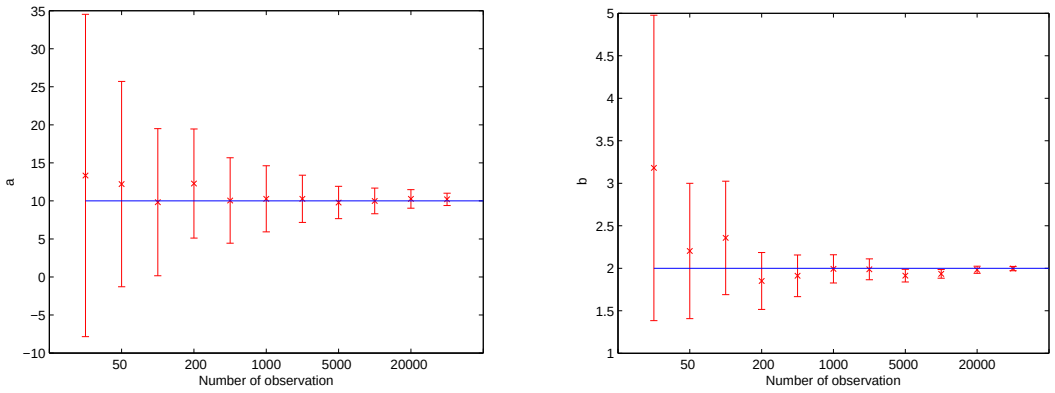
ACFs of the adaptive scheme and standard scheme with optimal scales are shown



(a) (b)

Figure 2.5: The ACFs of MCMC samples using adaptive scheme and standard M-H sampler with 'optimal' scales of proposal distribution.

in Fig. 2.5. It shows that the adaptive method is as efficient as standard scheme facilitated by the use of 'optimal' scale of the proposal distribution. We can also see that both the adaptive algorithms reach the optimal proposal scales, where σ_a^2 is around 0.5 and σ_b^2 is close to 0.01 as shown in Fig. 2.3 and Fig. 2.4. This adaptive strategy is easy to implement without knowing the optimal scales. Bedsides, it allows for quicker burn-in than standard algorithms and explores parameter space more efficiently and effectively.



(a) (b)

Figure 2.6: Estimation and its 95% confidence interval versus number of observations

Next, we will investigate the impact of the number of observations on Bayesian inference. In Fig. 2.6, the horizontal line represents the 'true' parameter values of the latent system while the error bars represent 95% confidence intervals which are symmetric to the estimates that are represented by crosses. Parameter estimates improve as the number of the observations increases, which is a general feature of statistic estimate.

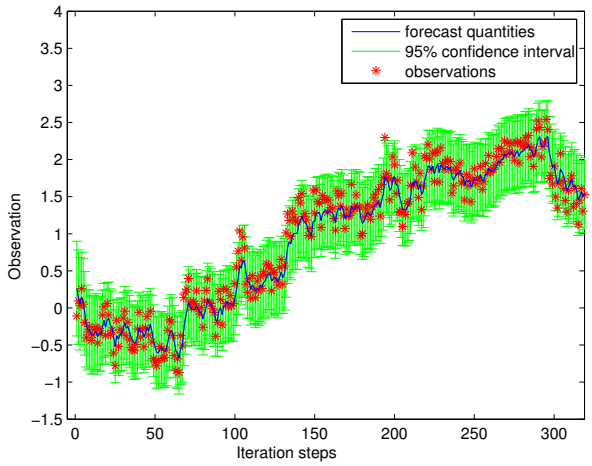


Figure 2.7: Forecast of the state with 95% confidence interval compared with observations

Apart from mean and variance of the posterior density, an alternative way to represent the accuracy and efficiency of the estimates is the difference between forecast proposed by Kalman filter and the actual observations. Fig. 2.7 plots the observations and predictive quantities with 95% confidence intervals.

2.5 The Most Coarse-Grained Model

We have presented the feature of MCMC algorithm in the previous section with a toy model. In this section, we are going to apply MCMC methods on studying coarse-grained models. The model we use to generate chain trajectories is exactly the same as the model introduced by Kremer and Grest [29]. We initialize the system with

particle density $\rho = 0.85$, each chain is of length $N = 16$. The system is simulated in a cubic box of 200 linear chains with periodic boundary conditions. The beads along the chain are connected by FENE springs. Excluded volume effects between monomers are governed by purely repulsive Lennard-Jones potential,

$$U_{FENE}(r) = \begin{cases} -\frac{r_0^2}{2} k \ln\left(1 - \left(\frac{r}{\bar{r}_0}\right)^2\right), & r \leq \bar{r}_0 \\ 0, & r > \bar{r}_0 \end{cases} \tag{2.37}$$

$$U_{LJ}(r) = \begin{cases} 4\epsilon\left(\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} + \frac{1}{4}\right), & \frac{r}{\sigma} \leq 2^{1/6} \\ 0, & \frac{r}{\sigma} > 2^{1/6}, \end{cases} \tag{2.38}$$

where $\bar{r}_0 = 1.5\sigma$ is maximum extension of the FENE spring and $k = 30\epsilon/\sigma^2$ is the spring constant. All the quantities including the Lennard-Jones (LJ) energy ϵ, Lennard-Jones radius σ and the mass of beads m are equal to unity. The Lennard-Jones time is $\tau_{LJ} = \sigma\sqrt{m/\epsilon}$. We use Langevin thermostat with $\zeta = 0.5$ to guarantee constant temperature which maintains $k_B T = \epsilon$.

The time step $dt = 0.012\tau_{LJ}$ is used for simulations and the system is run long enough to ensure the proper equilibration. 1000 configurations with time intervals $\delta_t = 1.2\tau_{LJ}$ are taken from molecular dynamics simulations. The observable is the center of mass of a single chain. We aim to capture essential features of the diffusion coefficient of the chain center of mass at both medium and large time scales ($t \geq \delta_t$). To investigate the impact of the choice of the discrete time step Δ_t shown in Eq. (2.28) on the estimates of the model parameters, three iteration schemes are facilitated using different iteration time steps $\Delta_t = 0.025\tau_{LJ}, 0.05\tau_{LJ}$ and $0.1\tau_{LJ}$, respectively. The observed time intervals $\delta_t > \Delta_t$ implies that we evaluate likelihood every n iteration steps, where $n = \delta_t/\Delta_t$. The simplest model consists of 2 beads described in Eq. (2.13), known as dumbbell model, is used to facilitate MCMC. We take the MCMC estimates as the input parameters and run simulations using equation of motion described in Eq. (2.13). The time-dependent diffusion coefficients $D(t)$ of the

first bead (observable) for each iteration schemes are plotted in Fig. 2.8.

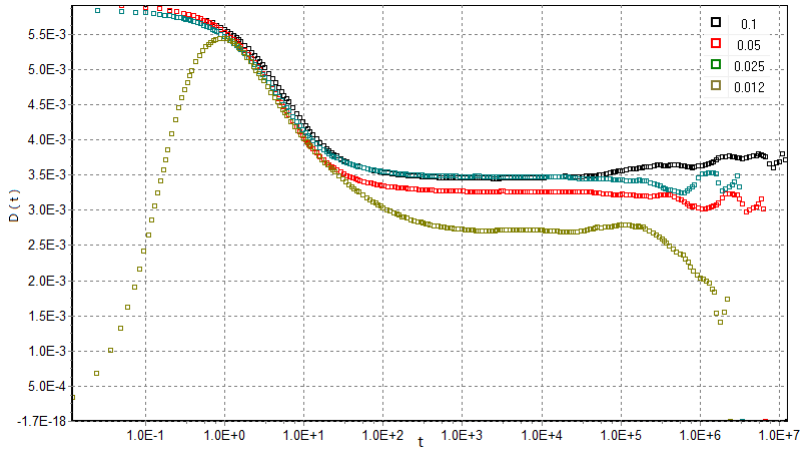


Figure 2.8: Diffusion coefficient $D(t)$ of dumbbell model (2-bead model) with frequent intervals $\delta_t = 1.2\tau_{LJ}$ using iteration schemes $\Delta_t = 0.02, 0.05$ and $0.1\tau_{LJ}$. The brown trace ($\Delta_t = 0.012\tau_{LJ}$) is the target $D(t)$ from MD simulations

Fig. 2.8 shows the diffusion coefficient $D(t)$ of the observable characterized by 2-bead model using iteration schemes with different time steps. The brown trace is the $D(t)$ obtained from MD simulations using with ($\Delta_t = 0.012\tau_{LJ}$). We can see that all the models are trying to 'fit' $D(t)$ between time interval $1.2\tau_{LJ} \sim 12\tau_{LJ}$. Note that in the multi-bead model, the diffusion coefficient converges to a nonzero constant as $t \to 0$, which can be derived from Eq. (2.26),

$$D(t) \propto \frac{2}{\sum_{j=0}^{N} \xi_j} + \sum_{j=1}^{N} 2Q_{0j}^2 d_j \tau_j, \text{ for } t \to 0. \tag{2.39}$$

Besides, the minimum observed time interval is $\Delta_t = 1.2\tau_{LJ}$ which implies that we cannot get any information about dynamic behaviours of the target chain at very short time scales $t < \Delta_t$. It is worth mentioning that the total observed time duration is $T = 1200\tau_{LJ}$ which may not be sufficient to capture the features of polymer dynamics at large time scales. When developing coarse-grained models, we usually wish to guarantee the consistency at large time scales first and then capture as more features as possible for medium and short time scales.

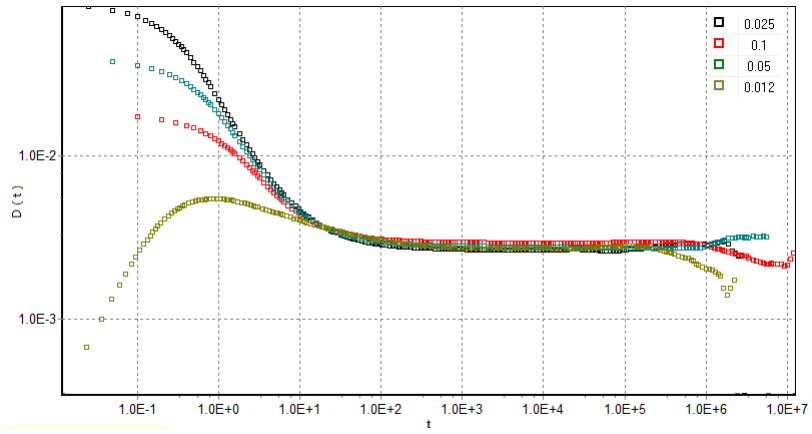


Figure 2.9: Diffusion coefficient $D(t)$ of dumbbell model (2-bead model) with frequent intervals $\delta_t = 12\tau_{LJ}$ using iteration schemes $\Delta_t = 0.02, 0.05$ and $0.1\tau_{LJ}$. The brown squares ($\Delta_t = 0.012\tau_{LJ}$) is the target $D(t)$ from MD simulations

To capture the dynamic behaviours at large time scales, we increase the observed time interval from $1.2\tau_{LJ}$ to $12\tau_{LJ}$ and the same number of configurations $L = 1000$ is used for modelling. Fig. 2.9 reveals that the model shows good agreement with MD simulations for $D(t)$ at $t > 12\tau_{LJ}$. This implies that if we want to fit the whole diffusion coefficient function using 2-bead model, observations with smaller time intervals and larger data size are essential. However, we find that the MCMC estimates using 10,000,000 configurations with observation interval $\delta_t = 1.2\tau_{LJ}$ are not satisfactory, which implies that 2-bead model is not adequate to describe the diffusion of the chain center of mass. We consider to increase the number of hidden particles and simulate a system with longer chains $N = 64$ as shown in Fig. 2.10.

It can be seen from Fig. 2.10 that the 2-bead model does not capture the diffusive behaviours at large time scales. The 3-bead model and 4-bead model predict the final plateau of $D(t)$ well with relatively small errors. Excellent agreement is shown between 4-bead model and MD simulations in the diffusion behaviours at time scales $10\tau_{LJ} \sim 1000\tau_{LJ}$. It is obvious that the more hidden beads we have, the better prediction of the diffusion coefficient $D(t)$ we will obtain. The problem is what is the sufficient number of beads to describe the dynamics of the center of mass of

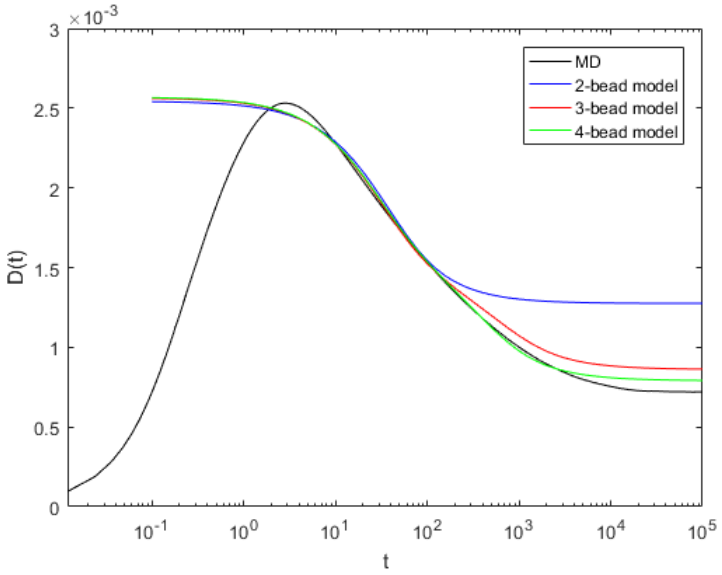


Figure 2.10: Analytical solutions of time-dependent diffusion coefficient $D(t)$ for 2-bead (blue), 3-bead (red) model and 4-bead (green) model. Frequent interval $\delta_t = 12\tau_{LJ}$; The parameters used to plot $D(t)$ are the MCMC estimates for each model

single chains. One needs to evaluate the increased model complexity and the benefits of increasing number of hidden variables. In Chapter 4, we will discuss a set of approaches for Bayesian model comparison which may be helpful to solve this problem.

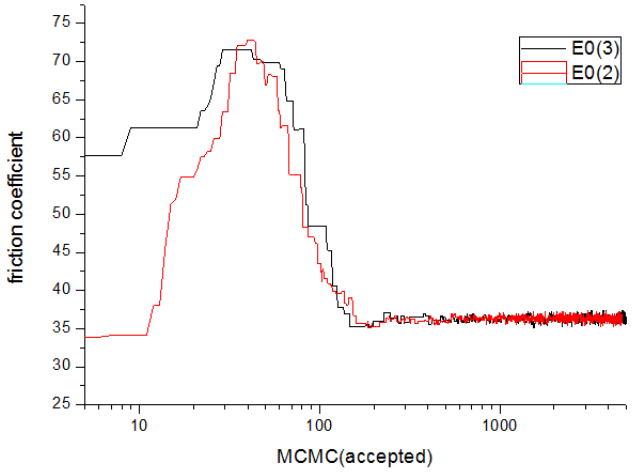


Figure 2.11: Estimate of E_0 (2-bead model and 3-bead model)

After looking at the agreement of the multi-bead model and MD simulations in

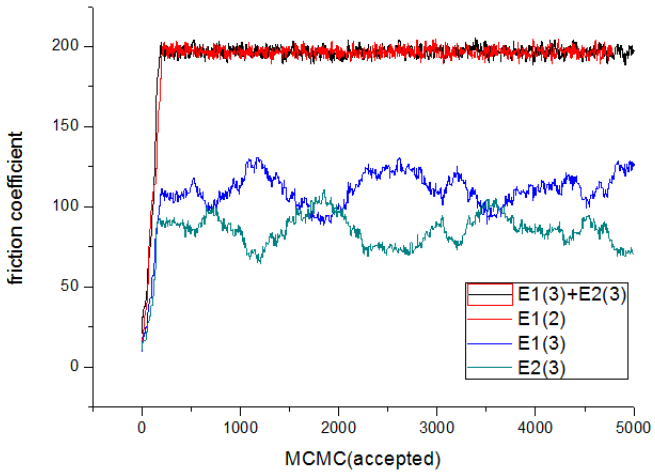


Figure 2.12: Estimates of friction coefficients of hidden beads 2-bead model and 3-bead model

the diffusion behaviours at different time scales, we also investigate the estimates of the models parameters provided by MCMC methods. Fig. 2.11 shows the trace of E_0 which is the friction coefficient of the first bead (observable) in MCMC iterations. We can see that there is excellent agreement in the estimate of E_0. It is also interesting to

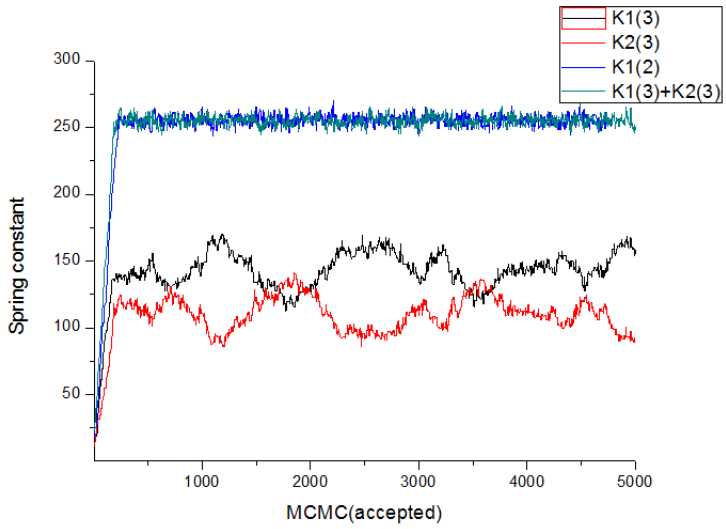


Figure 2.13: Estimate of spring constants (2-bead model and 3-bead model)

look at the estimates of the parameters for hidden variables. Fig. 2.12 and Fig. 2.13

compare the estimates of friction coefficients and spring constants for the hidden beads in 2-bead model and 3-bead model. In Fig. 2.12, the red line is the trace of friction coefficients of hidden beads in 2-bead. The blue and green lines show trace of friction coefficients of the second and third beads for 3-bead model, together with the black line which is the sum of the friction coefficients of these two beads. The fluctuations of blue line and green line imply that there is no distinct difference between the hidden beads in 3-bead model, i.e., the MCMC methods prefer a Rouse-like coarse-grained model to govern the dynamics of chain center of mass.

2.6 Summary

To conclude, this chapter is motivated by the need to develop novel coarse-grained model to simulate entangled polymer melts. We propose a multi-bead coarse-grained model to study the dynamics of chain center of mass diffusion. Unlike other coarse-grained approaches which usually take inter- and intra-chain interactions into account, our multi-bead coarse-grained model focuses on the chain center of mass diffusion, which will significantly reduce the number of free parameters in the coarse-grained model.

The multi-bead model is characterized by a sequence of linear stochastic equations with Gaussian noise, in which the parameters of interest are friction coefficients and spring constants. To obtain the estimates of the unknown parameters, we embed Kalman Filter into MCMC paradigm which is sufficient to draw inferences on the unknown parameters for linear system with Gaussian noise. Adaptive MCMC scheme is introduced to improve the efficiency of MCMC algorithm.

This chapter explores the performance of coarse-grained models with different hidden beads. We show that for a chain with chain length $N = 64$, a 3-bead model is sufficient to describe the chain center of mass diffusion. We also find that,in multi-bead model, it is preferable to have Rouse-like structure rather than asymmetric star structure.

Chapter 3

Single-Particle Dynamics with Memory Kernel

We have seen from the previous chapter that a linear model may not always simultaneously capture the dynamic features of the polymers at different time scales. As the chain length increases, more complicated behaviours will arise due to the chain uncrossibility. It is worth to investigate coarse-grained models with nonlinear interactions. In this chapter, we will focus on more general models, including multi-bead models involving non-linear dynamics and single-particle dynamics with memory kernel. At first, we consider a nonlinear system where standard Kalman filter does not work any more. We then introduce particle filter and the efficiency of this approach is discussed. After that, a more general model — single-particle dynamics with memory kernel is introduced. In the latter case, apart from the mean square displacement, we will also investigate the velocity autocorrelation function of the chain center of mass.

3.1 Introduction

Consider a multi-bead coarse-grained model where the interactions between beads are nonlinear, the equation of motion can be written as

$$d\mathbf{X} = -\boldsymbol{f}(\boldsymbol{X}, \boldsymbol{\beta})dt, \tag{3.1}$$

where $\boldsymbol{f}(\cdot)$ is a nonlinear function. This nonlinear system can be described as follows:

Suppose we have time series data $y_{1:L}$ which involved with a set of hidden states $x_{1:L}$. It is natural to assume that the state evolution is Markov, which means the prediction of the future state x_t depends only on the current x_{t-1}. The model evolution equation is

$$x_t = f(x_{t-1}, \beta_{t-1}), \quad t = 1, \cdots, L, \tag{3.2}$$

where β_{t-1} is a random variable that describes the model error. The conditional probability of the prediction is given by transition density $q(x_t|x_{t-1})$. Observation is described as

$$y_t = h(x_t, \eta_t), \quad t = 1, 2, ..., L, \tag{3.3}$$

where $h(\cdot)$ is the measurement operator, η_t is the observation error. If the measurement error is Gaussian with $\eta_t \sim \mathcal{N}(0, \sigma^2)$, the conditional probability of the observation given model state x_t can be written as

$$p(y_t|x_t) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(y_t - h(x_t))^2}{2\sigma^2}\right]. \tag{3.4}$$

Our goal is to infer the 'filtering distribution' $p\left(x_t|y_{1:t}\right)$ of x_t given all the previous observation $y_{1:t}$, $t = 1, 2, ..., L$. If the model is linear and all the distribution are Gaussian, one can reach the analytical form of the target distribution by using Kalman filter and conjugate prior. However, when the model is nonlinear, or the probability density is not Gaussian, we need to work out new methods to address this problem.

3.2 Particle Filter

Particle filters [30] [31] are a set of Monte Carlo methods originally used to solve filtering problems arising in signal processing and Bayesian statistical inference. The distribution of interest is the marginal of the latent variables x_t given all observations $y_{1:t}$ up to time t. Particle filters allow us to sequentially update the posterior distribution at time t without modifying the previously inference made on states $x_{1:t-1}$. The fundamental idea is to facilitate importance sampling on path space. At time t, importance sampling yields weighted points from distribution $p\left(x_t|y_{1:t}\right)$. For the p^{th} particle, sample $x_{t+1}^{(p)}$ is drawn from proposal distribution which yields $x_{t+1}^{(p)}|x_t^{(p)} \sim q(x_{t+1}|y_{1:t+1}, x_t)$. The particle would get unnormalised weight evaluated by the weight of $x_t^{(p)}$

$$\begin{aligned}
\widetilde{w}_{t+1}^{(p)} &= \frac{f(y_1|x_1^{(p)})p(x_1^{(p)})\prod_{i=1}^{t} p(x_{i+1}^{(p)}|x_i^{(p)})f(y_{i+1}|x_{i+1}^{(p)})}{q(x_1^{(p)}|y_1)\prod_{i=1}^{t} q(x_{i+1}^{(p)}|y_{1:i+1}, x_i^{(p)})} \\
&= w_t^{(p)} \frac{p(x_{t+1}^{(p)}|x_t^{(p)})f(y_{t+1}|x_{t+1}^{(p)})}{q(x_{t+1}^{(p)}|y_{1:t+1}, x_t^{(p)})}, \quad p = 1, 2, \cdots, N_p,
\end{aligned}$$

where N_p is the number of particles used for importance sampling. The normalised weights are given by

$$w_{t+1}^{(p)} = \frac{\widetilde{w}_{t+1}^{(p)}}{\sum_{p=1}^{N_P} \widetilde{w}_{t+1}^{(p)}}, \tag{3.5}$$

which provides a weighted particle approximation $\{w_t^{(p)}, x_t^{(p)}\}_{p=1}^{N_p}$ to the marginal distribution $p\left(x_t|y_{1:t}\right)$ at time t. The sampling procedure is described in Alg. 3. Particle Filters are applied to solve strongly nonlinear data-assimilation problems. They have a few strong assets, e.g., their ability to deal with nonlinear dynamics, the simplicity of their implementation and the advantage that their behaviours do not depend on a specification of the model state covariance matrix.

Algorithm 3 Particle Filter

procedure INITIALIZATION
  for $p = 1 : N_p$ **do**
    Sample $x_1^{(p)} \sim q(x_1|y_1)$
    Weight of $x_1^{(p)} : \widetilde{w}_1^{(p)} = \frac{f(y_1|x_1^{(p)})p(x_1^{(p)})}{q(x_1^{(p)}|y_1)}$.
  end for
  Normalised weights $w_1^{(p)} = \frac{\widetilde{w}_1^{(p)}}{\sum_{p=1}^{N_p} \widetilde{w}_1^{(p)}}$
end procedure
procedure ITERATION
  for $t = 1 : L - 1$ **do**
    for $p = 1 : N_p$ **do**
      Sample $x_{t+1}^{(p)}|x_t^{(p)} \sim q(x_{t+1}|y_{1:t+1}, x_t)$
      Weighted point :$\widetilde{w}_{t+1}^{(p)} = w_t^{(p)} \frac{p(x_{t+1}^{(p)}|x_t^{(p)})f(y_{t+1}|x_{t+1}^{(p)})}{q(x_{t+1}^{(p)}|y_{1:t+1}, x_t^{(p)})}$.
    end for
    Normalised weights $w_{t+1}^{(p)} = \frac{\widetilde{w}_{t+1}^{(p)}}{\sum_{p=1}^{N_p} \widetilde{w}_{t+1}^{(p)}}$
  end for
end procedure

3.3 Reducing the Variance in the Weights

Note that one of the main drawbacks of importance sampling is that when a particle moves away from the observations, the relative weight will become smaller if no correction made on the particle to pull it back to the observations. This always happens as the filter runs and the dimension of the space grows. In practical applications, one particle would always end up getting all of the weights in the end where the statistical information of the ensemble is not meaningful. This phenomenon is called filter degeneracy. Resampling strategy is useful to deal with the issue of particle degeneracy. In resampling scheme, the particle ensemble is resampled. Particles with very low weights are abandoned, while multiple copies of particles with high weights are kept for the posterior probability distribution in the sequential implementation in order to restore the total number of particles N_p. After resampling, all the new particles are of the same weight $w_i = \frac{1}{Np}$. There are several widely used resampling algorithms.

- Probabilistic resampling directly samples randomly from the density given by the weights $P(N_p, \{w_i\})$. The basic idea is to draw N_p independent uniforms $\{U_i\}_{1 \leq i \leq N_p}$ on the interval $(0, 1]$; The new i^{th} sample associated with U_i is set to be a copy of particle j if $U_i \in (W_{j-1}, W_j]$ where W is the cumulative sum of $\{w_i\}$.

