



Efficient Sampling of Rare Event Pathways

from simple models to nucleation

Daniele Moroni

Copyright ©2005 Universiteit van Amsterdam

Daniele Moroni - PhD Thesis
van't Hoff Institute for Molecular Science
Universiteit van Amsterdam
Nieuwe Achtergracht 166
1018WV - Amsterdam
The Netherlands

This document is available at

<http://www.science.uva.nl/~moroni/thesis.html>

The author can be reached at

moroni@science.uva.nl

Efficient Sampling of Rare Event Pathways

from simple models to nucleation

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. mr. P.F. van der Heijden
ten overstaan van een door het college voor promoties
ingestelde commissie, in het openbaar te verdedigen
in de Aula der Universiteit
op donderdag 28 april 2005, te 10:00 uur.

door

Daniele Moroni

geboren te Rome, Italië

Promotiecommissie:

Promotor: prof. dr. ir. B. Smit
Copromotor: dr. P.G. Bolhuis

Overige leden: prof. dr. C. Dellago
 dr. T.S. van Erp
 prof. dr. D. Frenkel
 prof. dr. B. Nienhuis
 dr. P.R. ten Wolde

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

The research reported in this thesis was carried out at the Van 't Hoff Institute for Molecular Sciences, Faculty of Science, University of Amsterdam (Nieuwe Achtergracht 166, 1018 WV, Amsterdam, The Netherlands) with financial support from a PIONIER project of the council for chemical sciences of the Netherlands Organization for Scientific Research (NWO-CW), and a grant from the University "La Sapienza", Rome, Italy.

a mia madre
a mio padre

Papers by the author related to this thesis:

Chapter 3

A novel path sampling method for the calculation of rate constants

Titus S. van Erp, Daniele Moroni, and Peter G. Bolhuis

J. Chem. Phys. **118**, 7762 (2003)

Investigating Rare Events by Transition Interface Sampling

Daniele Moroni, Titus S. van Erp, Peter G. Bolhuis

Physica A **340**, 395 (2004)

Chapter 4

Rate constants for diffusive processes by partial path sampling

Daniele Moroni, Peter G. Bolhuis, and Titus S. van Erp

J. Chem. Phys. **120**, 4055 (2004)

Chapter 5

Simultaneous computation of free energies and kinetics of rare events

Daniele Moroni, Titus S. van Erp, and Peter G. Bolhuis

Phys. Rev. E **71**, in press (2005)

Chapter 7

The interplay between structure and size in the critical crystal nucleus

Daniele Moroni, Pieter Rein ten Wolde, and Peter G. Bolhuis

submitted

Other papers by the author:

Reorientation dynamics in thin glassy films

Elio Cecchetto, Daniele Moroni, and Blandine Jerome

submitted

Dinâmica de um sistema não extensivo de rotores clássicos anisotrópicos acoplados

Ernesto P. Borges, Constantino Tsallis, Andrea Giansanti, Daniele Moroni

in Tendências da Física Estatística no Brasil,

Editoria Livraria da Física, São Paulo (2003)

Canonical solution of classical magnetic models with long-range couplings

Alessandro Campa, Andrea Giansanti, and Daniele Moroni

J. Phys. A **36**, 6897 (2003)

Metastable states in a class of long-range Hamiltonian systems

Alessandro Campa, Andrea Giansanti, and Daniele Moroni

Physica A **305**, 137 (2002)

Universal behaviour in the static and dynamic properties of the α -XY model

Andrea Giansanti, Daniele Moroni and Alessandro Campa

Chaos, Solitons and Fractals **13**, 407 (2002)

Classical spin systems with long-range interactions: universal reduction of mixing

Alessandro Campa, Andrea Giansanti, Daniele Moroni, and Constantino Tsallis

Phys. Lett. A **286**, 251 (2001)

Canonical solution of a system of long-range interacting rotators on a lattice

Alessandro Campa, Andrea Giansanti, and Daniele Moroni

Phys. Rev. E **62**, 303 (2000)

Contents

List of Symbols	viii
Introduction	1
1 Rare Events	3
1.1 The study of rare events	3
1.2 Random telegraph	4
1.3 Definitions	7
1.4 Transition State Theory	8
1.4.1 TST rates	9
1.4.2 TST rate expression	10
1.4.3 Notes to the TST rate expression	12
1.5 Bennett-Chandler procedure	13
1.5.1 Implementation of TST-BC procedure	14
1.5.2 Transmission coefficient	16
1.5.3 Alternative expressions for κ	19
1.5.4 Variational TST-BC	22
1.6 Other methods	23
1.6.1 Open-ended methods	24
1.6.2 Two-ended methods	24
1.7 Summary	25
2 Transition Path Sampling	26
2.1 The transition path ensemble	26
2.1.1 Path probability	27
2.1.2 Order parameters	28
2.2 Sampling the path ensemble	29
2.2.1 Shooting move	29
2.2.2 Shifting move	34
2.2.3 Path reversal	35

2.2.4	Joining the algorithms	36
2.3	Computing Rates	38
2.3.1	Umbrella sampling	38
2.3.2	Path ensemble average	39
2.4	The $(LJ)_7$ cluster	41
2.4.1	Potential Energy Surfaces	42
2.4.2	Choice of the order parameter	43
2.4.3	Rate constants	46
2.5	Finding the right reaction coordinate	48
2.5.1	Commitment probabilities	49
2.5.2	Committer distributions	51
2.6	Summary	51
3	Transition Interface Sampling	54
3.1	Theory	54
3.1.1	Interfaces and overall states	54
3.1.2	Rate constant	56
3.1.3	Effective positive flux	58
3.1.4	TIS rate expression	61
3.2	TIS algorithm	61
3.2.1	The flux calculation	61
3.2.2	The path sampling	62
3.2.3	Considerations	64
3.3	Numerical results	68
3.3.1	The model	68
3.3.2	Methodology	69
3.3.3	System with High Energy Barrier	69
3.3.4	System with Low Energy Barrier	73
3.4	Summary	78
4	Partial Path TIS	80
4.1	Theory	80
4.1.1	Illustration of the PPTIS concept	81
4.1.2	PPTIS formalism	83
4.2	PPTIS algorithm	85
4.2.1	Memory loss assumption	86
4.3	Numerical Results	88
4.3.1	The model	88
4.3.2	The Simulation	89

4.3.3	Comparing TIS and PPTIS	90
4.3.4	Validity of the memory loss assumption	91
4.4	Simple PPTIS	92
4.4.1	Theory	92
4.4.2	Numerical results	94
4.4.3	Resume	97
4.5	Parallel path swapping	98
4.6	Summary	100
5	Interface Sampling and Free Energies	102
5.1	Theory	102
5.2	Algorithm	105
5.3	Numerical results	107
5.4	Free energy as function of another order parameter	109
5.5	Summary	110
6	Efficiency of Interface Sampling	111
6.1	Scaling	111
6.1.1	TPS and TIS	112
6.1.2	Error analysis	113
6.1.3	TIS - Exponential barrier	114
6.1.4	TIS - Diffusive barrier	116
6.1.5	PPTIS	118
6.2	Transmission coefficients revisited	119
6.2.1	Transmission coefficient based on effective positive flux	119
6.2.2	Comparison with other methods	121
6.3	The acceptance ratio for TIS on flat diffusive barriers	123
6.4	Summary	124
7	Crystal Nucleation of LJ particles	125
7.1	The problem of nucleation	125
7.1.1	Rare event simulations of nucleation	127
7.2	Methodology	128
7.2.1	The system	128
7.2.2	LJ phase diagram	129
7.2.3	Choice of the order parameter	130
7.3	Rate constant	134
7.4	Mechanism	136
7.4.1	Path analysis	137

7.4.2	Transitions	138
7.4.3	Committer analysis	140
7.4.4	Transition state ensemble	143
7.5	Summary	149
Conclusion		150
A Rates as first passage times		152
A.1	Eigenvalue problem for the Fokker-Planck operator	153
A.2	Mean first passage time	154
A.3	Correlation function for reactive flux	158
A.4	A model	159
B Alternative TST rate expression		161
C RRKM theory		163
D Flux relation		167
E Recursive relations for PPTIS		169
E.1	Recursive relations for the illustrative example	169
E.2	Recursive relations for a general barrier	170
E.3	Recursive relations for simple PPTIS	171
E.4	Probability relation for symmetrical hopping model	172
F Biased and reweighted averages		174
G Classical Nucleation Theory		177
G.1	Nucleation barrier	177
G.2	Equilibrium distribution of cluster sizes	180
G.3	Nucleation rate	181
H NPH dynamics		183
I Bond order parameters		187
I.1	Use in the analysis of cluster structures	188
I.2	Use in the determination of the biggest cluster	190

J Generalized committors	192
K Algorithms	194
K.1 TIS	194
K.2 PPTIS	200
Bibliography	203
Samenvatting	208
Acknowledgements	211
To my friends	212
Index	214

List of Symbols

We refer to pages where the symbols are used. The order is alphabetical, lower case first, and ignoring typeface. There are three parts to the index: Latin, Greek, and Mathematical.

Latin symbols

b	barrier length	17, 114-123
$C(t)$	TST-BC correlation function	13
C	equilibrium constant	84
d	system dimension	7
\mathcal{D}	path differential	27
$f_t(x_0)$	time-propagator	27
$h_\Omega(x)$	characteristic function of state Ω	8
$h_{i,j}^b(x)$	backward characteristic function	55
$h_{i,j}^f(x)$	forward characteristic function	55
\mathcal{H}	Hamiltonian	174
H	Enthalpy	130
\mathbf{I}	moment of inertia tensor	43
$I_1 \leq I_2 \leq I_3$	moments of inertia	43
$k(t)$	reactive flux ($= \dot{C}(t)$)	13
k_{AB}, k_{BA}	rate constants	7, app. A
k_B	Boltzmann constant	7
\mathcal{K}	total kinetic energy	174
\mathbf{L}	total angular momentum	164
L_r	Fokker-Planck operator	153
M	diagonal mass matrix	7
$n_A(t r)$	time-dependent set measure	157
n	$d \cdot N$	7
n_I	(PP)TIS interfaces go from 0 to n_I	54
n_{big}	size of the biggest solid cluster	130
n	size of a solid cluster	179
n_{bulk}	number of bulk cluster particles	138
n_{surf}	number of surface cluster particles	138
N_W	number of windows	111
N	number of particles	7
\mathcal{N}	number of shooted paths	111

N_{cy}	number of path sampling MC cycles	69, 91
N_{bl}	number of independent simulation blocks	69, 91
N_{tr}	number of (independent) trajectories	17, 49
$p(x, t x', t')$	conditional probability density	5, 27, 152
$p(x, t; x', t')$	joint probability density	158
p_{st}	stationary distribution	153, 6
\mathcal{P}	path probability	27
\mathbf{p}	Cartesian momentum, dimension d	43, 174
p	Cartesian momentum, dimension n	26, 174
\mathbf{P}	total linear momentum	164
$P_{(m j)}^l$	4-interface probability	59
\mathbf{r}	Cartesian coordinate, dimension d	42, 174
r	Cartesian coordinate, dimension n	7
r_d	interdimer distance	68, 88
r_q	neighbor radius	130
$R(t)$	unnormalized transmission coefficient	14
\mathbf{R}_g^2	radius of gyration tensor	137
R_g^2	radius of gyration	137
$R_{g1}^2 \geq R_{g2}^2 \geq R_{g3}^2$	radius of gyration components	137
s	number of degrees of freedom	42, 164
s	hopping model has $s - 1$ states	81
t^{mfp}	mean first passage time	6, sec. A.2
t^{mr}	mean residence time	6, 9
T	temperature	7
\mathcal{T}	path or trajectory duration	27, 6
\mathcal{U}	total potential energy	174
U	internal energy	177
U	two-body interaction	43
U_{dw}	dimer double well potential	68
U_{ddw}	dimer diffusive double well potential	88
U_{WCA}	Weeks-Chandler-Andersen interaction	68
U_{dw}	double-well interaction	68
U_{LJ}	Lennard-Jones interaction	43
v	Cartesian velocity, dimension n	7
V	volume	177, 183
w	CNT jump probabilities	181
$W(x x')$	jump probabilities	5
W_{AB}, W_{BA}	random telegraph jump probabilities	5
x	phase space point, system state	26
y	random number in $[0, 1]$	63, 99
z_f	zone system, first zone	65
z_l	zone system, last zone	65

Greek symbols

α_i	algorithm probabilities	36
------------	-------------------------	----

x List of Symbols

β	inverse temperature ($1/k_B T$)	7
γ	Langevin friction coefficient	8
γ	surface tension	126, 177
$\delta(x)$	Dirac delta function	11
$\delta_{i,j}$	Kronecker delta	5
Δt	time increment	27
$\dot{\eta}$	white noise	8
$d\eta$	Wiener process	152
$\phi_{ij}(x)$	Effective positive flux	58
$\kappa(t), \kappa$	transmission coefficient	16
λ	reaction coordinate	15
λ^*	TST-BC dividing surface	15
λ, η	order parameters	28
μ	eigenvalue	164
π	scaled momenta	183
ρ	scaled coordinates	183
ρ_{st}	stationary distribution, path dynamics	31
$d\sigma$	surface element	16, 162
$\tau_{i,i\pm 1}$	hopping model, jump probabilities	81
τ_{stable}	lifetime of a stable state	3
τ_{trans}	duration of a transition	3
$\theta(x)$	Heaviside step function	11

Mathematical symbols

∇	gradient operator	8
Δ	Laplacian operator	152
\Re	real part	191
\Im	imaginary part	191
\dot{x}	time derivative	8
x^T	transpose	164
$\mathbf{1}$	unit matrix	43
$\langle \cdot \rangle$	ensemble average	7
$\langle \cdot \rangle_{\lambda=\lambda^*}$	constrained ensemble average	12
$\langle \cdot \rangle_w$	weighted ensemble average	59
\equiv	defined as	6

Abbreviations

BC	Bennett-Chandler (procedure)
bcc	body-centered cubic
CNT	Classical Nucleation Theory
fcc	face-centered cubic
hcp	hexagonal close packed
LJ	Lennard-Jones
MC	Monte Carlo

MD	Molecular Dynamics
MEP	Minimum Energy Path
MLF	Memory Loss Function
NCP	Number of Connections per Particle
PES	Potential Energy Surface
PPTIS	Partial Path Transition Interface Sampling
RC	Reaction Coordinate
RRKM	Rice, Ramsperger, Kassel, Marcus (theory)
TIS	Transition Interface Sampling
TPS	Transition Path Sampling
TSE	Transition State Ensemble
TST	Transition State Theory
WCA	Weeks-Chandler-Andersen

Introduction

*Alice opened the door and found
that it led into a small passage. . .*

Lewis Carroll

This thesis is concerned with computer simulations of rare events: physical or chemical transitions of a system from one stable state to another that occur with a frequency too low to be observed on a computer timescale. Rare events are not the passage of the Halley comet, but, for example, protein folding, conformational changes of molecules, chemical reactions (in solution), ion dissociation, cluster isomerizations, enzymatic reactions, diffusion in solids, complex surface reactions, and (crystal) nucleation during phase transitions (see Fig. 1). Popular computer simulation tools such as Molecular Dynamics (MD) generate dynamical trajectories by integration of the equations of motion for all the atoms in the system. However, since the fundamental molecular timestep is on the order of fs, there is a time gap between computers and reality, which even with current computers is of the order ¹ of $\mathcal{O}(10^{14})$: 100 s of computer time can only simulate 1 ps of reality. To observe an event that occurs once a μ s would then mean to wait for about 3 years. Thus, straightforward MD cannot be used and special methods must be devised.

The presence of stable states and rare transitions can be understood in terms of free-energy landscapes, where the stable states correspond to the free-energy minima and transitions to crossings of the barrier separating these minima. The traditional approach in the study of these processes, as in Transition State Theory (TST), is based on the calculation of the free energy as function of an intuitively chosen variable, the reaction coordinate, that describes the advance of the reaction. In chapter 1 of this thesis we define the problem of rare events and review TST together with its corrected version, the Bennett-Chandler procedure. The choice of a reaction coordinate is far from trivial in complex systems, and these methods might give bad results. Moreover, the static equilibrium free-energy calculation might disregard important kinetic aspects. All these factors lead to a wrong description of the transition. In order to avoid these effects, Chandler and coworkers created Transition Path Sampling (TPS), to which we dedicate chapter 2. The method gathers a collection of true dynamical trajectories connecting the stable states without any priori assumption of

¹ Assuming conventional MD and not *ab initio* MD, and a system of $\mathcal{O}(10^4)$ particles, which at unity density corresponds to a volume of $\mathcal{O}(10^6)\text{\AA}^3$.

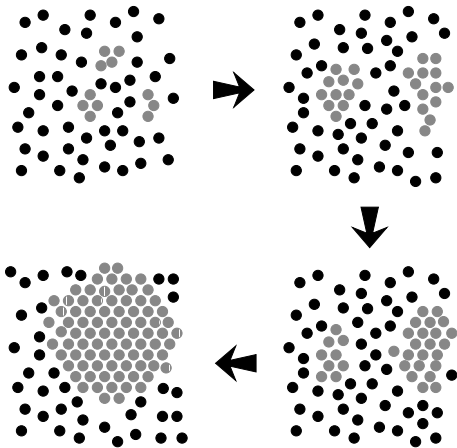


Figure 1: The concept of homogeneous liquid-solid nucleation. Small solid clusters are present in the undercooled liquid (top left). The nuclei grow to different structures (top right and bottom right) and finally a large solid cluster eventually grows to full solidification of the system. Experimental rates on liquid mercury at -120°C are around 10^3 nucleation drops per cubic meter per second [1]. Transposed in computer time this would take more than 10^{27} years. For comparison, the age of the universe is about 10^{10} years [2].

the reaction coordinate. From the ensemble of pathways rate constants can be deduced and the reaction mechanism can be extracted. A recent TPS study [3,4] on a 16-residue peptide was able to elucidate the folding mechanism of a β -hairpin [5] in water, showing the importance of the solvent as a lubricant in the folding process.