- Residual sampling assumes that there are n_i copies taken of particle i ($n_i = [N_p w_i]$), where $[\cdot]$ denotes the integer part. The rest of the particles needed to obtain ensemble size N_p are then drawn randomly from distribution $P(N_p - \sum n_i, \{\frac{N_p w_i - n_i}{N_p - \sum n_i}\})$ which is the same as Probabilistic resampling described above.

- Stratified resampling is based on a method of sampling from population used in statistical surveys. The basic idea is to equally divide interval $(0, 1]$ into n disjoint sets. The sequence of $\{U_i\}$ are drawn independently in each of these sub-intervals with $U_i \sim U(\frac{i-1}{N_p}, \frac{i}{N_p}]$. Then the method to draw new samples from $\{U_i\}$ is the same as the one used in probabilistic resampling

- Systematic resampling takes stratified resampling one step further by deterministically linking all the variables drawn in the sub-intervals. This is achieved by setting

$$U_i = \frac{i-1}{N_p} + u$$

where u is a single random variable drawn from the uniform distribution $U(0, 1/N_p]$.

Due to the random character of the sampling, so-called sampling noise is introduced. Douc et al. [32] proved that the conditional variance of probabilistic resampling is always larger than that of the rest of sampling schemes mentioned above. In practical applications, residual, stratified, and systematic resampling are generally found to provide comparable results. Systematic resampling is often preferred since it is simple to implement. In this chapter, we are going to use systematic resampling scheme.

3.4 The Proposal Distribution

To decrease the variance in the weights, apart from resampling to generate more copies of the particles with large weights, we can also try to increase the weight by improving the likelihood of each point. An efficient proposal density allows one to sample from the density that is conditioned on the new observations so that the predicted state is much closer to the observations than the prior density. It is important to note that the idea does not work in high-dimensional spaces with large numbers of independent observations since the conditional proposal density would be too difficult to sample from. This strategy is a one-time-step scheme, assuming observations available at each time step. If the observations are not available for each step t, an approach called Implicit Particle Filter [33] can be used. Implicit Particle Filter extends standard particle filter to multiple time steps between observations.

There are several approaches to construct efficient proposal densities. Ensemble Kalman filter (EnKF) [34] [35] is powerful to deal with problems involving nonlinear dynamics and Gaussian noises. Marginal Particle Filter (MPF) [36] performed filtering directly on the marginal distribution instead of on the joint space.

Note that using a clever proposal density is to generate particles that are close to the observations. The choice of proposal density does not introduce any bias on the approximation of target distribution. By doing this, we end up with particles that have much smaller variances in weights.

3.5 Particle Markov Chain Monte Carlo Methods

In the previous section we assumed that parameters of model evolution in Eq. (3.2) and observation operator in Eq. (3.3) are known. The joint distribution of data $y_{1:L}$ and hidden state $x_{1:L}$ given parameter $\boldsymbol{\theta}$ can be written as

$$p\left(x_{1:L}, y_{1:L}|\boldsymbol{\theta}\right) = p(y_1|x_1, \boldsymbol{\theta})p(x_1|\boldsymbol{\theta}) \prod_{i=1}^{L-1} p(x_{i+1}|x_i, \boldsymbol{\theta})p(y_{i+1}|x_{i+1}, \boldsymbol{\theta}), \tag{3.6}$$

Suppose we have a prior distribution $p(\boldsymbol{\theta})$ on parameter $\boldsymbol{\theta}$, and we would like to find the posterior distribution $\pi(\boldsymbol{\theta}|y_{1:L})$ for given data $y_{1:L}$. The posterior distribution can be written as

$$\pi(\boldsymbol{\theta}|y_{1:L}) \propto p(\boldsymbol{\theta})\, p(y_{1:L}|\boldsymbol{\theta}), \tag{3.7}$$

where $p(y_{1:L}|\boldsymbol{\theta})$ is known as marginal likelihood with integrating out the hidden states $x_{1:L}$,

$$p(y_{1:L}|\boldsymbol{\theta}) \;=\; \int_{x_{1:L}} p(y_{1:L}|x_{1:L},\boldsymbol{\theta})\, p(x_{1:L}|\boldsymbol{\theta})\, dx_{1:L},$$

where $p(x_{1:L}|\boldsymbol{\theta})\, p(y_{1:L}|x_{1:L},\boldsymbol{\theta})$ is given by

$$p(x_{1:L}|\boldsymbol{\theta})\, p(y_{1:L}|x_{1:L},\boldsymbol{\theta}) = p(x_1|\boldsymbol{\theta}) \prod_{t=2}^{L} p(x_t|x_{t-1},\boldsymbol{\theta}) \prod_{t=1}^{L} p(y_t|x_t,\boldsymbol{\theta}).$$

If the transition kernel $x_{t-1} \to x_t$ is linear and all the distributions involved are Gaussian distribution, the marginal likelihood $p(y_{1:L}|\boldsymbol{\theta})$ is tractable. In such case, the marginal likelihood can be derived using the Kalman filter. When the system is nonlinear, adopting the idea of standard MCMC, we can try sampling from

$$\pi(\boldsymbol{\theta}, x_{1:L}|y_{1:L}) \propto p(y_{1:L}|x_{1:L},\boldsymbol{\theta})\, p(x_{1:L}|\boldsymbol{\theta})\, p(\boldsymbol{\theta}).$$

In the i^{th} sweep of the sampler, parameter $\boldsymbol{\theta}^{(i)}$ and latent states $x_{1:L}^{(i)}$ are updated using the following framework,

$$\boldsymbol{\theta}^{(i)} \sim \pi\left(\boldsymbol{\theta}|x_{1:L}^{(i-1)}, y_{1:L}\right),$$

$$x_{1:L}^{(i)} \sim \pi\left(x_{1:L}|\boldsymbol{\theta}^{(i)}, y_{1:L}\right).$$

The algorithm might be inefficient when $\boldsymbol{\theta}$ and $x_{1:L}$ are strongly dependent in the posterior. Besides, hidden states $x_{1:L}$ are usually high dimensional and difficult to get efficient samples.

Particle MCMC (PMCMC) helps us get rid of this problem. Marginal particle

MCMC uses a particle filter to provide an unbiased estimate $\widehat{p}\left(y_{1:L}|\boldsymbol{\theta}_i\right)$ of $p\left(y_{1:L}|\boldsymbol{\theta}_i\right)$ which is based on the idea of pseudo-marginal approach (see section 1.7.7). By doing this, posterior dependence of $\boldsymbol{\theta}$ and $x_{1:L}$ no longer is a barrier, and PMCMC provides samples from the posterior on $x_{1:T}$ as a by-product.

Firstly, we are going to use the Ornstein–Uhlenbeck process shown in Eq. (2.9) to demonstrate how to implement particle MCMC.

The data we use are synthetic data obtained from model with parameter (a, b) : $(10, 0.5)$.

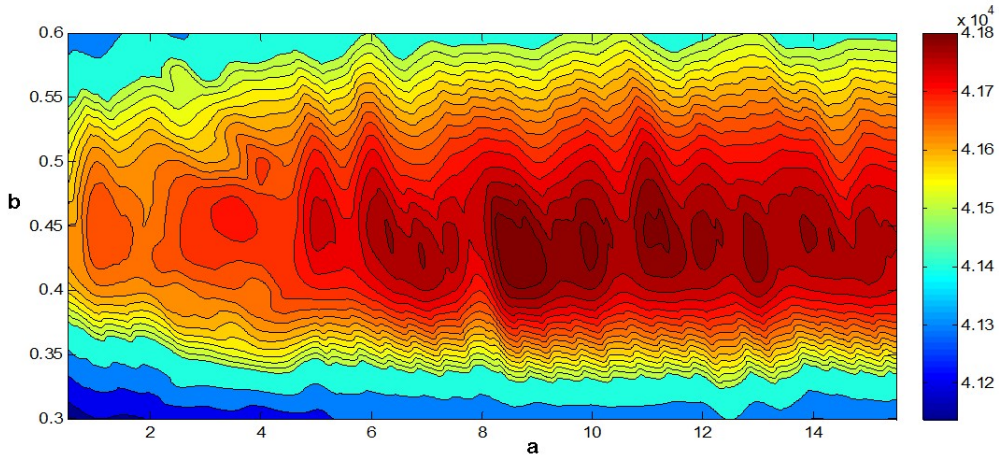


Figure 3.1: Likelihood plot over the parameter space

The number of observations $L = 50000$ and time step $dt = 0.001$. The total time of the simulation is $T = Ldt = 50$. Fig. 3.1 shows the likelihood density on parameter space. We can see that the likelihood function is sensitive to the accuracy of the noise term b. Therefore, if there is measurement error, it is worth investigating the impact of the scale of measurement error on evaluation of the likelihood. We artificially add different scales of errors on the observations and use particle filters to estimate the likelihood. The 'theoretical' solution would be the likelihood evaluated by Kalman filter without observation error. Fig. 3.2 shows the likelihood of estimate $(a, b) = (10, 0.5)$ computed using different number of particles on different scales of observation

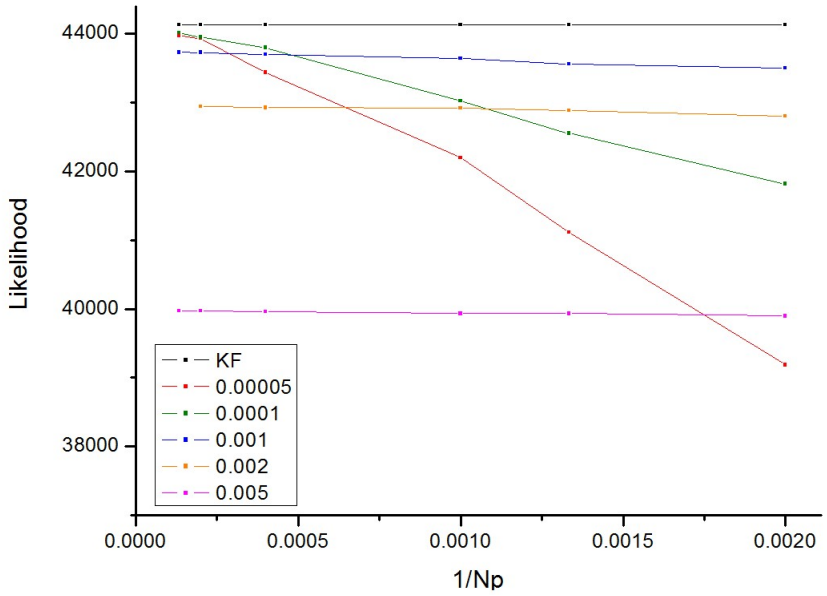


Figure 3.2: Likelihood probabilities versus the inverse of the number of particles $1/N_p$ for various observation error measurements: Black line is the likelihood evaluated by Kalman filter without observation error. The scales of the errors are characterized by the observation error variances.

errors. We can see that we need more particles to represent the proper density as the measurement of error decreases. Since most of particles suffer degeneracy issue if tolerance of the observation error is small.

Fig. 3.3 shows the likelihood function with respect to parameter a when b is known. The number of particles N_p varies from 600 to 5000. It can be seen that the likelihood would not increase significantly by increasing N_p for $N_p > 2000$. Note that PMCMC algorithm can be very computationally expensive since we need to run particle filter for each $\boldsymbol{\theta}^{(i)}$. The question then becomes whether it is better, for a fixed computational effort, to use more particles (larger N_p) per update, or use fewer particles and try more $\boldsymbol{\theta}^{(i)}$. Doucet et al. [37] suggests that the optimal number of particles must ensure the log-likelihood estimate has a variance of about 1.

Fig. 3.4 and Fig. 3.5 show the trace of MCMC iterations using KF and particle filter, respectively. The number of observations $L = 5000$ and time step $dt = 0.001$.

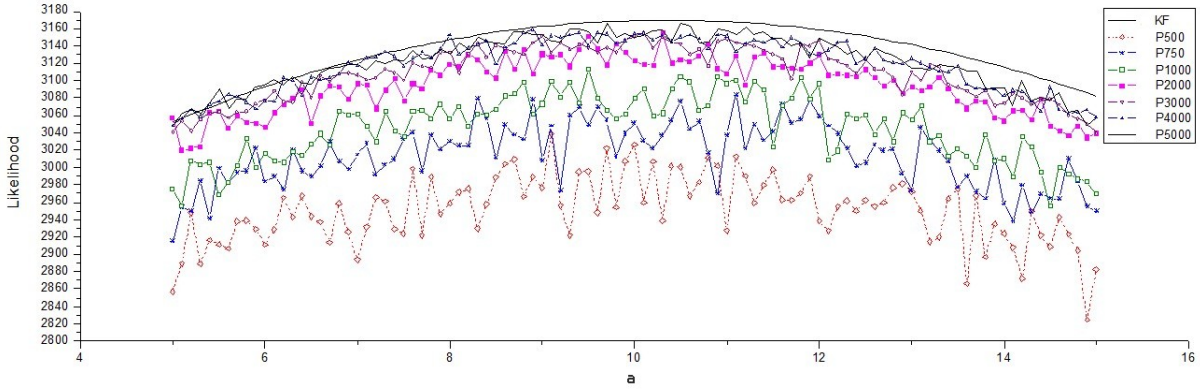


Figure 3.3: Approximation of likelihood of parameter a using different number of particles

We can see that in Fig. 3.4 the acceptance rate is high which means that the MCMC mixes well. However, PMCMC in Fig. 3.5 ends up with noisy likelihood which leads to low acceptance rate.

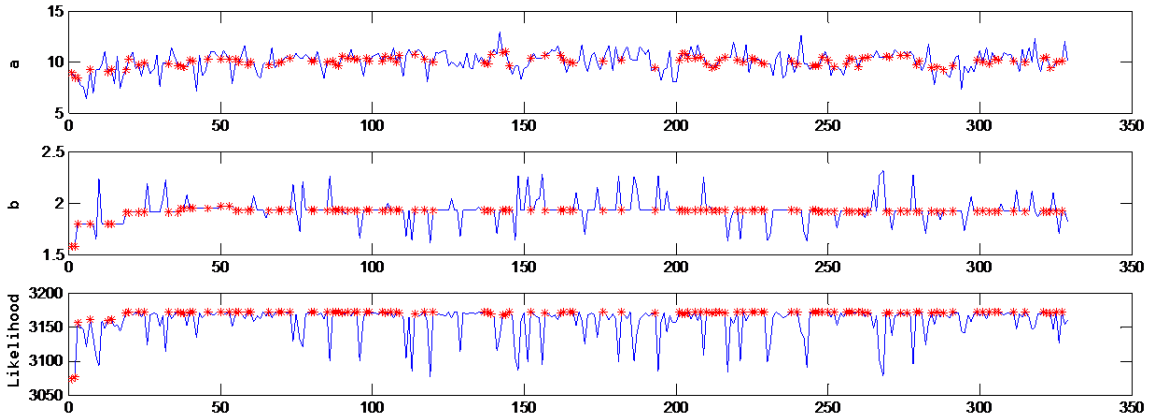


Figure 3.4: Standard MCMC iteration using KF; red points are accepted moves

3.6 Single-Particle Dynamics with Memory Kernel

As the level of coarse-graining increases, the potentials between supramolecular particles become weaker than the underlying interatomic interaction potentials. In the

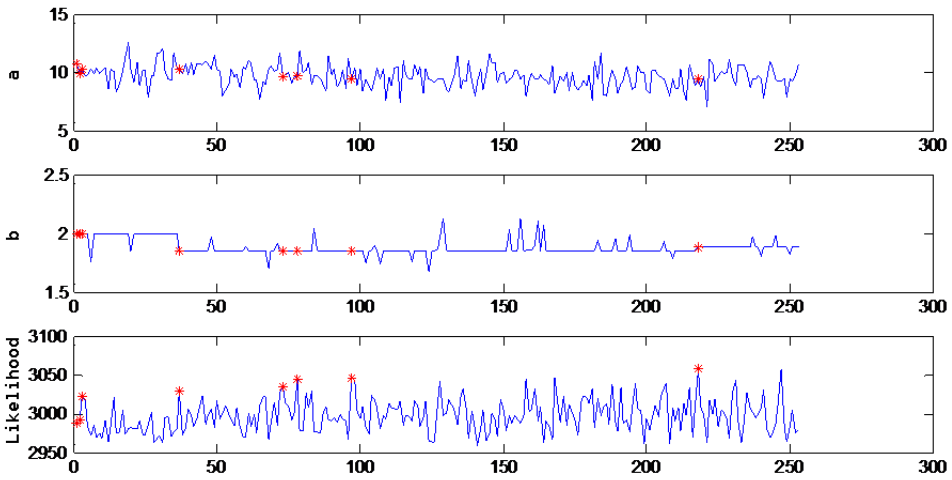


Figure 3.5: Particle MCMC iteration; red points are accepted moves

context of polymer dynamics, the uncrossability of chains gets lost at some degree of coarse-graining. Algorithms to prevent chain crossings have been suggested in the literature [38]. Lots of coarse-grained models just ignore the memory effects, by doing this, the models do have a good agreement with experiments, but they may lose the characteristics of slow dynamics. Therefore, these models perhaps are not suitable for studying long-time rheology. In a real polymeric system, the friction and random forces usually have the memory of time and uncorrelated and fully random dynamics are not adequate. Therefore, it is natural to set up a coarse-grained model with memory kernel.

Zwanzig [39] suggests that the motion of microscopic Brownian particles in viscoelastic materials is governed by the generalized Langevin equation (GLE),

$$m\frac{dV(t)}{dt} = -\int_0^t dt\varphi(t-t')V(t')dt' + \widetilde{F}(t), \tag{3.8}$$

where m is the mass of the particle, $V(t)$ is the velocity of the particle. $\widetilde{F}(t)$ is a zero-centered Gaussian noise which is correlated with the memory kernel $\varphi(t)$ through the

fluctuation-dissipation relation

$$\langle \widetilde{F}(t)\widetilde{F}(s)\rangle = k_B T\varphi(t-s), t > s, \tag{3.9}$$

where T is the absolute temperature, and k_B is the Boltzmann constant. We divide both sides by the mass of particle m and redefine the memory kernel to obtain

$$\frac{dV(t)}{dt} = -\int_0^t \xi(t-t')V(t')dt' + \sqrt{\frac{k_B T}{m}}F(t), \tag{3.10}$$

where $\langle F(t)F(s)\rangle = \xi(t-s)$ and $\xi(t-s) = \frac{\varphi(t-s)}{m}$, $t > s$.

In simple systems, the time correlation function for memory kernel could be described by the sum of exponentials with constant characteristic times. Let us start from the simplest memory kernel function with a single exponential decay.

$$\xi(t) = ce^{-\frac{t}{\tau}}, c = \frac{6\pi aG}{m}, \tag{3.11}$$

where a is the particle radius, shear modulus G and viscosity η determine the relaxation time $\tau = \eta/G$. In this case, the noise term $F(t)$ can be written as an Ornstein–Uhlenbeck process,

$$\frac{dF(t)}{dt} = -\frac{1}{\tau}F(t) + \sqrt{\frac{2c}{\tau}}f(t). \tag{3.12}$$

where $f(t)$ is white noise with zero mean and following covariance,

$$\langle f(t)f(s)\rangle = \delta(t-s) \tag{3.13}$$

Note that the Langevin equation for viscous diffusion is obtained in the limit $\tau \to 0$; that is $F(t)$ becomes (with $\xi = 6\pi a\eta$)

$$F(t) = \sqrt{\frac{2\xi}{m}}f(t), \tag{3.14}$$

To simplify Eq. (3.10), we introduce a new variable $Z(t)$ defined by

$$Z(t) = \int_0^t e^{-\frac{t-t'}{\tau}} V(t')dt', \tag{3.15}$$

which yields

$$\frac{dZ(t)}{dt} = -\frac{1}{\tau}Z(t) + V(t) \tag{3.16}$$

Then we have a sequence of differential equations

$$\begin{cases} \frac{dY(t)}{dt} = V(t), \\ \frac{dV(t)}{dt} = -cZ(t) + \sqrt{\frac{k_B T}{m}}F(t), \\ \frac{dZ(t)}{dt} = V(t) - \frac{1}{\tau}Z(t), \\ \frac{dF(t)}{dt} = -\frac{1}{\tau}F(t) + \sqrt{\frac{2c}{\tau}}f(t), \end{cases} \tag{3.17}$$

where $Y(t)$ is the observable, $V(t)$ is the velocity, $Z(t)$ represents an integral with respect to $V(t)$, $F(t)$, as mentioned before, is Gaussian color noise. Rewritten the above equations in vector form $X(t) = (Y(t), V(t), Z(t), F(t))$, we have a linear system of stochastic differential equations

$$\frac{dX(t)}{dt} = AX(t) + BW(t), \tag{3.18}$$

where

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -c & \sqrt{\frac{k_B T}{m}} \\ 0 & 1 & -\frac{1}{\tau} & 0 \\ 0 & 0 & 0 & -\frac{1}{\tau} \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{\frac{2c}{\tau}} \end{pmatrix}.$$

If the trajectory of chain center of mass is available, the above model can be viewed as a four-layer hidden Markov process with observable $Y(t)$ and latent states $V(t), Z(t)$ and $F(t)$. To make inference on the unknowns in the above model, a standard PMCMC method can be used to explore parameter space $\{c, \tau\}$. One can also consider Approximate Bayesian Computation (ABC) methods [40] to draw samples from the

posterior distribution with given observable Y or summary statistics of Y. ABC is useful in cases where likelihood function $P(Y|c,\tau)$ cannot be evaluated analytically or it is computationally expensive.

3.6.1 Laplace Transform of Velocity Autocorrelation Function

Multiplying both sides of Eq. (3.10) with $V(0)$ and performing an appropriate ensemble average $\langle\cdot\rangle$, one obtains the memory equation for the velocity autocorrelation function (VAF),

$$\frac{da(t)}{dt} = -\int_0^t \xi(t-t')a(t-t')dt', \tag{3.19}$$

where $a(t) = \langle V(0)V(t)\rangle$ appears, which is related to the diffusion constant by the integral relation

$$D = \frac{1}{3}\int_0^\infty a(t)dt. \tag{3.20}$$

Note that $\frac{1}{2}m\langle V^2(0)\rangle = \frac{1}{2}k_BT$, we have $a(0) = \langle V^2(0)\rangle = k_BT/m$. Again, suppose the memory kernel is a single exponential shown in Eq. (3.11), the Laplace transform of the memory kernel $\xi(t)$ can be written as

$$\widetilde{\xi}(z) = \int_0^\infty e^{-st}\xi(t)dt = \frac{c}{z+1/\tau}.$$

After a Laplace transform, Eq. (3.19) reads,

$$\widetilde{a}(z) = a_0[z+\widetilde{\xi}(z)]^{-1} = \frac{a_0}{z+\frac{c}{z+1/\tau}}, \tag{3.21}$$

where $\widetilde{a}(z)$ is the Laplace transform of $a(t)$. The inverse Laplace transform of $\widetilde{a}(z)$ is given by

$$a(t) = a_0 e^{-\frac{t}{2\tau}}\left(\cosh\left(\frac{t\sqrt{1-4c\tau^2}}{2\tau}\right) + \frac{\sinh\left(\frac{t\sqrt{1-4c\tau^2}}{2\tau}\right)}{\sqrt{1-4c\tau^2}}\right) \tag{3.22}$$

Now we have the analytical solution of the VAF of chain center of mass with respect to $\{c,\tau\}$. Let us consider another important quantity – diffusion coefficient D. Recall

that the Laplace transform of the time-domain integration is

$$\mathcal{L}\left[\int_0^t a(s)ds\right] = \frac{\widetilde{a}(z)}{z}.$$

Substitute the above formula into Eq. (3.20), for large t, the Final value theorem (see Appendix) gives

$$D = \frac{1}{3}\widetilde{a}(z)|_{z=0} = \frac{a_0}{3c\tau} = \frac{k_BT}{3mc\tau}, \tag{3.23}$$

$$C = \int_0^\infty sa(s)ds = -\frac{d\widetilde{a}(z)}{dz}|_{z=0} = \frac{k_BT}{mc^2\tau^2}(1-c\tau^2). \tag{3.24}$$

Substitute the analytical solution of $a(t)$ into the following formula in Eq. (3.22), we have the mean square displacement function of the chain center of mass

$$\begin{aligned}&\langle (Y(t)-Y(0))^2\rangle = 2\int_0^t (t-s)a(s)ds \\ &= \frac{2k_BT}{mc\tau}\left(t-\frac{1-c\tau^2}{c\tau}\right) - \frac{2k_BT}{m}\frac{e^{-\frac{t}{2\tau}}}{c^2\tau^2\Delta}\left[(\Delta c\tau^2-\Delta)\cosh\,(\frac{\Delta}{2\tau}t) + (3c\tau^2-1)\sinh(\frac{\Delta}{2\tau}t)\right],\end{aligned} \tag{3.25}$$

where $\Delta = \sqrt{1-4c\tau^2}$. We can see that Eq. (3.25) is consistent with the results given in Eq. (3.23) and (3.30) for large t

$$\langle (Y(t)-Y(0))^2\rangle|_{t\to\infty} = \lim_{t\to\infty} 2\int_0^t (t-s)a(s)ds = 6Dt - 2C. \tag{3.26}$$

Suppose the memory kernel $\xi(t)$ is the sum of several exponential modes,

$$\xi_M(t) = \sum_{i=1}^{M} c_i e^{-\frac{t}{\tau_i}}. \tag{3.27}$$

Laplace transform of $\xi(t)$ is

$$\widetilde{\xi}_M(z) = \int_0^\infty e^{-st}\xi_M(t)dt = \sum_{i=1}^{M}\frac{c_i}{z+1/\tau_i}.$$

Substitute Eq. (3.27) into Eq. (3.19), after a Laplace transform, we have

$$\widetilde{a}_M(z) = a_0[z + \widetilde{\xi}_M(z)]^{-1} = \frac{a_0}{z + \sum_{i=1}^{M} \frac{c_i}{z+1/\tau_i}}. \tag{3.28}$$

It is challenging to analytically tackle $a(t)$ for multi-mode memory kernel as we have done in Eq. (3.22). Solving VAF $a(t)$ and MSD function $\langle (Y(t)-Y(0))^2 \rangle$ for M-mode memory kernel involves roots of a polynomial of degree $M+1$. However, the Laplace transform $\widetilde{a}_M(z)$ shown in Eq. (3.28) makes it possible to investigate the properties of multi-mode memory kernel models. Moreover, the diffusion coefficient for large t can also be derived from $\widetilde{a}_M(z)$,

$$D = \frac{1}{3}\widetilde{a}_M(z)|_{z=0} = \frac{k_B T}{3m \sum_{i=1}^{M} c_i \tau_i}, \tag{3.29}$$

$$C = -\frac{d\widetilde{a}_M(z)}{dz}|_{z=0} = \frac{k_B T}{m \sum_{i=1}^{M} c_i^2 \tau_i^2}(1 - \sum_{i=1}^{M} c_i \tau_i^2). \tag{3.30}$$

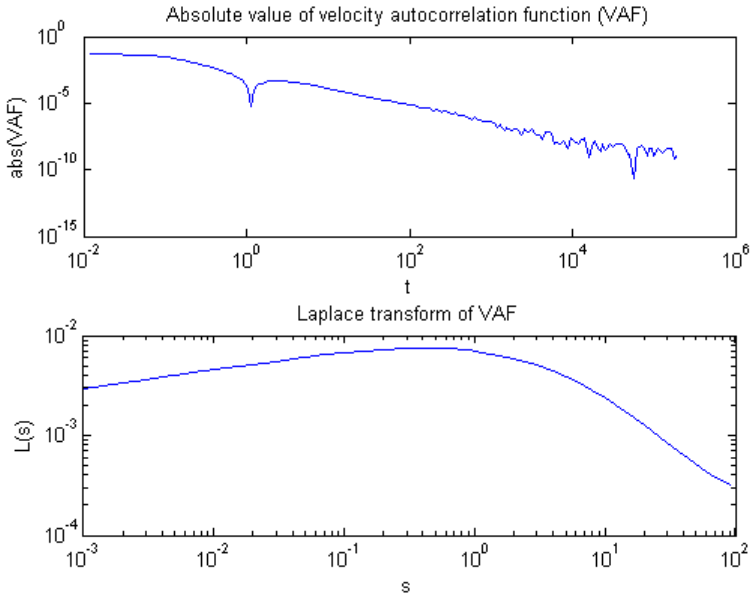


Figure 3.6: Absolute value of VAF and its Laplace transform

3.7 Experiments

We run molecular dynamics simulations using Kremer-Grest system introduced earlier in section 2.5, where the particle density is $\rho = 0.85$ and chain length $N = 64$. We simulate the system consisting of 200 linear chains in a cubic box with periodic boundary conditions. The Fig. 3.6 shows the absolute value of VAF and its Laplace transform in double logarithmic scales. Due to a lack of analytical solution of VAF for the multi-mode model, we cannot extract information about parameters directly. However, with the help of Laplace transform, one can apply numerical approximation to $\tilde{a}(z)$ to get a rough estimation of parameters a_0, c_i and τ_i, $i = 1..., M$. This will help us to set reasonable initial values of parameters in MCMC. Fig. 3.7 and Fig.