In this thesis we show how to improve the efficiency of path sampling. The resulting new method, which we called Transition Interface Sampling (TIS), is the subject of chapter 3. It takes the name from the use of phase space surfaces between the states, the interfaces. By computing fluxes through the interfaces with paths of variable duration, TIS speeds up the rate constant calculation. For diffusive systems we devised a variation of TIS, called Partial Path TIS (PPTIS), which effectively exploits the loss of long time correlation and is described in chapter 4. In chapter 5 we show how the interface sampling methods, which are based on the generation of dynamical trajectories, can at the same time produce the free-energy profile of the rare event, which is an equilibrium property. Chapter 6 concludes the discussion of interface sampling methods with theoretical derivations for the scaling of the efficiency. The last part of this thesis, chapter 7, is an application of TIS and PPTIS to one of the real cases mentioned above: crystal nucleation. We study a system of Lennard-Jones particles, generating paths from an undercooled liquid to the solid. We compute the rate constant, the free-energy profile, and give an explanation of the mechanism, fully exploiting the power of the path sampling methodology. Finally, in the appendices we treat several theoretical details.

1

Rare Events

*Blue Moon
You saw me standing alone*

Frank Sinatra

We present an introduction to rare events in complex systems of physical importance. We give a heuristical definition of the problem, and gradually refine it to a more rigorous picture. We review the historical Transition State Theory and its corrected improvement known as Bennett-Chandler procedure. They form the basis of Transition Path Sampling and Interface Sampling methods, which will be discussed in the following chapters. We conclude mentioning some other methods that have appeared in literature.

1.1 The study of rare events

Consider the situation depicted in Fig. 1.1. Two subsets A and B of the phase space exist such that if a dynamical trajectory is started in one of them, it stays there for a long time τ_{stable} until it finds its way to escape. It then undergoes a transition to the other state B in a time

$$\tau_{trans} \ll \tau_{stable}. \quad (1.1)$$

This *separation of timescales* is the effect of energy barriers or entropic bottlenecks that partition the system into stable states of lifetime $\sim \tau_{stable}$.

The states can be considered as two valleys separated by a mountain ridge with high and narrow passages. A blind and slow walker in one valley will wander considerably before finding a way to escape. When it does, it will relax comparably fast into the other basin. An example is a Lennard-Jones cluster and the isomerization transition between two conformational minima, which we will study in section 2.4. Another example is a supercooled liquid, which is stable until solid clusters of a critical size are formed. Then they grow indefinitely to full solidification. This is called liquid-solid nucleation and will be considered in chapter 7. More examples can be conformational changes of molecules including protein folding, or chemical reactions in general.

The object of research is the transition itself, that is, we are interested in

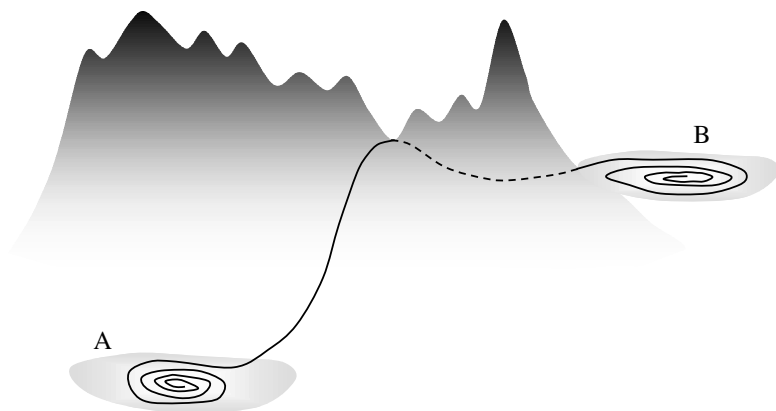


Figure 1.1: Representation of a trajectory in the phase space. It starts in A and spends there a time long compared to the time to switch to B . The separation of timescales is due to an energy barrier separating the states.

- **the mechanism:** understanding the relevant features of the process, and the identification of a (set of) coordinates, called the reaction coordinate, that explains how the reaction proceeds.
- **the transition states:** what are the dividing passages, what is the relevant change that the system must undergo to switch state
- **the rate constants:** the transition probabilities per unit time. For the process $A \rightarrow B$ we call it k_{AB} . It can be considered as the frequency of the event, so that k_{AB}^{-1} is the lifetime of state A . Corresponding concepts hold for the reversed process and k_{BA} . We make these definitions more rigorous in the following sections.

In computer simulations the application of straightforward Molecular Dynamics (MD) would in principle yield all this kinetical information we are looking for. However the state lifetime τ_{stable} depends exponentially on barrier heights, and can become very large. We have not attempted to quantify the ratio $\tau_{trans}/\tau_{stable}$ in Eq. (1.1). Indeed, the practical cases show quite a range of values, as it will become evident in the rest of the thesis. A common feature is the fact that the expectation time for a transition can easily exceed current computer capabilities by many orders of magnitude. In this case the transition becomes a *rare event* and special methods must be employed.

1.2 Random telegraph

Since we are interested in transitions between two stable regions, a basic and widely used way to model it, is a two-state system. Though a simplified scheme it can

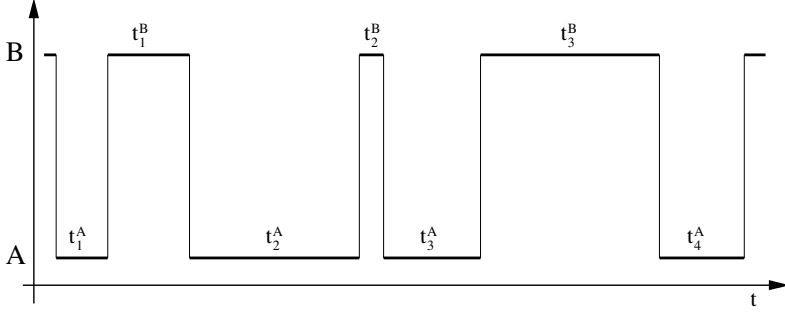


Figure 1.2: A schematic representation of a dynamical trajectory of a random telegraph. Residence periods t_i^A, t_i^B in the states are separated by instantaneous switches between them, with a transition probability per unit time given by W_{AB} for the process $A \rightarrow B$ and W_{BA} for the reversed process.

highlight much of the essence of the problem. Indeed it has proved very useful in modelling many other physical systems as well.

Consider a Markov random process $X(t)$ which can only assume the values $x = A, B$. Let the jump probabilities be

$$W(x|x') = \begin{cases} W_{AB} & \text{if } x = B, x' = A, \\ W_{BA} & \text{if } x = A, x' = B, \end{cases} \quad (1.2)$$

and $W(A|A) = 1 - W(B|A)$, $W(B|B) = 1 - W(A|B)$ follow from normalization. Here $W(x|x')$ is the transition probability per unit time of jumping to x provided that you were in x' . The process is usually known as random telegraph [6] and its behavior is schematically represented in Fig. 1.2.

The basic quantity in the study of Markov processes is the probability

$$p(x, t|x', 0) = \begin{aligned} &\text{Probability of being in } x \text{ at time } t \\ &\text{provided that you were in } x' \text{ at time } 0 \end{aligned} \quad (1.3)$$

For a random telegraph the corresponding Master Equation for (1.3) is

$$\frac{d}{dt} p(A, t|x', 0) = -W_{AB} p(A, t|x', 0) + W_{BA} p(B, t|x', 0) \quad (1.4a)$$

$$\frac{d}{dt} p(B, t|x', 0) = W_{AB} p(A, t|x', 0) - W_{BA} p(B, t|x', 0) \quad (1.4b)$$

subject to the initial condition

$$p(x, 0|x', 0) = \delta_{x,x'}. \quad (1.5)$$

Summing equations (1.4) we get the conserved quantity

$$p(A, t|x', 0) + p(B, t|x', 0) = 1 \quad (1.6)$$

6 1. Rare Events

where the constant value of 1 comes from normalization. The solution is easily written

$$p(A, t|x', 0) = p_{st}(A) + e^{-(W_{AB}+W_{BA})t} [p(A, 0|x', 0) - p_{st}(A)] \quad (1.7a)$$

$$p(B, t|x', 0) = p_{st}(B) + e^{-(W_{AB}+W_{BA})t} [p(B, 0|x', 0) - p_{st}(B)] \quad (1.7b)$$

where the stationary probabilities $p_{st}(x) = p(x, +\infty|x', 0)$ are independent of the initial point x' and given by

$$p_{st}(A) = W_{BA}/(W_{AB} + W_{BA}) \quad (1.8a)$$

$$p_{st}(B) = W_{AB}/(W_{AB} + W_{BA}) \quad (1.8b)$$

as can be seen directly from (1.4).

Now imagine to observe realizations of a random telegraph, without knowing the underlying jump matrix W . We want to measure the transition probabilities. One way could be to use the definition

$$W(x|x') = \lim_{dt \rightarrow 0} p(x, dt|x', 0)/dt \quad (1.9)$$

For example, one can consider an ensemble of processes starting in A , and check if there is a transition to B in a time dt . Averaging the number of transitions over the total number of samples (times dt) converges to W_{AB} . Alternatively one can make use of ergodicity. If a trajectory visits all the possible states of the system given enough time, the averages on initial conditions can be replaced by time averages, see also sec. 1.3. With reference to Fig. 1.2, one could observe a single realization for a long time. Dividing the trajectory into time slices of length dt , each slice beginning in A is part of the ensemble. Calling $N_{A \rightarrow B}(\mathcal{T})$ the number of transitions $A \rightarrow B$ during time \mathcal{T} , it follows

$$W_{AB} = \lim_{\mathcal{T} \rightarrow \infty} \frac{N_{A \rightarrow B}(\mathcal{T})}{t_{tot}^A(\mathcal{T})} \quad (1.10)$$

where $t_{tot}^A(\mathcal{T}) = \sum_i t_i^A$ is the total time spent in A during \mathcal{T} . Note that $t_{tot}^A(\mathcal{T})$ and not \mathcal{T} appears in (1.10). Since to each transition corresponds a reversed one, using just \mathcal{T} would mean $W_{AB} = W_{BA}$. But the importance of a state is characterized by the time spent in the state. Indeed Eq. (1.10) can be rewritten as

$$W_{AB}^{-1} = \lim_{\mathcal{T} \rightarrow \infty} \frac{1}{N_{A \rightarrow B}(\mathcal{T})} \sum_{i=1}^{N_{A \rightarrow B}(\mathcal{T})} t_i^A \equiv t_A^{mr} \quad (1.11)$$

where t_A^{mr} is the mean residence time in state A .

There is a third quantity of importance in this model. Consider the problem of starting in A and computing the time after which the system first enters B . This is usually called the *first passage time* from A to B , and it is a standard and important topic in the theory of stochastic processes [6, 7]. Mathematically it is treated putting an absorbing boundary condition at B and solving an equation for t_A^{mfp} , the mean first passage time (see also appendix A). For the random telegraph the result is [6]:

$$t_A^{mfp} = W_{AB}^{-1} \quad (1.12)$$

It is equal to the mean residence time t_A^{mr} , because transitions between A and B are instantaneous. This is not the case in real systems and we will come back to this in section 1.5.

Summarizing, in this model we have three equivalent ways of defining the rate constant k_{AB} for the process $A \rightarrow B$:

1. transition probability per unit time

$$k_{AB} = W_{AB} \quad (1.13)$$

2. inverse mean residence time in A

$$k_{AB} = (t_A^{mr})^{-1} \quad (1.14)$$

3. inverse mean first passage time to B

$$k_{AB} = (t_A^{mfp})^{-1} \quad (1.15)$$

With this in mind, we now turn to consider real systems in the full phase space and see how these ideas can be applied. In particular, we will show which of the above definitions still holds, and which one is the most sensible.

1.3 Definitions

Before we tackle the transition problem for real systems, we have to prepend some definitions. Consider a system of N particles in dimension d , and let $n = dN$. We denote with $r \in \mathbb{R}^n$ the vector of Cartesian coordinates and with $v \in \mathbb{R}^n$ the corresponding velocities. Suppose the system admits an equilibrium distribution such as the Gibbs distribution

$$\rho(r, v) = \rho(r) \cdot \rho(v) = Z_r^{-1} e^{-\beta \mathcal{U}(r)} \cdot Z_v^{-1} e^{-\frac{\beta}{2} v^T M v} \quad (1.16)$$

$$Z = Z_r \cdot Z_v = \int_{\mathbb{R}^n} e^{-\beta \mathcal{U}(r)} dr \cdot \int_{\mathbb{R}^n} e^{-\frac{\beta}{2} v^T M v} dv \quad (1.17)$$

where $\mathcal{U}(r)$ is some given potential, $\beta = 1/k_B T$ is the inverse temperature, k_B the Boltzmann constant, and M is the diagonal mass matrix

$$M = \text{diag}(\overbrace{m_1 \dots m_1}^d, \dots, \overbrace{m_N \dots m_N}^d). \quad (1.18)$$

The ensemble average of an observable $f(r, v)$ is defined by

$$\langle f(r, v) \rangle \equiv \int_{\mathbb{R}^n \times \mathbb{R}^n} f(r, v) \rho(r, v) dr dv \quad (1.19)$$

Define for a set $\Omega \subseteq \mathbb{R}^n$

$$\int_{\Omega} \rho(r) dr = \langle h_{\Omega}(r) \rangle \quad (1.20)$$

8 1. Rare Events

where h_Ω is the set characteristic function, i.e. $h_\Omega(r) = 1$ if $r \in \Omega$ and 0 elsewhere. We could in principle consider subsets of the full phase space (r, v) but we will limit us to the configuration space only. We will come back to this in section 2.1 and when dealing with Interface Sampling methods, in chapter 3. The regions A, B are stable if

$$\langle h_A \rangle + \langle h_B \rangle \simeq 1 \quad (1.21)$$

and they are separated by a region of low probability, called the buffer or transition region. In other words the system at equilibrium is found in one of the states with high probability, but is difficult to go from one to the other.

In this thesis we will consider deterministic (e.g. Hamiltonian) and stochastic dynamics. To be general, we include both in the Langevin dynamics

$$\dot{r} = v \quad (1.22a)$$

$$M\dot{v} = -\nabla\mathcal{U}(r) - \gamma v + \sqrt{2\gamma\beta^{-1}}\dot{\eta} \quad (1.22b)$$

where γ is the friction coefficient, and $\dot{\eta}$ a white noise [6]. This dynamics is consistent with the distribution (1.16), see appendix A. Assuming ergodicity we can replace time averages with ensemble average as

$$\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T f(r(t), v(t)) dt = \int_{\mathbb{R}^n \times \mathbb{R}^n} \rho(r, v) f(r, v) dr dv \quad (1.23)$$

and definition (1.20) can be also regarded as the fraction of time spent in region Ω .

When working in the phase space, we use coordinates and momenta $p = Mv$, and we denote the system phase point $x = (r, p)$. If the initial condition of (1.22) is x_0 we denote the evolution at time t with x_t . For $\gamma = 0$ we recover Hamiltonian dynamics and x_t is the evolution of x_0 through the Liouville operator.

For non-zero γ , the dynamics (1.22) mimics a system in contact with a thermal bath, and the evolution of the system is a stochastic process. This picture is not restrictive. Even when considering deterministic dynamics, statistical mechanics requires considering time averages, or by ergodicity, ensemble averages. A deterministic evolution becomes a stochastic process because of the initial conditions extracted from the ensemble distribution $\rho(r, v)$. In this chapter we will therefore treat our systems as stochastic processes, in order to use the powerful theoretical methods, and we do not address the limit of our findings as $\gamma \rightarrow 0$, whose mathematics is far beyond this thesis.

1.4 Transition State Theory

Let us go back to the situation depicted in Fig. 1.1 and apply the rate constant definitions at the end of previous section 1.2. Since the transition takes place in a non-vanishing time, the direct definition (1.13) of jump probability loses its meaning, because the limit (1.9) is always zero. However we can recover (1.13) if the two phase-space subsets A and B are adjacent, i.e. if $B = A^c$, the complement of A . When this is the case, leaving A means entering B , the transition is instantaneous, and definitions (1.13) and (1.14) coincide again.

Transition State Theory (TST) is the oldest attempt to describe bistability in ergodic systems [8,9]. TST exactly computes the mean residence time in the states, but is only an approximation of the transition event. We proceed following [10].

1.4.1 TST rates

Transition State Theory assumes $B = A^c$, i.e. expands the sets A, B of Fig. 1.1 until they touch, see Fig. 1.3(a). Obviously (1.21) becomes

$$\langle h_A \rangle + \langle h_{B=A^c} \rangle = 1. \quad (1.24)$$

Define the mean residence time in A as (see Fig.1.4)

$$t_A^{mr} = \lim_{T \rightarrow \infty} \frac{2}{N(T)} \int_0^T h_A(r(t)) dt \quad (1.25)$$

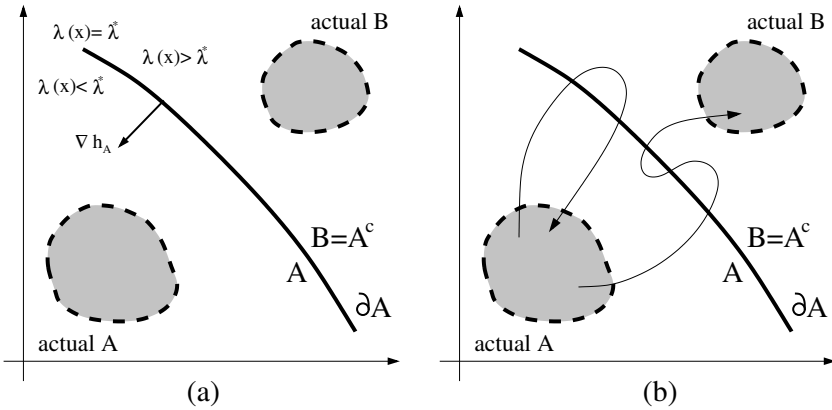


Figure 1.3: (a) the TST assumption: the two actual states are substituted with two adjacent states separated by ∂A . The vector ∇h_A normal to the dividing surface ∂A is also shown. (b) failure of the assumption: some trajectories might exist that are wrongly considered transitions (upper curve), or are overcounted (lower curve).