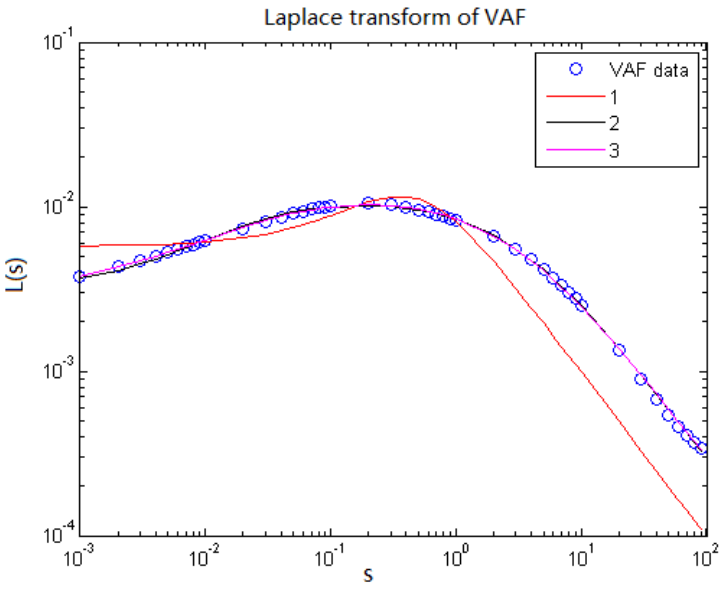


Figure 3.7: Models with different number of memory modes to fit the Laplace transform of VAF

3.8 show the goodness fit of Laplace transform of VAF $\tilde{a}(z)$ and diffusion coefficient $D(t)$ for models with different memory modes. We can see that single-mode memory kernel model has poor performance. Models with multi- mode memory kernel are capable of capturing long-term diffusive behaviours. Increasing number of modes

does not improve the goodness of fitting $D(t)$ at short time scales. It is shown in Fig. 3.8 that, though the curve has the same slope at short time scales, multi-mode models still cannot fit $D(t)$ well. This is due to the poor estimate of the initial value of the velocity auto-correlation, a_0. To improve the approximation of $D(t)$ at short time scales, an alternative would be running MD simulation with smaller time step to increase data points at very short time scales which will require much longer simulation time to get equilibrium.

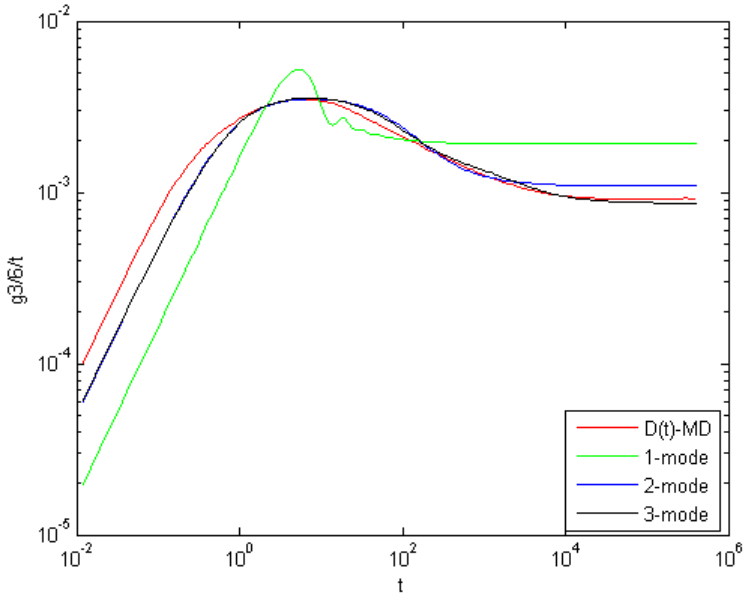


Figure 3.8: Time-dependent diffusion coefficient $D(t)$ of models with different number of memory modes

In order to capture the diffusive behaviours at short time scales, we perform least square fitting on VAF at short time scales ($t < 10$) to get numerically optimal parameters for the single mode memory kernel model. Fig. 3.9 shows the analytical solutions of the time-dependent diffusion coefficient $D(t)$ characterized by different models. The parameters used to evaluate $D(t)$ in multi-bead models are MCMC estimates for each model. Parameters of single-mode memory kernel model are obtained by numerical approximation of VAF. Fig. 3.8 and Fig. 3.9 show that multi-bead

model and multi-mode memory kernel model can captures feature of the long-term dynamics when we have enough beads (or modes). Models with memory kernel have advantages on capturing diffusive features at relatively short time scales. However, the memory kernel $\xi(t)$ is characterized by the sum of several exponential modes, we probably can not get further information on the latent system in MD simulations. While the multi-bead models are the structured-based coarse-grained models, it is possible to extract the physical descriptions about the hidden variables, e.g., entanglement effects.

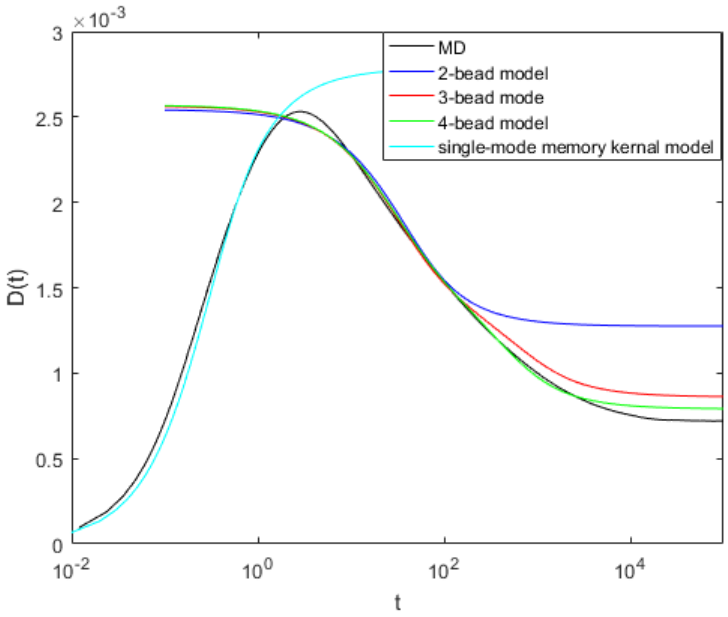


Figure 3.9: Time-dependent diffusion coefficient $D(t)$ characterized by different models; Analytical solutions obtained from 2-bead (blue), 3-bead (red) and 4-bead (green) model and single mode memory kernel model (cyan) with observation interval $\Delta_t = 12$.

3.8 Summary

In this chapter, we have investigated two different models dealing with nonlinear dynamics in order to develop coarse-grained models to study entangled polymer melts. The first model is the extension of multi-bead model developed in Chapter 2 with

nonlinear interactions. The latter model is described by the generalized Langevin equation with memory kernel. We first show how to implement particle MCMC to draw inference on unknown parameters for the extension of multi-bead model. We introduce resampling schemes to reduce the variance in the particle weights which will help us to avoid filter degeneracy. A set of resampling strategies and their advantages together with limitations are discussed. A more generic model associated with observation error is proposed and the impact of the variance of the observation noise has been investigated.

In order to evaluate the efficiency of the particle MCMC, we apply both standard MCMC and particle MCMC on the linear toy model with Gaussian model error. We find that particle MCMC ends up with noisy likelihood which is introduced by the resampling scheme and the fact that the number of particles is not large enough. It is worth to note that, in some cases, huge computational complexity will arise as the number of particles increases.

Some recent work on "Coupling of Particle Filters" [41] and "The Correlated Pseudo-Marginal Method" [42] have been devoted to getting around this issue by making the errors in nearby estimates be correlated. However, it turns out that these algorithms lack generality and schemes obtained from one model are not suitable to apply to other models directly.

In the second model, we construct a four-layer hidden Markov process. A standard particle MCMC method can be used to explore parameter space. However, we have demonstrated that it is challenging to implement particle MCMC to explore the parameter space due to the noisy likelihood. An alternative method can be Approximate Bayesian Computation (ABC) method [40]. ABC is very useful in cases where the likelihood is not tractable or it is computationally expensive to evaluate.

If the velocity autocorrelation function is available, one can apply Laplace transform and perform numerical approximation or ABC to get the estimates of unknown parameters. We compare the linear multi-bead model and the model with memory

kernel and find that for given MD simulations, the former is in good agreement with the MD simulation on the center of mass diffusion at large time scale. While the latter model shows excellent agreement with the MD simulation at very short time scales. It is worth to note the data for modelling is extracted from simulations of Kremer-Grest model where the dynamics at very short time scales is of less importance compared to that at large time scales. Therefore, the multi-bead models which are capable of capturing long-term dynamics are more promising than the latter model.

Chapter 4

Application of Monte Carlo Methods for Studying Polymer Entanglements

In this chapter, we introduce a new method for detecting entanglements in polymer melts obtained from molecular dynamics simulations based on the slip-spring model and the Bayesian inference. A set of slip-springs will be introduced to each target polymer for modelling the entanglements along its backbone. The total number, the effective lengths and the locations of the anchor points of the slip-springs will be decided by different Bayesian statistical methods. The Bayesian alternative is capable of computing the posterior distribution of different models and also provide uncertainty analysis on the estimation of model parameters.

4.1 Introduction

The dynamics of long polymer chains in melt or concentrated solutions is governed by intermolecular interactions [1] [43]. The uncrossibility between the chains significantly slows down their dynamics and leads to more complicated rheological properties than in dilute solutions. There is long-term discussion on the phenomenon of entanglements in polymer physics. The fundamental questions to answer include

what is the microscopic picture of entanglements and what is the best framework to quantitatively describe entanglement effects. The work we have done may contribute to answering these questions.

Tube model is considered as the most successful theoretical model so far for studying the dynamics of entanglements [1] [44] [45]. It introduces the mean-filed concept of the tube which represents the topological constraints imposed by the surrounding chains on a target chain as a tube. The motion of the chain is thus considered as being confined in the tube-like region. By doing so, the tube model successfully represents a multi-chain problem with a single-chain model. The tube model assumes that at small time scales, where t is smaller than the Rouse time of an entanglement strand τ_e, the dynamics of the chain shows Rouse-like behaviour. For $t \gg \tau_e$, the motion of the chain follows one-dimensional Rouse motion along the tube. A fully quantitative analysis of this 1D motion can be found in [46].

In addition, the effects of constraint release (CR) and contour length fluctuations (CLF) are taken into account. Since the tube is made from the neighbouring chains which are fluctuating and moving around, the tube should also be allowed to move around. CLF is the fluctuation of the primitive path length or contour length of the tube, and CR describes the effect of the finite lifetime of the topological constraints, arising from the motion of the surrounding chains. The most detailed treatment of CR Rouse motion is the self-consistent theory developed by Rubinstein and Colby where the mobilities of the Rouse tube segments are assigned randomly following a probability distribution determined from the tube survival relaxation spectrum [47]. Likhtman and McLeish incorporated this approach into their tube theory together with the improved treatment of CLF and the consideration of the longitudinal stress relaxation along the tube [46].

Tube models are successful in providing qualitative and sometimes semi-quantitative predictions on many dynamic properties of entangled polymers [1] [48]. However, its quantitative prediction power is still limited mainly due to the incomplete description

of constraint release effects, which requires the microscopic picture of entanglements. There are several numerical methods for detecting entanglements. Everaers et al. developed the primitive path analysis (PPA) method [6], in which the contour lengths of the tubes are found by turning off the thermal motion of the monomers, eliminating excluded volume interactions between monomers on the same chain and keeping the ends of the chains fixed while maintaining interchain uncrossability. All the chains end up with shrinking to the shortest paths with many kinks.

Likhtman introduced a microscopic definition of polymer entanglements [12]. The basic idea is to create a contact map between the mean paths of each pair of chains (i, j). If the contact persists for a long enough period of time, e.g., $3\tau_e$, the two chains are considered to be entangled. Note that under this definition, the number of entanglements may show some quantitative, but not qualitative, change depending on the cut-off distance used for constructing in the contact map and the smallest lifetime of the persistent contact [13]. The results of the entanglement analysis are typically introduced into the existing theoretical models and then tested against experimental data.

4.2 The Slip-Spring Model

The slip-spring model, developed by Likhtman [49], is a computationally efficient method for studying dynamic properties of entangled polymers to much large time and length scales than the MD simulations based on fine- or coarse-grained bead-spring models.

The slip-spring model is a single-chain model for describing the dynamics of entangled polymers. The chain of interest is represented by a Rouse chain of $N + 1$ beads. The effects of the entanglements is described by a set of virtual springs of N_s monomers. On average, there is a slip-link every N_e monomers. One end of each of these springs is fixed in space, and the other end is connected to the Rouse chain by

a slip-link. The chain can only slide through these slip-links which ensures that the long time dynamics is dominated by reptation. This model contains all mechanisms included in the tube model. Apart from the parameters describing the Rouse model, there are extra three adjustable parameters related to entanglements: N_e, N_s, and ξ_s, which is the friction of the slip-links when sliding along the chain. The values of N_s and N_e can be tuned to fit the slip-springs model to MD data. The value of ξ_s is usually fixed at $0.1\xi_0$ [49], where ξ_0 is the bead friction of the Rouse chain.

The stochastic equations of motion for both the beads and the slip-links can be derived from the Hamiltonian of the system. Disentanglement of the chain from a slip-link and reentanglement with a slip-link are accomplished by the destruction and the creation of slip-links at the chain ends. Constraint release is included by approximating the entanglements as binary events. The total number of entanglements is a constant during the simulations. This is reasonable for a large number of chains at equilibrium; in practice, the large number translates to running the simulations with at least 10 chains that affect each other's dynamics only through CR. Some of the mechanisms, like reptation, CR, and the sliding movement of the slip-links, can be individually deactivated to isolate their contribution to the relaxation of the chains. This is essential in the current work for separating out the CR effects on the segmental dynamics. The slip-springs equations of motion are solved numerically to the desired accuracy and with no uncontrolled approximations by means of Brownian dynamics simulations. Further details of the model can be found in [49] [50].

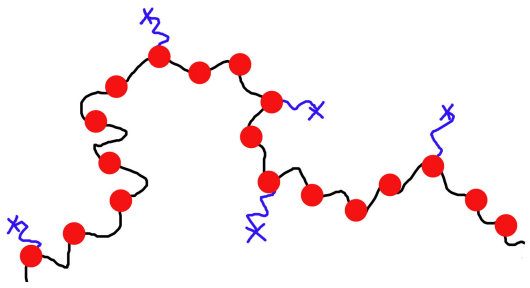


Figure 4.1: The slip-spring model

In this work, we study melt systems of concatenated polymer rings. The reason we are interested in ring polymers rather than linear chains is that in the linear chains systems, the entanglements continuously destroyed and created, while in concatenated ring polymers, the number of entanglements is fixed, making it easier for testing of the analysis method.

For a free ring polymer with N beads, the probability distribution function of the bead positions $\{\mathbf{R}_i\}_{i=0}^{N-1}$ is solely determined by the harmonic potential energy,

$$P(\{\mathbf{R}_i\}) = \frac{1}{Z_0}\exp\left[-\frac{3}{2b^2}\left(\sum_{i=0}^{N-2}\left(\mathbf{R}_{i+1}-\mathbf{R}_i\right)^2+\left(\mathbf{R}_{N-1}-\mathbf{R}_0\right)^2\right)\right], \tag{4.1}$$

where the partition function of the free ring is given by

$$Z_0 = \int d\mathbf{R}_0 \int d\mathbf{R}_1 .. \int d\mathbf{R}_{N-1}\exp\left[-\frac{3}{2b^2}\left(\sum_{N=0}^{N-2}\left(\mathbf{R}_{i+1}-\mathbf{R}_i\right)^2+\left(\mathbf{R}_{N-1}-\mathbf{R}_0\right)^2\right)\right]. \tag{4.2}$$

In a melt of ring polymers with the degree of polymerization much larger than the entanglement length N_e ($N \gg N_e$), each ring is entangled with surrounding rings due to uncrossibility. In the framework of the slip-springs model, the topological constraint on a given polymer ring can be represented by a set of slip-springs. Each slip-spring is modelled as a Rouse chain of N_s bonds with one end of the chain fixed in space called anchor points while the other end can slide along the polymer chain. In this work, we adopt a discrete version of the slip-springs model, meaning that the slipping-ends or slip-links are sitting on the individual monomers of the polymer chain and can only change their positions from one monomer to another.

The potential energy of a slip-spring with one end sitting on a monomer at position $\boldsymbol{r}_i$ and the other end at the anchor points $\boldsymbol{a}_i$ can be written as

$$U_i = \frac{3k_BT}{2N_sb^2}(\boldsymbol{r}_i-\boldsymbol{a}_i)^2, \tag{4.3}$$

where N_s is the degree of polymerization of the spring.

4.2.1 Partition Function

Suppose that there are M slip-springs restricting the motion of a ring polymer. The j^{th} slip-spring has its anchor point at $\boldsymbol{a}_j$, length of the virtual slip-spring is N_{s_j}. Considering the location that the end of the slip-spring, namely the slip-link, sits on, it is important to note that Likhtman introduced a continuous variable $x = 0...N-1$ along the chain which allows each slip-link to be anywhere on the chain [49]. In that work, Likhtman calculated the distribution of the anchoring points and developed stochastic equation of motion for the chain and the slip-links position along the chain. We adapt a discrete version of the model described in [49] by assuming that the slipping-end can only sit on particular monomer with coordinate $\mathbf{R}_{s_j}$, where s_j is the monomer number. The slipping-end or slip-link can only move from one monomer to another in a discrete way s_j, such that s_j is an integer in between 0 and $N-1$. For a given set of anchor point positions $\boldsymbol{a} = \{\boldsymbol{a}_j\}_{j=1}^{M}$ and length of slip-spring $\boldsymbol{N}_s = \{N_{s_j}\}_{j=1}^{M}$, the probability distribution function of the dynamic variables $(\boldsymbol{R}, \boldsymbol{s})$ is

$$P\left(\boldsymbol{R}, \mathbf{s}|\boldsymbol{a}\right) = \frac{1}{Z(\boldsymbol{a})} \exp\left[-\frac{3}{2b^2}\left(\sum_{i=0}^{N-2}\left(\mathbf{R}_{i+1}-\mathbf{R}_i\right)^2 + \left(\mathbf{R}_{N-1}-\mathbf{R}_0\right)^2 + \sum_{j=1}^{M}\frac{\left(\mathbf{R}_{s_j}-\boldsymbol{a}_j\right)^2}{N_{s_j}}\right)\right], \tag{4.4}$$

where $\mathbf{s} = \{s_j\}_{j=1}^{M}$, $\boldsymbol{R} = \{\mathbf{R}_j\}_{j=0}^{N-1}$. For simplicity, we denote the partition function for given $\boldsymbol{a}$ as

$$Z(\boldsymbol{a}) = \sum_{s_1=0}^{N-1}\sum_{s_2=0}^{N-1}...\sum_{s_M=0}^{N-1} Z_{ns}(\boldsymbol{a}, \mathbf{s}). \tag{4.5}$$

where $Z_{ns}(\boldsymbol{a}, \mathbf{s})$ is the partition function for given $\boldsymbol{a}, \boldsymbol{N}_s$ and $\boldsymbol{s}$, which can be written as

$$\begin{aligned} Z_{ns}(\boldsymbol{a}, \mathbf{s}) = &\int d\mathbf{R}_0 \int d\mathbf{R}_1 .. \int d\mathbf{R}_{N-1} \times \\ &\exp\left[-\frac{3}{2b^2}\left(\sum_{i=0}^{N-2}\left(\mathbf{R}_{i+1}-\mathbf{R}_i\right)^2 + \left(\mathbf{R}_{N-1}-\mathbf{R}_0\right)^2 + \sum_{j=1}^{M}\frac{\left(\mathbf{R}_{s_j}-\boldsymbol{a}_j\right)^2}{N_{s_j}}\right)\right]. \end{aligned} \tag{4.6}$$

In the case that both the anchor point positions $\boldsymbol{a}$, length of slip-spring $\boldsymbol{N}_s$ and the monomers that the slip-links sit on $\boldsymbol{s}$, are fixed, e.g., in the Warner-Edwards model, the probability distribution function of $\boldsymbol{R}$ can be written as

$$P\{\boldsymbol{R}|\boldsymbol{a},\mathbf{s}\} = \frac{1}{Z_{ns}(\boldsymbol{a},\mathbf{s})}\times \exp\left[-\frac{3}{2b^2}\left(\sum_{i=0}^{N-2}(\mathbf{R}_{i+1}-\mathbf{R}_i)^2) + (\mathbf{R}_{N-1}-\mathbf{R}_0)^2 + \sum_{j=1}^{M}\frac{(\mathbf{R}_{s_j}-\boldsymbol{a}_j)^2}{N_{s_j}}\right)\right]. \quad (4.7)$$

To compute $Z_{ns}(\boldsymbol{a},\mathbf{s})$, one can separate the integrals in Eq. (4.6) over the monomer coordinates $\mathbf{R}_j$ into two groups. One group contains the coordinates of the monomers that are connected to slip-links, namely $\mathbf{R}_{s_j}, j = 1,2,...,M$. The other group contains bead positions without slip-links.

Assuming that no bead has more than one slip-link, the M monomers with slip-links divided the ring polymer into M segments. The segment between monomer numbers s_j and s_{j+1} has $n_j = s_{j+1} - s_j$ bonds or $n_j + 1$ monomers (assuming that $s_1 < ... < s_{i-1} < s_i < ... < s_M$). For convenience, we renumber these monomers as $i = 0,1,2,...,n_j$. The integration over the coordinates of the monomers without slip-springs can be done in the same way as for an ideal n_j bonds with two end monomers fixed at $\mathbf{R}_0$ and $\mathbf{R}_{n_j}$. Recall the probability distribution function given in Eq. (1.14), we have

$$\int d\mathbf{R}_1 .. \int d\mathbf{R}_{n_j-1} \exp\left[-\frac{3}{2b^2}\left(\sum_{i=0}^{n_j-1}(\mathbf{R}_{i+1}-\mathbf{R}_i)^2)\right)\right] = \left(\frac{2\pi b^2}{3}\right)^{\frac{dn_j}{2}}\left(\frac{3}{2\pi n_j b^2}\right)^{d/2}\exp\left[-\frac{3}{2n_j b^2}\left(\mathbf{R}_{n_j}-\mathbf{R}_0\right)^2\right], \quad (4.8)$$

where d is the space dimension. The expression in Eq. (4.8) is the probability distribution function of the end-to-end distance of an ideal chain of n_j bonds with two end-

monomers positioned at $\mathbf{R}_0$ and $\mathbf{R}_{n_j}$ multiplying by a normalization factor $\left(\frac{2\pi b^2}{3}\right)^{\frac{dn_j}{2}}$.

Therefore the integration over the positions of the monomers not connected to slip-links can be replaced by the multiplication of the probability distribution functions of the end-to-end distances of M chain segments. The partition function $Z_{ns}(\boldsymbol{a}, \boldsymbol{s})$ in Eq. (4.6) then takes the form of

$$\begin{aligned} Z_{ns}(\boldsymbol{a}, \mathbf{s}) = & \left(\prod_{j=1}^{M} n_j\right)^{-d/2} \left(\frac{3}{2\pi b^2}\right)^{Md/2} \int d\mathbf{R}_{s_1} .. \int d\mathbf{R}_{s_M} \cdot \\ & \exp\left[-\frac{3}{2b^2}\left(\sum_{j=1}^{M-1} \frac{1}{n_j}\left(\mathbf{R}_{s_{j+1}} - \mathbf{R}_{s_j}\right)^2 + \frac{1}{n_M}\left(\mathbf{R}_{s_M} - \mathbf{R}_{s_1}\right)^2 + \sum_{j=1}^{M} \frac{1}{N_{s_j}}\left(\mathbf{R}_{s_j} - \boldsymbol{a}_j\right)^2\right)\right] \\ = & \left(\prod_{j=1}^{M} n_j\right)^{-d/2} \left(\frac{3}{2\pi b^2}\right)^{Md/2} \int d\mathbf{R}_{s_1} .. \int d\mathbf{R}_{s_M} \exp\left[-\beta\left(\sum_{i,j}^{M} A_{ij}\mathbf{R}_{s_i} \cdot \mathbf{R}_{s_j} + \sum_{i=1}^{M} b_i \cdot \mathbf{R}_{s_i} + c\right)\right], \end{aligned} \tag{4.9}$$

where we have dropped the normalization factor $(\frac{2\pi b^2}{3})^{\frac{dN}{2}}$ which is constant for a given Rouse ring. In the second step, we have defined

$$\beta = \frac{3}{2b^2}, \qquad b_i = -\frac{2}{N_{s_i}}\boldsymbol{a}_i, \qquad c = \sum_{i=1}^{M} \frac{\boldsymbol{a}_i^2}{N_{s_i}},$$

$$\boldsymbol{A} = \begin{pmatrix} \frac{1}{n_1} + \frac{1}{n_M} + \frac{1}{N_{s_1}} & -\frac{1}{n_1} & 0 & \cdots & 0 & -\frac{1}{n_M} \\ -\frac{1}{n_1} & \frac{1}{n_1} + \frac{1}{n_2} + \frac{1}{N_{s_2}} & -\frac{1}{n_2} & \cdots & 0 & 0 \\ 0 & -\frac{1}{n_2} & \frac{1}{n_2} + \frac{1}{n_3} + \frac{1}{N_{s_3}} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \cdots & \frac{1}{n_{M-2}} + \frac{1}{n_{M-1}} + \frac{1}{N_{s_{M-1}}} & -\frac{1}{n_{M-1}} \\ -\frac{1}{n_M} & 0 & \cdots & \cdots & -\frac{1}{n_{M-1}} & \frac{1}{n_{M-1}} + \frac{1}{n_M} + \frac{1}{N_{s_M}} \end{pmatrix}. \tag{4.10}$$

Especially in the case of $M = 2$, matrix $\boldsymbol{A}$ can be simplified as follows:

$$\boldsymbol{A} = \begin{pmatrix} \frac{1}{n_1} + \frac{1}{n_M} + \frac{1}{N_{s_1}} & -\frac{1}{n_1} - \frac{1}{n_M} \\ -\frac{1}{n_1} - \frac{1}{n_M} & \frac{1}{n_1} + \frac{1}{n_M} + \frac{1}{N_{s_M}} \end{pmatrix}. \tag{4.11}$$

In the simplest model with $M = 1$, without the normalization factor $(\frac{2\pi b^2}{3})^{\frac{dN}{2}}$, the partition function $Z_{ns}(a, s)$ can be written as

$$\begin{aligned} Z_{ns}(\boldsymbol{a}, \mathbf{s})_{M=1} &= \left(\frac{1}{N}\right)^{d/2} \left(\frac{3}{2\pi b^2}\right)^{d/2} \int d\mathbf{R}_{s_1} \exp\left[-\frac{3}{2b^2}\frac{(\mathbf{R}_{s_1} - \boldsymbol{a}_1)^2}{N_{s_1}}\right] \\ &= \left(\frac{1}{N}\right)^{d/2} \left(\frac{3}{2\pi b^2}\right)^{d/2} (2\pi b^2 N_{s_1}/3)^{d/2} \\ &= \left(\frac{N_{s_1}}{N}\right)^{d/2}. \end{aligned} \tag{4.12}$$

The physical meaning of the elements in matrix $\boldsymbol{A}$ can be understood from the effective elasticity of the chain segments or slip-springs connected to the monomers $\boldsymbol{s}$ where the slip-links sit on. For example, the diagonal term A_{jj} indicates that monomer s_j is connected with two segments of lengths n_j and n_{j+1} along the polymer and a slip-spring of length N_{s_j}. The elastic contributions from the three components can be represented by a virtual chain whose effective elasticity equals to the sum of that of the three elastic springs in parallel. The off-diagonal terms are simply the elasticity of the chain segments connecting monomers s_j and s_{j+1}. The elements A_{1M} and A_{M1} come from the connectivity of the ring polymers.

The integral in Eq. (4.9) can be conveniently computed by separating the variables in each direction and using the n-dimensional Gaussian integral for the scalar form of the integrals

$$\int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}\sum_{i,j=1}^{M} A_{ij}x_i x_j\right) d^M\boldsymbol{x} = \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}\right) d^M\boldsymbol{x} = \sqrt{\frac{(2\pi)^M}{\det\boldsymbol{A}}},$$

or, more precisely, the n-dimensional Gaussian integral with linear term

$$\int \exp\left(-\frac{1}{2}\sum_{i,j=1}^{M} A_{ij}x_i x_j + \sum_{i=1}^{M} b_i x_i\right) d^M\boldsymbol{x} = \sqrt{\frac{(2\pi)^M}{\det\boldsymbol{A}}} \exp\left(\frac{1}{2}\boldsymbol{b}^T\boldsymbol{A}^{-1}\boldsymbol{b}\right),$$

because $\boldsymbol{A}$ is a symmetric positive-definite matrix. Noting that A_{ij} and b_i in these Gaussian integrals differ from those used in Eq. (4.9) by a factor of 2β and $-\beta$, respectively, we obtain the partition function

$$Z_{ns}(\boldsymbol{a},\mathbf{s},M) = \left(\prod_{j=1}^{M} n_j\right)^{-d/2} \left(\frac{3}{2\pi b^2}\right)^{Md/2} (\det\, \boldsymbol{A})^{-d/2} \left(\frac{\pi}{\beta}\right)^{Md/2} \exp\left(\frac{1}{4}\beta \sum_{i,j=1}^{M} (b_i A_{ij}^{-1}) \cdot b_j - \beta c\right)$$
$$= \left(\prod_{j=1}^{M} n_j\right)^{-d/2} (\det\, \boldsymbol{A})^{-d/2} \exp\left[\frac{3}{2b^2}\left(\sum_{i,j=1}^{M}\left(\frac{\boldsymbol{a}_i}{N_{s_i}} A_{ij}^{-1}\right) \cdot \frac{\boldsymbol{a}_j}{N_{s_j}} - \sum_{i=1}^{M} \frac{\boldsymbol{a}_i^2}{N_{s_i}}\right)\right], \quad (4.13)$$

where the constant factor of $\left(\frac{2\pi b^2}{3}\right)^{\frac{Md}{2}}$ has been cancelled with $\left(\frac{\pi}{\beta}\right)^{Md/2}$. Note that in Eq. (4.13), it is not necessary to calculate the inverse of matrix A, the term $A^{-1}\boldsymbol{a}$ can be determined by solving the system of equations $A\mathbf{u} = \boldsymbol{a}$, from which we obtain $A^{-1}\boldsymbol{a} = \mathbf{u}$.

4.3 Bayesian Inference on the Slip-Spring Model

Fixed Dimensional Problem

For given number of slip-springs M, parameters which need to be estimated are the positions of the anchor points $\boldsymbol{a} = \{\boldsymbol{a}_j\}_{j=1}^{M}$, length of each slip-spring $\boldsymbol{N}_s = \{N_{s_j}\}_{j=1}^{M}$. Here the indices of the ring monomers $\boldsymbol{s} = \{s_j\}_{j=1}^{M}$, where the spring-links sit on, can be thought as being a set of hidden variables. In such fixed dimensional problem, the set of unknown parameters reduces from $\{\boldsymbol{a}, \boldsymbol{N}_s, M\}$ to $\{\boldsymbol{a}, \boldsymbol{N}_s\}$. Using trajectories form MD simulations, standard MCMC algorithm can be employed to investigate the posterior distribution $P(\boldsymbol{a}, \boldsymbol{N}_s|\boldsymbol{R})$.

In Bayesian statistics, the posterior probability $P(\boldsymbol{a}, \boldsymbol{N}_s|\boldsymbol{R})$ of the slip-spring parameters $\{\boldsymbol{a}, \boldsymbol{N}_s\}$ given chain coordinates $\boldsymbol{R}$ is related to the likelihood $P(\boldsymbol{R}|\boldsymbol{a}, \boldsymbol{N}_s)$ and the prior $P(\boldsymbol{a}, \boldsymbol{N}_s)$ as

$$P(\boldsymbol{a}, \boldsymbol{N}_s|\boldsymbol{R}) = \frac{P(\boldsymbol{R}|\boldsymbol{a}, \boldsymbol{N}_s)P(\boldsymbol{a}, \boldsymbol{N}_s)}{P(\boldsymbol{R})}. \quad (4.14)$$

Recall in Eq. (1.31), the posterior probability can be written in the form of

Posterior probability $\propto$ Likelihood $\times$ Prior probability.

The prior distribution of anchor point a is assumed to be Gaussian centered at the chain centre of mass. Prior of slip-spring length is Gamma distribution.

In standard MCMC simulation with given M, the acceptance probability of move only depends on the ratio of the posterior distribution before and after the move. $P(\boldsymbol{R})$ do not enter explicitly in the acceptance criterion. If we propose a move within the model, an MCMC step involves proposing a new anchor point $\widetilde{\boldsymbol{a}}_j$, tuning length of slip-spring and adjusting slip-link position $\widetilde{s}_j$ for the j^{th} slip-spring, then we will accept it with the probability

$$\min\left(1, \frac{P(\widetilde{\boldsymbol{a}}, \widetilde{\boldsymbol{N}}_s|\boldsymbol{R})}{P(\boldsymbol{a}, \boldsymbol{N}_s|\boldsymbol{R})}\right) = \min\left(1, \frac{P(\boldsymbol{R}|\widetilde{\boldsymbol{a}}, \widetilde{\boldsymbol{N}}_s)}{P(\boldsymbol{R}|\boldsymbol{a}, \boldsymbol{N}_s)} \cdot \frac{P(\widetilde{\boldsymbol{a}}, \widetilde{\boldsymbol{N}}_s)}{P(\boldsymbol{a}, \boldsymbol{N}_s)}\right). \tag{4.15}$$

According to Eq. (4.7), the ratio of the likelihoods can be written as

$$\frac{P(\boldsymbol{R}|\widetilde{\boldsymbol{a}}, \widetilde{\boldsymbol{s}})}{P(\boldsymbol{R}|\boldsymbol{a}, \boldsymbol{s})} = \frac{Z_{ns}(\boldsymbol{a}, \boldsymbol{s})}{Z_{ns}(\widetilde{\boldsymbol{a}}, \widetilde{\boldsymbol{s}})} \exp\left[-\frac{3}{2b^2}\sum_{j=1}^{M}\left(\frac{(\widetilde{\boldsymbol{R}}_{\widetilde{s}_j} - \widetilde{\boldsymbol{a}}_j)^2}{\widetilde{N}_{\widetilde{s}_j}} - \frac{(\boldsymbol{R}_{s_j} - \boldsymbol{a}_j)^2}{N_{s_j}}\right)\right]. \tag{4.16}$$

There are three mainly types of trial moves in the standard MCMC propagation algorithm:

- Changing the anchor point position $\boldsymbol{a}_{s_j}$ of a randomly selected slip-spring
- Changing Rouse chain length N_{s_j} of a randomly selected slip-spring
- Changing the monomer index s_j of the slipping-end of a randomly selected slip-spring

Note that the $\boldsymbol{a}_j \in R^3$, $N_{s_j} \in R$, s_j represents monomer index which the slip-spring sits on, $s_j = 0, ..., N-1$. Therefore acceptance of each MCMC attempt is determined by Eq. (4.15) and Eq. (4.16). In the case where the number of slip-springs is known, one can implement standard MCMC algorithm (see section 2.4) to explore the fixed-dimensional parameter space. In this work, we are going to focus

on drawing Bayesian inference on different model spaces. For the trans-dimensional move involving increasing/decreasing the number of slip-springs, the corresponding acceptance rate involves evaluating the ratio of posterior on different parameter space. We are going to discuss it later in section 4.5.

4.4 Model Comparison

Our goal is to perform Bayesian inference on the posterior probability of a sequence of slip-link models consisting of different number of slip-springs. In Bayesian community, when people have gathered a set data and would like to investigate the mechanism of generating the data or something of interest behind the data, they usually have a series of possible models available, and then make inference on which model is the most suitable one and what is the best parameters to represent the system; We treat all the unknowns, including the model indicator and its parameters, as random variables and aim to make inference over the uncertainty of models and parameters based on posterior distributions, which is a joint conditional distribution of all the unknowns and a given set of observed data.

Assume a countable set of models $\mathcal{M}$ are available, Bayesian model comparison problem is to make inference on model indicator m in $\mathcal{M}$ and corresponding parameter vector $\boldsymbol{\theta}_m \in \boldsymbol{\Theta}_m$, where the dimension n_m of parameter space is determined by the model indicator m. It is worth noting that n_m may vary from one model to another, and there is no need to restrict ourselves in nested models where $\boldsymbol{\Theta}_m \subset \boldsymbol{\Theta}'_m$. For the goal of making inference about the uncertainty of models, the model indicator m would be of no difference to other parameters. With given observed data Y on hands, we are going to make inference about posterior distribution $p(m, \boldsymbol{\theta}_m|Y)$ over joint unknowns $(m, \boldsymbol{\theta}_m)$.

Bayesian inference for the model comparison is based on the framework of a Bayesian hierarchical model. Suppose that we are given a prior $p(m)$ of model indicator

m. For a particular model m, the prior of parameter vector $\boldsymbol{\theta}_m$ together with likelihood $p(Y|m, \boldsymbol{\theta}_m)$ for data Y are available. In some cases, priors $p(m)$ and $p(\boldsymbol{\theta}_m)$ may not be written separately but written in the form of joint prior $p(m, \boldsymbol{\theta}_m)$ instead. It is obvious that this requires only an additional specification of $p(m, \boldsymbol{\theta}_m) = p(m) \times p(\boldsymbol{\theta}_m)$, and does not cost extra effort to deal with it.

Once we have the marginal likelihood $p(Y|m)$ of the data with given model, it is straightforward to obtain the posterior distribution $\pi(m|Y)$ up to an unknown normalizing constant by performing Bayesian inference

$$\pi(m|Y) = \frac{p(m)p(Y|m)}{\sum_{m=1}^{M} p(m)p(y|m)}, \tag{4.17}$$

where the marginal likelihood $p(Y|m)$ is an integral over $\boldsymbol{\theta}_m$

$$p(Y|m) = \int_{\boldsymbol{\theta}_m} p(\boldsymbol{\theta}_m|m)p(Y|\boldsymbol{\theta}_m, m)d\boldsymbol{\theta}_m. \tag{4.18}$$

The posterior distribution across models shows the suitability of a model m for modelling the data, and posterior odds in favour of model m over m' are given by

$$\frac{\pi(m|Y)}{\pi(m'|Y)} = \frac{p(Y|m)}{p(Y|m')} \times \frac{p(m)}{p(m')},$$

where the pairwise comparison of marginal likelihood of two models is called Bayes factor, which is written as

$$K(m, m') = \frac{p(Y|m)}{p(Y|m')}. \tag{4.19}$$

In many cases, a uniform prior distribution over models is adopted. Therefore, the relative strength of model m and m' for modelling given data y can be fully represented by Bayes factor $K(m, m')$. The larger $K(m, m')$ is, the stronger the evidence is in favour of model m over m'. Jefferys [51] provides a scale for interpretation of $K(m, m')$.

In most of the statistical models, the marginal likelihood is intractable, since this may involve high dimensional integration and the function to be integrated is

usually complicated. Note that it is not necessary to process integration to compute the marginal likelihood $p(Y|m)$ if the prior is a conjugate prior for the likelihood function. From Bayes' theorem, the posterior density of parameter $\boldsymbol{\theta}_m$ is given by the product of likelihood $p(Y|\boldsymbol{\theta}_m, m)$ and prior $p(\boldsymbol{\theta}_m|m)$, normalized by the marginal likelihood $p(Y|m)$

$$p(\boldsymbol{\theta}_m|Y, m) = \frac{p(Y|\boldsymbol{\theta}_m, m)p(\boldsymbol{\theta}_m|m)}{p(Y|m)}. \tag{4.20}$$

A conjugate prior means for a given likelihood function, the posterior distribution belongs to the same distribution family as the prior probability distribution does. To make it more specific, we take the normal distribution and the inverse gamma distribution for example. Suppose that for a given set of observation $Y = \{y_i\}_{i=1}^L$, the likelihood is given by

$$p(y_{1:n}|\boldsymbol{\theta}_m, m) = \left(\frac{1}{\sqrt{2\pi\boldsymbol{\theta}_m}}\right)^L \exp\left(-\frac{\sum_{i=1}^L (y_i - \mu)^2}{2\boldsymbol{\theta}_m}\right), \tag{4.21}$$

where μ is known and $\boldsymbol{\theta}_m$ is the unknown variance with a prior distribution of $\boldsymbol{\theta}_m \sim \mathcal{IG}(\alpha, \beta)$, which gives

$$p(\boldsymbol{\theta}_m|m) = \frac{\beta^\alpha}{\Gamma(\alpha)}\boldsymbol{\theta}_m^{-\alpha-1}\exp\left(-\frac{\beta}{\boldsymbol{\theta}_m}\right), \tag{4.22}$$

where $\Gamma(\cdot)$ denotes the gamma function. Therefore, the posterior density of $\boldsymbol{\theta}_m$ is

$$\begin{aligned}
p(\boldsymbol{\theta}_m|y_{1:L}, m) &= \frac{p(y_{1:L}|\boldsymbol{\theta}_m, m)p(\boldsymbol{\theta}_m|m)}{p(y_{1:L}|m)} \\
&= \frac{1}{p(y_{1:L}|m)}\left(\frac{1}{\sqrt{2\pi\boldsymbol{\theta}_m}}\right)^L \exp\left(-\frac{\sum_{i=1}^L (y_i - \mu)^2}{2\boldsymbol{\theta}_m}\right)\frac{\beta^\alpha}{\Gamma(\alpha)}\boldsymbol{\theta}_m^{-\alpha-1}\exp\left(-\frac{\beta}{\boldsymbol{\theta}_m}\right) \\
&= Z \times \boldsymbol{\theta}_m^{-\alpha-1-L/2}\exp\left(-\frac{\beta + \frac{1}{2}\sum_{i=1}^L (y_i - \mu)^2}{\boldsymbol{\theta}_m}\right),
\end{aligned}$$

where $Z = \frac{1}{p(y_{1:L}|m)}\left(\frac{1}{\sqrt{2\pi}}\right)^L \frac{\beta^\alpha}{\Gamma(\alpha)}$ is the normalising constant. We can see that the density kernel of the posterior is the density kernel of inverse gamma distribution, which is exact the distribution family of prior $p(\boldsymbol{\theta}_m|m)$. The normalized constant Z

is determined by the condition that the integral of a probability density must be one. Therefore, we obtain

$$A \times \int_0^{\infty} \boldsymbol{\theta}_m^{-\alpha-1-L/2} \exp\left(-\frac{\beta + \frac{1}{2}\sum_{i=1}^{L}(y_i-\mu)^2}{\boldsymbol{\theta}_m}\right) d\boldsymbol{\theta}_m = 1.$$

It can be is easily derived that $Z = \frac{\beta'^{\alpha'}}{\Gamma(\alpha')}$, where $\alpha' = \alpha + L/2$ and $\beta' = \beta + \frac{1}{2}\sum_{i=1}^{L}(y_i - \mu)^2$. Hence the marginal likelihood can be written as

$$\begin{aligned} p(y_{1:L}|m) &= \left(\frac{1}{\sqrt{2\pi}}\right)^L \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{1}{Z} \\ &= \left(\frac{1}{\sqrt{2\pi}}\right)^L \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha + L/2)}{\left(\beta + \frac{1}{2}\sum_{i=1}^{L}(y_i-\mu)^2\right)^{\alpha+L/2}}. \end{aligned} \tag{4.23}$$

In many cases, it is usually challenging to find a suitable conjugate prior for the likelihood function because the integration of marginal likelihood can be hard to evaluate in terms of the convergence. In practice, there are several efficient methods to approximate intractable marginal likelihood of a statistical model.

4.4.1 Numerical Methods to Approximate Marginal Likelihood

Laplace Approximation

Laplace Approximation is a widely-used method to approximate marginal likelihood [52]. The basic idea is to apply Gaussian approximation on the parameter posterior around the Maximum-a-Posteriori (MAP) estimate $\hat{\theta}_m$. for simplicity, let d denotes the dimension of parameter vector $\boldsymbol{\theta}_m$, we obtain

$$p(\boldsymbol{\theta}_m|Y,m) \approx (2\pi)^{-\frac{d}{2}}|A|^{\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{\theta}_m - \hat{\boldsymbol{\theta}}_m)^T A(\boldsymbol{\theta}_m - \hat{\boldsymbol{\theta}}_m)\right),$$

where A is the $d \times d$ Hessian of the log posterior $p(\boldsymbol{\theta}_m|Y,m)$ evaluated at $\hat{\boldsymbol{\theta}}_m$, written as

$$A_{ij} = -\frac{d^2}{d\theta_{mi}d\theta_{mj}} \ln p(\boldsymbol{\theta}_m|Y,m)|_{\boldsymbol{\theta}_m=\hat{\boldsymbol{\theta}}_m}.$$

Then the marginal likelihood can be written as

$$p(Y|m) = \frac{p(\boldsymbol{\theta}_m, Y|m)}{p(\boldsymbol{\theta}_m|Y,m)}.$$

Evaluating $\ln p(Y|m)$ at $\hat{\boldsymbol{\theta}}$, we obtain the Laplace approximation

$$\ln p(Y|m) \approx \ln p(\hat{\boldsymbol{\theta}}_m|m) + \ln p(Y|\hat{\boldsymbol{\theta}}_m, m) + \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|A|. \tag{4.24}$$

This approximation holds adequately when the posterior density is high-peak at the MAP estimate $\hat{\boldsymbol{\theta}}_m$. Finding $\hat{\boldsymbol{\theta}}_m$ can be done by using standard optimization algorithms such as gradient search. It also requires computing the second derivative matrix and inverting it to obtain A, which is usually hard to calculate.

Bayesian Information Criterion

Another well-known tool for statistical model selection is Bayesian Information Criterion (BIC) [53]. The BIC tries to minimize the impact of the prior, hence the MAP estimate $\hat{\boldsymbol{\theta}}_m$ is replaced with a value $\tilde{\theta}_m$ which maximizes the likelihood. This can be obtained from Laplace approximation when the prior does not dominate. In other words, the approximation is reasonable if the size of data L is much larger then the number of parameters. In this case, Matrix A grows as LA_0 for some fixed As_0 when n increases, so we have

$$\ln|A| \to \ln|nA_0| = \ln n^d|A_0| = d\ln n + \ln|A_0|.$$

Retaining only the terms that grow with n, Eq. (4.24) can be reduced to

$$\ln\, p(Y|m) \approx \ln\, p(Y|\tilde{\boldsymbol{\theta}}_m, m) - \frac{d}{2}\ln n.$$

When dealing with model selection, we pick the model with the highest BIC score. Note that there is no prior in this estimate, this approximately only valid for large n. Since large data size will decrease the dependence of posterior on the prior. when the size of data is large enough, all the posterior parameters would be well represented. Another limitation of BIC is that it cannot deal with a complex model in high dimension space. There are also several approaches available, such as Variational Bayes [54], Expectation Propagation [55].

Note that the methods mentioned above somehow require approximations, they are sensible only when the approximations are appropriate. Since the models we are dealing with are complex, we would not consider them further. Instead, we are going to look at Monte Carlo methods.

4.4.2 Monte Carlo Methods to Approximate Marginal Likelihood

Monte Carlo methods are widely used approaches to approximate probability distribution that is hard to evaluate or sample from. There are mainly three classes of Monte Carlo methods to perform the Bayesian model comparison. The first method which is also the simplest one is to estimate the marginal likelihood for each model separately, especially when there is no link between the parameters of different models. It is only valid when the number of models is relatively small since it requires to design a separate Monte Carlo algorithm for each model. If there exists a strong connection between two models, performing direct estimation on the Bayes factor is a good choice. However, when we have a large number of models to investigate, doing the pairwise comparison is not practical. We can also set the index of the model as an extra parameter, in this way, Monte Carlo methods can be used to simultaneously explore the model and parameter space. We will look at each of these three methods in the following sections.

When there exists no link or only weak link between two models, a straightfor-

ward way to perform model selection is to estimate the marginal likelihood of each model separately. For example, when performing model comparison between geometric and Poison to model discrete data. In such cases, the parameter space $\boldsymbol{\theta}_m$ is of different dimension, and the estimate of one model is of no help with estimating the other model. Hence, a natural choice would be to evaluate the marginal likelihood separately. This only works for a small number of models.

There are various methods available to implement marginal likelihood approximation, such as Chib's method, Harmonic mean estimator, Importance Sampling and Annealed Importance Sampling.

Chib's Method

The basic idea of Chib's method is to rearrange the Bayes' formula

$$p(Y|m) = \frac{p(Y|\boldsymbol{\theta}_m, m)p(\boldsymbol{\theta}_m|m)}{p(\boldsymbol{\theta}_m|Y, m)}.$$

Evaluating $\ln p(Y|m)$ at a high posterior estimate $\hat{\boldsymbol{\theta}}_m$

$$\ln p(Y|m) = \ln p(Y|\hat{\boldsymbol{\theta}}_m, m) + \ln p(\hat{\boldsymbol{\theta}}_m|m) - \ln \hat{p}(\hat{\boldsymbol{\theta}}_m|Y, m),$$

where $\hat{p}(\hat{\boldsymbol{\theta}}_m|Y, m)$ is an estimate of the posterior density $\hat{p}(\boldsymbol{\theta}_m|Y, m)$ at $\hat{\boldsymbol{\theta}}_m$.

The most challenging part of the algorithm is the estimate of $\hat{p}(\hat{\boldsymbol{\theta}}_m|Y, m)$. Chib's make it easily achieved by approximating the posterior $p(\hat{\boldsymbol{\theta}}|Y, m)$ with the MC average of samples from Gibbs sampler. With broad applicability of Gibbs sampling, Chib's method can be easily applied to high dimensional space. However, it is also caused concern about the limit of storage, which requires generating samples from a full conditional density and store the samples latter averaging.

When it comes to the case of a mixture, hidden Markov models and other models involving component identifying, the difficulty and complexity of the algorithm would rise due to the time needed for Gibbs sampler to visit all the possible components.