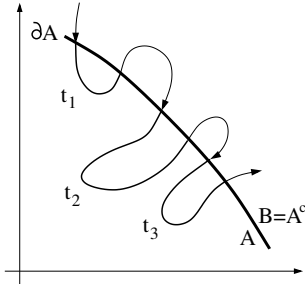


Figure 1.4: Computation of mean residence time. With reference to Eq. (1.25), here $N(T) = 6$, so that $t_A^{mr} = 2/6 \cdot \sum_i t_i = (t_1 + t_2 + t_3)/3$.

where $N(\mathcal{T})$ is the number of times that, within a time \mathcal{T} , the trajectory crosses ∂A , the boundary of A , named the *dividing surface*. The factor 2 accounts for ingoing and outgoing crossings. Similar definition holds for $t_{A^c}^{mr}$. We can rewrite the mean residence time definitions as

$$t_A^{mr} = \langle h_A \rangle / \nu^{TST} \quad t_{A^c}^{mr} = \langle h_{A^c} \rangle / \nu^{TST} \quad (1.26)$$

where

$$\langle h_A \rangle = \lim_{\mathcal{T} \rightarrow \infty} \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} h_A(r(t)) dt = \int_A \rho(r) dr \quad (1.27)$$

is the fraction of time that the trajectory spends in A , (1.24) holds, and

$$\nu^{TST} = \lim_{\mathcal{T} \rightarrow \infty} \frac{N(\mathcal{T})}{2\mathcal{T}} \quad (1.28)$$

is the half mean frequency of crossing the boundary ∂A . As a result, using definition (1.14), we get for the rate constants

$$k_{AB} = (t_A^{mr})^{-1} = \frac{\nu^{TST}}{\langle h_A \rangle} \quad k_{BA} = (t_{A^c}^{mr})^{-1} = \frac{\nu^{TST}}{\langle h_{A^c} \rangle}. \quad (1.29)$$

There are problems with this expression. Since a trajectory that has just crossed the boundary ∂A may have a high probability of re-crossing it right away, the successive transitions between A and A^c are in general correlated. A random telegraph model derived from the transition probabilities (1.29) completely disregards those correlations and the model will only be a poor approximation. The situation is depicted in Fig 1.3(b): when we try to extend the actual states A, B to A, A^c there might appear false and overcounted trajectories. Transition State Theory will be valid only if the dividing surface is perfectly chosen such that each crossing of the surface corresponds to a real transition and each transition has only one crossing with this surface. When the dynamics is reversible, each crossing point corresponding to a true trajectory from actual A to actual B , will result in a trajectory from actual B to actual A when reversing the momenta. As a result, on the TST dividing surface each point has equal probability of ending in the actual stable state A or B .

Finding the best dividing surface is very much the essence of the problem of rare events. We will come back to this very important point several times, but for the moment we just assume we have one and let us see what this implies.

1.4.2 TST rate expression

Let us express (1.28) in a different way. As pictured in Fig. 1.5, $|\dot{h}_A(r(t))|$ is a sum of positive delta functions concentrated at the times when $r(t)$ crosses ∂A . Therefore we can write

$$\begin{aligned} \nu^{TST} &= \lim_{\mathcal{T} \rightarrow \infty} \frac{1}{2\mathcal{T}} \int_0^{\mathcal{T}} |\dot{h}_A(r(t))| dt \\ &= \lim_{\mathcal{T} \rightarrow \infty} \frac{1}{2\mathcal{T}} \int_0^{\mathcal{T}} |\nabla h_A(r) \cdot v(t)| dt \\ &= \frac{1}{2} \int_{\mathbb{R}^n \times \mathbb{R}^n} dr dv \rho(r, v) |\nabla h_A(r) \cdot v| \end{aligned} \quad (1.30)$$

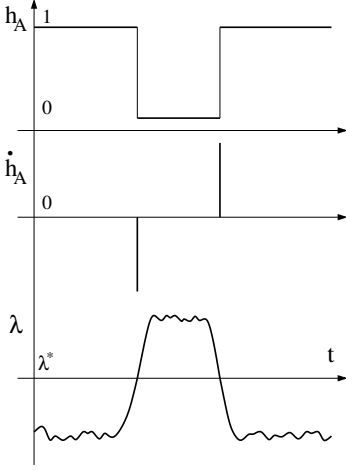


Figure 1.5: Behavior of $h_A(r(t))$, $\dot{h}_A(r(t))$, $\lambda(r(t))$ along a trajectory $r(t)$.

where we used Eq. (1.22a) and ergodicity. We now parametrize the set A using a continuous function $\lambda(r) : \mathbb{R}^n \rightarrow \mathbb{R}$ such that (see again Fig.1.5)

$$\begin{aligned} A &= \{r \in \mathbb{R}^n : \lambda(r) < \lambda^*\} \\ \partial A &= \{r \in \mathbb{R}^n : \lambda(r) = \lambda^*\} \end{aligned} \quad (1.31)$$

As a consequence we can write $h_A(r) = \theta(-\lambda(r) + \lambda^*)$, where $\theta(\cdot)$ is the Heaviside step function. Then $\nabla h_A(r) = -\delta(\lambda(r) - \lambda^*)\nabla\lambda$ is a vector on the surface ∂A , normal to the surface and pointing inward, see Fig. 1.3(a), and thus (1.30) becomes

$$\begin{aligned} \nu^{TST} &= \frac{1}{2} \int_{\mathbb{R}^n \times \mathbb{R}^n} dr dv \rho(r, v) \delta(\lambda(r) - \lambda^*) |\nabla\lambda \cdot v| \\ &= \frac{1}{2} \int_{\mathbb{R}^n \times \mathbb{R}^n} dr dv \rho(r, v) \delta(\lambda(r) - \lambda^*) |\dot{\lambda}| \\ &\equiv \frac{1}{2} \langle \delta(\lambda(r) - \lambda^*) |\dot{\lambda}| \rangle \\ &= \langle \delta(\lambda(r) - \lambda^*) \dot{\lambda} \theta(\dot{\lambda}) \rangle \end{aligned} \quad (1.32)$$

where the last equality follows from $\rho(r, v) = \rho(r, -v)$. This is the usual TST rate expression [11].

Putting together (1.29) and (1.32), (1.20) the final expression for the rate constant is written as

$$k_{AB}^{TST} = \frac{\langle \delta(\lambda(r) - \lambda^*) \dot{\lambda} \theta(\dot{\lambda}) \rangle}{\langle \theta(\lambda^* - \lambda(r)) \rangle} \quad (1.33)$$

Then, if we introduce the free energy

$$e^{-\beta F(\lambda^*)} \equiv \langle \delta(\lambda(r) - \lambda^*) \rangle = \int_{\mathbb{R}^n} dr \rho(r) \delta(\lambda(r) - \lambda^*) \quad (1.34)$$

we can rewrite it as

$$\begin{aligned}
 k_{AB}^{TST} &= \frac{\langle \delta(\lambda(r) - \lambda^*) \dot{\lambda} \theta(\dot{\lambda}) \rangle}{\langle \delta(\lambda(r) - \lambda^*) \rangle} \frac{\langle \delta(\lambda(r) - \lambda^*) \rangle}{\langle \theta(\lambda^* - \lambda(r)) \rangle} \\
 &= \langle \dot{\lambda} \theta(\dot{\lambda}) \rangle_{\lambda=\lambda^*} \frac{e^{-\beta F(\lambda^*)}}{\int_{-\infty}^{\lambda^*} e^{-\beta F(\lambda')} d\lambda'}
 \end{aligned} \tag{1.35}$$

where we have introduced the notation $\langle \cdot \rangle_{\lambda=\lambda^*}$ for the constrained average.

1.4.3 Notes to the TST rate expression

Formula (1.35) is the basis for an implementation in practical cases. This will be discussed in the context of the Bennett-Chandler procedure, in sec. 1.5. For now let us discuss some qualitative aspects.

Given a parametrization $\lambda(r)$ we still have to choose a value for λ^* . We make use of the free energy F , Eq. (1.34), which is essentially the probability of being at $\lambda(r) = \lambda^*$. Since the TST adjacent states A, A^c are supposed to contain the actual stable states A, B , see Fig. 1.3, the free energy as function of λ will display two minima left and right of λ^* , separated by a maximum, see Fig. 1.6. We remarked in sec. 1.4.1 that the dividing surface should be chosen so as to avoid false and multiple recrossings. It seems natural to choose for λ^* the maximum separating the stable states. Then we can make an estimate for the second term in (1.35). Since the free energy has the form of a double well we can approximate the integral with a Gaussian integral around the minimum λ_A to the left of λ^* . The result is

$$k_{AB}^{TST} \propto e^{-\beta(F(\lambda^*) - F(\lambda_A))} \equiv e^{-\beta \Delta F} \tag{1.36}$$

where we have highlighted the exponential dependence on the energy difference, which is the cause of the rarity of the event. This expression shows that the second term in (1.35) involves essentially a free energy computation from A up to a region of minimum probability. TST has turned the dynamical problem of the computation of the rate constant into the equilibrium static problem of computing a free energy difference. We will come back to this in sec. 1.5.4.

Before concluding let us remark that in formula (1.35) the free energy is not the

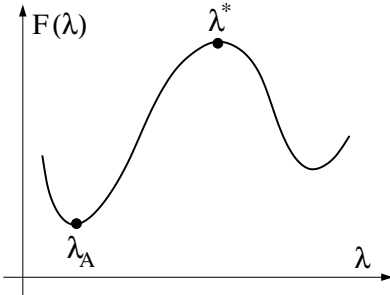


Figure 1.6: Typical free energy, Eq. (1.34) for a parametrization $\lambda(r)$ such that the TST states A, A^c contain the actual stable states A, B .

whole story. We still miss the first term

$$\langle \dot{\lambda} \theta(\dot{\lambda}) \rangle_{\lambda=\lambda^*} = \frac{1}{2} \langle |\dot{\lambda}| \rangle_{\lambda=\lambda^*}. \quad (1.37)$$

If we choose $\lambda(r) = r_1$ equal to a coordinate, then $\dot{\lambda} = v_1$ and the average reduces to a one-dimensional integral which is easy to evaluate (see also sec. B) and gives $\sqrt{k_B T / 2\pi m}$ where m is the mass attached to coordinate 1. In general, however, the term must be fully calculated. We will address this expression (1.37) again as well, in sec. 1.5.2.

1.5 Bennett-Chandler procedure

TST assumes a dividing surface between the stable states A, B and defines them as adjacent sets separated by this surface. However in order to approximate a system with a random telegraph one has to assume quasi-Markov transitions, i.e. successive transitions must be sufficiently decorrelated. To ensure this decorrelation between the surface crossings, it is more practical to define sets A, B that are sufficiently separated, i.e. there exists a buffer region between them that has a large volume but negligible probability.

The situation is shown in Fig. 1.1 and (1.21) holds. We mentioned already that the rate constant definition (1.13) is meaningless, because the transition takes place in a finite time. Definition (1.14) still makes sense, but is not related to a transition probability, because now leaving A does not mean entering B . The sensible definition is based on (1.15).

We show in the appendix A that a real system can be approximated with a random telegraph if one takes for the jump probabilities the inverse mean first passage times. The demonstration is based on the spectral properties of the Fokker-Planck operator associated to the conditional probability (1.3). The assumption (1.1) of separation of timescales is equivalent to the assumption of a *gap* in the spectrum of the operator, and the equations (1.4) can be derived from basic principles.

The Bennett-Chandler (BC) procedure [12,13] can be derived using the correlation function

$$C(t) \equiv \frac{\langle h_A(0) h_B(t) \rangle}{\langle h_A \rangle} \quad (1.38)$$

where $h_A(t) = h_A(r(t))$. It is also shown in the appendix A.3 that as a consequence of the separation of timescales (or equivalently of the spectral gap), for times t such that $\tau_{trans} \ll t \ll \tau_{stable}$, it can be approximated as

$$C(t) \simeq (t_A^{mf})^{-1} t = k_{AB} t \quad (1.39)$$

The result implies that the first derivative $k(t) \equiv \dot{C}(t)$, called the *reactive flux*, has a constant value equal to the rate constant.

Next we apply this correlation function in the context of TST, where the states are adjacent, and separated by the surface $\lambda(r) = \lambda^*$. Then, as in section 1.4.2

$$h_A(r) = \theta(\lambda^* - \lambda(r)) \quad h_B(r) = \theta(\lambda(r) - \lambda^*) \quad (1.40)$$

14 1. Rare Events

To express $\dot{C}(t)$ in a more convenient way, we make use of time translation invariance. Since $\langle h_A(0)h_B(t) \rangle = \langle h_A(-t)h_B(0) \rangle$,

$$\begin{aligned} \frac{d}{dt} \langle h_A(0)h_B(t) \rangle &= \frac{d}{dt} \langle h_A(-t)h_B(0) \rangle \\ &= -\langle \dot{h}_A(-t)h_B(0) \rangle \\ &= -\langle \dot{h}_A(0)h_B(t) \rangle \end{aligned} \quad (1.41)$$

Using then (1.40), we get

$$\dot{C}(t) = \frac{\langle \delta(\lambda^* - \lambda_0) \dot{\lambda}_0 \theta(\lambda_t - \lambda^*) \rangle}{\langle \theta(\lambda^* - \lambda_0) \rangle} \quad (1.42)$$

where $\lambda_t = \lambda(r(t))$. This expression is the basis for a computational implementation, sec. 1.5.1. For $t = 0$ exactly, $\dot{C}(0) = 0$ because it becomes an odd function of velocities (here sets are defined in coordinate space only, remember sec. 1.3). However for small ϵ we can write $\lambda_t = \lambda^* + \dot{\lambda}_0 \epsilon$ because of the presence of the δ -function. Then

$$\begin{aligned} \dot{C}(\epsilon) &\simeq \frac{\langle \delta(\lambda^* - \lambda_0) \dot{\lambda}_0 \theta(\dot{\lambda}_0 \epsilon) \rangle}{\langle \theta(\lambda^* - \lambda_0) \rangle} \\ &= \frac{\langle \delta(\lambda^* - \lambda_0) \dot{\lambda}_0 \theta(\dot{\lambda}_0 \text{sign}(\epsilon)) \rangle}{\langle \theta(\lambda^* - \lambda_0) \rangle} \end{aligned} \quad (1.43)$$

which in the limit $\epsilon \rightarrow 0$, remembering (1.33), implies

$$\dot{C}(0^+) = k_{AB}^{TST} \quad (1.44)$$

$$\dot{C}(0^-) = -\dot{C}(0^+) \quad (1.45)$$

where in the second we used $\theta(-x) = 1 - \theta(x)$ and $\langle \dot{\lambda}_0 \delta(\lambda^* - \lambda_0) \rangle = 0$. Indeed since the sets are defined in coordinate space, $C(t)$ only contains variables even under time inversion [6], and it is immediately seen that $C(t)$ is an even function of time, and $\dot{C}(t)$ an odd one. Finally $C(0) = 0$ directly from the definition (1.38). The behavior of $C(t)$ and $\dot{C}(t)$ is summarized in Fig. 1.7.

1.5.1 Implementation of TST-BC procedure

Formula (1.42) is easily rewritten as

$$\begin{aligned} \dot{C}(t) &= \frac{\langle \delta(\lambda^* - \lambda_0) \dot{\lambda}_0 \theta(\lambda_t - \lambda^*) \rangle}{\langle \delta(\lambda^* - \lambda_0) \rangle} \frac{\langle \delta(\lambda^* - \lambda_0) \rangle}{\langle \theta(\lambda^* - \lambda_0) \rangle} \\ &= \langle \dot{\lambda}_0 \theta(\lambda_t - \lambda^*) \rangle_{\lambda_0 = \lambda^*} \frac{e^{-\beta F(\lambda^*)}}{\langle \theta(\lambda^* - \lambda_0) \rangle} \\ &\equiv R(t) \frac{e^{-\beta F(\lambda^*)}}{\langle \theta(\lambda^* - \lambda_0) \rangle} \end{aligned} \quad (1.46)$$

where we have introduced the unnormalized transmission coefficient $R(t)$.

The actual implementation in a computer simulation consists of three steps:

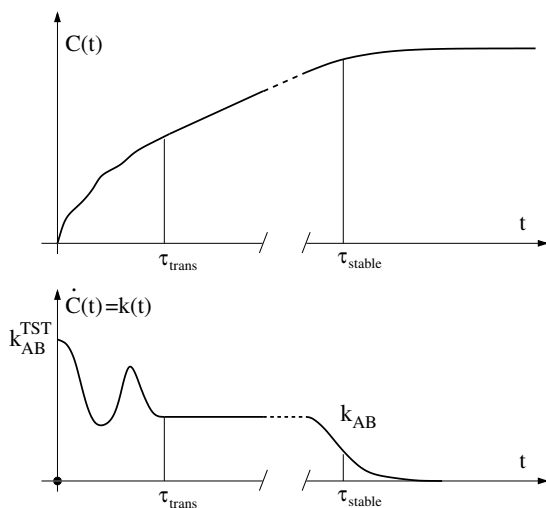


Figure 1.7: Correlation function for the Bennett-Chandler procedure, and its time derivative. Note the difference with the TPS correlation function $\dot{C}(t)$, see Fig. 2.7, for which in place of the jump at $t = 0$ there is a continuous behavior from the value $\dot{C}(0) = 0$ due to the existence of a buffer region and a finite minimum transition time.

1. **Choice of a reaction coordinate.** We must choose a parametrization (1.31), i.e. the function $\lambda(r)$ and the value λ^* , so that $A = \{r : \lambda(r) < \lambda^*\}$ and $B = A^c = \{r : \lambda(r) > \lambda^*\}$. This function must be able to detect a transition, because A and A^c are supposed to contain the actual stable states. A reasonable behavior for the function would be to maintain two different range of values when in A, B and continuously and monotonously change from one to the other under the transition, and only under the true transition, see Fig. 1.5. It is then usually called a *reaction coordinate*, and it is supposed to describe the advancement of the transition event. The actual choice in practical cases is derived from chemical and physical intuition for the specific transition under investigation. Common reaction coordinates in chemical reactions are dihedral angles, or interatomic distances, that can track the approaching of reactants or the detaching of products. For crystal nucleation it could be the size of the growing solid cluster.

We must stress that the TST-BC procedure does not give any information on how to choose a reaction coordinate. It just *assumes* there is one. Then when a $\lambda(r)$ is chosen, we noticed already in sec.1.4.3 that a reasonable choice for λ^* is the maximum in the free energy $F(\lambda)$. We will come back to this in sec. 1.5.4.