Harmonic Mean Estimator

When the posterior is not analytically tractable, the most common way is using MCMC methods to perform inference on posterior. However, MCMC does not lead us directly to the marginal likelihood. Newton and Raftery proposed Harmonic mean estimator [56] by applying importance sampling to obtain harmonic mean estimator. The basic idea is to obtain an estimator of marginal likelihood based on samples drawn from the posterior distribution, which approximates the estimate of $\frac{1}{p(Y|m)}$ with

$$\frac{1}{p(Y|m)} = \int_{\boldsymbol{\theta}_m} \frac{1}{p(Y|\boldsymbol{\theta}_m, m)} p(\boldsymbol{\theta}_m|Y, m) d\boldsymbol{\theta}_m. \tag{4.25}$$

Suppose samples $\boldsymbol{\theta}_m^{(1)}, \boldsymbol{\theta}_m^{(2)}, ...\boldsymbol{\theta}_m^{(K)}$ are drawn from posterior $p(\boldsymbol{\theta}_m|Y, m)$, the estimate of $\frac{1}{p(Y|m)}$ is given by the average of inverse likelihood at each sample $\boldsymbol{\theta}_m^{(i)}, i = 1, 2, ...K$,

$$\frac{1}{p(Y|m)} \approx \frac{1}{K} \sum_{i=1}^{K} \frac{1}{p(Y|\boldsymbol{\theta}_m^{(i)}, m)}. \tag{4.26}$$

Recall that the expectation of reciprocal likelihood can be written as

$$\begin{aligned} E\left(\frac{1}{p(Y|\boldsymbol{\theta}_m, m)}\right) &= \int_{\boldsymbol{\theta}_m} \frac{1}{p(Y|\boldsymbol{\theta}_m, m)} p(\boldsymbol{\theta}_m|Y, m) d\boldsymbol{\theta}_m = \int_{\boldsymbol{\theta}_m} \frac{1}{p(Y|\boldsymbol{\theta}_m, m)} \frac{p(Y|\boldsymbol{\theta}_m, m) p(\boldsymbol{\theta}_m|m)}{p(Y|m)} d\boldsymbol{\theta}_m \\ &= \frac{1}{p(Y|m)} \int_{\boldsymbol{\theta}_m} p(\boldsymbol{\theta}_m|m) d\boldsymbol{\theta}_m = \frac{1}{p(Y|m)}. \end{aligned}$$

Therefore the Law of Large Numbers guarantees the convergence of the approximation in Eq. (4.26) if one uses a sufficiently large number of samples from the posterior distribution.

However, it is easy to notice that this approximation does not work well in practice. As is known to all that in Bayesian statistics, the posterior distribution of a model is normally quite narrower than the prior and not significantly sensitive to the prior, since the prior is usually set to be broad enough to ensure it covers the region with high likelihood. The posterior would not be sensitive to the prior. Hence when we take $\pi(\boldsymbol{\theta}|Y)$ as proposal while $p(\boldsymbol{\theta})$ as the target, the estimate would have

large variance.

Importance Sampling is a efficient method to approximate the marginal likelihood. We have discussed it in section 1.7.6. Comparing with IS, MCMC has broader applicability, since we can tackle high dimensional models with a single component update. However, difficulties would arise to design a proposal that mixes the Markov chain well. In IS, it is hard to design a suitable proposal in high dimensions. Here is some generalization of IS available, such as SMC, we are going to discuss it in later section 4.7.

4.4.3 Methods for Bayesian Model Comparison

In some cases, there is a distinct link between the parameters of one model and that of another model. The estimate of one model would help with drawing inference on the other. For example, we would like to infer the number of components m of a Gaussian mixture model given a set of observations. When the inference on a model with $m = 5$ is available, which includes the estimators of the parameters' mean and variance for these 5 components. When it comes to drawing inference about $m = 6$, instead of inferring from a rather flat prior, one can make use of posterior of $m = 5$.

In these cases, rather than estimating each model separately, it might be more efficient to use what has been learned about one model to help with estimating a neighbouring model. Thus we consider such algorithms, which perform pairwise model comparison and estimate the ratio of marginal likelihoods - the Bayes' factor - directly.

We have discussed mainly three classes of methods for Bayesian model comparison:

- Apply Monte Carlo methods to explore the joint parameter and model space;
- Investigate each model separately;
- Estimate on one model based on the inference on another model.

According to the number of models of interest and the relation between models, one can choose the most suitable approach. In this thesis, we are going to look at the first two methods.

4.5 Reversible Jump MCMC

The most widely used Monte Carlo method to explore the joint model and parameter space is reversible jump MCMC (RJMCMC), which is capable of exploring the joint space if deterministic relations between the parameters of different models are available. In this way, the model index m can be considered as an auxiliary variable and RJMCMC algorithm can be viewed as an extension of the standard Monte Carlo algorithm on more general state spaces.

We notice that in slip-spring model, the entanglements are represented by a number of slip-spring. The models we are looking at are defined on similar parameters space $\{\boldsymbol{a}, \boldsymbol{N}_s, M\}$ with the different dimension n_m. If we have knowledge of posterior distribution of model with m slip-springs, the inference is useful for investigating the posterior distribution over $m-1$ and $m+1$. It would be ideal if we are able to set an invariant relation between neighbouring models. When we attempt to jump from model m to $m+1$, a straightforward way is to add a new slip-spring and meanwhile keep the existing slip-springs unchanged. In this thesis, we will briefly describe how to implement RJMCMC, further details can be found in [57] [58].

Ideally, if the marginal likelihood $p(Y|m)$ in Eq. (4.18) is tractable, the acceptance ratio of jump on model space from m to m' can be written as

$$\gamma(m, m^{'}) = \frac{p(m^{'})p(Y|m^{'})q(m|m^{'})}{p(m)p(Y|m)q(m'|m)}. \tag{4.27}$$

In reality, $p(Y|m)$ usually turns out to be intractable. Therefore, we need to find alternatives to propose jump between models. Reversible jump MCMC (RJMCMC) method is an efficient method that explores the joint posterior of model and parame-

ters. Developed by Green in 1995 [57], RJMCMC performs across-model simulation, by which the algorithm is able to make a move from one model to another. With the help of RJMCMC, we are able to construct a Markov chain with an invariant probability distribution that we are interested in. The proportion of times that different models are visited gives an estimate of posterior of each model.

Suppose we are going to explore the joint parameter and model space $(\boldsymbol{\theta}, \mathcal{M})$, and the invariant distribution of interest is the posterior $p(\boldsymbol{\theta}_m, m|Y)$. Now the challenge is to how to set up an MCMC algorithm that is capable to explore the joint space. We are going to implement RJMCMC with M-H framework (see section 1.7.5) to construct Markov chain that targets the posterior $p(\boldsymbol{\theta}_m, m|Y)$. In standard MCMC, the M-H algorithm involves moves between two states x and x', including $(x \to x')$ and inverse move $(x' \to x)$. Here the models are usually defined on different parameters spaces which are usually of different dimension. The trick to deal with this issue is to set up links between parameter spaces. In this way, we are able to view all the distributions in the same parameter space after using a transformation of parameters without introducing any bias.

It is important to note that we cannot set up unique mapping if the parameter spaces are of different dimension. To get rid of this restriction, when the move is made from a lower dimensional space to a higher dimensional space, a set of new variables are introduced. Here we are going to design a pair of jump that allows the MCMC move from one model to another. Consider a move from model m to model m', which means we propose a move from joint state $(\boldsymbol{\theta}_m, m) \to (\boldsymbol{\theta}_{m'}, m')$ and backward move $(\boldsymbol{\theta}_{m'}, m') \to (\boldsymbol{\theta}_m, m)$. For the forward move, we introduce extra random numbers u which is drawn from distribution ϕ_m. The proposed parameters $\boldsymbol{\theta}_{m'}$ can be expressed by an invertible function h_m with respect to $(\boldsymbol{\theta}_m, u)$

$$(\boldsymbol{\theta}_{m'}, u_{m'}) = h_m(\boldsymbol{\theta}_m, u_m),$$

where random numbers $u_{m'}$ is distributed from $\phi_{m'}$ which is needed for the reverse move from m' to m, using function h'_m which is the reverse function of h_m. The acceptance rate of a particular move $(\boldsymbol{\theta}_m, m) \to (\boldsymbol{\theta}_{m'}, m')$ is

$$\gamma = \min\left\{1, \frac{p(\boldsymbol{\theta}_{m'}, m'|Y)\phi_{m'}(u_{m'})q(m|m')}{p(\boldsymbol{\theta}_m, m|Y)\phi_m(u_m)q(m'|m)} J_{m\to m'}\right\}, \tag{4.28}$$

where $q(m'|m)$ the probability to propose a move from model m to m', and J is the Jacobian of the transformation h_m:

$$J_{m\to m'} = \left|\frac{\partial h_m}{\partial(\boldsymbol{\theta}_m, u_m)}\right|. \tag{4.29}$$

The standard RJMCMC update is described in Alg. 4

Algorithm 4 Reversible jump MCMC update

1. Propose trans-dimension move:
   Given state $(m, \boldsymbol{\theta}_m)$, sample $m' \sim q(\cdot|m)$ and $u_m \sim \phi_m(\cdot)$;
2. Specify the new parameters:
   $(\boldsymbol{\theta}_{m'}, u_{m'}) = h_m(\boldsymbol{\theta}_m, u_m)$
3. Evaluate ratio of target density:

$$\gamma = \min\left\{1, \frac{p(\boldsymbol{\theta}_{m'}, m'|Y)\phi_{m'}(u_{m'})q(m|m')}{p(\boldsymbol{\theta}_m, m|Y)\phi_m(u_m)q(m'|m)} J_{m\to m'}\right\}$$

4. Accept the transition with probability $\min\{1, \gamma\}$.

This algorithm allows us to explore the joint model and parameter space at the same time. By designing reversible jump over different model space, we obtain a series of samples from a joint distribution of parameters from different models. It can be seen from Eq. (4.27) that the acceptance rate involves the Bayes factor between two models. While in Eq. (4.28), we find that it evaluates the acceptance rate for a single move, which means that we are using a single point importance sampling estimator to approximate the Bayes factor, which may result with poor acceptance rate. Next, we are going to use a Gaussian mixture example to demonstrate how RJMCMC works.

4.5.1 Mixture Model

In this section, we will introduce a mixture model to show how to implement reversible jump MCMC. By allowing the number of mixtures k to vary, we can apply reversible jump MCMC to explore the posterior distribution of both m and the corresponding mixture parameters. The basic framework of a mixture model for independence observation y_i can be written as

$$y_i \sim \sum_{j=1}^{m} w_j f(\cdot|\theta_j), \quad , i = 1, \cdots, L, \tag{4.30}$$

where $f(\cdot|\theta_j)$ is the probability density function for a class of given distribution family that characterized by parameter θ_j.

The goal is to make inference on the number of components m and the corresponding parameter θ_j together with the component weight w_j. Note that the component weights are normalized with $\sum_{j=1}^{m} w_j = 1$. Assume the mixture is Gaussian with $\theta_j = \{\mu_j, \sigma_j^2\}$ where the probability density function $f(x|\theta_j)$ can be written with respect mean μ_j and variance σ_j^2

$$f(x|\mu_j, \sigma_j^2) = \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left(-\frac{(x-\mu_j)^2}{2\sigma_j^2}\right). \tag{4.31}$$

The priors on the parameters are thought to be independent. The number of mixture components m is usually taken from informative priors. In this thesis, we set uniform distribution as the prior of m. The weight w_j is taken from symmetric Dirichlet, $w_j \sim D(\delta, \delta, \cdots, \delta)$. Mean μ_j and variance σ_j^{-2} of the mixture are drawn from normal and gamma density, respectively,

$$\mu_j \sim \mathcal{N}(\xi, \kappa^2), \quad \sigma_j^{-2} \sim \Gamma(\alpha, \beta).$$

4.5.2 Birth/Death Move

Consider model state changing from $(m, \boldsymbol{\theta}_m)$ to $(m+1, \boldsymbol{\theta}^*_{m+1})$, a new component characterized by $\{w', \mu', \sigma'^2\}$ is introduced to keep bijection between models,

$$w' \sim be(1, m), \quad \mu_j \sim \mathcal{N}(\xi, \kappa^2), \quad \sigma_j^{-2} \sim \Gamma(\alpha, \beta).$$

The acceptance rate can be written as

$$\gamma = \frac{p(m+1)q(m|m+1)p(\theta^*_{m+1}|m+1)p(Y|\theta^*_{m+1}, m+1)}{p(m)q(m+1|m)p(\boldsymbol{\theta}_m|m)p(Y|\boldsymbol{\theta}_m, m)p(w', \mu', \sigma'^2)} \frac{1}{m+1} \frac{(m+1)!}{m!} J_{m\to m+1}, \tag{4.32}$$

where we have

$$\boldsymbol{\theta}_m = \left(\{w_i\}_{i=1}^m, \{\mu_i\}_{i=1}^m, \{\sigma_i^2\}_{i=1}^m\right),$$

$$\boldsymbol{\theta}^*_{m+1} = \left(\{w_i^*\}_{i=1}^{m+1}, \{\mu_i^*\}_{i=1}^{m+1}, \{\sigma_i^{*2}\}_{i=1}^{m+1}\right).$$

When generating a new component, existing weight are re-normalized with constant $(1-w')$

$$w_i^* = (1-w')w_i \text{ for } i = 1, \cdots m.$$

The corresponding mean and variance keep unchanged, that means $(\mu_i^*, \sigma_i^{*2}) = (\mu_i, \sigma_i^2)$ for $1 \le i < m+1$. The factor of $\frac{1}{m+1}$ is associated with randomly deleting one of the components when transiting from $m+1$ to m. The Jacobian of the transformation is

$$J_{m\to m+1} = (1-w')^{m-1}, \tag{4.33}$$

Then Eq. (4.32) can be rewritten as

$$\gamma = \frac{p(m+1)d_{m+1}p(Y|\boldsymbol{\theta}^*_{m+1}, m+1)}{p(m)b_m p(Y|\boldsymbol{\theta}_m, m)be_{1,m}(w')} \frac{P(\{w^*\})}{P(\{w\})}(1-w')^{m-1}, \tag{4.34}$$

where d_{m+1} is the transit probability from $m+1$ to m and b_m the probability to attempt move from m to $m+1$. The prior of model indicator M prior can be Poisson or uniform distribution. The acceptance probability for model transiting from m to

$m+1$ is

$$\gamma_{m\to m+1} = \min\{1, \gamma\}.$$

For the corresponding inverse move from $m+1$ to m, the acceptance probability is $\min\{1, \gamma^{-1}\}$.

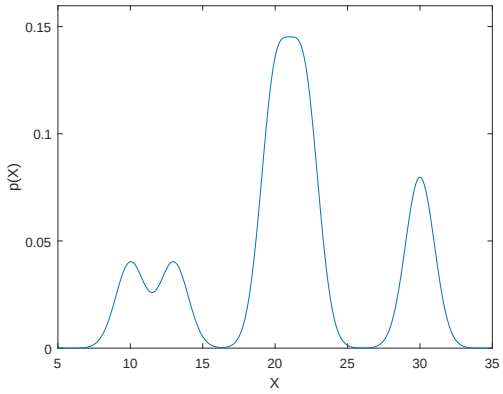


Figure 4.2: The probability density function of five modes Gaussian Mixture Model

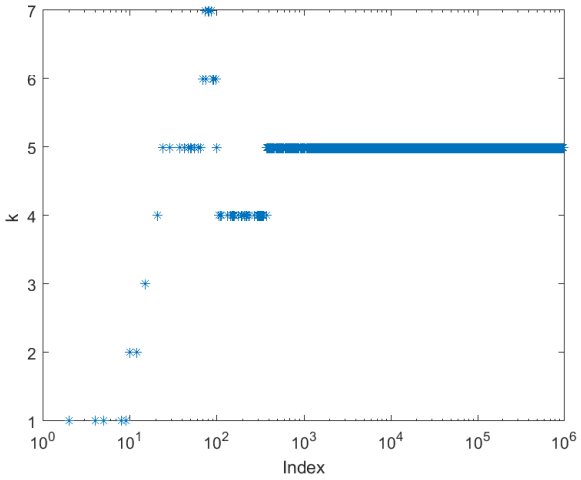


Figure 4.3: RJMCMC with birth/death move explores model space

We perform RJMCMC on Gaussian mixture model with data $\{y_i\}_{i=1}^{500}$ given by model $m = 5$ with $w = \{0.1, 0.1, 0.3, 0.3, 0.2\}$, $\mu = \{10, 13, 20, 22, 30\}$, $\sigma^2 = \{1, 1, 1, 1, 1\}$.

The probability density function of the mixture model is shown in Fig. 4.2. The prior of m is assumed to be uniform distribution $U[1, 10]$. Firstly, we randomly draw m_0 from $U[1, 10]$, and initialize component parameters $\{w_j\}_{j=1}^{m_0}$, $\{\mu_j\}_{j=1}^{m_0}$ and $\{\sigma_j^2\}_{j=1}^{m_0}$ which are randomly drawn from the priors. Fig. 4.3 shows that the RJMCMC does make some successful move between models at first.After only about 1000 iterations, the MCMC gets stuck at $m = 5$.

Table 4.1: True underlying parameters

parameters					
j	1	2	3	4	5
w_j	0.1	0.1	0.3	0.3	0.2
μ_j	10	13	20	22	30
σ_j^2	1	1	1	1	1

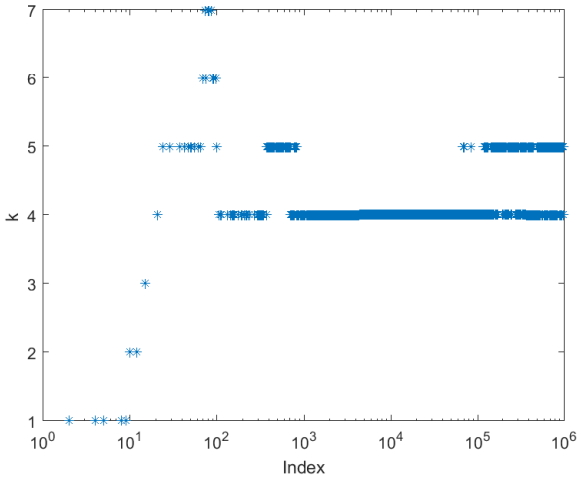


Figure 4.4: RJMCMC with birth/death move explores model space

When taking a look at the weights of each component with mixture $m = 5$, we find that one of the weights is much smaller than the others (see Table.4.2 where $j = 3$). We plot the number of significant components whose weights are larger than 1/30 as shown in Fig. 4.4. It is obvious that between MCMC iterations $t = 2000$ to 20000,

Table 4.2: Estimate of a single MCMC iteration

Estimate					
j	1	2	3	4	5
w_j	0.2025	0.1998	0.0024	0.2088	0.3865
μ_j	11.3049	30.0506	21.8627	19.7869	21.6937
σ_j^2	1.5533	1.0185	24.2553	0.7955	1.1888

Table 4.3: True underlying parameters

True parameters		
j	1	2
w_j	0.5	0.5
μ_j	10	20
σ_j^2	1	1

one of the components has extremely small weight $w_j = 0.0024$ and relatively large variance $\sigma_j^2 = 24.2553$. The moves of randomly deleting one of the components are always rejected. It is also unlikely to accept the trans-dimensional move by randomly drawing a new component from the prior. We run another toy model with parameters shown in Table 4.3. Again, it gets stuck at $m = 4$ as shown in Fig. 4.6. We extract the parameter on $m = 4$, and the estimate is given in Table 4.4.

Table 4.4: One of RJMCMC outputs

Estimate				
j	1	2	3	4
w_j	0.246	0.255	0.240	0.259
μ_j	10.022	19.747	20.276	10.027
σ_j^2	0.814	1.056	0.861	1.154

We can see that there are obvious overlaps between modes. It separates a single component in Table 4.3 into two with similar mean and variance. We believe that there is no difference between these two presentations. This four-mode model can be

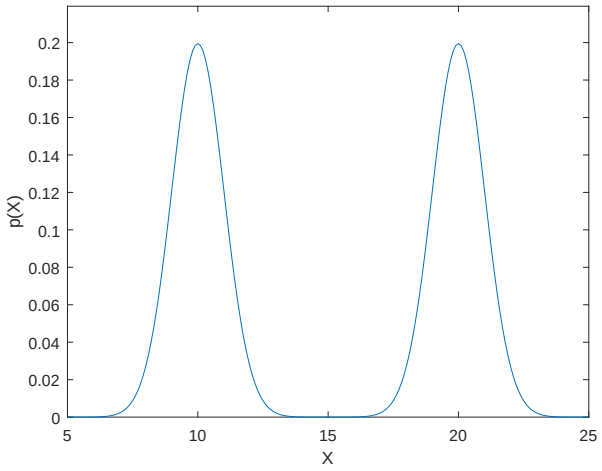


Figure 4.5: The probability density function of two modes Gaussian Mixture Model

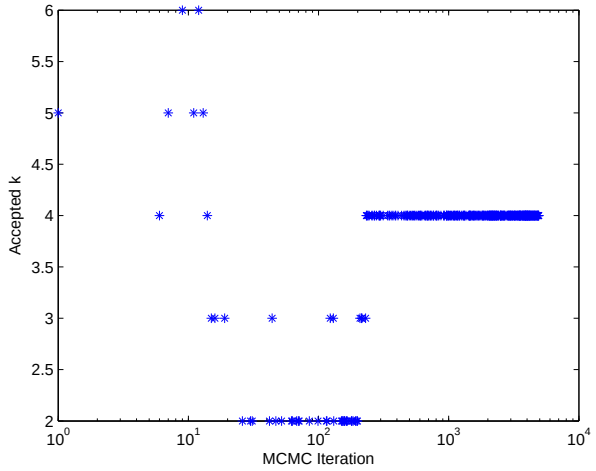


Figure 4.6: RJMCMC explores model space

fully represented by a two-mode model. In fact, we wish to use fewer free parameters to present the models, which is essential to model determination. One can either avoid this situation by setting a tolerance of minimum distance between the mean of new components or manually combining two components when we find there is nothing particularly distinguishable between these two components in certain critical standard before looking at the posterior distribution of the model.

In our case, RJMCMC with birth/death move is not efficient which takes too long to get sensible answers. The inefficiency of algorithms is due to the fact that we are using single particle IS to approximate the Bayes factor in the acceptance ratio in Eq. (4.27). Therefore, we need other efficient methods to approximate the Bayes factor properly. We will look at this by applying annealed IS methods instead of IS in section 4.6. In the next section, I am going to use a transformation to bring the proposal closer to the target. We will see that it is useful for improving the IS estimate of the Bayes factor.

4.5.3 Combine/Split Move

Instead of adding or deleting a component to propose jump between the model space, an alternative is to introduce combine/split move which can maintain the moment invariant property for the model. It would be more likely to accept the move by combining two modes into rather than randomly delete one of the modes directly. We would like to find a bijective transformation between different models while the first and second moments of the mixture models keep unchanged. The first moment of the Gaussian mixture model can be written as

$$E(X) = \int x \sum_{i=1}^{m} w_i p_i(x) dx = \sum_{i=1}^{m} w_i \mu_i, \quad \text{for all } m = 1, ..., M, \tag{4.35}$$

where the weights are normalized with $\sum_{i=1}^{m} w_i = 1$. Merge one of the components j_1 with another j_2 to get a new component j^*, while other components keep unchanged. In such case, the number of components decreases from $m + 1$ to m. The second-moment invariant property can be expressed as follows:

$$E(X^2) = \int x^2 \sum_{i=1}^{m} w_i p_i(x) dx = \sum_{i=1}^{m} w_i (\mu_i^2 + \sigma_i^2), \quad \text{for all } m = 1, ..., M.$$

The model transition between state m and $m+1$. According to moments invariant, we obtain

$$\sum_{i=1}^{m} w_{m,i}\mu_{m,i} = \sum_{i=1}^{m+1} w_{m+1,i}\mu_{m+1,i}, \tag{4.36}$$

$$\sum_{i=1}^{m} w_{m,i}(\mu_{m,i}^2 + \sigma_{m,i}^2) = \sum_{i=1}^{m+1} w_{m+1,i}(\mu_{m+1,i}^2 + \sigma_{m+1,i}^2), \tag{4.37}$$

where $w_{m,i}$, $\mu_{m,i}$, $\sigma_{m,i}^2$ are weight, mean and variance of the i^{th} component in model m. In the split move, we assume that other components will keep weighs, means, and variances unchanged, which means that most expressions will be cancelled in Eq. (4.36) and (4.37). For simplicity, we assume that component j is split into two components, j_1 and j_2. We obtain

$$\begin{cases} w_j = w_{j_1} + w_{j_2}, \\ w_j\mu_j = w_{j_1}\mu_{j_1} + w_{j_2}\mu_{j_2}, \\ w_j(\mu_j^2 + \sigma_j^2) = w_{j_1}(\mu_{j_1}^2 + \sigma_{j_1}^2) + w_{j_2}(\mu_{j_2}^2 + \sigma_{j_2}^2). \end{cases} \tag{4.38}$$

The are three degrees of freedom in achieving this move, so we need to generate a three-dimensional random vector u to specify the new parameters. Here, we use beta distributions

$$u_1 \sim be(2,2), \quad u_2 \sim be(2,2), \quad u_3 \sim be(1,1).$$

Random variables drawn from the beta distribution must satisfy that $0 < u_1, u_2, u_3 < 1$. Deduce from Eq. (4.38) and introduce dimension-matching parameters u, we have

$$\begin{aligned} w_{j_1} = w_j u_1, \quad & w_{j_2} = w_j(1-u_1), \\ \mu_{j_1} = \mu_j - u_2\sigma_j\sqrt{w_{j_2}/w_{j_1}}, \quad & \mu_{j_2} = \mu_j + u_2\sigma_j\sqrt{w_{j_1}/w_{j_2}}, \\ \sigma_{j_1}^2 = u_3(1-u_2^2)\sigma_j^2 w_j/w_{j_1}, \quad & \sigma_{j_2}^2 = (1-u_3)(1-u_2^2)\sigma_j^2 w_j/w_{j_2}. \end{aligned}$$

We run RJMCMC with combine/split move. Data is given by the underlying model in Table 4.1. The number of observations is $L = 500$. We extract 2500 iterations after RJMCMC has reached equilibrium. The trace plot of RJMCMC

and total time spent on each model are shown in Fig. 4.7. There is a significant improvement in the efficiency of trans-dimensional moves due to the benefits from the moment-matching trick in the split/combine move in the mixture case. Note that the object is to implement MCMC that is capable of jumping between model spaces in slip-spring models. Unfortunately, there is no obvious way to do anything analogous in the slip-spring case. It is challenging to find a bijective transformation to combine two slip-springs into one. We need to find alternatives to implement RJMCMC that allow us to perform trans-dimensional jump efficiently in slip-spring model when using birth/death move.

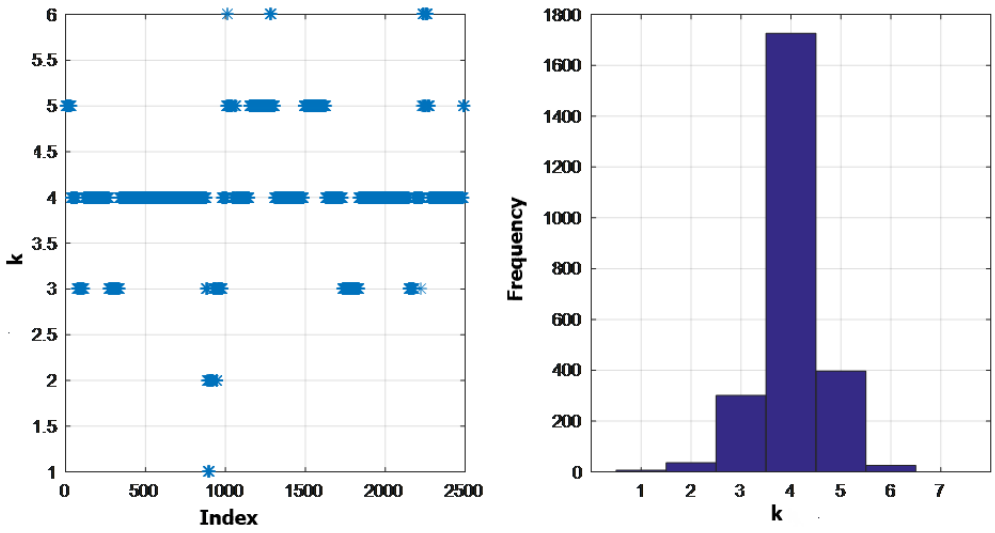


Figure 4.7: RJMCMC with combine/split move explores model space

4.6 Annealed Importance Sampling RJMCMC

We have seen from the previous section that the efficiency of RJMCMC mainly depends on the variance of the Bayes factor estimator in the acceptance ratio. Therefore, we hope to improve the efficiency of RJMCMC by using a more accurate estimator. Karagiannis and Andrieu [59] proposed to do this using annealed importance sampling (AIS) instead of IS. AIS RJMCMC is a new method aiming to implement the

reversible jump MCMC efficiently. We know that the inefficiency of RJMCMC is due to the high variance of the IS estimator of the acceptance probability, which leads to low acceptance rates for dimension changing moves. The idea of AIS is that since the variance of IS estimators depends on the distance between the proposal $q(\cdot|\theta)$ and the target π, it is possible to improve the IS estimator by placing some bridging distributions to reduce the distance between the proposal and the target.

For the transition from model m to model m', we are going to introduce a set of annealing densities that construct a smoother path from the proposal to the target. The annealing densities is a sequence of intermediate distributions $\{\rho_t(\cdot; m \to m')\}$, which bridge densities $\pi(\boldsymbol{\theta}_m|m)\phi_m(u_m)$ and $\pi(\boldsymbol{\theta}'_m|m')\phi_{m'}(u_{m'})$. Here index t is referred as time, $t = 0, \ldots, T$. The annealing densities $\{\rho_t(\cdot\ ; m \to m')\}$ is characterized by

$$\rho_0 = \pi(\boldsymbol{\theta}_m|m)\phi_m(u_m)J_{m\to m'},$$
$$\rho_T = \pi(\boldsymbol{\theta}_{m'}|m')\phi_{m'}(u_{m'}),$$

where $J_{m\to m'}$ is the Jacobian of transformation $\phi_{m\to m'}$. In this thesis, we are going to use geometric annealing distributions to construct intermediate densities $\{\rho_t(\cdot\ ; m \to m')\}$, for $0 < t < T$. Introducing $\{\gamma_t\}_0^T \in [0,1]$, with $\gamma_t = 0, \gamma_T = 1$, we define annealing densities $\{\rho_t(\cdot\ ; m \to m')\}$ as follows:

$$\rho_t = [\pi(\boldsymbol{\theta}_m|m)\phi_m(u_m)J_{m\to m'}]^{1-\gamma_t} \times [\pi(\boldsymbol{\theta}_{m`}|m')\phi_{m'}(u_{m'})]^{\gamma_t}.$$

In such case, the annealed weight can be written as

$$r_{m\to m'}^{(0:T-1)} = \prod_{t=0}^{T-1} \left(\frac{\pi(m', \boldsymbol{\theta}_{m'}^{(t)})}{\pi(m, \boldsymbol{\theta}_m^{(t)})} \frac{\phi_{m'}(u_{m'})}{\phi_m(u_m)} \right)^{(\gamma_{t+1}-\gamma_t)}. \tag{4.39}$$

More general methods to construct annealing densities can be found in [59] [60].

The bridge can be constructed easily if we could sample independently from each of the bridging distributions. By estimating the ratio between each of the successive

bridging distributions and combining them, we obtain the acceptance ratio of exploring different joint space. However, we usually cannot sample independently from each of the bridging distributions. The alternative is to use an MCMC move instead, with the initial point being the point generated from the previous intermediate distribution. Neal [20] states that the entire procedure yields an unbiased estimate of the overall ratio.

Now we will plug in the AIS estimate into RJMCMC. The initial point of the first distribution will be given by the current state of the RJMCMC $\{\boldsymbol{\theta}_m, U_{m'\to m'}, m\}$. After plugging in the AIS estimator, the acceptance probability of move from $(m, \boldsymbol{\theta}_m)$ to $\left(m', \boldsymbol{\theta}_{m'}^{(T-1)}\right)$ is given by

$$\gamma_{m\to m'} = \min\left\{1, \frac{q(m|m')}{q(m'|m)} r_{m\to m'}^{(0:T-1)}\right\}. \tag{4.40}$$

The pseudo-code of the AIS RJMCMC algorithm is presented as follows:

Algorithm 5 AIS RJMCMC transdimensional algorithm

1. Propose trans-dimension move:

Given state $(m, \boldsymbol{\theta}_m)$, sample $m' \sim q(\cdot|m)$ and $u_m \sim \phi_m(\cdot)$;

2. Specify the new parameters:

$(\boldsymbol{\theta}_{m'}, u_{m'}) = h_m(\boldsymbol{\theta}_m, u_m)$

3. Annealing procedure:

for $i = 1 : T - 1$ **do**

Sample $(\boldsymbol{\theta}_{m'}^{(t)}, u_{m'}^{(t)})$ from target density ρ_t :

$$\rho_t = [\pi(\boldsymbol{\theta}_m|m)\phi_m(u_m)J_{m\to m'}]^{1-\gamma_t} \times [\pi(\boldsymbol{\theta}_{m'}|m')\phi_{m'}(u_{m'})]^{\gamma_t}$$

endfor

4. Annealing importance weight:

Compute the annealing importance weight $r_{m\to m'}^{(0:T-1)}$

5. Accept the transition with probability $\min\{1, \alpha_{m\to m'}\}$

$$\gamma_{m\to m'} = \min\left\{1, \frac{q(m|m')}{q(m'|m)} r_{m\to m'}^{(0:T-1)}\right\}$$

By introducing a number of intermediate states between two models, we can get rid of the problem of stuck in certain model space and visit the space with lower posterior probability.

Andrieu and Vihola [23] indicate that the properties of pseudo-marginal algorithms improves when the variance of the estimator within the algorithm improves, and remark that a similar result is likely also to hold for AIS-RJMCMC compared to RJMCMC. Note that when we do more AIS steps, we may have to do fewer MCMC iterations. The results in Karagiannis and Andrieu [59] indicate that it is worthwhile to use a few bridging distributions, then the benefit drops off. It is an open question as to how to optimally choose the number of intermediate distributions, but we note that there is work on an analogous question in pseudo-marginal, in choosing the number of importance points. By doing AIS RJMCMC, we are limited to a single AIS importance point. If we use more points to approximate the ratio, we will lose the

correctness of the algorithm and end up with a noisy algorithm [61].

4.6.1 Application of AIS-RJMCMC on Mixture Model

We run AIS RJMCMC with different annealing scheme $T = 1, 2, 5, 10, 50$. To investigate the independence of the sampler using different annealing scheme, we present the integrated autocorrelation time (IAT) of each scheme in Table 4.5.

Table 4.5: IAT for different annealing time T

IAT						
t	1	2	5	10	50	100
IAT	61.75	46.26	25.29	16.77	12.45	12.83

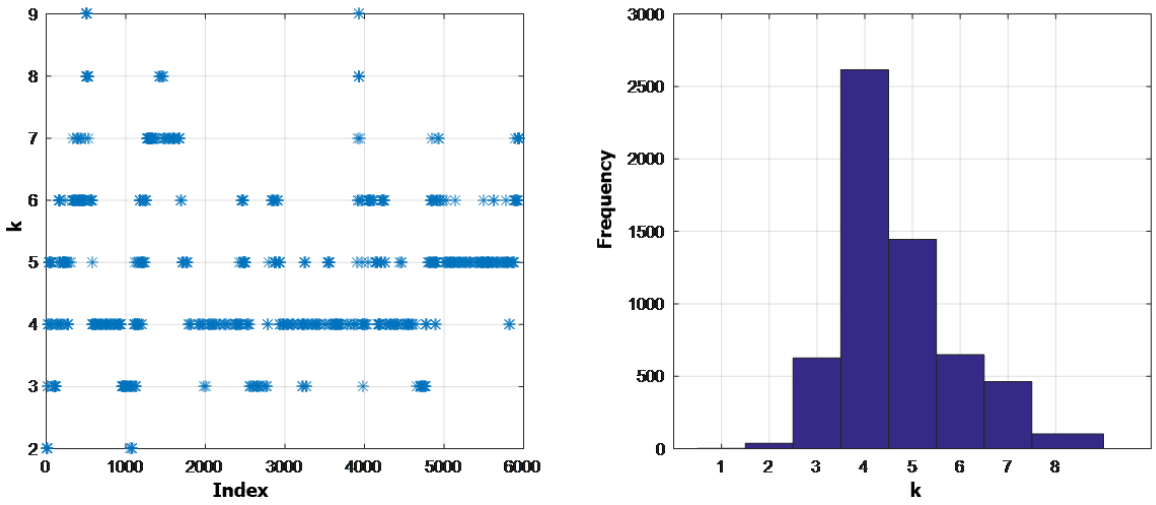


Figure 4.8: AIS RJMCMC with birth/death move explores mixture model space. Annealing time $T = 50$

IAT is a criterion indicating the decrease in rate of convergence introduced by dependent sampling. We can see that the efficiency of the algorithm does improve significantly as the anneal time increases. However, the benefit will drop off and it is not worthy increasing the anneal time comparing the benefit and the extra computational effort it requires. Fig. 4.8 shows 6000 iterations after AIS RJMCMC has reached

equilibrium. The sampler is implemented with $T = 50$ using combine/split move. Data is give by underlying model Table 4.1. Number of observations is $N = 500$.

AIS scheme significantly improve the efficiency of RJMCMC sampler. Next, we are going to apply AIS RJMCMC on the slip-spring models.

4.6.2 Application of AIS-RJMCMC on the Slip-Spring Model

We run a molecular dynamic simulation of 150 concatenated polymer rings, each polymer consists of 512 monomers. The system is initialized with particle density $\rho = 0.85$ in a cubic box with periodic boundary conditions. The beads along the chain are connected by FENE springs. Excluded volume effect between monomers is governed by purely repulsive Lennard-Jones potential. The time step of the simulation is $dt = 0.012\tau_{LJ}$. To obtain a system which consists of concatenated polymer rings, we first switch off the inter-chain excluded volume interactions which allows the chains to cross with each other. After the system is run long enough to reach proper equilibrium, we switch on the inter-chain excluded volume interactions. In this way, we manually set up the initial configuration of system which makes each ring polymer concatenate with 10 ring polymers on average. We extract the configurations of mean path of the polymers every 10^6 MD steps. The number of configurations used for analysis is $L = 500$. The parameters of interest are the number of slip-springs M, the location of anchor points $\boldsymbol{a}$ and the length of slip-spring $\boldsymbol{N}_s$. The monomer indices $\boldsymbol{s}$ that slip link sit on are hidden variables. The length of each slip-springs $\boldsymbol{N}_s$ is allow to be varied.

Parameters $\boldsymbol{a}, \boldsymbol{N}_s$ are assumed to be independent of each other. Anchor points $\boldsymbol{a}$ is supposed to be drawn from Gaussian distribution

$$p(\boldsymbol{a}_j) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\boldsymbol{a}_j - \boldsymbol{\mu})^2}{2\sigma^2}\right).$$

The length of slip-spring $\boldsymbol{N}_s$ is drawn from Gamma distribution. The prior of model index m is chosen from a uniform distribution between 1 and a pre-specific number M. In this work, we set the priors of parameters as $M = 20$, $\boldsymbol{a}_j \sim \mathcal{N}(0, 5)$ and $N_{s_j} \sim \mathcal{G}(2, 3)$. The probability $p(m'|m)$ of proposing a move from model m to m' is given by

$$q(m'|m) = \begin{cases} 1/3, & |m' - m| \leq 1, 1 < m < M \\ 1/2, & |m' - m| \leq 1, m = 1 \text{ or } m = M \\ 0, & otherwise. \end{cases}$$

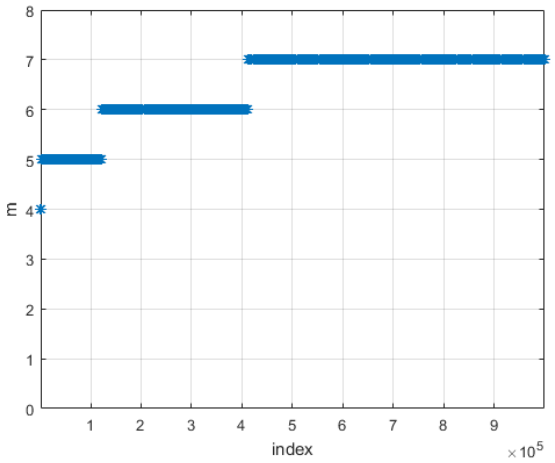


Figure 4.9: AIS RJMCMC with birth/death move explores slip-spring model space. Annealing time $T = 50$

We implement AIS RJMCMC on the slip-spring model with annealing scheme $T = 50$. Fig. 4.9 shows AIS RJMCMC iterations after 1000 steps. The total running steps is 10^6. We can see that it takes particularly long time for the sampler jumping from one model space to the other. The poor mixing property of the algorithm mainly comes from the complexity of the hidden states $\boldsymbol{s}$. When proposing a trans-dimensional move from model m to $m + 1$, it requires to infer the positions of slip links for individual configurations for the newly added slip-spring. By doing this, we introduce two parameters a_{m+1} and $N_{s_{m+1}}$ and a set of hidden varia-

bles $\{s^i_{m+1}\}, i = 1, \cdots, L$, which causes huge gap in likelihood and makes it hard to explore the joint model and parameter space. We have seen in Fig. 4.9 that the sampler gets stuck at $m = 7$. It indicates we can not construct a smooth path bridging target densities $\pi(\boldsymbol{\theta}_m|m)$ and $\pi(\boldsymbol{\theta}_{m'}|m')$ within 50 annealing steps for our slip-spring model. Therefore, trans-dimensional moves are likely to be rejected and models with low posterior density are rarely visited. One may obtain a better mixing sampler by using more intermediate step in the annealing scheme. In our case, this can be computationally expensive. We have seen in mixture case that using combine/split move will improve the efficiency of the algorithm. However, it requires a reversible pairwise move that is difficult to find an invariant quantity from a physical point of view for the slip-spring model.

4.7 Sequential Monte Carlo Sampler

We have seen from the previous section that large state space problem is involved in AIS RJMCMC, so we will look at Sequential Monte Carlo (SMC) samplers instead. Since we consider only a small number of models, the number of slip-springs M is usually smaller than 20.

SMC sampling is a class of methods that combine importance sampling and resampling strategies. It is based on compute simulation to compute posterior distribution. Unlike MCMC methods, SMC methods do not require complex configuration and is easy to implement in parallel.

The idea of importance sampling is to represent the probability density with a set of particles $\{x_i\}_{i=1}^{N_p}$. Base on this idea, all the statistical measurements that we are interested can be easily calculated. It is a straightforward implementation of Bayes' theorem. Firstly, the prior distribution can be expressed with respect to the particles as follows:

$$p(x) = \sum_{i=1}^{N_p} \frac{1}{N_p} \delta(x - x_i).$$

According to Bayes Theorem,

$$p(x|y) = \frac{p(y|x)p(x)}{\int p(y|x)p(x)dx},$$

which gives

$$p(x|y) = \sum_{i=1}^{N_p} w_i \delta(x - x_i),$$

where the weights are given by

$$w_i = \frac{p(y|x_i)}{\sum_{j=1}^{N_p} p(y|x_i)}, \quad i = 1, \cdots, N_p,$$

where $p(y|x_i)$ is the likelihood of observation y given parameter x_i. the weight of a particle means the relevant importance (contribution) of a particle compared to all the other particles. For example, if one would like to calculate the average of a function $f(\cdot)$ which can be approximated by

$$E[f(x)] = \int f(x)p(x)dx \sim \sum_{i=1}^{N_p} w_i f(x_i).$$

Let π be a target distribution of interest on measurable space E, and $\{\pi_t\}_{t=1}^{T}$ be a sequence of target distributions on E with $\pi_T = \pi$. Del Moral et al. [62] demonstrate that it is not practical doing sequential importance sampling in the cases where the proposal for target π_{t+1} is a transition kernel applied to points distributed from π_t. This is mainly due to the need for calculating an integral involved in evaluating the density of the proposal. The solution in [62] is to make the problem similar to the one that particle filtering tackles where the sequence of distributions has increasing dimension. We are going to have a brief description of how it works:

In SMC samplers, a sequence of artificial targets are introduced,

$$\widetilde{\pi}_t(x_{1:t}) = \pi_t(x_t) \prod_{s=2}^{t} L_{s-1}(x_{s-1}|x_s),$$

on the space E^n, through the introduction of backward kernels $L_t(\cdot)$, $t = 2, .., T$. The

basic idea is to apply a particle filtering methodology to this sequence of artificial targets. If, at time t, the forward kernel $K_t(.|x_{t-1})$ is used to update the point x_{t-1}, and γ_t is the unnormalised form of target π_t, then the incremental weight is given by

$$\widetilde{w}_t = \frac{\gamma_t(x_t)L_{t-1}(x_{t-1}|x_t)}{\gamma_{t-1}(x_{t-1})K_t(x_t|x_{t-1})}.$$

In the case where K_t is a π_t-invariant MCMC kernel, with a suitable choice of L_{t-1}, K_t can be canceled with L_{t-1}, the incremental weight becomes

$$\widetilde{w}_t = \frac{\gamma_t(x_{t-1})}{\gamma_{t-1}(x_{t-1})}. \tag{4.41}$$

For further details, see references in [62].

For simplicity, we will stick to using MCMC kernels, and using the incremental weight in equation (4.41). Note that in order to get around particle degeneracy, we also need to resample if necessary. When using MCMC kernels, we may perform this directly after normalising the weights. Note that AIS is exactly this algorithm, but without resampling scheme since it is implemented using only one particle. This algorithm yields an empirical approximation of π_t

$$\hat{\pi}_t = \sum_{p=1}^{N_p} w_t^{(p)} \delta_{\boldsymbol{\theta}_t^{(p)}}, \tag{4.42}$$

where $\delta_{\boldsymbol{\theta}}$ is a Dirac mass at $\boldsymbol{\theta}$, and an estimate of its normalising constant is given by

$$\hat{Z}_t = \prod_{s=1}^{t} \sum_{p=1}^{N_p} w_s^{(p)} \frac{\pi_{s+1}\left(\boldsymbol{\theta}_s^{(p)}\right)}{\pi_s\left(\boldsymbol{\theta}_s^{(p)}\right)}. \tag{4.43}$$

The normalising constant in our case is the marginal likelihood $p(y|m)$, so we may use this method to estimate the marginal likelihood.

Now, we are going to talk about how to choose the sequence of distributions. For a particular model m, our target is to approximate the posterior $\pi(\boldsymbol{\theta}|y,m)$ which is intractable. Instead of drawing samples to approximate $\pi(\boldsymbol{\theta}|y,m)$ directly, we

construct a sequence of distributions $\{\pi_t(\boldsymbol{\theta})\}_{t=0}^{T}$ with $\pi_0(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}|m)$ and $\pi_T(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}|y, m)$ to build a smooth path between the prior and the target distribution.

$$\pi_t(\boldsymbol{\theta}) \propto \pi(\boldsymbol{\theta}|m)p(y|\boldsymbol{\theta}, m)^{a(t)}, \tag{4.44}$$

where the number of intermediate distributions T and the annealing scheme $a(t) \in [0, 1]$ may vary from model to model. Firstly, we draw samples $\{\boldsymbol{\theta}^p\}_{p=1}^{N_p}$ from prior $\pi(\boldsymbol{\theta}|m)$.

Then the weight of each point can be written as

$$\tilde{w}_1^p = w_0^p \frac{\pi_1(\boldsymbol{\theta}_0^p)}{\pi_0(\boldsymbol{\theta}_0^p)}, \tag{4.45}$$

where we assume that all the points are allocated with the same weight at $t = 0$ with $w_0^i = \frac{1}{N_p}$. Then we normalize the weight w_1,

$$w_1^{(p)} = \frac{\widetilde{w}_1^{(p)}}{\sum_{p=1}^{N_p} \widetilde{w}_1^{(p)}}. \tag{4.46}$$

If the variance of the weights are large, we have discussed this issue in section 3.3 that resampling is an effective approach to reduce the weight variance.

Algorithm 6 SMC sampler

for $t = 2 : T$ **do**

Renew weight $\tilde{w}_t = w_{t-1} \frac{\pi_t(\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})}$

Normalize weight

$$w_t^{(p)} = \frac{\widetilde{w}_t^{(p)}}{\sum_{p=1}^{N_p} \widetilde{w}_t^{(p)}}$$

Resample if necessary

for $p = 1 : N_p$ **do**

Run MCMC to obtain $\boldsymbol{\theta}_t^p \sim K_{\pi_t}(\cdot|\boldsymbol{\theta}_{t-1}^p)$

endfor

endfor

Next, we will look at how to adaptively tune $a(t)$. The choice of adaptive scheme

is quite flexible. The aim of performing adaptive scheme is to make sure that the the distance of neighbouring targets is neither too far, which will lead the MCMC move to be rejected. It cannot be too close which would decrease the efficiency of the algorithm. Del Moral et al. [63] introduced a strategy to monitor the efficiency of each iteration with conditional effective sample size (CESS).

$$\text{CESS} = \frac{N_p \left(\sum_{p=1}^{N_p} w_t^{(p)} \omega_{t+1}^{(p)}\right)^2}{\sum_{p=1}^{N_p} \left(w_t^{(p)} \omega_{t+1}^{(p)}\right)^2}, \tag{4.47}$$

where $\omega_{t+1}^{(p)}$ is the incremental weight evaluating the ratio in Eq. (4.43). CESS can monitor the discrepancy between π_{t-1} and π_t. One can decrease the distance between π_{t-1} and π_t to ensure that CESS$= \kappa$, $0 < \kappa < 1$. The choice of κ relies on the complex of the models. In this work, we adapt $\kappa = 0.9$.

We can also use adaptive strategies in MCMC kernels to improve the efficiency of SMC sampler. It allows us introduce apply adaptation in MCMC kernels when exploring the space based on the past history of the sampler. A straightforward way to adapt the proposal variance used in M-H step according to sample variance of the current particles. In this work, we adapt the proposal scale using twice of the sample variances suggested by Beaumont et al. [64]. Note that we investigate models with different number on slip-springs. The ensemble sample variances evaluating from all the particles would be in much larger than that of an individual slip-spring. However, we do not keep unique order of the slip-springs for all the particles. Note that the for each particle the order of the slip-spring may change. When evaluating the sample variance of anchor point a_j, we need to know the order number of a_j on each particle. Clustering a set of objects in such a way that objects in the same group are more similar to each other than to those in other groups. k-means clustering is one of the well-known clustering methods that can exactly distribute each point into one of k clusters defined by centroids. We assume that there is no overlap between entanglements. Therefore, it is easy to apply k-means clustering based on

the spatial location of anchor points a_j to group components into clusters. Then we can adaptively take cluster variances as proposal variance.

4.8 Application of SMC and Slip-Spring Model on Detecting Entanglements

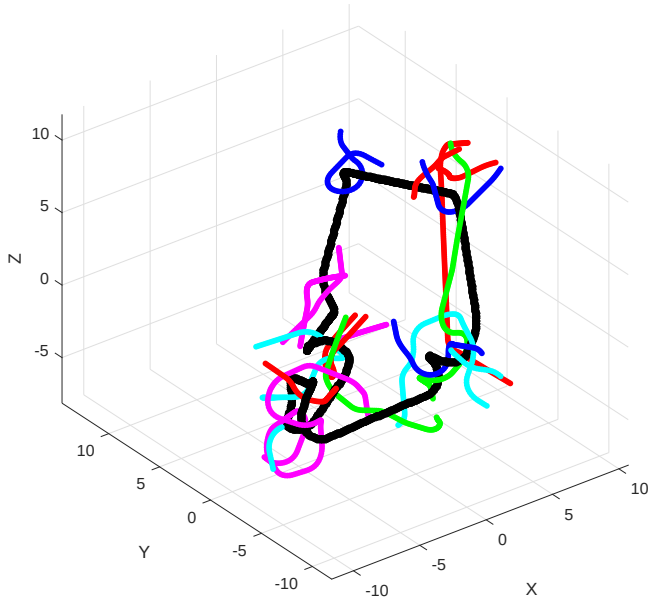


Figure 4.10: Contour length of a ring polymer (thick black line) as found by the primitive path analysis. The primitive paths of some surrounding chains are also included for reference.

The trajectories of concatenated ring polymers obtained from the MD simulations described in section 4.6.2 are employed for entanglement analysis using the combined SMC and slip-spring model method. Firstly, we want to learn about the concatenated typology of the target ring polymer. We run a new set of MD simulations as described in section 4.6.2 which takes one of the configurations obtained from the standard MD simulation as the initial configuration. Then we turn off the thermal fluctuations of the

monomers, eliminate excluded volume interactions between monomers on the same chain while maintaining interchain uncrossability. All the chains end up with shrinking to the shortest paths with many kinks as shown in Fig. 4.10. The tube-like cloud constructed by the instantaneous configurations of the target polymer as collected from 500 MD frames which are evenly separated by the time interval $t = 10,000\tau_{LJ}$ is shown in Fig. 4.11, together with the primitive path of this chain same as that shown in Fig. 4.10. It is natural to assume that clusters with high monomer densities in the cloud, as reflected by the deeper colours, may correspond to the locations of the entanglements. It can be seen that the kinks of the given primitive paths do not necessarily correspond to the high-density clusters, meaning that the shrinking process has artificially destroyed the equilibrium configurations of the polymers. The number of kink or so-called entanglements may not be consistent with the definition of persistent contacts. Therefore primitive path analysis is probably not adequate to provide a complete picture about the locations of entanglements.