2. **Free energy calculation.** Because of the definition (1.34) the free energy could be computed in a normal MD or MC simulation directly through the probability histogram of λ , see Eq. (1.34). However if λ^* is at the free energy maximum, the system will rarely visit up to the minimum-probability region of λ^* (that was the whole problem after all!). Fortunately this step can be solved flawlessly. There exist powerful methods like Umbrella Sampling [14] and Blue Moon Sampling [15, 16] that, based on biasing techniques, can overcome this problem.

3. **Transmission coefficient.** The last step is the calculation of $R(t)$. This implies computing a constrained average, on the surface $\lambda(r) = \lambda^*$ and can be achieved using constrained dynamics [17] or umbrella sampling. However in numerical simulation it is important to distinguish between expressions that are correct in principle, and those that are correct and computationally efficient. There are indeed computational problems with the transmission coefficient, and we dedicate to it the next section.

1.5.2 Transmission coefficient

Let us rewrite (1.46) by normalizing $R(t)$ to its value at $t = 0^+$. With the same reasoning used in Eq. (1.43) we have

$$R(0^+) = \langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \rangle_{\lambda_0=\lambda^*} \quad (1.47)$$

and we can rewrite (1.46) as

$$\begin{aligned} \dot{C}(t) &= \frac{R(t)}{R(0^+)} R(0^+) \frac{e^{-\beta F(\lambda^*)}}{\langle \theta(\lambda^* - \lambda_0) \rangle} \\ &= \frac{\langle \dot{\lambda}_0 \theta(\lambda_t - \lambda^*) \rangle_{\lambda_0=\lambda^*}}{\langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \rangle_{\lambda_0=\lambda^*}} \langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \rangle_{\lambda_0=\lambda^*} \frac{e^{-\beta F(\lambda^*)}}{\langle \theta(\lambda^* - \lambda_0) \rangle} \\ &= k_{AB}^{TST} \frac{\langle \dot{\lambda}_0 \theta(\lambda_t - \lambda^*) \rangle_{\lambda_0=\lambda^*}}{\langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \rangle_{\lambda_0=\lambda^*}} \\ &\equiv k_{AB}^{TST} \kappa(t) \end{aligned} \quad (1.48)$$

where we used (1.35). The time dependence of the correlation function $C(t)$ is now in $\kappa(t)$, which because of the properties of $\dot{C}(t)$, Fig. 1.7, will plateau to a value

$$\kappa = k_{AB}/k_{AB}^{TST}, \quad (1.49)$$

called the transmission coefficient. There is a simple meaning attached to this quantity. The TST rate constant is basically a flux through the surface ∂A defined by $\lambda(r) = \lambda^*$, which can be seen using $k_{AB} = \nu^{TST}/\langle h_A \rangle$, Eq. (1.29), and rewriting (1.30) as

$$\nu^{TST} = \frac{1}{2} \int_{\mathbb{R}^n} dv \int_{\partial A} d\sigma(r) \rho(r, v) |\hat{n} \cdot v| \quad (1.50)$$

where $\hat{n} = \nabla h_A(r)/|\nabla h_A(r)|$ is the normal to the surface, and $d\sigma(r) = \delta(\lambda(r) - \lambda^*) |\nabla \lambda| dr$ the surface element. As shown in Fig. 1.3b, the problem with this expression is the over-counting of trajectories. The correct rate can be expressed using [18, 19]

$$\nu = \frac{1}{2} \int_{\mathbb{R}^n} dv \int_{\partial A} d\sigma(r) \rho(r, v) |\hat{n} \cdot v| \chi(r, v) \quad (1.51)$$

where $\chi(r, v)$ is non-zero only for true reactive trajectories, i.e. paths from actual state A to actual state B or viceversa. If $N_R(r, v)$ is the number of crossing of ∂A between the times outside A and B , one has [10]

$$\chi(r, v) = \frac{1}{2} \left(1 - (-1)^{N_R(r, v)} \right) / N_R(r, v) \quad (1.52)$$

The first factor is 1 only if N_R is odd, i.e. the trajectory connects the states, and zero otherwise. The second factor N_R^{-1} accounts for the over-counting, since a reactive trajectory just counts once, regardless of the recrossings back and forth the surface ∂A . Hence, the ratio $\nu/\nu^{TST} = k_{AB}/k_{AB}^{TST} = \kappa$ is inversely proportional to the number of recrossings. As a result $0 \leq \kappa \leq 1$, which implies the general inequality

$$k_{AB} \leq k_{AB}^{TST}. \quad (1.53)$$

This inequality is at the basis of variational principles to improve TST, as we will see in sec.1.5.4.

Diffusive motion

The numerical implementation of (1.48) suffers from efficiency problems. As shown in [17], running N_{tr} independent trajectories from λ^* results in a relative error in the transmission coefficient that scales as

$$\frac{\sigma_\kappa}{\kappa} \sim \frac{1}{\kappa \sqrt{N_{tr}}} \quad (1.54)$$

which becomes problematic for small κ .

For diffusive barrier crossings, κ is typically small when the barrier separating the states is flat or the number of recrossings is high. Consider the model of sec. A.4. From Eq. (A.47) we see that the relation $\gamma \propto \kappa^{-1}$ holds for the friction coefficient γ , which means γ proportional to the number of recrossings. And when γ is high, or $|U''(s)|$ small, i.e. flatter barrier, κ decreases. We can even adapt the model assuming a complete flat barrier. Let the potential $U(r)$ as in Fig. 1.8, with h the barrier height and b the barrier width. We can apply formula (A.43) for the mean first passage time, including γ and with no approximation

$$\begin{aligned} (\beta\gamma)^{-1} t^{mfp}(r \rightarrow B) &= \int_r^b dy e^{\beta U(y)} \int_{-\infty}^y dz e^{-\beta U(z)} \\ &= \int_r^b dy e^{\beta U(y)} \int_{-L}^y dz e^{-\beta U(z)} \\ &= \int_r^b dy e^{\beta U(y)} [\theta(y)(L + e^{-\beta h} y) + \theta(-y)(L + y)] \\ &= \theta(-r) \left[L(e^{\beta h} b - r) + \frac{b^2 - r^2}{2} \right] \\ &\quad + \theta(r) \left[L e^{\beta h} (b - r) + \frac{b^2 - r^2}{2} \right] \end{aligned} \quad (1.55)$$

which for $\exp \beta h \gg 1$ has the behavior shown in Fig. 1.9. Again there is a plateau, which allows us to identify

$$k_{AB} = \frac{k_B T}{L\gamma} \frac{e^{-\beta h}}{b} = \frac{D}{b} \frac{e^{-\beta h}}{L} \quad (1.56)$$

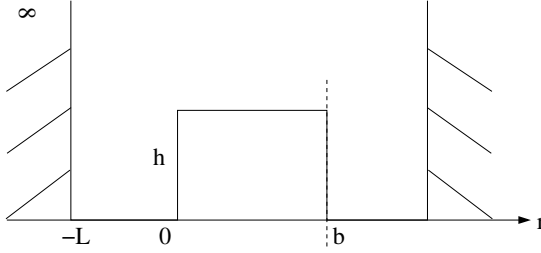
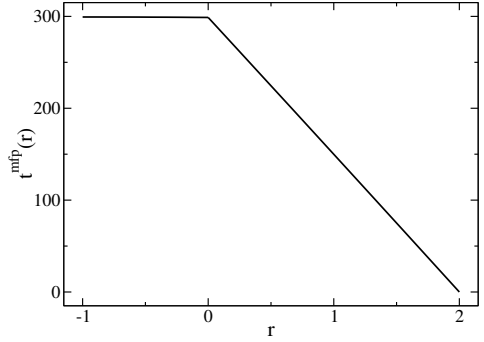


Figure 1.8: Onedimensional diffusive potential. The barrier height and width are h and b respectively. The dashed line indicates the absorbing boundary used for the calculation of the mean first passage time in Eq. (1.55).

Figure 1.9: Mean first passage time for the model of Fig. 1.8, formula (1.55). We use here $\beta h = 5$, $b = 2$, and $L = 1$, which is just a unit of length. The plateau at $2e^5 \simeq 300$ is the inverse rate constant.



In the first passage we have used (1.55) at $r = 0^-$ and neglected $b^2/2$ with respect to $L \exp(\beta h)b$. In the second expression we have used the Einstein relation $k_B T/\gamma = D$ to express the diffusion coefficient D . If we take L as unit of length, the rate constant is basically given by the exponential factor related to the initial barrier step, and the factor $1/b$ to reach the other end of the flat potential.

If we compute the TST rate with $\lambda = r$ and $\lambda^* = 0$, from (1.35) we get

$$k_{AB}^{TST} = R(0^+) \frac{e^{-\beta h}}{L} = \sqrt{\frac{k_B T}{2\pi m}} \frac{e^{-\beta h}}{L} \quad (1.57)$$

so that

$$\kappa = \frac{k_{AB}}{k_{AB}^{TST}} = \frac{1}{R(0^+)} \frac{D}{b} = \frac{\sqrt{2\pi m k_B T}}{b\gamma} \quad (1.58)$$

Eq. (1.58) shows that κ is inversely proportional to the length of the barrier and the number of recrossings. For diffusive systems κ becomes small and according to (1.54) the relative error becomes large. For instance for a transmission coefficient of 0.1 we would need to generate about 10^4 trajectories in order to get a 10% relative error. Finally, in Eq. (1.58) we also note that the unnormalized transmission coefficient $\kappa R(0^+) = k_{AB} * L / \exp(-\beta h)$ (see Eq. (1.48)) is equal to D/b , the ratio of the diffusion constant to the barrier length.

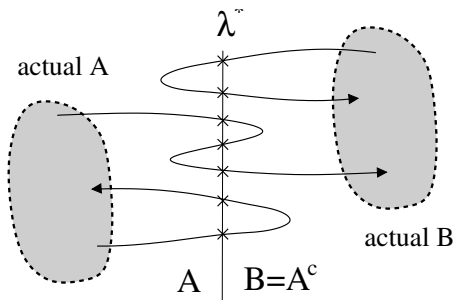


Figure 1.10: Illustration of the difference in counting in the transmission coefficient Eqs. (1.48), (1.59), and (1.65). For simplicity, assume that the system consists of three kinds of possible trajectories, as shown by this figure, that cross the dividing surface with the same speed $\dot{\lambda}$ orthogonal to the surface. To the seven phase points on the surface (from top to bottom) the numerator of Eq. (1.48) with t greater than the transition time τ_{trans} will assign the values $[-\dot{\lambda}, \dot{\lambda}, \dot{\lambda}, -\dot{\lambda}, \dot{\lambda}, 0, 0]$, while these are $[0, 0, \dot{\lambda}, -\dot{\lambda}, \dot{\lambda}, 0, 0]$ for Eq. (1.59) and $[0, 0, \dot{\lambda}, 0, 0, 0, 0]$ for Eq. (1.65). The sum of these give the same result $\dot{\lambda}$. Evaluation of Eq. (1.65) in an actual computer algorithm requires the fewest MD steps as only phase points similar to the 3rd and 7th phase points would need the integration until reaching stable state regions. For instance, the fifth crossing point can be assigned zero already as soon as one detects that its backward trajectory recrosses the surface.

1.5.3 Alternative expressions for κ

Improving the BC expression

Because of the bad scaling of the relative error there have been proposed several ways to improve the computation of κ [20]. The problem with expression (1.48) is that the numerator counts trajectories with a positive but also with a negative weight. The latter trajectories leave the surface at $t = 0$ with a negative velocity $\dot{\lambda}(x_0)$, but are eventually found at the B side of the surface after a (few) recrossing(s). However, untrue $B \rightarrow B$ trajectories do not contribute to the rate because the positive and negative terms cancel ¹ (See Fig. 1.10). Similarly, the $A \rightarrow B$ trajectories with multiple λ^* crossings are effectively counted only once [21]. Although Eq. (1.42) gives the correct rate constant, it is rather unsatisfactory to sample only trajectories forward in time not knowing which contribute to the rate and which do not. Therefore, alternative expressions for the rate constant have been proposed taking the past into account. Here, they are referred to as the BC2 [13, 12] expression

$$\kappa^{BC2}(t) = \frac{\langle \dot{\lambda}_0 \theta(\lambda^* - \lambda_{-t}) \theta(\lambda_t - \lambda^*) \rangle_{\lambda_0 = \lambda^*}}{\langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \rangle_{\lambda^*}} \quad (1.59)$$

¹This cancellation might seem to be not apparent if a trajectory recrosses the same surface but with a different velocity. Still, this is the case. The absolute value of the flux of a trajectory is at each intersecting surface the same. A lower crossing velocity $\dot{\lambda}$ is compensated by a higher probability to measure the crossing point as the trajectory spends more time at the surface.

and the positive flux PF [22] expression

$$\kappa^{pf}(t) = \frac{\langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \theta(\lambda_t - \lambda^*) \rangle_{\lambda_0=\lambda^*}}{\langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \rangle_{\lambda_0=\lambda^*}} - \frac{\langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \theta(\lambda_{-t} - \lambda^*) \rangle_{\lambda_0=\lambda^*}}{\langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \rangle_{\lambda_0=\lambda^*}} \quad (1.60)$$

In Eq. (1.59) the Heaviside functions guarantee that only true $A \rightarrow B$ events are counted. Still, the numerator in Eq. (1.59) contains negative terms: those phase points x_0 with a negative velocity $\dot{\lambda}(x_0)$ and with corresponding backward and forward trajectory that ends up in A and B , respectively. Eq. (1.60) counts only positive crossings, but cancellation with a negative term can occur when the backward trajectory also ends up at the B side of the dividing surface. At first sight, Eq. (1.60) seems to overcount $A \rightarrow B$ trajectories with multiple λ^* crossings. However, if one realizes that each $A \rightarrow B$ trajectory has an equivalent trajectory $B \rightarrow A$ by reversing the time, an overall cancellation of positive and negative terms ensures a proper final outcome.

There are also similar expressions by Berne [23, 24] and a relation by Hummer [25] that counts both positive and negative crossings with a positive weight, but only if the corresponding trajectory ends at opposite sides of the surface and with a weight lower than $|\dot{\lambda}|$ if its trajectory has more than just one crossing.

Ruiz-Montero method

Another way to improve the BC expression (1.48) was devised by Ruiz-Montero et al. [17]. Consider again the BC correlation function $C(t)$, Eq. (1.38). As shown in appendix A, the convergence of $\dot{C}(t)$ to the rate k_{AB} does not rely on the TST definition of the stable states, Eqs. (1.40). Using two general complementary characteristic functions h_A, h_B such that $h_A + h_B = 1$ also in the transition region, and applying the general relation (1.41), we can rewrite

$$\dot{C}(t) = \frac{\langle \dot{\lambda}_0 h'_B(\lambda_0) h_B(\lambda_t) \rangle}{\langle h_A \rangle} \quad (1.61)$$

where $h'_B(0) = \partial h_B / \partial \lambda$ and we used the chain rule to differentiate $h_B(\lambda)$. In the TST-BC procedure $h_B(\lambda) = \theta(\lambda - \lambda^*)$ and the expression (1.42) for $\dot{C}(t)$ is derived. However, one can take different forms for the function h_B in order to improve the relative error (1.54). Ruiz-Montero et al. decided to use a function that is linearly increasing in the transition region between A and B . This choice was based on the following consideration. As mentioned in appendix A, Eq. (1.38) can also be derived applying linear response theory on the phenomenological random telegraph equations (1.4). A perturbation proportional to h_A is taken, which changes equilibrium in favor of state A , and the response h_B is derived. Consider then the onedimensional example of the previous subsection: a flat square barrier of length b separating state A for $r < 0$ and state B for $r > b$ (see Fig. 1.8). We showed that the mean first passage time decreases linearly as the starting point varies on the barrier from A to B (see Fig. 1.9). It can be shown that the steady state distribution function $p_{st}(r)$ obeying the stationary Fokker-Planck equation associated to this problem, also follows a linear decreasing

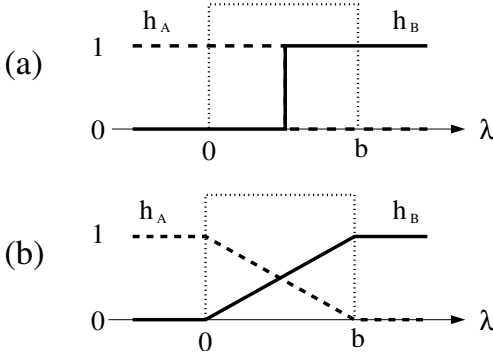


Figure 1.11: Different forms for the characteristic function $h_B(\lambda)$ (solid line) in the case of a potential given by a square barrier of length b (dotted line). The complementary characteristic function $h_A(\lambda) = 1 - h_B(\lambda)$ is also shown as a dashed line. (a) BC choice, $h_B(\lambda)$ is a step function. (b) the choice of Ruiz-Montero et al., $h_B(\lambda)$ is a linear increasing function. As a result it is less noisy than the Heaviside function and allows a faster approach to the steady state.

behavior. The idea of Ruiz-Montero et al. was then to use an initial perturbation $h_A(\lambda)$ of the same form as the steady state distribution, in order to help the slow diffusive approach to the stationary regime. The difference with the BC approach is pictured in Fig. 1.11. As a result, Eq. (1.61) can be rewritten as (compare also Eq. (1.56))

$$k_{AB} = \frac{D}{b} \frac{\langle \delta(\lambda - \lambda^*) \rangle}{\langle h_A \rangle} \quad (1.62)$$

where D is defined from the velocity autocorrelation function as

$$D = \int_0^\infty \langle \dot{\lambda}(0) \dot{\lambda}(t') \rangle_{\lambda_0 = \lambda^*} dt' \quad (1.63)$$

and because of the Green-Kubo relation, can be identified with the diffusion coefficient. The integral in Eq. (1.63) extends to $t = \infty$, but is in practice limited to the decay time of the correlation function, which for a flat barrier is a constant, independent of the barrier length, and shorter than the time to leave the top of the barrier. One can show then [17, 11] that the relative error on the transmission coefficient $\kappa = k_{AB}/k_{AB}^{TST}$ reduces to

$$\frac{\sigma_\kappa}{\kappa} \sim \frac{1}{\sqrt{N_{tr}}} \quad (1.64)$$

and gains a factor κ with respect to Eq. (1.54).

This analysis however could be restrictive to the case of a truly flat barrier. The calculation of κ is equivalent to the calculation of a diffusion coefficient and assumes that $\kappa(t)$ shows a plateau for times while trajectories are still on the barrier. In the case of a general barrier, the same principle can be applied of choosing a perturbation function $h_A(\lambda)$ that is as close as possible to the steady state distribution. It turns out that the best choice is such that $h'_A(\lambda)$ is proportional to the factor $\exp(\beta F(\lambda))$, where $F(\lambda)$ is the free energy profile. The final expression to be used in practice resembles Eq. (1.62), but the velocity autocorrelation function is computed weighing the velocities with such exponential factor. However, if the barrier is not completely flat, Eqs. (1.61) and (1.63) might converge at a time t when the system has left the barrier, and then $h_B(\lambda)$ is the same as normal TST-BC (see Fig. 1.11). This affects

the error analysis in such a way that the scaling of Eq. (1.64) might turn back into that of Eq. (1.54). A more careful analysis, with tests on real systems, should be performed to check this possibility.

Effective positive flux

To this list of alternative ways to compute transmission coefficients, we add our expression, that uses the *effective positive flux formalism*, the idea at the basis of the interface sampling methods of chapters 3 and 4. The formula for the transmission coefficient reads

$$\kappa^{TIS}(t) = \frac{\langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \chi_{AB}^{epf}(x_0) \rangle_{\lambda_0=\lambda^*}}{\langle \dot{\lambda}_0 \theta(\dot{\lambda}_0) \rangle_{\lambda_0=\lambda^*}}, \quad (1.65)$$

where the function χ_{AB}^{epf} counts only reactive trajectories by following the evolution forward in time until the trajectory reaches stable states A , and backward in time until it reaches stable state B , or recrosses the surface λ^* . The definition of $\chi_{AB}^{epf}(x_0)$ is

$$\chi_{AB}^{epf}(x_0) = \begin{cases} 1 & \text{if integrating backward } x_t \text{ hits actual } A \text{ before } \lambda^* \\ & \text{and integrating forward hits actual } B \text{ before actual } A, \\ 0 & \text{otherwise,} \end{cases} \quad (1.66)$$

where x_0 is a phase space point on the surface λ^* , and x_t is the evolution of the trajectory at time t . Since $\chi_{AB}^{epf}(x_0)$ is independent of time and can only assume two values, it is easy to estimate the relative error on κ . For a boolean variable $\sigma_\kappa^2 = \kappa(1 - \kappa) \simeq \kappa$, since κ is small for diffusive cases. Then we obtain

$$\frac{\sigma_\kappa}{\kappa} \sim \frac{1}{\sqrt{\kappa N_{tr}}}. \quad (1.67)$$

Surprisingly, the effective positive flux counting strategy is not so common. To our knowledge only another slightly different expressions of a transmission coefficient based on the effective positive flux has been proposed in Ref. [26]. In all other expressions found in the literature the counting of recrossings is not avoided, but the final rate constant follows through cancellation of many negative and positive terms.

In sec. 6.2 we will give an expression for $\chi_{AB}^{epf}(x_0)$ and address again the problem of computing the transmission coefficient. We will try to compare on qualitative grounds some of the expressions encountered in this section, adding the contribution of the Interface Sampling methods to be presented in the following chapters.

1.5.4 Variational TST-BC

The variational inequality (1.53) tells us that the best reaction coordinate is the one that minimizes k_{AB}^{TST} . This is also a definition for the best dividing surface [9, 27]. Truhlar and coworkers have formulated many different versions of variational TST, where the full calculation of the rate constant is repeated for different choices of

the transition state [28]. Since any recrossing of the dividing surface leads to an overestimate in the rate constant, one might assume that minimizing the number of recrossings will give the lowest rate estimate. Makarov and Metiu have showed that this may not be the case [29]. A more rigorous formulation is in terms of the free energy, as first pointed out by Chandler [12], who making use of expression (1.35), transposed (1.53) into: the best reaction coordinate is the one that maximizes the free energy F . This reasoning however neglects the first term (1.37).²

It is shown in appendix B that an alternative expression for the TST rate is

$$k_{AB}^{TST} = \frac{1}{\langle h_A \rangle} \sqrt{\frac{k_B T}{2\pi m}} \int_{\lambda(q)=\lambda^*} d\sigma(r) \rho(r) \equiv \sqrt{\frac{k_B T}{2\pi m}} \frac{e^{-\beta G(\lambda^*)}}{\langle h_A \rangle} \quad (1.68)$$

where the integral is on the surface ∂A , and $d\sigma(r)$ is the surface element. We assume all particle masses equal to m . The free energy G so defined has the important property of being geometrically invariant. Under reparametrization of A and its surface, it does not change. This is not the case for F . The two are related by

$$e^{-\beta G(\lambda^*)} = \langle |\nabla \lambda| \rangle_{\lambda=\lambda^*} e^{-\beta F(\lambda^*)} \quad (1.69)$$

This expression is the most appropriate free energy definition in the context of TST. Now the variational principle can be applied to (1.68), and it reads: the best reaction coordinate is the one that maximizes the free energy G . In principle, variational equations can be written for $\lambda(r)$ [10], but they are too formidable a task to be solved. Attempts have been made restricting the class of functions $\lambda(r)$ to simple ones, such as hyperplanes [30]. For a given parametrization $\lambda(r)$, the best value for λ^* is the maximum of $G(\lambda^*)$. This recovers and precises the qualitative reasoning of sec. 1.4.3 and sec. 1.5.1, where we suggested that the best choice of λ^* is the maximum of F . However, only when $|\lambda(r)|$ is constant (e.g. linear parametrization), will F and G be the same. In general, a λ^* that maximizes G could, for example, be a bit to the right of the maximum of F . Even though in practice the difference will not be much, it is in principle wrong to work only in terms of the free energy F .

1.6 Other methods

Before switching to transition path sampling, we review some alternative approaches to rare events that can be found in literature.

An important distinction is between methods that focus on the study of the potential energy surface and those that do not. The first approach has its natural development in the context of TST. Transition pathways are defined as *minimum energy paths* (MEP), and a theorem by Murrell and Laidler identifies transition states as saddles of index one [31]. Therefore the proper reaction coordinates are along the direction of the unstable modes of relevant saddles down to the minima, which identify the stable states. At each saddle point the TST dividing surface is taken to be the hyperplane going through the saddle point with the normal given by the displacement vector of the unstable mode. TST is then often applied within the harmonic

²and the denominator in (1.35), which can however be considered roughly constant and equal to the measure of stable state A .

approximation, where the partition function of the system in the dividing surface and in the initial state is approximated by a harmonic partition function [32] (see also sec. 2.4.1).

The MEP approach has been used several times and refined up to recent publications. However, in complex systems at finite temperature, concepts like the minimum energy path or the lowest saddle point are not very useful. The potential energy surface is not just a smooth landscape of two valleys separated by a passage, but is exponentially populated with saddles and local minima [33, 34]. The reaction is rather described by an ensemble of paths. Similarly, one cannot speak of a particular transition state but only of an ensemble of transition states [35].

With this distinction in mind we give a (naturally incomplete) list of methods, distinguishing them by their implementation approach.

1.6.1 Open-ended methods

This class of methods focuses on escaping the initial state without making assumptions on the final state. This can be achieved by, for instance, artificially increasing the frequency of the rare event in a controlled way. The methods of Voter and collaborators follow this approach: hyperdynamics [36, 37] aims at lowering the energy difference between the top of the barrier and the initial basin, the parallel replica method [38] exploits the power of parallel processing to extend the molecular simulation time, and temperature-accelerated dynamics [39, 40] speeds up the event by raising the temperature. The idea of driving energy into the system to escape the basin of the energy minimum in which the system is initially prepared is also at the basis of conformational flooding [41], the Laio-Parrinello method [42, 43], and the enhanced sampling of a given reaction coordinate [44]. Another possible route is to coarse-grain the molecular dynamics on the fly and explore the resulting free-energy landscape [45]. Several methods are devoted to the exploration of the full potential energy surface through all its minima and saddle points. Examples are eigenvector following [46, 47], the activation-relaxation technique of Barkema and Mousseau, [48], the dimer method of Henkelmann and Jónsson [49], the kinetic Monte Carlo (MC) approach [50, 51, 52], and the discrete path sampling of Wales [53, 34].

Finally we also add to this class the clustering algorithms of Schütte et al. [54, 55]. In sec. A.1 we gave a glimpse of the significance and capabilities of the spectral analysis of the Fokker-Planck operator. Based on its rigorous mathematical theory, Schütte et al. derived algorithms that regroup sets of configurations according to the eigenvalues and eigenfunctions of a properly constructed operator, called the proximity matrix.

1.6.2 Two-ended methods

When the initial and final state are known it is possible to generate paths connecting the two in the form of a discretized chain of states. This is the basis of a second class, the so-called two point boundary methods. One option is to find the minimal energy path on the potential energy surface, as in the Nudged Elastic Band method of Jónsson and collaborators [56, 57, 58, 59] and in the zero-temperature string method of E et al. [60]. When the system is too complex the latter replaces the concept of MEP with that of minimum *free* energy path. the finite-temperature string method [61, 62] exploits the fact that in some thermally averaged potential the small features of the

energy below the thermal scale can be smoothed out. Another possibility is to find a true dynamical path by minimizing a suitably chosen action [63], or to use modified stochastic equations of motion that guide the system from the initial to the final state [64]. Similar techniques by Elber and Olender [65,66,67] and Doniach et al [68] sample discretized stochastic pathways based on the Onsager-Machlup action. We also mention the topological method of Tănase-Nicola and Kurchan [69,70] in which they suggest to use TPS in combination with saddle point searching vector walkers.

Usually these two-ended methods require an initial path to initiate the algorithm. Some simple interpolation schemes are used or it is often taken from a path created by an open-ended method. The milestoning [71] method of Faradjian and Elber assumes the existence of a suitable path and builds a chain of states to overcome diffusive problems. Milestoning is quite similar to Partial Path Transition Path Sampling, which is the subject of chapter 4. We defer to then a comparative discussion of the two methods.

1.7 Summary

We pointed out the problem of rare events in sec. 1.1. After reading the intermediate sections, one might be inclined to identify it with the calculation of the rate constant. All in all, the quest for this number has inspired most of the numerical and theoretical framework that we have tried to outline in this chapter. However, in real systems, no method computes a rate constant, simply because such a definition does not exist. Methods can compute a mean residence time (TST), a correlation function (BC), or, as we will show in the following chapter, a mean first passage time (TIS). When computing a rate constant, one is in fact trying to construct the best random telegraph approximation (i.e. the best jump probability W_{AB}) that describes the transition process.

In sec. 1.1 we also said that the study of rare events is the study of the transition mechanism: finding out how the process actually takes place, including the identification of the transition states. At the basis of the whole TST-BC procedure lies the first step in sec. 1.5.1: the choice of a reaction coordinate. The traditional belief is just that a proper choice of $\lambda(r)$, based on physical and/or chemical intuition, will do fine. The reaction coordinate is supposed to describe the advance of the transition from A to B . For TST, the dynamical information is retrieved from the free energy F or G (see sec. 1.5.4) as function of this reaction coordinate. In the BC procedure, the dynamical calculation of the transmission coefficient corrects TST and converges to the correct rate. We might have the final rate constant, but the mechanistic picture of the transition event is biased by the *a priori* choice of the reaction coordinate. Even if the RC intuitively looks fine, we could be wrong, or simply not just right enough. We might be disregarding some important aspects of the event. In the study of nucleation in chapter 7 this is the case. We need therefore a procedure that tells us which RC to use. We cannot expect such information from the Bennett-Chandler procedure, which simply assumes we are looking at the correct reaction way. Driven by this need new methods have been created such as Transition Path Sampling, and its improvement Interface Sampling. We will see in the following chapters how they address this problem.

2

Transition Path Sampling

omnes viae Romam ducunt
(all the paths lead to Rome)

Ancient saying

Transition Path Sampling is essentially a Monte Carlo random walk in the path space of the transition trajectories. From the ensemble of these paths one can get all the information requested in sec 1.1. The method inspired the development of Interface Sampling, which will be discussed in the following chapters. Interface Sampling uses a different approach to the computation of the rate constant, but the study of mechanism and transition states as well as the computational tools follow directly from TPS. We review TPS here following [35], and in sec 2.4 we add an application to the rearrangement of a seven particle Lennard-Jones cluster in 3 dimensions.

2.1 The transition path ensemble

Again, the framework is that of Fig. 1.1: two states A and B are stable on a timescale much longer than the duration of the transitions between them. In order to define an ensemble of paths, we would like to associate to every trajectory of particular duration a probability density functional. In this way, we restrict it then only to the paths of importance, the transitions.

Let us reconsider the definitions of sec. 1.3. Denote with x the set of variables describing the state of the system. The actual definition depends on the choice of the dynamics. For Hamiltonian or Langevin dynamics $x = \{r, p\}$ is the set of coordinates and momenta, for stochastic dynamics in the high friction limit it might consist of configurations only, for a lattice system and Monte Carlo dynamics x denotes the state of each lattice site. We will present a formalism that is valid for all kinds of dynamics, but for most applications, in the rest of this thesis, we will mainly consider deterministic motion. Instead of the Langevin dynamics (1.22) we assume that the evolution is described by a set of ordinary homogeneous differential equations

$$\dot{x} = \Gamma(x). \quad (2.1)$$

The solution x_t at time t is uniquely determined by the initial condition x_0 at time 0 and can be written as

$$x_t = f_t(x_0) \quad (2.2)$$

which defines the propagator f_t . In the standard Hamiltonian picture (2.1) is derived from a Hamiltonian $\mathcal{H}(x) = \mathcal{H}(r, p)$ according to $\Gamma(r, p) = \{\partial\mathcal{H}/\partial p, -\partial\mathcal{H}/\partial r\}$. In the general form, Eq. (2.1) includes also dynamics based on the extended Lagrangian of Car and Parrinello [72], or various thermostatted systems [73].

We remind that deterministic motion does not mean a deterministic system, because of the ensemble of initial conditions. We assume the system admits an equilibrium distribution $\rho(x)$, which does not necessarily have to be canonical. The other definitions of sec. 1.3 still hold in the proper phase space x . In particular, the characteristic function h_Ω is now defined on a general subset of the full phase space.

2.1.1 Path probability

We denote a trajectory of length \mathcal{T} by a discretized sequence of states

$$x(\mathcal{T}) \equiv \{x_0, x_{\Delta t}, x_{2\Delta t}, \dots, x_{\mathcal{T}}\} \quad (2.3)$$

where consecutive states are separated by a small time increment Δt . The sequence is called a *path* and is made up of $L = \mathcal{T}/\Delta t + 1$ states, called *time slices*. The pathlength \mathcal{T} is a fixed time and chosen a priori. Even though the time evolution of the system is continuous, such a discretization is convenient for numerical implementation and allows us to use probabilities.

The statistical weight or probability $\mathcal{P}[x(\mathcal{T})]$, of a path $x(\mathcal{T})$ depends on the distribution of initial conditions and on the propagation rules of the dynamics. We assume the dynamics is a Markov process to which is attached the probability (see Eq. A.2)

$$\begin{aligned} p(x_t \rightarrow x_{t+\Delta t}) &\equiv p(x_{t+\Delta t}, t + \Delta t | x_t, t) \\ &= \text{Probability of being in } x_{t+\Delta t} \text{ at time } t + \Delta t \\ &\quad \text{provided that you were in } x_t \text{ at time } t \end{aligned} \quad (2.4)$$

The dynamical path probability is expressed as a product of short-time probabilities

$$\mathcal{P}[x(\mathcal{T})] = \rho(x_0) \prod_{i=0}^{\mathcal{T}/\Delta t - 1} p(x_{i\Delta t} \rightarrow x_{(i+1)\Delta t}) \quad (2.5)$$

where $\rho(x_0)$ is the distribution of initial conditions, mostly an equilibrium distribution but can also be a non-equilibrium one. The *transition path ensemble* is then defined as the subset of trajectories that connect states A and B :

$$\mathcal{P}_{AB}[x(\mathcal{T})] \equiv Z_{AB}^{-1}(\mathcal{T}) h_A(x_0) \mathcal{P}[x(\mathcal{T})] h_B(x_{\mathcal{T}}) \quad (2.6a)$$

$$Z_{AB}(\mathcal{T}) \equiv \int \mathcal{D}x(\mathcal{T}) h_A(x_0) \mathcal{P}[x(\mathcal{T})] h_B(x_{\mathcal{T}}) \quad (2.6b)$$

where in the normalizing factor the path-integral notation $\int \mathcal{D}x(\mathcal{T})$ indicates a summation over all pathways $x(\mathcal{T})$. In our discretized case it corresponds to an integration

over states at each time slice of the path. The constraint $h_A(x_0)h_B(x_T)$ ensures that paths that do not connect A and B have zero weight in this ensemble. We remark that the definition (2.6) only contains *true* dynamical trajectories, without any bias of unphysical forces or constraints.

In the case of deterministic dynamics

$$p(x_t \rightarrow x_{t+\Delta t}) = \delta[x_{t+\Delta t} - f_{\Delta t}(x_t)] \quad (2.7)$$

and using (2.5), Eq. (2.6) becomes

$$\mathcal{P}_{AB}[x(\mathcal{T})] = Z_{AB}^{-1}(\mathcal{T}) \rho(x_0) h_A(x_0) h_B(x_T) \prod_{i=0}^{\mathcal{T}/\Delta t - 1} \delta[x_{(i+1)\Delta t} - f_t(x_{i\Delta t})] \quad (2.8a)$$

$$Z_{AB}(\mathcal{T}) = \int dx_0 \rho(x_0) h_A(x_0) h_B(x_T) \quad (2.8b)$$

where integrations over the states along the path have been carried out at all times except zero.

2.1.2 Order parameters

The definition of a state by $h_A(x)$ or $h_B(x)$ relies in practice on a parametrization such as (1.31). A high-dimensional set in the full phase space is reduced to a onedimensional range of a real-valued function $\lambda(x)$, called the *order parameter*.