To identify the locations of the entangles on a target chain, we introduce M slip-springs, each with one end initially located on a randomly selected monomer of the chain, the other end anchored at a random position in space and the initial end-to-end distance of $\sqrt{\boldsymbol{N}_s}b$.

We implement the SMC sampler with $N_p = 1000$ to this model system for the number of slip-springs $M = 1, 2, \cdots, 14$. The prior distributions are the same as specified in section 4.6.2. The logarithm marginal evidence estimated using different m is given in Fig. 4.12. We can see that the marginal evidence increases significantly as the number of slip-springs M increasing from 1 to 3. It indicates that each of the newly added components does improve the performance of the slip-spring based model. As we keep on adding slip-springs to the model system, the marginal evidence witnesses slight fluctuations between $M = 3$ and $M = 5$. After that, the marginal evidence rises again and reach a peak value at $M = 7$. After exhibiting some fluctuations between $7 \sim 12$, the marginal likelihood begins to decrease from $M = 13$ and onwards. The

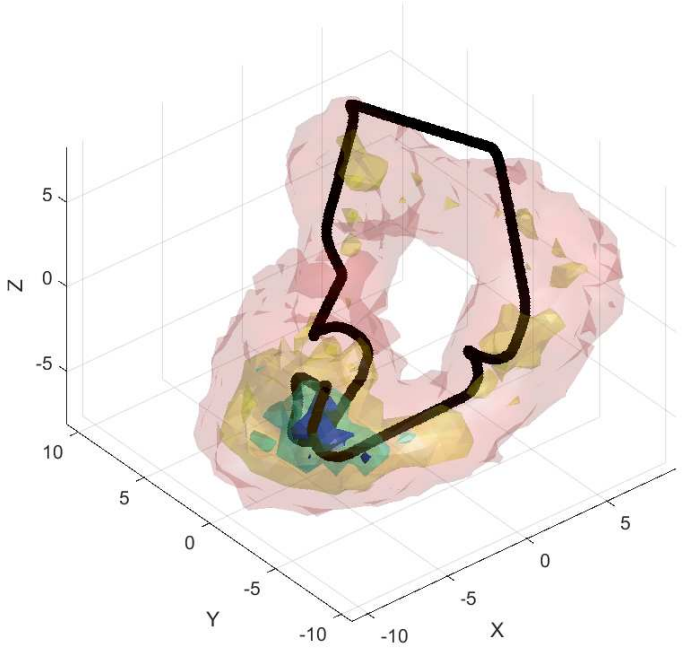


Figure 4.11: Tube-like cloud constructed by the instantaneous configurations of the target polymer as collected from 500 MD frames, together with the primitive path of this chain (thick black line) same as that shown in Fig. 4.10.

SMC sampler suggests that there are probably $7 \sim 12$ entanglements that restrict the lateral motion of the test ring polymer. Although it might appear that $M = 7$ is sufficient, one should keep in mind that the marginal likelihood takes into account both the likelihood and the model complexity. Note that the model complexity rises as the number of slip-springs increases. Therefore the fluctuation shown between 7 and 12 reveals that the likelihood is still increasing a bit up to this point. The marginal likelihood drops off after $M = 12$ indicates that the benefit to the likelihood is not sufficient compared to the increased model complexity.

Fig. 4.13 shows a snapshot of the analysed model system with 11 slip-springs,

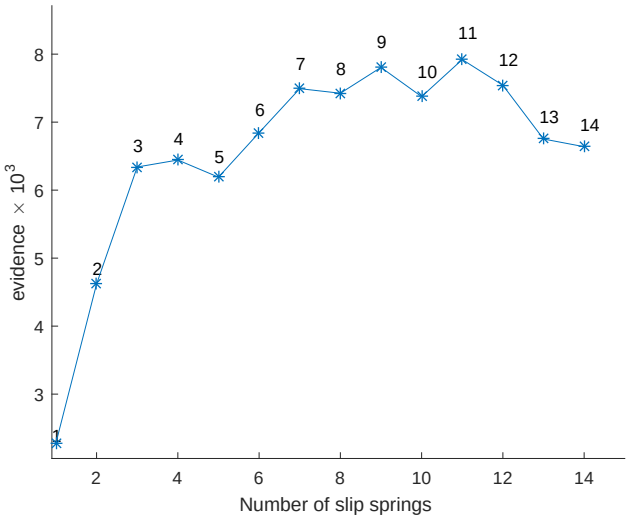


Figure 4.12: Marginal evidence versus the number of slip-springs

together with the configuration cloud and primitive path of the target chain. The instantaneous positions of the anchor points of the slip-springs as obtained from the SMC sampler shown as scattered red points for providing direct evidence on the mean positions and variances of the estimate. The mean positions of the slip-links are characterize by cyan spheres. The values printed along with the slip-springs are their mean end-to-end distances $\sqrt{\langle \boldsymbol{N}_s \rangle}$ of the slip-springs.

We can see from Fig. 4.13 that all the clusters with high density on the dense cloud have been detected by the SMC sampler. The primitive path analysis suggests that there are at least 12 kinks on the contour length. Our algorithm indicates that we can use 12 or fewer slip-springs to represent the entanglements. It is possible that there are more than 12 polymer rings concatenated with the target ring. However, some of them do not have persistent interaction with the target ring presumably due to the existence of other ring segments in between them, which are therefore not reflected by high-density clusters in the configuration cloud.

The average end-to-end distance of each slip-spring and the standard deviation of the estimate are given in Table 4.6.

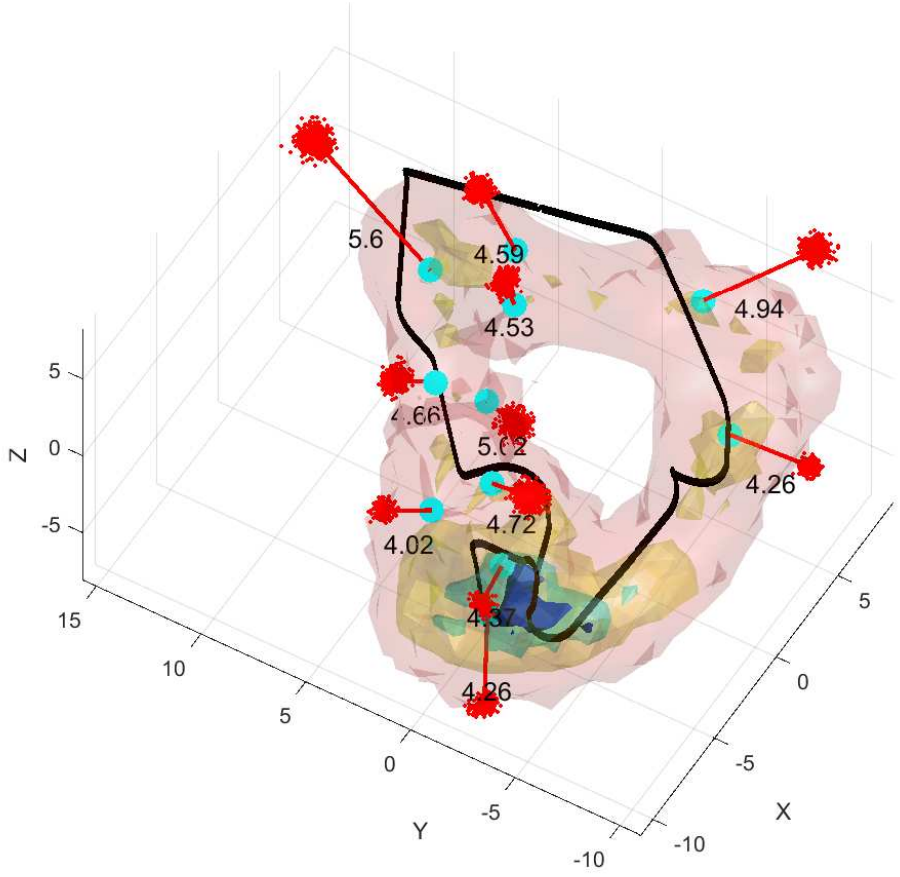


Figure 4.13: Snapshot of the slip-spring positions as estimated by the SMC sampler, together with the configuration cloud and the contour length (thick black line) of the target ring polymer. The scattered red points represent instantaneous positions of the anchor points during the SMC sampling, and the large cyan spheres represent the mean position of the slip-links. The short red lines link the mean positions of the anchor point and slip-links of each slip-spring, and the numbers added next to them are the mean end-to-end distance of the springs.

To visualize how accurate the estimate is, we plot the projections of the dense cloud and slip-springs on the XY-plane in Fig. 4.14. It is natural that some of the high-density clouds are characterized by more than one slip-spring. Since the high-density clusters arise from persistent contact between the target chain and neig-

Table 4.6: Parameter estimate and relevant uncertainty ($M = 11$)

j	1	2	3	4	5	6	7	8	9	10	11
std($\boldsymbol{a}_j$)	0.2159	0.2231	0.2620	0.2494	0.2645	0.3095	0.2163	0.1709	0.1969	0.2890	0.2376
std(N_{s_j})	0.0307	0.1200	0.5179	0.0309	0.0627	0.0358	0.0191	0.0273	0.0429	0.0477	0.0615
$\sqrt{\langle N_{s_j} \rangle}$	4.5872	4.2576	4.9420	4.659	4.7207	5.5966	4.0235	4.2583	4.3701	5.0170	4.5298

hbouring chains, it is very likely that there are several neighbouring chains entangled with the target chain in the high-density area.

It is interesting to note that concatenated ring polymers is a tricky situation where the number of entanglement are fixed. Consequently, CR and CLF, which are essential in the study of entanglements, are not taken into consideration in such case. Therefore, it is important to consider applying SMC algorithms to describe the entanglement dynamics in more general cases. In the following section, we will look at the application of SMC algorithms on linear polymers.

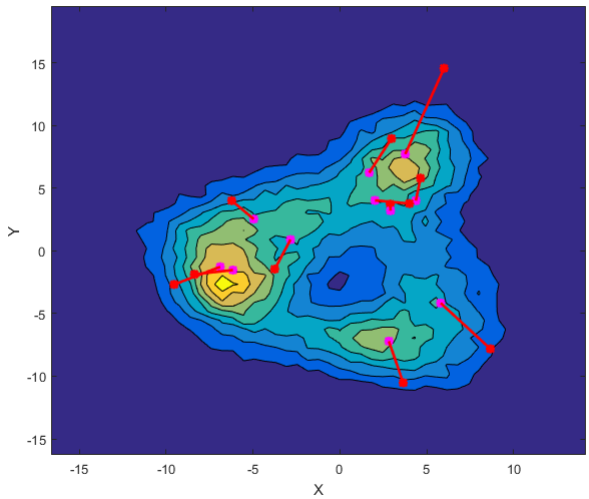


Figure 4.14: Projections of the chain monomer density profile and slip-springs on the XY-plane

4.9 Application of SMC on Linear Polymers

We run molecular dynamics simulations of polymer melts using the Kremer-Grest bead-spring model introduced in section 2.5, where the number of chain $N_c = 250$ and the particle number density ρ is 0.85. To guarantee there are sufficiently large number of entanglements on each chain for analysis, we increase the chain length N to 1023.

In the melts consisting of of linear polymers, the number of entanglements on each chain can fluctuate over time due to the combined reptation, CLF and CR effects. Firstly, we will investigate the CLF effects in a given time window. We aim to analyse MD trajectories consisting of 300 configurations where the time interval between two neighbouring configurations is $\tau_{\text{int}} = 1200\tau_{LJ}$. In Fig. 4.15 we plot the configuration cloud of a single target chain based on all 300 configurations and its mean path obtained at the simulation times corresponding to the first and the last configurations by averaging the instantaneous configurations of the chain over a time duration τ_{int}. . It is observed in Fig. 4.15 that left hand side of the target chain has retracted from the original tube and entered a new tube mainly due to CLF effects.

It is important to note that we assume that when applying the slip-spring model and SMC methods to study entanglements, the total number of slip-springs in the system is fixed. To tackle this problem, we consider only the central part of the chain conformation and the fluctuations of the chain ends are ignored. Wang and Larson [65] demonstrated how to evaluate the maximum number of monomers escape from the initial tube due to CLF [46]. The fraction of monomers released by CLF for given time duration t is given by

$$\Phi(t) = 1.5(t/\tau_e)^{1/4}/M, \tag{4.48}$$

where τ_e is the entanglement time and M is the number of entanglements of the chain. Wang, Likhtman and Larson found $\tau_e \approx 2950\tau_{LJ}$ for the flexible chains from

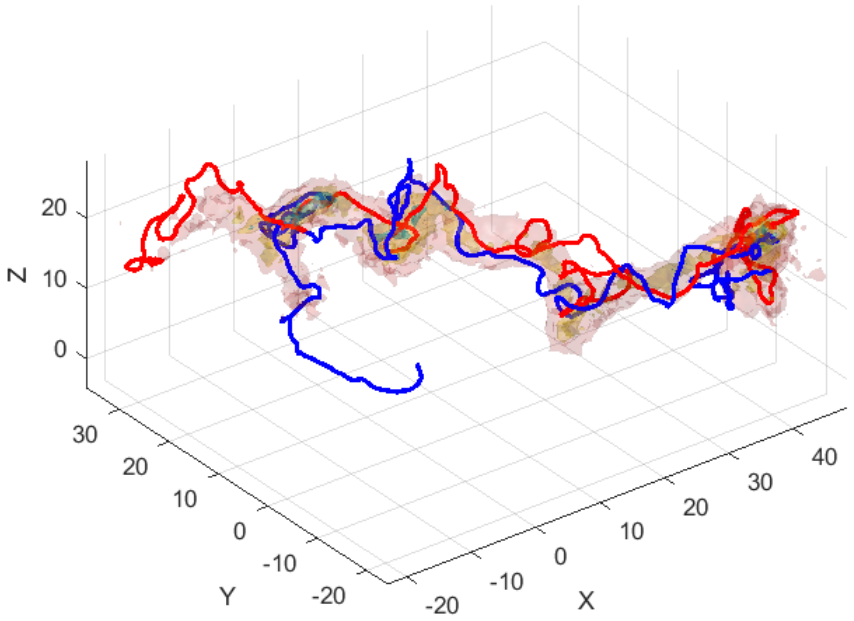


Figure 4.15: Snapshot of mean paths together with the configuration cloud of a target linear polymer. The red line is a snapshot of mean path of the chain and the blue line is the snapshot of its mean path after time $360000\tau_{LJ}$.

the mean square displacements of chain middle monomers [66]. To estimate the number of entanglements M, we apply PPA method [6] [67] by fixing the ends of the chains and switching off the intra-chain excluded volume interactions, which gives us the average number of kinks per chain $M \approx 25$. The corresponding entanglement strand length was estimated to be in the range of 60 to 85, giving the average number of entanglements per chain $M \approx 12 \sim 17$. Therefore, on each side of the chain, the number of monomers affected by CLF is given by

$$N\Phi(t)/2 \approx 150 \sim 213 \tag{4.49}$$

which implies that within time interval $360000\tau_{LJ}$, for a linear chain with chain length $N = 1023$, there are approximately $300 \sim 425$ monomers escaped from the initial tube due to the effect of CLF. Thus in such a time window, the central part of the chain

with $600 \sim 724$ monomers feels very weak CLF effects. To improve the accuracy of the estimate of the number and location of the slip-springs, in the following section, we will focus on the 512 monomers in the central part of the chain to avoid the influence of CLF effects.

4.9.1 Partition Function

For a free Rouse chain with $N+1$ beads, the probability distribution function of the bead positions $\{\mathbf{R}_i\}_{i=0}^{N}$ is solely determined by the harmonic potential energy,

$$P(\{\mathbf{R}_i\}) = \frac{1}{Z_0}\exp\left[-\frac{3}{2b^2}\left(\sum_{i=0}^{N-1}\left(\mathbf{R}_{i+1}-\mathbf{R}_i\right)^2\right)\right], \tag{4.50}$$

where

$$Z_0 = \int d\mathbf{R}_0 \int d\mathbf{R}_1 .. \int d\mathbf{R}_N \exp\left[-\frac{3}{2b^2}\left(\sum_{i=0}^{N-1}\left(\mathbf{R}_{i+1}-\mathbf{R}_i\right)^2\right)\right]$$

is the partition function of the free linear chain.

For a given set of anchor point positions $\boldsymbol{a} = \{\boldsymbol{a}_j\}_{j=1}^{M}$ and length of slip-spring $\boldsymbol{N}_s = \{N_{s_j}\}_{j=1}^{M}$, the probability distribution function of the dynamic variables $(\boldsymbol{R}, \boldsymbol{s})$ is

$$P\{\boldsymbol{R}, \boldsymbol{s}|\boldsymbol{a}\} = \frac{1}{Z(\boldsymbol{a})}\exp\left[-\frac{3}{2b^2}\left(\sum_{i=0}^{N-1}\left(\mathbf{R}_{i+1}-\mathbf{R}_i\right)^2 + \sum_{j=1}^{M}\frac{\left(\mathbf{R}_{s_j}-\boldsymbol{a}_j\right)^2}{N_{s_j}}\right)\right]. \tag{4.51}$$

We note that the distribution of a vector between any two monomers is given in Eq.

(4.8). thus the partition function $Z_{ns}(\boldsymbol{a}, \boldsymbol{s})$ takes the form of

$$
\begin{aligned}
Z_{ns}(\boldsymbol{a}, \mathbf{s}) = & \left(\prod_{j=0}^{M} n_j\right)^{-d/2} \left(\frac{3}{2\pi b^2}\right)^{(M+1)Md/2} \int d\mathbf{R}_0 \int d\mathbf{R}_{s_1} \cdots \int d\mathbf{R}_{s_M} \int d\mathbf{R}_N \times \\
& \exp\left[-\frac{3}{2b^2}\left(\sum_{j=1}^{M-1} \frac{1}{n_j}\left(\mathbf{R}_{s_{j+1}} - \mathbf{R}_{s_j}\right)^2 + \frac{1}{n_0}\left(\mathbf{R}_{s_1} - \mathbf{R}_0\right)^2 + \frac{1}{n_M}\left(\mathbf{R}_N - \mathbf{R}_{s_M}\right)^2 + \sum_{j=1}^{M} \frac{1}{N_{s_j}}\left(\mathbf{R}_{s_j} - \boldsymbol{a}_j\right)^2\right)\right] \\
= & \left(\prod_{j=0}^{M} n_j\right)^{-d/2} \left(\frac{3}{2\pi b^2}\right)^{(M+1)Md/2} \int d\mathbf{R}_0 \int d\mathbf{R}_{s_1} \cdots \int d\mathbf{R}_{s_M} \int \mathrm{d}\mathbf{R}_N \times \\
& \exp\left[-\beta\left(\sum_{i=0,j=0}^{M+1} A_{ij} \mathbf{R}_{s_i} \cdot \mathbf{R}_{s_j} + \sum_{i=0}^{M+1} b_i \cdot \mathbf{R}_{s_i} + c\right)\right], \qquad (4.52)
\end{aligned}
$$

where $\boldsymbol{A}$ is given by

$$
\boldsymbol{A} = \begin{pmatrix}
\frac{1}{n_0} & -\frac{1}{n_0} & 0 & 0 & 0 & 0 \\
-\frac{1}{n_0} & \frac{1}{n_0} + \frac{1}{n_1} + \frac{1}{N_{s_1}} & -\frac{1}{n_1} & \cdots & 0 & 0 \\
0 & -\frac{1}{n_1} & \frac{1}{n_1} + \frac{1}{n_2} + \frac{1}{N_{s_2}} & \cdots & 0 & 0 \\
0 & 0 & -\frac{1}{n_2} & \cdots & \vdots & \vdots \\
\vdots & \vdots & \ddots & \ddots & 0 & \vdots \\
0 & 0 & \cdots & \frac{1}{n_{M-2}} + \frac{1}{n_{M-1}} + \frac{1}{N_{s_{M-1}}} & -\frac{1}{n_{M-1}} & 0 \\
0 & 0 & \cdots & -\frac{1}{n_{M-1}} & \frac{1}{n_{M-1}} + \frac{1}{n_M} + \frac{1}{N_{s_M}} & -\frac{1}{n_M} \\
0 & 0 & 0 & 0 & -\frac{1}{n_M} & \frac{1}{n_M}
\end{pmatrix}.
$$

In the second step, we have defined

$$
\begin{aligned}
& \beta = \frac{3}{2b^2}, \qquad b_i = -\frac{2}{N_{s_i}} \boldsymbol{a}_i, \qquad c = \sum_{j=1}^{M} \frac{\boldsymbol{a}_j^2}{N_{s_j}}, \quad i = 1, ..., M, \\
& b_0 = 0, \qquad b_M = 0.
\end{aligned}
$$

It is important to note that our entanglement analysis is only carried out for the central part of a chain with 512 monomers. The two ends of this part of the chain are not free but connected with the rest of the original chain segments instead. Their motion are thus subject to topological constraints analogous to those connected to

slip-springs. Therefore we have rewritten their positions in Eq. 4.52 as

$$\mathbf{R}_{s_0} = \mathbf{R}_0, \quad \mathbf{R}_{s_{M+1}} = \mathbf{R}_N.$$

We note that no extra virtual spring energy needs to be added in the partition function calculation, because these two monomers can still be occupied by the slip-links added during the SMC sampling. It is interesting to note that the partition function of linear polymers shown in Eq. (4.52) has a very similar form to that in Eq. (4.9) for concatenated ring polymers. Thus the partition function $Z_{ns}(\boldsymbol{a}, \mathbf{s})$ can be easily obtained by Eq. (4.13).

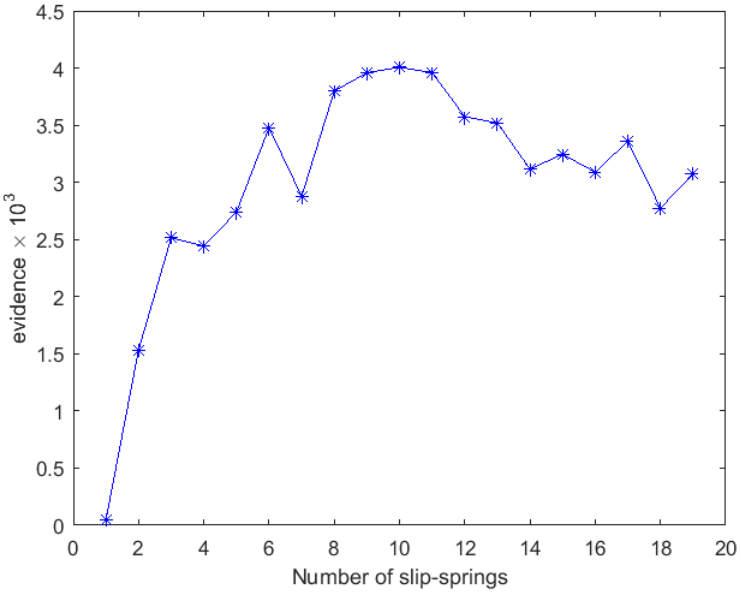


Figure 4.16: Marginal evidence versus the number of slip-springs

Next we will apply SMC sampler given by Alg. 6 to linear polymers. The implementation of the SMC sampler is described in section 4.7. We use the same number of particles $N_p = 1000$ and CESS $\kappa = 0.9$ for each method. Fig. 4.16 shows the logarithm of the marginal likelihood for models characterized by different number of slip-springs. Similar to the concatenated ring case, the log-marginal likelihood

estimate shows fluctuations due to sample noise introduced by the SMC sampler. However, it is obvious that the log-marginal likelihood exhibits a clear peak between $m = 6$ and $m = 13$.

Fig. 4.17 shows a snapshot of the slip-spring positions as estimated by the SMC sampler with $m = 12$, together with the configuration cloud and the contour length (black line) of the central part of target polymer. The contour length is obtained in the following manner. We take one of the configurations obtained from the standard MD simulation as the initial configuration. The ends of the chains are fixed in space. Then we turn off the thermal fluctuations of the monomers, eliminate excluded volume interactions between monomers on the same chain while maintaining interchain uncrossability. In the end, the target chain ends up with many kinks and shrinks to the shortest paths as shown in Fig. 4.17.

The scattered red points represent instantaneous positions of the anchor points during the SMC sampling, and the large cyan spheres represent the mean position of the slip-links. We can see that the SMC sampler with 12 slip-springs is capable of capturing nearly all the significant high-density clusters in the configuration cloud. Fig. 4.18 provides the projection on XY-plane of snapshot.

We note that the number of $m = 12$ slip-springs on a chain segment of 512 monomers seems to agree with the total number of $M = 25$ kinks as found in PPA for the whole chain of 1024 monomers. On the other hand, Fig. 4.16 shows that the marginal evidence reaches the plateau regime at about $m = 6 \sim 8$, which is consistent with the number of entanglements estimated by using $N_e = 60 \sim 85$. It has been shown by Everaers et al. [6] that the number of entanglements per chain is not directly determined by the number of kinks, but instead by the random walk feature of the primitive path constructed by the straight lines linking the kinks. The effective number of entanglement is thus much smaller than the number of kinks found in PPA. The results in Figures 4.16 - 4.18 imply that different from the PPA method which destroys the local structure of the chains, the SMC methods allow us

to find both the number and unperturbed positions of the topological constraints, but also the minimum number of entanglements required to describe the primitive path. The latter could be referred to the number of entanglements generally used in tube theories.

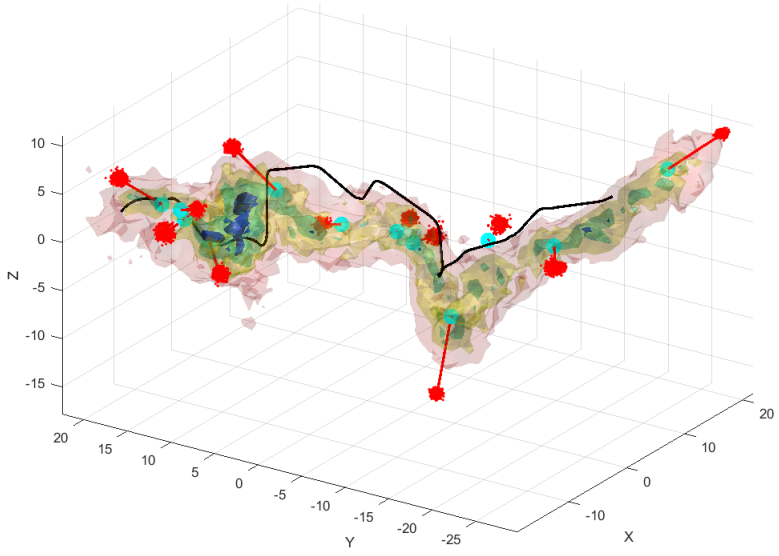


Figure 4.17: Snapshot of the slip-spring positions as estimated by the SMC sampler with $m = 12$, together with the configuration cloud and the contour length (black line) of the central part of target linear polymer. The scattered red points represent instantaneous positions of the anchor points during the SMC sampling, and the large cyan spheres represent the mean position of the slip-links. The short red lines link the mean positions of the anchor point and slip-links of each slip-spring.