At variance with the definition of reaction coordinate, sec. 1.5.1, the order parameter does not have to characterize the progress of the transition, but we only require that it properly characterizes the stable states A and B . The phase space regions defined by the order parameter should encompass the basins of attraction, but also distinguish them, see Fig. 2.1. Basins of attraction are the distinct sets of points from which trajectories relax to the corresponding stable state. In practice, definitions are based on physical intuition, similar to the choice of the reaction coordinate, and adjusted by trial and error. Nevertheless, the situation depicted in Fig. 2.1 can be difficult to detect.

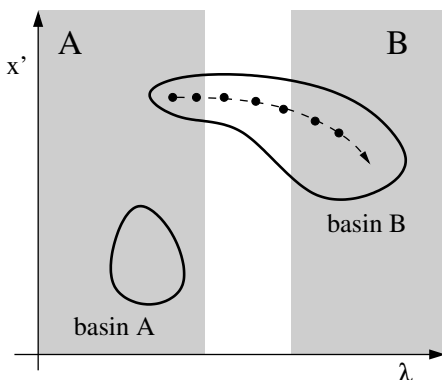


Figure 2.1: Phase space projected along an order parameter λ and orthogonal coordinates, here called x' . Basins of attraction of the stable states are portrayed, and the two λ intervals defining the A and B are showed in gray. Even though they accomodate most of the fluctuations, there is overlap of the definition of A with the basin of attraction of B that might lead to wrongly identify as transitions paths like the one in figure.

2.2 Sampling the path ensemble

We want to generate trajectories $x(\mathcal{T})$ with a frequency proportional to their weight (2.6). We apply the Monte Carlo idea of constructing a random walk stochastic process corresponding to the desired equilibrium distribution.

Assume we have a path $x^{(o)}(\mathcal{T})$ belonging to the path ensemble, i.e. a transition. Suppose we generate from this another path $x^{(n)}(\mathcal{T})$ ¹ with probability $P_{gen}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})]$. We next accept the newly generated path with probability $P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})]$. A simple way to construct an appropriate acceptance probability is based on detailed balance of moves in trajectory space. Since the overall conditional probability for this Markov process is $P_{gen}P_{acc}$, we impose (see also Eq. (A.10))

$$\begin{aligned} \mathcal{P}_{AB}[x^{(o)}(\mathcal{T})]P_{gen}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})]P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = \\ \mathcal{P}_{AB}[x^{(n)}(\mathcal{T})]P_{gen}[x^{(n)}(\mathcal{T}) \rightarrow x^{(o)}(\mathcal{T})]P_{acc}[x^{(n)}(\mathcal{T}) \rightarrow x^{(o)}(\mathcal{T})] \end{aligned} \quad (2.9)$$

Since $x^{(o)}(\mathcal{T})$ belongs to the ensemble, $h_A(x_0^{(o)}) = h_B(x_{\mathcal{T}}^{(o)}) = 1$. Using (2.6) we get for the ratio of the acceptance probabilities

$$\begin{aligned} \frac{P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})]}{P_{acc}[x^{(n)}(\mathcal{T}) \rightarrow x^{(o)}(\mathcal{T})]} &= h_A(x_0^{(n)})h_B(x_{\mathcal{T}}^{(n)}) \\ &\frac{\mathcal{P}[x^{(n)}(\mathcal{T})]}{\mathcal{P}[x^{(o)}(\mathcal{T})]} \frac{P_{gen}[x^{(n)}(\mathcal{T}) \rightarrow x^{(o)}(\mathcal{T})]}{P_{gen}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})]} \end{aligned} \quad (2.10)$$

The \mathcal{P} probabilities come from (2.5), P_{gen} depend on the algorithm that generates a new path from an old one, see following sections. The condition (2.10) can be satisfied using the Metropolis rule [74]

$$\begin{aligned} P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] &= h_A(x_0^{(n)})h_B(x_{\mathcal{T}}^{(n)}) \\ &\min \left[1, \frac{\mathcal{P}[x^{(n)}(\mathcal{T})]}{\mathcal{P}[x^{(o)}(\mathcal{T})]} \frac{P_{gen}[x^{(n)}(\mathcal{T}) \rightarrow x^{(o)}(\mathcal{T})]}{P_{gen}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})]} \right] \end{aligned} \quad (2.11)$$

Accordingly only paths that connect A and B are accepted.

Resuming, a path sampling simulation is the realization of two Markov processes. One is the dynamics, and takes place in the phase space x of the system. On top of that the path sampling is built, a Markov process in the transition path space $x(\mathcal{T})$. We now move to see how this MC walk in path space is implemented in practice.

2.2.1 Shooting move

This powerful and efficient algorithm [75] is the main tool for the realization of the path sampling and is essential in Interface Sampling methods, see sec. 3.2.2.

The algorithm is as follows. A time slice $x_{t'}^{(o)}$, $0 \leq t' \leq \mathcal{T}$, of the old path $x^{(o)}(\mathcal{T})$ is selected at random. The state is modified in some way, for example by simply changing the momenta. Then a new path is created evolving (shooting) backward

¹Here superscripts stand for 'old' and 'new'.

30 2. Transition Path Sampling

and forward in time from the modified state $x_{t'}^{(n)}$ until $t = 0$ and $t = T$ respectively. The new path $x_{t'}^{(n)}$ is accepted with a probability (2.11), which implies that the path is rejected if it does not connect A and B . The procedure is schematically depicted in Fig. 2.2 (a).

We can now expand (2.11). The generation probability P_{gen} is written as

$$P_{gen}[x^{(o)}(T) \rightarrow x^{(n)}(T)] = p_{gen}[x_{t'}^{(o)} \rightarrow x_{t'}^{(n)}] \prod_{i=t'/\Delta t}^{T/\Delta t-1} p\left(x_{i\Delta t}^{(n)} \rightarrow x_{(i+1)\Delta t}^{(n)}\right) \prod_{i=1}^{t'/\Delta t} \bar{p}\left(x_{i\Delta t}^{(n)} \rightarrow x_{(i-1)\Delta t}^{(n)}\right) \quad (2.12)$$

where p, \bar{p} are the small time step probabilities describing the evolution forward and backward in time, and $p_{gen}[x_{t'}^{(o)} \rightarrow x_{t'}^{(n)}]$ is the probability to obtain state $x_{t'}^{(n)}$ by modification of state $x_{t'}^{(o)}$. Here p_{gen} contains the probability of selecting timeslice t' , equal to $\Delta t/T$ in the case of uniform choice. The backward small time step probability \bar{p} also deserves some explanation. Given definition (2.4) for the forward small time step probability, \bar{p} is defined as

$$\bar{p}(x \rightarrow y) = p(\bar{x} \rightarrow \bar{y}) \quad (2.13)$$

where \bar{x} denotes the transformation of x under time-reversal. Namely, if $x = \{r, p\}$, then $\bar{x} = \{r, -p\}$. We remark that in practice not all integration schemes have a simple way of reversing the evolution, such as inversion of the momenta and forward integration in time. For this reason time-reversible algorithms as the Verlet algorithm and in general those based on the Trotter factorization of the Liouville propagator [76], are definitively preferable.

Substituting (2.12) into (2.11) we get

$$P_{acc}[x^{(o)}(T) \rightarrow x^{(n)}(T)] = h_A(x_0^{(n)})h_B(x_T^{(n)}) \min \left[1, \frac{\rho(x_0^{(n)})}{\rho(x_0^{(o)})} \frac{p_{gen}(x_{t'}^{(n)} \rightarrow x_{t'}^{(o)})}{p_{gen}(x_{t'}^{(o)} \rightarrow x_{t'}^{(n)})} \prod_{i=0}^{t'/\Delta t-1} \frac{p(x_{i\Delta t}^{(n)} \rightarrow x_{(i+1)\Delta t}^{(n)})}{\bar{p}(x_{(i+1)\Delta t}^{(n)} \rightarrow x_{i\Delta t}^{(n)})} \frac{\bar{p}(x_{(i+1)\Delta t}^{(o)} \rightarrow x_{i\Delta t}^{(o)})}{p(x_{i\Delta t}^{(o)} \rightarrow x_{(i+1)\Delta t}^{(o)})} \right] \quad (2.14)$$

where factors have cancelled out because the trial trajectory was generated using the same propagation rules of the underlying dynamics.

This formula is a general expression for the shooting algorithm. In the following we will make assumptions in order to simplify it further and eventually concentrate on deterministic dynamics. For a more general implementation in various cases, e.g. stochastic dynamics, we refer to [35].

dynamics	x	$\mathcal{H}'(x)$
NVE	r, p	$\mathcal{H}(r, p)$
NVT [†]	r, p, η, p_η	$\mathcal{H}(r, p) + p_\eta^2/2Q + dNk_B T \eta$
NPT [*]	$r, p, \eta, p_\eta, V, p_\epsilon$	$\mathcal{H}(r, p) + \sum_{k=1}^M p_{\eta_k}^2/2Q_k + (dN + 1)k_B T \eta_1$ $+ \sum_{k=2}^M \eta_k + P_{ext}V + p_\epsilon^2/2W$
NPH [‡]	$\boldsymbol{\rho}, \boldsymbol{\pi}, V, p_V$	$\mathcal{H}(\boldsymbol{\rho}, \boldsymbol{\pi}) + P_{ext}V + p_V^2/2W$

Table 2.1: Some generalized dynamics, corresponding phase space x and conserved generalized Hamiltonian \mathcal{H}' . Here \mathcal{H} is the standard Hamiltonian, kinetic plus potential energy.

[†] Nosé-Hoover algorithm, look at [73]. ^{*} Martyna-Tobias-Klein algorithm, look at [73].

[‡] NPH-Andersen algorithm, look at [77], $\boldsymbol{\rho}, \boldsymbol{\pi}$ are the scaled coordinates and momenta, see also sec. H.

Assumption I: microscopic reversibility

It is a very general consequence of microscopic reversibility that the dynamics conserve a stationary distribution $\rho_{st}(x)$, i.e. the small time step probabilities satisfy

$$\frac{p(x \rightarrow y)}{\bar{p}(y \rightarrow x)} = \frac{\rho_{st}(y)}{\rho_{st}(x)} \quad (2.15)$$

In case of deterministic generalized hamiltonian dynamics the stationary distribution is $\rho_{st}(x) = \delta(\mathcal{H}'(x) - E)$, with \mathcal{H}' the generalized conserved energy. In table 2.1 we summarize the conserved variable for some well known deterministic MD algorithms. Other possible conserved quantities $\Lambda_k(x)$, $k = 1 \dots n_c$, such as the total linear or angular momentum, can also enter the stationary distribution, which takes the form

$$\rho_{st}(x) = \delta(\mathcal{H}'(x) - E) \prod_1^{n_c} \delta(\Lambda_k(x) - C_k). \quad (2.16)$$

where E and C_k are constants.

Another case of interest is the Andersen thermostat [77] for constant temperature MD, which also conserves a canonical stationary distribution, but is not related to a conserved generalized Hamiltonian.

Assumption II: symmetric slice modification

If one assumes, besides microscopic reversibility, a symmetric generation probability

$$p_{gen} \left(x_{t'}^{(o)} \rightarrow x_{t'}^{(n)} \right) = p_{gen} \left(x_{t'}^{(n)} \rightarrow x_{t'}^{(o)} \right) \quad (2.17)$$

the acceptance rule (2.14) simplifies further into

$$P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = h_A(x_0^{(n)})h_B(x_{\mathcal{T}}^{(n)}) \min \left[1, \frac{\rho \left(x_0^{(n)} \right) \rho_{st} \left(x_0^{(o)} \right) \rho_{st} \left(x_{t'}^{(n)} \right)}{\rho \left(x_0^{(o)} \right) \rho_{st} \left(x_0^{(n)} \right) \rho_{st} \left(x_{t'}^{(o)} \right)} \right] \quad (2.18)$$

32 2. Transition Path Sampling

where the fraction in the min term depends only on time 0 and t' . Before considering two specific cases of interest we make some general remarks.

Devising a symmetrical algorithm that modifies a state $x_{t'}^{(o)}$ into $x_{t'}^{(n)}$ requires some care. In particular, the modification has to be consistent with the initial conditions distribution $\rho(x)$, which might contain additional constraints. In the case of linear constraints, such as null total momentum and angular momentum, a common procedure is the following. In this procedure we assume that only momenta are changed. Consider the phase space point $x_{t'}^{(o)} = \{r_{t'}^{(o)}, p_{t'}^{(o)}, \xi_{t'}^{(o)}\}$, where $\xi_{t'}^{(o)}$ represents the additional variables of generalized dynamics, see table 2.1

1. Select $n = dN$ momentum displacements δp from a onedimensional Gaussian distribution and add them to $p_{t'}^{(o)}$, so that $p' = p_{t'}^{(o)} + \delta p$.
2. Impose the linear constraints using the procedure of [78]. The m constraint equations define a $n - m$ linear manifold in \mathbb{R}^n through its m dimensional normal bundle. In order to project the new momenta p' onto this manifold, first construct an orthonormal basis for the normal bundle through a Gram-Schmidt procedure, and then subtract the components along this basis set. If considering only vanishing total linear momentum, this step is equivalent to subtract $\sum_i \mathbf{p}_i'/N$ from all single particle momenta.
3. When necessary, rescale the momenta by a constant factor to obtain the total desired energy E . If the constraints are of vanishing total momentum and the center of mass is in the origin, this step leaves the constraints unchanged.

For further details we refer again to [35].

Two cases of interest

First, suppose that the distribution of initial points is the same as the stationary distribution, i.e. $\rho(x) = \rho_{st}(x)$, then Eq. (2.18) simply reduces to

$$P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = h_A[x_0^{(n)}]h_B[x_{\mathcal{T}}^{(n)}] \min \left[1, \frac{\rho(x_{t'}^{(n)})}{\rho(x_{t'}^{(o)})} \right]. \quad (2.19)$$

In the case the stationary distribution is as in (2.16), and if the generating algorithm is consistent with this distribution, the acceptance criterion becomes

$$P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = h_A[x_0^{(n)}]h_B[x_{\mathcal{T}}^{(n)}] \quad (2.20)$$

The path sampling procedure is now as follows. Choose a slice, change the phase point, rescale the momenta to meet the conditions in (2.16) and integrate the evolution backward and forward in time. If the new path is reactive (connects A and B) accept, otherwise reject it. Note that in practice the generalized energy E in (2.16) is not perfectly conserved by the integration algorithm, and to satisfy detailed balance, one should rescale to the energy of the shooting time slice t' . This, however, can lead to a drift in the energy and in practice it can be better to rescale to a fixed

quantity. The error is assumed negligible. Most important, (2.20) holds even when the dynamics is not Hamiltonian. We will use it for NVE simulations, where both the dynamics and the distribution of initial conditions are microcanonical, but also for the simulation of a nucleation process at constant pressure-constant enthalpy. In the case of NVT-Andersen algorithm, where (2.16) does not hold, the path sampling procedure is similar. The difference is that momenta are not rescaled, but accepted or rejected on the basis of the Metropolis rule, as given by Eq. (2.19).

Second, suppose that the distribution of initial conditions is not equal to the stationary distribution, i.e. $\rho(x) \neq \rho_{st}(x)$. Consider again (2.18). To the dynamical Eq. (2.2) is associated the transformation of a volume element

$$dx_t = J(x_t; x_0)dx_0, \quad (2.21)$$

where $J(x_t; x_0) \equiv |\partial f_t(x_0)/\partial x_0| = |\partial x_t/\partial x_0|$ is the Jacobian of the transformation. The dynamical system (2.1), even if not preserving the probability measure (2.21), nor the distribution $\rho_{st}(x)$, does preserve the probability itself:

$$\rho_{st}(x_t)dx_t = \rho_{st}(x_0)dx_0 \quad (2.22)$$

and as a consequence

$$\rho_{st}(x_t) = \rho_{st}(x_0)J(x_t; x_0)^{-1} \quad (2.23)$$

Inserting (2.23) into (2.18) gives

$$\begin{aligned} P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] &= h_A(x_0^{(n)})h_B(x_{\mathcal{T}}^{(n)}) \\ &\min \left[1, \frac{\rho(x_0^{(n)})}{\rho(x_0^{(o)})} \frac{|\partial x_{t'}^{(o)}/\partial x_0^{(o)}|}{|\partial x_{t'}^{(n)}/\partial x_0^{(n)}|} \right] \end{aligned} \quad (2.24)$$

In the case $\rho(x) = \rho_{st}(x)$ this equation is equivalent to (2.19), but if we express the dependence on the slices at time 0, the Jacobian appears. In Eq. (2.20), however, which is most convenient in practice, there is no Jacobian involved.

Suppose now that the dynamics is area-preserving, i.e. $J = 1$, such as in Hamiltonian systems. Then, Eq. (2.24) becomes

$$P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = h_A[x_0^{(n)}]h_B[x_{\mathcal{T}}^{(n)}] \min \left[1, \frac{\rho(x_0^{(n)})}{\rho(x_0^{(o)})} \right] \quad (2.25)$$

which depends only on the relative weights of initial conditions. This means that in principle the Metropolis acceptance rule should be applied to the initial slice x_0 and not to the shooting timeslice, cf (2.19). One has to integrate first all the way to time 0. However, simplification may occur, for example in the typical case of NVE dynamics with a *canonical* distribution $\rho(x_0)$. When $\rho(x_0) \propto \exp(-\beta E)$ and the energy E is constant on the path, the criterion can be applied to any slice, in particular the one at time t' . In reality this is slightly wrong, because the energy is not perfectly conserved along the path by the integration scheme, but in practice it is usually assumed that the error is negligible.

Efficiency

The shooting move creates new paths for the random walk by a perturbation of the old path. The acceptance of the move can be controlled by changing the magnitude of the phase point displacement δx (δp in the case of the algorithms proposed above). If $\delta x = 0$ one simply reproduces the old path² and the acceptance is automatically 1. A low δx then basically results in sampling the same path again. On the other hand if δx is very large one samples very different paths, but most of these paths will be rejected because the reactive trajectories $A \rightarrow B$ are unstable, in the sense that they can easily change into trajectories of type $A \rightarrow A$ or $B \rightarrow B$. An efficient sampling produces decorrelated paths in the smallest number of Monte Carlo cycles. It was shown in [79] that, just as in conventional Monte Carlo simulation, the most efficient sampling is given by an acceptance around 40%.

This holds for reactive trajectories of fixed length, but we can expect it to be valid also for TIS trajectories of fluctuating length. When the system is diffusive, however, the shooting move runs into trouble. Because of the Lyapunov instability of the underlying dynamics even the smallest δx might lead to completely different and unreactive trajectories. In this case the acceptance plateaus to a fixed value and the algorithm becomes intrinsically inefficient. For this reason we developed PPTIS, which will be discussed in chapter 4.

2.2.2 Shifting move

In TPS paths start in A and end in B with fixed length \mathcal{T} . Part of the path is in the stable states, and part is in the transition region. Therefore translating the path in time might create a new path that still connects the states. This leads to the idea of the shifting move, depicted in Fig. 2.2 (b). A segment of length $\Delta\mathcal{T}$ is deleted from the beginning (end) of the path and a corresponding segment is grown at the end (beginning) by applying the rules of the dynamics. If $\delta\mathcal{T} > 0$ the result is a forward translation in time and if $\Delta\mathcal{T} < 0$ it is backward in time. The generation probabilities for a forward and backward shift are written, by considering the newly created portions of path,

$$P_{gen}^f[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = \prod_{i=(\mathcal{T}-\Delta\mathcal{T})/\Delta t}^{\mathcal{T}/\Delta t-1} p\left(x_{i\Delta t}^{(n)} \rightarrow x_{(i+1)\Delta t}^{(n)}\right) \quad (2.26a)$$

$$P_{gen}^b[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = \prod_{i=1}^{\Delta\mathcal{T}/\Delta t} \bar{p}\left(x_{i\Delta t}^{(n)} \rightarrow x_{(i+1)\Delta t}^{(n)}\right) \quad (2.26b)$$

while the common portions of the old and the new path are identical. We impose the detailed balance condition (2.9) with the forward probabilities for the move $x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})$ and the backward ones for the reversed move $x^{(n)}(\mathcal{T}) \rightarrow x^{(o)}(\mathcal{T})$. Assuming a symmetrical generation of $\Delta\mathcal{T}$ for forward and backward shift, and assuming microscopic reversibility (2.15), after cancellation of common terms the acceptance

²because of round-off errors a different path might be created even if $\delta x = 0$

criterion for shifting is

$$P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = h_A(x_0^{(n)})h_B(x_{\mathcal{T}}^{(n)}) \quad (2.27)$$

$$\min \left[1, \frac{\rho(x_0^{(n)})}{\rho(x_0^{(o)})} \frac{\rho_{st}(x_0^{(n)})}{\rho_{st}(x_0^{(o)})} \right]$$

As in the case of the shooting move two cases are of interest. First, suppose $\rho(x) = \rho_{st}(x)$, then (2.27) simplifies to

$$P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = h_A(x_0^{(n)})h_B(x_{\mathcal{T}}^{(n)}) \quad (2.28)$$

which is valid even for a generalized Hamiltonian dynamics. Second, suppose $\rho(x) \neq \rho_{st}(x)$ but the Jacobian of the transformation (2.23) is 1. Then (2.27) becomes

$$P_{acc}[x^{(o)}(\mathcal{T}) \rightarrow x^{(n)}(\mathcal{T})] = h_A(x_0^{(n)})h_B(x_{\mathcal{T}}^{(n)}) \min \left[1, \frac{\rho(x_0^{(n)})}{\rho(x_0^{(o)})} \right] \quad (2.29)$$

which depends on the slice at time 0. In the case of a canonical distribution and NVE Hamiltonian dynamics, however, $\rho(x)$ depends only on the energy, which is conserved by the dynamics. Thus, the Metropolis acceptance criterion can be applied before the expensive integration of the dynamics.

The shifting algorithm is as follows. Given a path, choose a time displacement $\Delta\mathcal{T}$ from a symmetrical distribution around 0. If necessary check the Metropolis criterion in the last term of (2.29). Then integrate the proper new part of the path at the beginning (if $\Delta\mathcal{T} < 0$) or at the end (if $\Delta\mathcal{T} > 0$). If the new path still connects A and B accept, otherwise reject it.

The shifting move is a basic ingredient of path sampling, because considerably improves the sampling of the transition ensemble [75]. The move is also computationally cheap, because it requires integration of the dynamics only over a short time $\Delta\mathcal{T}$. If all the slices are saved in memory, then reversing a shift move is costless. The shooting move on the contrary is very expensive, requiring the creation of a new path with each attempt.

The shifting move is called a *diffusion move* when $\Delta\mathcal{T} = \pm\Delta t$ with probability 1/2. In fact if this short shift is applied frequently, it results in a sort of diffusion of the path, which smoothes the computed time-observables, e.g. $C(t)$, see sec. 2.3.

2.2.3 Path reversal

The transition path ensemble (2.6) can be modified to include trajectories that start in A , visit B , but not necessarily stay in B until $t = \mathcal{T}$. This modification becomes useful when computing rates, see sec. 2.3. Define

$$H_B[x(\mathcal{T})] \equiv \max_{0 \leq t \leq \mathcal{T}} h_B(x_t) \quad (2.30)$$

which is 0 only if the path never visited B and 1 otherwise. Replacing h_B with H_B in (2.6) we have the new transition path ensemble

$$\mathcal{P}_{AB}^*[x(\mathcal{T})] \equiv [Z_{AB}^*(\mathcal{T})]^{-1} h_A(x_0) \mathcal{P}[x(\mathcal{T})] H_B[x(\mathcal{T})] \quad (2.31a)$$

$$Z_{AB}^*(\mathcal{T}) \equiv \int \mathcal{D}x(\mathcal{T}) h_A(x_0) \mathcal{P}[x(\mathcal{T})] H_B[x(\mathcal{T})] \quad (2.31b)$$

The shooting and shifting algorithm are still able to sample this ensemble. In addition, if a path starts in A , visits B and goes back to A , we can get a new path by just applying time-reversal. Inverting all the velocities of the slices and reading the path backwards we have another path of the ensemble. This move is computationally inexpensive because it requires no integration of the dynamics. One simply checks if the path ends in A , and if it does reverse the momenta and accept, otherwise reject.

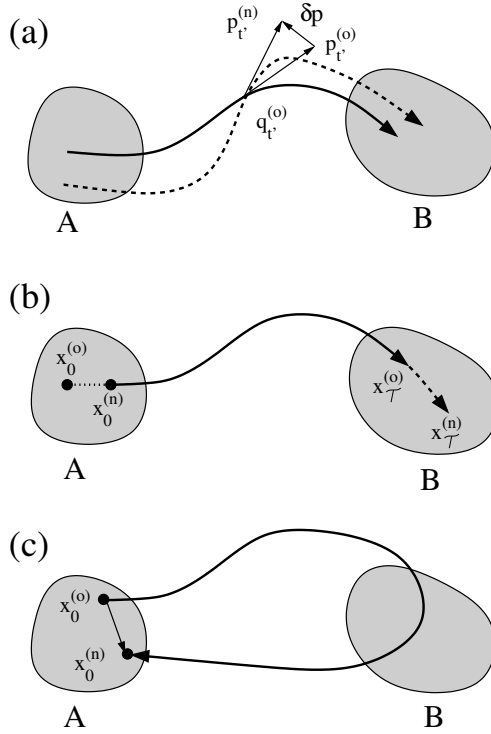


Figure 2.2: TPS algorithms: (a) shooting (b) shifting (c) path reversal

2.2.4 Joining the algorithms

In a transition path sampling simulation we use all the above mentioned algorithms. Assigning a probability α_i to each of them, with $\sum_i \alpha_i = 1$, the complete TPS algorithm reads

1. take an initial path belonging to the transition path ensemble
2. draw a random number $y \in [0, 1]$ and see in which range falls



If $\sum_{j=1}^{i-1} \alpha_j < y < \sum_{j=1}^i \alpha_j$ choose generating algorithm i

3. generate a new path using the chosen algorithm. Accept or reject according to the rules of the algorithm
4. if the new path is accepted, it becomes the current one. Otherwise the old path is retained as the current one again.
5. sample observables on the path
6. repeat from step 2

Note that step 2 involves a random number to satisfy overall detailed balance. One cannot fix a sequence of algorithms, e.g. a shoot and a shift, but one can specify $\alpha_{\text{shoot}} = \alpha_{\text{shift}} = 0.5$.

We remark that in the above algorithm the first step requires an initial path already belonging to the ensemble. In conventional MC, for example when simulating a liquid, one starts from a lattice configuration which, even if not equilibrated, belongs to the ensemble, i.e. its weight is low but not 0. Eventually an equilibration run will generate configurations with higher weight. In TPS even the very first configuration requires special care, because a path is needed with weight different from 0, i.e. a transition. Producing it from an MD run is usually difficult because the event is rare. The trajectory does not have to be a dynamical one, so you could use interpolation schemes between initial and final states, in order to approximate a transition state. Shooting from it one can hope to end in both A and B . Biased MD simulations are another option, including running at high-temperature and subsequent cooling down. However, all in all there is no general rule to create a first path [35]. This is a lack in a very important step of the procedure. We will see in chap. 3 how the problem is naturally overcome in TIS.

Finally, we mention some other algorithms to sample the path ensemble that have appeared in literature [80]: the local algorithm for stochastic dynamics, which samples slices individually; the dynamical algorithm, based on the path action; and an algorithm based on Configurational Biased Monte Carlo, which exploits the analogy between polymers and stochastic paths.

TPS has also been used in connection with parallel tempering [81]. Different path sampling simulations are carried at different temperatures and in addition to the moves of step 2 a move is present that exchanges paths between the various simulations. Ideal for implementation on parallel computers, the idea enhances considerably the sampling at low temperatures [35].

2.3 Computing Rates

In sec. 1.5 we showed that in a system with separation of timescales the time derivative of the correlation function $C(t)$, Eq. (1.38), displays a plateau equal to the rate constant k_{AB} . The fact was then used in the context of TST-BC where the states are adjacent, but the proof Eq. (1.39) we give in appendix A.3 does not use a reaction coordinate and holds for the general situation of Fig. 1.1. TPS is a method that computes the correlation function $C(t)$, and hence the rate constant.

2.3.1 Umbrella sampling

Because of the definition of the path ensemble (2.5), an ensemble average $\langle A(t) \rangle$ can be recast into an average over all the trajectories

$$\langle A(t) \rangle_{AB} = \frac{\int \mathcal{D}x(t) \mathcal{P}[x(t)] A(t)}{\int \mathcal{D}x(t) \mathcal{P}[x(t)]} \quad (2.32)$$

The correlation function $C(t)$ can then be rewritten as

$$\begin{aligned} C(t) &= \frac{\langle h_A(0) h_B(t) \rangle}{\langle h_A \rangle} \\ &= \frac{\int \mathcal{D}x(t) \mathcal{P}[x(t)] h_A(x_0) h_B(x_t)}{\int \mathcal{D}x(t) \mathcal{P}[x(t)] h_A(x_0)} \end{aligned} \quad (2.33)$$

where the denominator is in fact t independent and equal to the equilibrium average $\langle h_A \rangle$. Having defined an order parameter $\lambda(x)$ (sec. 2.1.2) we can choose the set B as

$$B = \{x : \lambda_{min}^B < \lambda(x) < \lambda_{max}^B\} \quad (2.34)$$

As a consequence, we can insert the characteristic function $h_B(x)$ in Eq. (2.33) and obtain (2.33) as

$$\begin{aligned} C(t) &= \frac{1}{\langle h_A \rangle} \int \mathcal{D}x(t) \mathcal{P}[x(t)] h_A(x_0) \int_{\lambda_{min}^B}^{\lambda_{max}^B} d\lambda \delta[\lambda - \lambda(x_t)] \\ &= \int_{\lambda_{min}^B}^{\lambda_{max}^B} d\lambda \frac{\int \mathcal{D}x(t) \mathcal{P}[x(t)] h_A(x_0) \delta[\lambda - \lambda(x_t)]}{\int \mathcal{D}x(t) \mathcal{P}[x(t)] h_A(x_0)} \\ &= \int_{\lambda_{min}^B}^{\lambda_{max}^B} d\lambda \langle \delta[\lambda - \lambda(x_t)] \rangle_A \\ &\equiv \int_{\lambda_{min}^B}^{\lambda_{max}^B} d\lambda P_A(\lambda, t) \end{aligned} \quad (2.35)$$

where $\langle \cdot \rangle_A$ denotes an average on paths beginning in A . The function $P_A(\lambda, t)$ is the probability that a trajectory started in A reaches λ at time t . Since we are interested in the function up to values of λ in the range (2.34) and $t \ll \tau_{stable}$, the computation of $P_A(\lambda, t)$ involves overcoming a low probability region.

The same problem is found in regular free energy calculations, and a solution is the introduction of an artificial bias, by e.g. the umbrella sampling method [14]. Let us review the method in the case of a bias in the form of hard windows. Given a system state x , an equilibrium distribution $\rho(x)$ and an order parameter $\lambda(x)$, we want to compute

$$P(\lambda) = \frac{\int dx \rho(x) \delta[\lambda - \lambda(x)]}{\int dx \rho(x)} = \langle \delta[\lambda - \lambda(x)] \rangle \quad (2.36)$$

We therefore partition the phase space, dividing the total range of λ , into slightly overlapping hard windows

$$W_i = \{x : \lambda_L^{(i)} < \lambda(x) < \lambda_R^{(i)}\} \quad (2.37)$$

such that $\cup_i W_i$ is the total phase space, but the sets are not disjoint. Next, we consider

$$P_{W_i}(\lambda) = \frac{\int dx \rho(x) h_{W_i}(x) \delta[\lambda - \lambda(x)]}{\int dx \rho(x) h_{W_i}(x)} = \langle \delta[\lambda - \lambda(x)] \rangle_{W_i} \quad (2.38)$$

The probability (2.38) is proportional to (2.36) when $\lambda \in W_i$ and zero otherwise. The proportionality factor is the measure of the set $\langle h_{W_i} \rangle$, which is difficult to compute because there are regions W_i of low probability. Even if we start a simulation with a configuration belonging to such a low probability region, there will be a tendency to move away from this region. However, we can constrain the simulation to remain in the region W_i and compute (2.38). In case of MC trial move we simply reject a trial moves when it leaves the region. This is equivalent to biasing with a square well potential equal to 0 if $x \in W_i$ and ∞ otherwise. As a consequence, if we compute the window average (2.38) for a couple of slightly overlapping regions, the two probability histograms (2.38) must be the same in the common region. We can rescale one to the other using the probability in the common areas. Repeating this for all subsequent windows we can reconstruct the whole probability histogram (2.36), and normalize it to one. This procedure is successful if the windows are chosen narrow enough for the sampling (2.38) to gather enough statistics in each window.

The principle can be straightforwardly applied to the sampling of the path probability $P_A(\lambda, t)$, which corresponds to average (2.36). The path sampling analog of (2.38) is

$$\begin{aligned} P_{AW_i}(\lambda, t) &= \frac{\int \mathcal{D}x(t) \mathcal{P}[x(t)] h_A(x_0) h_{W_i}(x_t) \delta[\lambda - \lambda(x_t)]}{\int \mathcal{D}x(t) \mathcal{P}[x(t)] h_A(x_0) h_{W_i}(x_t)} \\ &= \langle \delta[\lambda - \lambda(x_t)] \rangle_{AW_i} \end{aligned} \quad (2.39)$$

which is a probability histogram in λ computed in the ensemble of paths starting in A and ending in W_i . Computing (2.39) in all the windows W_i and rematching as described above, $P_A(\lambda, t)$ can be obtained, and from it $C(t)$ using (2.35). The procedure is illustrated in Fig. 2.3 for the isomerization of a 7-particle Lennard-Jones cluster.

2.3.2 Path ensemble average

In principle, the umbrella sampling procedure could be repeated for every t to get the full correlation function $C(t)$. In practice, this is too expensive and is not necessary

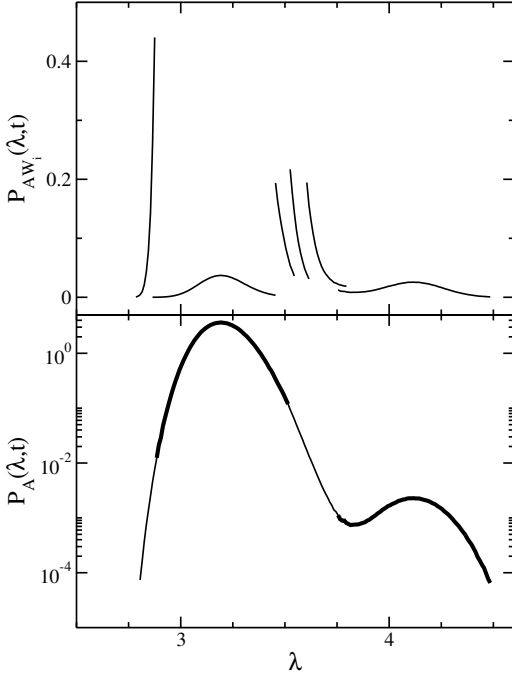


Figure 2.3: Calculation of $P_A(\lambda, t)$, the probability of a trajectory being in λ at time t provided that it started in A at time 0. The illustrated process is the isomerization of 7-particle LJ cluster at $E/N = -2.072$ (see sec. 2.4). The calculation employs the umbrella sampling window technique. Top: calculation of the distribution in the windows. Bottom: rematching all curves. The states A and B are indicated by bold curves.

as the computation can be simplified using a convenient factorization [80, 79].