4.10 Summary

This chapter is motivated by the need of developing novel approaches to detect the entanglements along the polymer's backbone. We aim to draw inference on the number, the effective lengths and the locations of the anchor points of the slip-springs for given chain trajectories. A sequence of Bayesian statistical methods, e.g, RJMCMC,

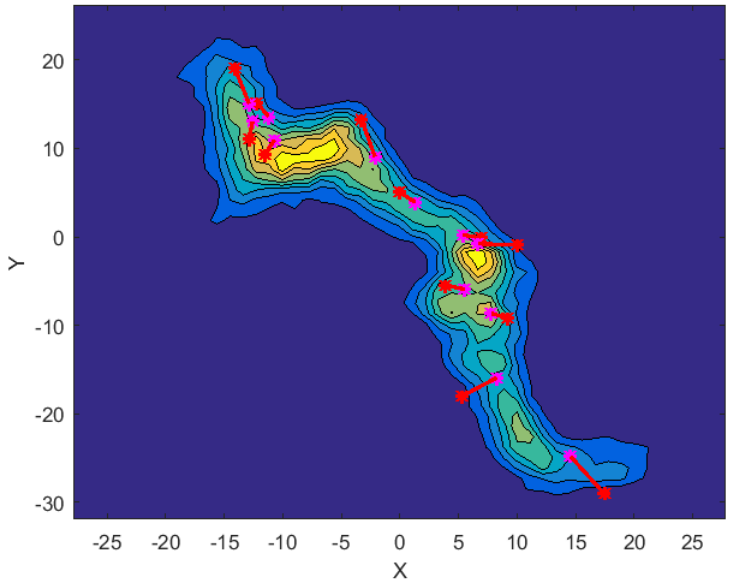


Figure 4.18: Projections of the chain monomer density profile and slip-springs on the XY-plane

AIS RJMCMC and SMC, are applied to implement model comparison over a set of potential model. One of the significant advantages of the Bayesian paradigm is that it allows us to compute the posterior distribution of different models and provides uncertainty analysis on the estimation of model parameters at the same time.

First, we commence from describing the slip-spring model in a discrete manner where the slip-links are only allowed to sit on the coordinates of the monomers. The exact solution of the partition function is derived by making use of Gaussian integral. Different methods for model comparison are reviewed and their advantages and limitations are discussed. A Gaussian mixture model is introduced to demonstrate how to implement RJMCMC to explore the joint parameter and model space. We find that the RJMCMC can easily get stuck at particular model space because of a lack of proper transdimension-move. AIS RJMCM explores the joint parameter and model space more efficiently by introducing intermediate distribution between the proposal and the target. However, when applying AIS RJMCM on slip-spring model, we find

it is challenging to jump from one model to another since it is hard to find a bijective transformation to combine two slip-springs into one slip-spring.

An attractive alternative for model comparison is SMC which allows us to investigate each model independently and provides an approximation of marginal likelihood for model comparison. One of the possible limitations of SMC for studying entanglement effects is the assumption that the potential model and its parameters are fixed. However, owing to the reptation, CR and CLF effects, the tube made from the neighbouring chains is fluctuating and moving around. To avoid the impacts of CR and CLF, we first look concatenated ring polymers. The SMC based slip-spring model successfully captures all the significant entanglements shown in the configuration cloud and provides the log-marginal likelihood for model comparison. We then study melt systems consisting of entangled linear chains. To minimize the impacts of CLF, we focus on the central part of the the liner chain and the monomers at each end of the chain are discarded. In this way, all the significant entanglements along the target segment of the linear chain are represented by slip-springs. In the future, we intend to apply SMC and slip-spring model to capture more properties of the entanglements.

Chapter 5

Conclusions

This chapter summarizes the main contributions in this thesis and analyses the limitations of the models we have developed. Also discussed in this chapter are the possible solutions for those limitations and potential future research directions.

5.1 Contributions

In this thesis, the overall research goal is to develop novel models for studying the dynamics of entangled polymers. We have proposed three stochastic models to study the diffusion process of the chain center of mass. We have also applied slip-spring model based on Bayesian statistics and Monte Carlo methods to detect the existence of entanglements and infer the number and location of the potential entanglements.

The first substantial contribution in this thesis is applying slip-spring model based on SMC sampler in studying entanglements. Extensive efforts have been devoted to defining the microscopic pictures of entanglements and trying to detect entanglements at microscopic scales. The slip-spring model can be considered as one of the most promising models for describing entanglement dynamics. With the help of Bayesian statistical methods, the slip-spring model is capable of detecting the number of entanglements. Besides, the location of the entanglements and the fluctuations of the tube that constrains the motion of the target chain are well described by the locations

of the anchor points and the lengths of the slip-springs, respectively.

We also demonstrate how to implement RJMCMC and AIS RJMCMC for exploring joint model and parameter space. It is important to note that the entanglement analysis based on slip-spring model in this work is the first practical application of AIS RJMCMC. We report that AIS RJMCMC may not be suitable for the models that lack appropriate bijective transformations between parameters in different models.

The second substantial contribution in this thesis is developing the linear multi-bead coarse-grained models which have potential to describe the dynamics of entanglement polymers. We have applied these models to reproduce the dynamics of the center of mass of single chains in MD simulations. The multi-bead coarse-grained models have much less free parameters compared with existed coarse-grained models, which allows us to run sufficiently long enough simulations for a reasonably long period of time. It is interesting to find that, for modelling linear polymers, the multi-bead models prefer Rouse-like topology rather than asymmetric star structure. We also demonstrate how to implement MCMC to draw inference on the models and their parameters and discuss strategies on how to improve the efficiency of the MCMC sampler.

The third substantial contribution in this thesis is to investigate coarse-grained models with nonlinear interactions. We find that the single-particle model with memory kernel is capable of capturing the dynamic properties of polymer chains studied in MD simulation at very short time scales while it fails to recover the diffusion process of the chain center of mass at large time scales. We show that particle MCMC is a promising Monte Carlo method for finding model parameters as long as we have powerful computational tools for massively parallel computing.

5.2 Limitations

Although the models we developed in this thesis have proved to be effective in analysing dynamics of unentangled and entangled polymers, there are a number of limitations with these models.

First, due to the fact that entanglements are caused by the uncrossibility of the neighbouring chains and all the chains are fluctuating and moving around over time, which implies that the number of entanglements in the system is not fixed. We assume that the number of slip-springs in a given model is fixed, which fails to capture the effects of CR and CLF. For analysing entanglements on a linear polymer, we have to discard a set of monomers at each end of the chain to ensure that the CLF has weak impact on analysis results.

Another limitation is the noise on the log-marginal likelihood estimator over slip-spring models. One can either increase the number of particles or increase the CESS value, which will increase the computational complexity in return. Due to limited computational power and limited time for simulation, it is challenging to find a solution to minimize the impact of noise on the estimate.

Turning attention to the multi-bead model, we can only perform parameter estimation on linear system using standard MCMC methods. It is challenging to draw inference on nonlinear system where particle MCMC fails to explore the parameter space. A more effective method is needed for obtaining smooth likelihood.

5.3 Future Work

In this section we point out the possible solutions to the limitations we have discussed and also provide some potential future research directions.

The entanglement analysis should be performed over a large ensemble of target chains in order to examine the accuracy of combined the slip-spring model and Bayesian statistics. It is also important to provide quantitative information on the

entanglement strength using the probability distribution of the slip-spring lengths. It would be more interesting and also more challenging to test if our slip-spring model works for detecting entanglements for whole polymers with free-ends. In the slip-spring model, the anchor points are assumed to be fixed in space, which might not be sufficient for describing the effects of the CR and CLF. It is very worthwhile devoting effort to extending the current application to study the dynamics of the entanglements which are related to reptation, CLF and CR effects. This can be done by dividing a large number of trajectories into subsets and finding the locations of the slip-springs in each sub-time period. By looking at the number and location of the anchor points, we can track the dynamics of the entanglements along the contour lengths of the polymers.

On the other hand, we could turn our attention to the multi-chain system. We could allow the two ends of a slip-spring to slide on two different polymers which are potentially entangled, which may provide useful information for the development of multi-chain slip-spring model [68] [69] [70] [71]. It important to note that this choice will rely on the assumption that entanglements are binary events [13]. The method we have discussed in Chapter 4 is more general which does not depend on the binary assumption.

List of Symbols

N	Polymer length, the number of bonds per chain
b	Average bond length
k	Spring constant, $k = 3k_BT/(2b^2)$
$\mathbf{R}$	Position vector of monomer
$\mathbf{r}$	Bond vector
$\mathbf{R}_e$	End-to-end vector
$\mathbf{R}_g$	Radius of gyration
$\mathbf{R}_{cm}$	Center of mass
$\mathbf{W}$	Wiener process
d	Dimension
ξ	Friction coefficient
D	Diffusion coefficient
ρ	Particle density
U	Potential
τ	Relaxation time
τ_R	Rouse time
τ_{LJ}	Lennard-Jones time

L	Number of observations
Θ	Unknown parameter set
N_p	Number of particle
w	Weight
$\pi(\cdot)$	Target distribution
K	Length of Markov chain, sample size drawn from the target distribution
Y	Observed data set
$Z(\cdot)$	Partition function
k	Spring constant, $k = 3k_BT/(2b^2)$
N_c	Number of Chains
N_e	Number of entanglements
τ_e	entanglement time
N_s	Chain length of each slip-spring, the number of bonds in each slip-spring
M	number of slip-springs
$\mathbf{a}_j$	Anchor point of s_j fixed end of j^{th} slip-spring
s_j	Monomer index connected to j^{th} slip-spring an integer in between 0 and $N-1$
n_j	Segment length between s_j and s_{j+1}

Abbreviations

ABC	Approximate Bayesian Computation
AIS	Anneal importance sampling
BF	Bayesian Factor
BIC	Bayesian Information Criterion
CESS	Conditional effective sample size
CR	Constraint release
CLF	Contour length fluctuations
EnKF	Ensemble Kalman filter
ESS	Effective sample size
FENE	Finitely extensible nonlinear elastic
GIMH	Grouped Independence Metropolis-Hastings
HMM	Hidden Markov Model
IAT	Integrated autocorrelation time
IS	Importance sampling
KF	Kalman filter
LJ	Lennard-Jones

MAP	Maximum-a-Posteriori
MC	Monte Carlo
MCMC	Markov chain Monte Carlo
MD	Molecular Dynamics
M-H	Metropolis-Hastings
MLE	Maximum likelihood estimate
MPF	Marginal Particle Filter
MSD	Mean squared displacement
PF	Particle filter
PMCMC	Particle Markov chain Monte Carlo
PPA	Primitive path analysis
RJMCMC	Reversible jump Markov chain Monte Carlo
SMC	Sequential Monte Carlo
VAF	Velocity autocorrelation function

References

[1] M. Doi and S. F. Edwards, *The theory of polymer dynamics*, vol. 73. Oxford University Press, 1988.

[2] M. Rubinstein and R. Colby, *Polymers Physics*. Oxford, 2003.

[3] R. Kubo, "The fluctuation-dissipation theorem," *Reports on Progress in Physics*, vol. 29, no. 1, p. 255, 1966.

[4] P. E. Rouse Jr, "A theory of the linear viscoelastic properties of dilute solutions of coiling polymers," *The Journal of Chemical Physics*, vol. 21, no. 7, pp. 1272–1280, 1953.

[5] W. W. Graessley, "Entangled linear, branched and network polymer systems—molecular theories," in *Synthesis and Degradation Rheology and Extrusion*, pp. 67–117, Springer, 1982.

[6] R. Everaers, S. K. Sukumaran, G. S. Grest, C. Svaneborg, A. Sivasubramanian, and K. Kremer, "Rheology and microscopic topology of entangled polymeric liquids," *Science*, vol. 303, no. 5659, pp. 823–826, 2004.

[7] M. Kröger, "Shortest multiple disconnected path for the analysis of entanglements in two-and three-dimensional polymeric systems," *Computer Physics Communications*, vol. 168, no. 3, pp. 209–232, 2005.

[8] C. Tzoumanekas and D. N. Theodorou, "Topological analysis of linear polymer melts: a statistical approach," *Macromolecules*, vol. 39, no. 13, pp. 4592–4604, 2006.

[9] S. D. Anogiannakis, C. Tzoumanekas, and D. N. Theodorou, "Microscopic description of entanglements in polyethylene networks and melts: Strong, weak,

pairwise, and collective attributes," *Macromolecules*, vol. 45, no. 23, pp. 9475–9492, 2012.

[10] A. E. Likhtman, "The tube axis and entanglements in polymer melts," *Soft Matter*, vol. 10, no. 12, pp. 1895–1904, 2014.

[11] J. Cao, J. Qin, and S. T. Milner, "Simulating constraint release by watching a ring cross itself," *Macromolecules*, vol. 47, no. 7, pp. 2479–2486, 2014.

[12] A. E. Likhtman and M. Ponmurugan, "Microscopic definition of polymer entanglements," *Macromolecules*, vol. 47, no. 4, pp. 1470–1481, 2014.

[13] J. Cao and Z. Wang, "Microscopic picture of constraint release effects in entangled star polymer melts," *Macromolecules*, vol. 49, no. 15, pp. 5677–5691, 2016.

[14] W. Edwards, H. Lindman, and L. J. Savage, "Bayesian statistical inference for psychological research.," *Psychological Review*, vol. 70, no. 3, p. 193, 1963.

[15] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[16] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[17] A. E. Gelfand and A. F. Smith, "Sampling-based approaches to calculating marginal densities," *Journal of the American Statistical Association*, vol. 85, no. 410, pp. 398–409, 1990.

[18] G. O. Roberts and J. S. Rosenthal, "General state space Markov chains and MCMC algorithms," *Probability Surveys*, vol. 1, pp. 20–71, 2004.

[19] S. Agapiou, O. Papaspiliopoulos, D. Sanz Alonso, and A. Stuart, "Importance sampling: computational complexity and intrinsic dimension," *arXiv preprint arXiv:1511.06196*, 2015.

[20] R. M. Neal, "Annealed importance sampling," *Statistics and Computing*, vol. 11, no. 2, pp. 125–139, 2001.

[21] C. Andrieu and G. O. Roberts, "The pseudo-marginal approach for efficient Monte Carlo computations," *The Annals of Statistics*, pp. 697–725, 2009.

[22] M. A. Beaumont, "Estimation of population growth or decline in genetically monitored populations," *Genetics*, vol. 164, no. 3, pp. 1139–1160, 2003.

[23] C. Andrieu and M. Vihola, "Convergence properties of pseudo-marginal Markov chain Monte Carlo algorithms," *The Annals of Applied Probability*, vol. 25, no. 2, pp. 1030–1077, 2015.

[24] D. T. Gillespie, "Exact numerical simulation of the Ornstein-Uhlenbeck process and its integral," *Physical Review E*, vol. 54, no. 2, p. 2084, 1996.

[25] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[26] G. Bishop and G. Welch, "An introduction to the Kalman filter," *Proc of SIGGRAPH, Course*, vol. 8, no. 27599-23175, p. 41, 2001.

[27] P. S. Maybeck, *Stochastic models, estimation, and control*, vol. 3. Academic Press, 1982.

[28] G. O. Roberts and J. S. Rosenthal, "Optimal scaling for various Metropolis-Hastings algorithms," *Statistical Science*, vol. 16, no. 4, pp. 351–367, 2001.

[29] K. Kremer and G. S. Grest, "Dynamics of entangled linear polymer melts: A molecular-dynamics simulation," *The Journal of Chemical Physics*, vol. 92, no. 8, pp. 5057–5086, 1990.

[30] N. J. Gordon, D. J. Salmond, and A. F. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," in *IEE Proceedings F (Radar and Signal Processing)*, vol. 140, pp. 107–113, IET, 1993.

[31] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[32] R. Douc and O. Cappé, "Comparison of resampling schemes for particle filtering," in *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on*, pp. 64–69, IEEE, 2005.

[33] A. Chorin, M. Morzfeld, and X. Tu, "Implicit particle filters for data assimilation," *Communications in Applied Mathematics and Computational Science*, vol. 5, no. 2, pp. 221–240, 2010.

[34] G. Burgers, P. Jan van Leeuwen, and G. Evensen, "Analysis scheme in the ensemble Kalman filter," *Monthly Weather Review*, vol. 126, no. 6, pp. 1719–1724, 1998.

[35] G. Evensen, "The ensemble Kalman filter: Theoretical formulation and practical implementation," *Ocean Dynamics*, vol. 53, no. 4, pp. 343–367, 2003.

[36] M. Klaas, N. De Freitas, and A. Doucet, "Toward practical N2 Monte Carlo: The marginal particle filter," *arXiv preprint arXiv:1207.1396*, 2012.

[37] A. Doucet, M. Pitt, G. Deligiannidis, and R. Kohn, "Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator," *Biometrika*, vol. 102, no. 2, pp. 295–313, 2015.

[38] S. H. Klapp, D. J. Diestler, and M. Schoen, "Why are effective potentials 'soft'?," *Journal of Physics: Condensed Matter*, vol. 16, no. 41, p. 7331, 2004.

[39] R. Zwanzig, *Nonequilibrium statistical mechanics*. Oxford University Press, 2001.

[40] J. K. Pritchard, M. T. Seielstad, A. Perez-Lezaun, and M. W. Feldman, "Population growth of human Y chromosomes: a study of Y chromosome microsatellites.," *Molecular Biology and Evolution*, vol. 16, no. 12, pp. 1791–1798, 1999.

[41] P. E. Jacob, F. Lindsten, and T. B. Schön, "Coupling of Particle Filters," *arXiv preprint arXiv:1606.01156*, 2016.

[42] G. Deligiannidis, A. Doucet, and M. K. Pitt, "The correlated pseudo-marginal method," *arXiv preprint arXiv:1511.04992*, 2015.

[43] P. G. de Gennes, "Reptation of a polymer chain in the presence of fixed obstacles," *The Journal of Chemical Physics*, vol. 55, no. 2, pp. 572–579, 1971.

[44] P. G. de Gennes, "Reptation of a polymer chain in the presence of fixed obstacles," in *Simple Views On Condensed Matter*, pp. 149–157, World Scientific, 2003.

[45] T. McLeish, "Tube theory of entangled polymer dynamics," *Advances in Physics*, vol. 51, no. 6, pp. 1379–1527, 2002.

[46] A. E. Likhtman and T. C. McLeish, "Quantitative theory for linear dynamics of linear entangled polymers," *Macromolecules*, vol. 35, no. 16, pp. 6332–6343, 2002.

[47] M. Rubinstein and R. H. Colby, "Self-consistent theory of polydisperse entangled polymers: Linear viscoelasticity of binary blends," *The Journal of Chemical Physics*, vol. 89, no. 8, pp. 5291–5306, 1988.

[48] M. Doi, "Explanation for the 3.4 power law of viscosity of polymeric liquids on the basis of the tube model," *Journal of Polymer Science Part C-Polymer Letters*, vol. 19, no. 5, pp. 265–273, 1981.

[49] A. E. Likhtman, "Single-chain slip-link model of entangled polymers: Simultaneous description of neutron spin- echo, rheology, and diffusion," *Macromolecules*, vol. 38, no. 14, pp. 6128–6139, 2005.

[50] J. Ramírez, S. K. Sukumaran, and A. E. Likhtman, "Significance of cross correlations in the stress relaxation of polymer melts," *The Journal of Chemical Physics*, vol. 126, no. 24, p. 244904, 2007.

[51] H. Jeffreys, *The theory of probability*. OUP Oxford, 1998.

[52] L. Tierney and J. B. Kadane, "Accurate approximations for posterior moments and marginal densities," *Journal of the American Statistical Association*, vol. 81, no. 393, pp. 82–86, 1986.

[53] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[54] M. J. Beal, *Variational algorithms for approximate Bayesian inference*. University of London, 2003.

[55] T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pp. 362–369, Morgan Kaufmann Publishers Inc., 2001.

[56] M. A. Newton and A. E. Raftery, "Approximate Bayesian inference with the weighted likelihood bootstrap," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 3–48, 1994.

[57] P. J. Green, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.

[58] D. I. Hastie and P. J. Green, "Model choice using reversible jump Markov chain Monte Carlo," *Statistica Neerlandica*, vol. 66, no. 3, pp. 309–338, 2012.

[59] G. Karagiannis and C. Andrieu, "Annealed importance sampling reversible jump MCMC algorithms," *Journal of Computational and Graphical Statistics*, vol. 22, no. 3, pp. 623–648, 2013.

[60] A. Gelman and X. L. Meng, "Simulating normalizing constants: From importance sampling to bridge sampling to path sampling," *Statistical Science*, pp. 163–185, 1998.

[61] P. Alquier, N. Friel, R. Everitt, and A. Boland, "Noisy Monte Carlo: Convergence of Markov chains with approximate transition kernels," *Statistics and Computing*, vol. 26, no. 1-2, pp. 29–47, 2016.

[62] P. Del Moral, A. Doucet, and A. Jasra, "Sequential Monte Carlo samplers," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 3, pp. 411–436, 2006.

[63] P. Del Moral, A. Doucet, and A. Jasra, "An adaptive sequential Monte Carlo method for approximate Bayesian computation," *Statistics and Computing*, vol. 22, no. 5, pp. 1009–1020, 2012.

[64] M. A. Beaumont, J. M. Cornuet, J. M. Marin, and C. P. Robert, "Adaptive approximate Bayesian computation," *Biometrika*, vol. 96, no. 4, pp. 983–990, 2009.

[65] Z. Wang and R. G. Larson, "Constraint release in entangled binary blends of linear polymers: A molecular dynamics study," *Macromolecules*, vol. 41, no. 13, pp. 4945–4960, 2008.

[66] Z. Wang, A. E. Likhtman, and R. G. Larson, "Segmental dynamics in entangled linear polymer melts," *Macromolecules*, vol. 45, no. 8, pp. 3557–3570, 2012.

[67] S. Shanbhag and M. Kröger, "Primitive path networks generated by annealing and geometrical methods: Insights into differences," *Macromolecules*, vol. 40, no. 8, pp. 2897–2903, 2007.

[68] T. Uneyama and Y. Masubuchi, "Multi-chain slip-spring model for entangled polymer dynamics," *The Journal of Chemical Physics*, vol. 137, no. 15, p. 154902, 2012.

[69] V. C. Chappa, D. C. Morse, A. Zippelius, and M. Müller, "Translationally invariant slip-spring model for entangled polymer dynamics," *Physical Review Letters*, vol. 109, no. 14, p. 148302, 2012.

[70] M. Langeloth, Y. Masubuchi, M. C. Böhm, and F. Müller-Plathe, "Recovering the reptation dynamics of polymer melts in dissipative particle dynamics simulations via slip-springs," *The Journal of Chemical Physics*, vol. 138, no. 10, p. 104907, 2013.

[71] A. Ramírez-Hernández, F. A. Detcheverry, B. L. Peters, V. C. Chappa, K. S. Schweizer, M. Müller, and J. J. De Pablo, "Dynamical simulations of coarse grain polymeric systems: Rouse and entangled dynamics," *Macromolecules*, vol. 46, no. 15, pp. 6287–6299, 2013.

Appendix

Laplace transform

The Laplace transform is a frequency-domain approach for continuous time signals. The Laplace transform of a function $f(t)$, defined for all real numbers $t \geq 0$, is the function $F(s)$, which is a unilateral transform defined by

$$F(s) = \int_0^\infty f(t)e^{-st}\,dt$$

where s is a complex number frequency parameter $s = \sigma + i\omega$, with real numbers σ and ω.

Initial value theorem:

$$f(0^+) = \lim_{s\to\infty} sF(s). \tag{1}$$

Final value theorem:

$$f(\infty) = \lim_{s\to 0} sF(s), \tag{2}$$

The final value theorem is useful because it gives the long-term behaviour without having to perform partial fraction decompositions or other difficult algebra.

Linearity: The Laplace transform of a sum is the sum of Laplace transforms of each term.

$$\mathcal{L}\{f(t) + g(t)\} = \mathcal{L}\{f(t)\} + \mathcal{L}\{g(t)\}$$

The Laplace transform of a multiple of a function is that multiple times the Laplace transformation of that function.

$$\mathcal{L}\{af(t)\} = a\mathcal{L}\{f(t)\}$$

The Laplace transform is linear, i.e., homogeneity and superposition hold.

Differentiation in s-Domain

$$\mathcal{L}[tf(t)] = -\frac{d}{ds}F(s), \tag{3}$$

This can be proven by differentiating the Laplace transform:

$$\frac{d}{ds}F(s) = \int_{-\infty}^{\infty} f(t)\frac{d}{ds}e^{-st}dt = \int_{-\infty}^{\infty}(-t)f(t)e^{-st}dt$$

Repeat this process we get

$$\mathcal{L}[t^n f(t)] = (-1)^n \frac{d^n}{ds^n}F(s)$$

Integration in time domain

$$\mathcal{L}\left[\int_0^t f(\tau)d\tau\right] = \frac{F(s)}{s}, \tag{4}$$

This can be proven by realizing that

$$x(t) * u(t) = \int_{-\infty}^{\infty} x(\tau)u(t-\tau)d\tau = \int_{-\infty}^{t} x(\tau)d\tau$$

where $u(t)$ is the Heaviside step function. Therefore by convolution property we have

$$\mathcal{L}[x(t) * u(t)] = X(s)\frac{1}{s}$$

Using final value theorem, integral $\int_0^t f(\tau)d\tau$ is give by

$$\int_0^{\infty} f(\tau)d\tau = F(0) \tag{5}$$

To calculate Laplace transform of integration $tf(t)$ in time domain,applying Eq.(3) and Eq.(4) gives us

$$\mathcal{L}\left[\int_0^t \tau f(\tau)d\tau\right] = -\frac{1}{s}\frac{d}{ds}F(s),$$

Using final value theorem, integral $\int_0^{\infty} \tau f(\tau)d\tau$ is give by

$$\int_0^{\infty} \tau f(\tau)d\tau = -\frac{dF(s)}{ds}|_{s=0} \tag{6}$$