Let $t' < t$ and rewrite $C(t)$ as

$$\begin{aligned}
 C(t) &\equiv \frac{\langle h_A(0)h_B(t) \rangle}{\langle h_A \rangle} \\
 &= \frac{\langle h_A(0)h_B(t) \rangle}{\langle h_A(0)h_B(t') \rangle} \frac{\langle h_A(0)h_B(t') \rangle}{\langle h_A \rangle} \\
 &= \frac{\langle h_A(0)h_B(t) \rangle}{\langle h_A(0)h_B(t') \rangle} C(t')
 \end{aligned} \tag{2.40}$$

This expression shows that if one knows $C(t')$ at time t' , one can calculate $C(t)$ at all other times t by multiplying by the factor $\langle h_A(0)h_B(t) \rangle / \langle h_A(0)h_B(t') \rangle$. Since the function $H_B[x(\mathcal{T})]$, eq. (2.30), is zero only if $h_B(x_t)$ vanishes for all x_t of the path, we can insert it into the ensemble averages. We obtain then for the first factor

$$\begin{aligned}
 \frac{\langle h_A(0)h_B(t) \rangle}{\langle h_A(0)h_B(t') \rangle} &= \frac{\langle h_A(0)h_B(t) \rangle}{\langle h_A(0) \rangle} \frac{\langle h_A(0) \rangle}{\langle h_A(0)h_B(t') \rangle} \\
 &= \frac{\langle h_A(0)h_B(t)H_B[x(\mathcal{T})] \rangle}{\langle h_A(0)H_B[x(\mathcal{T})] \rangle} \frac{\langle h_A(0)H_B[x(\mathcal{T})] \rangle}{\langle h_A(0)h_B(t')H_B[x(\mathcal{T})] \rangle} \\
 &= \frac{\langle h_B(x_t) \rangle_{AB}^*}{\langle h_B(x_{t'}) \rangle_{AB}^*}
 \end{aligned} \tag{2.41}$$

where the notation specifies a path average in the path ensemble (2.31). Substituting

(2.41) into (2.40) we finally obtain

$$C(t) = \frac{\langle h_B(x_t) \rangle_{AB}^*}{\langle h_B(x_{t'}) \rangle_{AB}^*} C(t') \quad (2.42)$$

$$k(t) \equiv \dot{C}(t) = \frac{\langle \dot{h}_B(t) \rangle_{AB}^*}{\langle \dot{h}_B(t') \rangle_{AB}^*} C(t') \quad (2.43)$$

We can now summarize the TPS procedure for the calculation of the rate constant.

1. Calculate the average $\langle h_B(t) \rangle_{AB}^*$ in the path ensemble (2.31), i.e. paths that start in A and visit B at least once
2. If the time derivative $d\langle h_B(t) \rangle_{AB}^*/dt$ displays a plateau go to next step, otherwise repeat step 1 with a longer time t
3. Calculate the correlation function $C(t')$ for fixed $t' \in [0, t]$ using umbrella sampling
4. Determine $C(t) = C(t') \langle h_B(t) \rangle_{AB}^* / \langle h_B(t') \rangle_{AB}^*$ in the entire interval $[0, t]$.
5. Compute the derivative $\dot{C}(t)$. The rate constant k_{AB} is the value of the plateau

Finally, we derive the correspondence with the reactive flux formalism (BC procedure) of sec. 1.5. Suppose the states A and B are adjacent. We have then

$$C(t') \simeq C(0) + \dot{C}(0^+)t' = k_{AB}^{TST}t' \quad (2.44)$$

$$\langle h_B(t') \rangle_{AB}^* \simeq \langle h_B(0) \rangle_{AB}^* + \langle \dot{h}_B(0) \rangle_{AB}^* t' = \langle \dot{h}_B(0) \rangle_{AB}^* t' \quad (2.45)$$

where we used (1.44) and $\langle h_B(0) \rangle_{AB}^* = 0$ because by definition of the ensemble the paths begin in $A \neq B$. Then (2.43) for small $t' > 0$ becomes

$$k(t) = \frac{\langle \dot{h}_B(t) \rangle_{AB}^*}{\langle \dot{h}_B(0) \rangle_{AB}^*} k_{AB}^{TST} \quad (2.46)$$

and by comparison with (1.48) we have

$$\kappa(t) = \frac{\langle \dot{h}_B(t) \rangle_{AB}^*}{\langle \dot{h}_B(0) \rangle_{AB}^*} \quad (2.47)$$

i.e. the transmission coefficient is $\langle \dot{h}_B(t) \rangle_{AB}^*$ normalized to its value at $t = 0^+$ [79].

2.4 The $(LJ)_7$ cluster

In this section we present an application of TPS to the rearrangement of a cluster of seven particles interacting through a pairwise Lennard-Jones potential. We study the transition from the global minimum to the second lowest energy minimum. For such a simple system it makes sense to explore the potential energy surface (PES). We dedicate the next section to this, but when applying TPS we will show that the method does not rely on any knowledge of the PES. We conclude with a connection between the two approaches.

2.4.1 Potential Energy Surfaces

We describe N particles in dimension $d = 3$ using the potential energy function $\mathcal{U}(\mathbf{r}_1 \dots \mathbf{r}_N)$, with Cartesian positions $\mathbf{r}_i \in \mathbb{R}^d$. For a cluster of N particles in vacuum with no additional conditions, we require the function to have the following properties

1. Invariance under space translation, if all particle positions are displaced by a common vector
2. Invariance under space rotation, if all particle positions are rotated with the same transformation
3. Invariance under space inversion, if all coordinates are inverted through a space-fixed origin
4. Invariance under equivalent particle permutation, if two particles of the same kind are exchanged

Properties 1, 2 imply in three dimensions that the number of degrees of freedom is reduced to $s = 3N - 6$, due to conservation of total linear and angular momentum. An exception are systems composed of linear molecules for which $s = 3N - 5$ because we do not need to conserve the component of angular momentum along the linear axis. If a bulk system is considered and periodic boundary conditions are applied, we lose property 2 and consequently $s = 3N - 3$. For a system composed of N_A atoms of type A , N_B atoms of type B , etc., it follows from properties 3, 4 that the number of permutation-inversion isomers of any given configuration is

$$2 \times N_A! \times N_B! \times \dots \quad (2.48)$$

but this number is actually reduced by a factor equal to the order of the prevailing point group [33].

The important features of potential energy surfaces are the stationary points, for which $\nabla \mathcal{U} = 0$. A number of studies of PES for specific systems has resulted in general methods to locate stationary points [34]. Usually stationary points are classified through their index, that is the number of negative eigenvalues of the corresponding Hessian matrix $\partial^2 \mathcal{U} / \partial r_i \partial r_j$. With this definition points of index 0 correspond to minima. Points of index 1 are the highest energy points of minimum energy paths between minima and they are therefore taken as definition of transition states [31]. Higher index points are not important as far as transitions between minima are concerned. For completeness, a study on Lennard-Jones clusters showed that the number of stationary points of index I has a gaussian shape with a maximum for $I \simeq N - 2$ and a cut-off at $I = 2N - 4$, beyond which there are no stationary points [82]. There is no strict rule for the growth in the number of stationary points corresponding to different structures, i.e. excluding inversions and permutations (2.48). The evolution in the number of stationary points is system dependent, but empirical observations and theoretical arguments suggest that the number of structurally distinct minima is likely to increase as [33]

$$N_{min} = e^{aN^b} \quad a > 0, b = O(1) \quad (2.49)$$

which shows that in these finite cluster systems, the PES landscape becomes rapidly complex. Great effort has been spent for decades to locate the minima of clusters bound by specific potentials modelling real interactions [83]. In the competitive quest for the global minimum extensive databases have been created of cluster stationary points, in some cases up to $N = 1000$ [84].

Here we are interested in potential energy functions given by pairwise interactions, $\mathcal{U} = \sum_{ij} U(\mathbf{r}_i, \mathbf{r}_j)$. Because of the above requirements 1 and 2, it takes the form

$$\mathcal{U}(\mathbf{r}_1 \dots \mathbf{r}_N) = \frac{1}{2} \sum_{ij=1}^N U(r_{ij}) = \sum_{i<j} U(r_{ij}) \quad (2.50)$$

where $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$ is the interparticle distance. We take the pairwise interaction $U(r)$ in the form of a Lennard-Jones (LJ) potential [85]

$$U_{LJ}(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right] \quad (2.51)$$

Besides its long history of didactical application as the typical pair interaction, the LJ potential is extensively used to model parts of more complicated potentials, and it is a reasonably good model for noble gases, such as Argon. Here, and in the following we consider all the particles with equal mass m and we use reduced units, so that ϵ , σ and m are unity. The LJ unit of time $(m\sigma^2/\epsilon)^{1/2}$ is therefore also unity.

The system consists of $N = 7$ particles in dimension $d = 3$, called $(LJ)_7$. Its PES has been completely determined. As far as transitions are concerned, it consists of 4 minima and 12 saddle points of index one, the full exploration being first completed by Tsai and Jordan [86]. The PES is shown in Fig. 2.4 In this study we are interested in the transition from state A , the basin of the global minimum pentagonal bipyramid, to state B , the basin of the second lowest one, a capped octahedron. We could have chosen other transitions but we preferred this because it shows no intermediates. We now turn to see how TPS can compute the rate constant of this process.

2.4.2 Choice of the order parameter

Denoting with $(\mathbf{p}_1 \dots \mathbf{p}_N)$ the momenta of the particles, we write the full Hamiltonian

$$\mathcal{H} = \sum_{i=1}^N \frac{\mathbf{p}_i^2}{2} + \sum_{i<j} 4 [r_{ij}^{-12} - r_{ij}^{-6}] \quad (2.52)$$

To identify an order parameter able to distinguish the pentagonal bipyramid from the capped octahedron we inspected both the potential energy and the moments of inertia. The latter are defined as the eigenvalues of the inertia tensor [88]

$$\mathbf{I} = \sum_{i=1}^N r_i^2 \mathbf{1} - \mathbf{r}_i \mathbf{r}_i \quad (2.53)$$

where $\mathbf{1}$ is the 3×3 identity matrix and the second term uses the dyadic product. We also assume that the center of mass of the cluster is in the origin. Eigenvalues

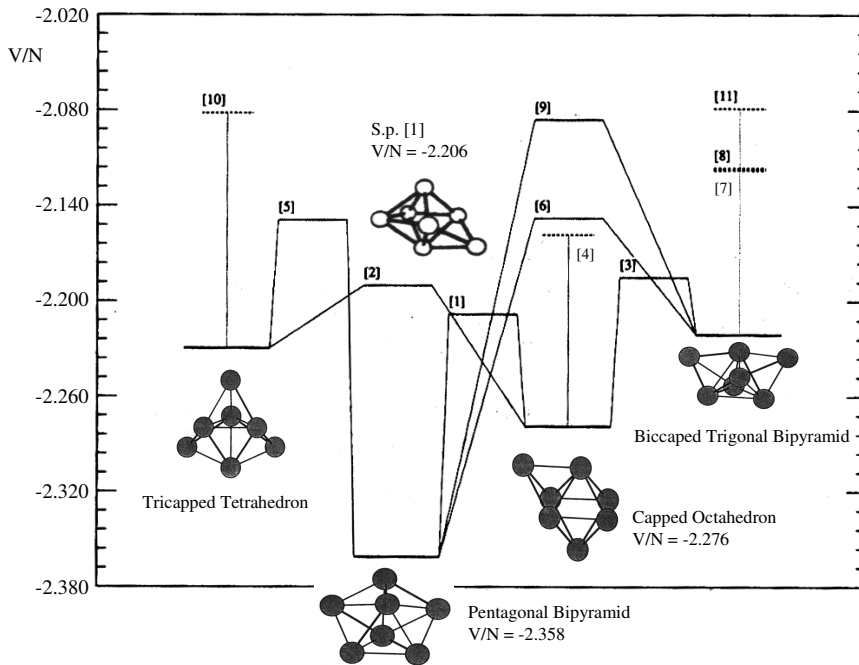


Figure 2.4: The PES of $(LJ)_7$. The high-energy 12th saddle point is omitted. Minima are represented and saddles labeled in square brackets. The first saddle point is also represented as the transition state for the transition we are analyzing, from the pentagonal bipyramid to the capped octahedron. Some saddles connect different minima, and the connections are showed. Other saddles (dashed lines) connect permutational isomers of the same structure, are termed degenerate and display symmetrical features. The figure is adapted from [86, 87].

are usually identified in ascending order $I_1 \leq I_2 \leq I_3$. We report in table 2.2 their values for the four $(LJ)_7$ minima together with the values of the potential energy. Note that this is at variance with the usual custom in chemical reactions of using as shape indicators the eigenvalues of the radius of gyration tensor, see sec. 7.4.1.

We simulate the system at constant total energy E by integrating the equations of motion corresponding to (2.52) using a Verlet integrator with time step $\Delta t = 0.005$. The total linear and angular momentum were set to 0. We report in Fig. 2.5 an MD run at the total energy $E/N = -2.0$, started from a configuration equilibrated from the global minimum. The energy is high enough to observe several transitions in the total simulation time. We note that even if the potential energy slightly changes during the transitions, it is not able to distinguish them. The moments of inertia, on the other hand, are much better indicators. We report in the same figure our guesses for the basin of attraction of the configurations, based on the zero-temperature values of table 2.2. These guesses proved to be correct in a later quenching analysis. For the transition we are interested in ($BP \rightarrow CO$ in figure 2.5) we observe that I_2 can distinguish the initial and final states. We have run distinct NVE simulations in the

Minimum	\mathcal{U}/N	I_1	I_2	I_3
Pentagonal Bipyramid	-2.358	2.94	2.94	4.57
Capped Octahedron	-2.276	2.50	4.09	4.09
Tricapped Octahedron	-2.228	3.24	3.24	4.69
Bicapped Trigonal Bipyramid	-2.219	2.31	4.53	4.75

Table 2.2: Potential energy per particle and moments of inertia for the four minima of $(LJ)_7$. Note the first three all have two degenerate eigenvalues of the inertia tensor, corresponding to cylindrical symmetry.

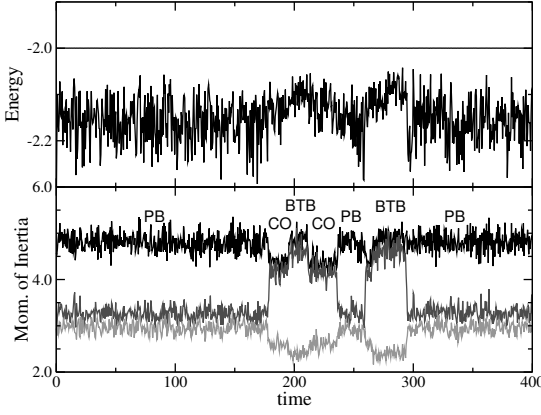


Figure 2.5: MD run started in the basin of the pentagonal bipyramid. Top: total and potential energy. Total energy is conserved at $E/N = -2.0$. Bottom: moments of inertia I_3 black, I_2 dark grey, I_1 light gray. The abbreviations indicate: PB pentagonal bipyramid, CO capped octahedron, BTB bicapped trigonal bipyramid.

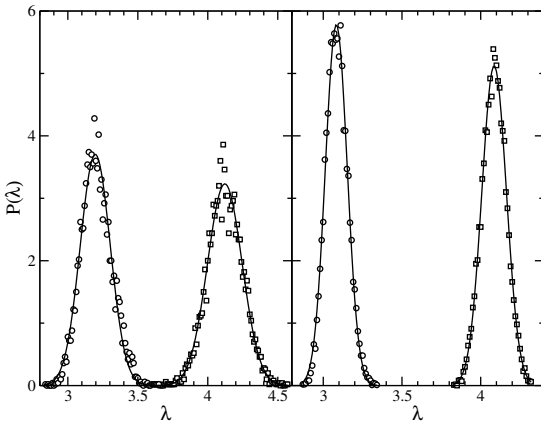


Figure 2.6: Histograms of $\lambda = I_2$ for MD runs started in the pentagonal bipyramid (circles) and the capped octahedron (squares) for the highest and lowest total energy considered: Left: $E/N = -2.072$, Right: $E/N = -2.205$. The lines are gaussian fits.

initial and final states at four different total energies (see table 2.3), we histogrammed the values of I_2 , and we report them in Fig. 2.6. The highest energy was chosen in order to compare with previous results in literature [89], the lowest one was just above the transition state, to test the limits of TPS. For all energies the I_2 histograms are indeed well separated.

2.4.3 Rate constants

We define stable states A and B using gaussian fits on the histograms of I_2 (see Fig. 2.6). A configuration is identified as a stable state if its I_2 value is around 3σ of the mean value of the stable state. See table 2.3 for details.

E/N	-2.072	-2.100	-2.200	-2.205
A	3.20 ± 0.32	3.17 ± 0.29	3.12 ± 0.25	3.08 ± 0.21
B	4.12 ± 0.37	4.11 ± 0.33	4.10 ± 0.25	4.09 ± 0.23

Table 2.3: Ranges of I_2 defining states A and B for the four energies considered.

Following the procedure of sec. 2.3.2 we computed $\langle h_B(t) \rangle_{AB}^*$ with a path ensemble simulation. The trajectory length was $\mathcal{T} = 5.0$ and we used the shooting, shifting, and diffuse move with probabilities 20%, 40%, 40% respectively (see sec. 2.2.4). In the shooting algorithm we took care of constraining zero total linear and angular momentum (sec. 2.2.1). The momentum displacement in the shooting move and the time origin displacement in the shifting move were chosen such that the acceptance ratio was of 40%. To save memory time, slices of paths were stored every ten MD steps. The total number of harvested paths was 10^6 . We report the results in Fig. 2.7, $\langle h_B(t) \rangle_{AB}^*$ and its time derivative. Clearly, the length \mathcal{T} was long enough for all the curves to display a plateau.

Subsequently, we applied the umbrella sampling technique (see sec.2.3.1) to compute $C(t')$ at time $t' = \mathcal{T}$. We could have chosen a shorter t' to save CPU time, but in such a small system (7 particles) it does not really matter. The whole calculation took only one hour per energy choice on a 1GHz AMD Athlon. We report in Fig. 2.8 the reconstructed probability function $P_A(\lambda = I_2, t' = \mathcal{T})$. After integration over the range defining B , we get the value of $C(t' = \mathcal{T})$. Finally, using the last point of $\langle \dot{h}_B(t) \rangle_{AB}^*$ as the plateau value, we get the rate constant k_{AB} from the formula

$$k_{AB} = \frac{\langle \dot{h}_B(\mathcal{T}) \rangle_{AB}^*}{\langle h_B(\mathcal{T}) \rangle_{AB}^*} C(\mathcal{T}) \quad (2.54)$$

All these results are summarized in table 2.4.

Since we have a detailed knowledge of the PES, we can also apply TST using as order parameter the direction provided by the eigenvector of the unstable mode of the transition state, the saddle at $\mathcal{U}_{TS}/N = -2.206$. The quantities in (1.33) can be evaluated in the microcanonical ensemble using an harmonic approximation. This harmonic approach is common in the theory of unimolecular reactions, and is usually known as RRKM theory, see appendix C. The resulting formula for the

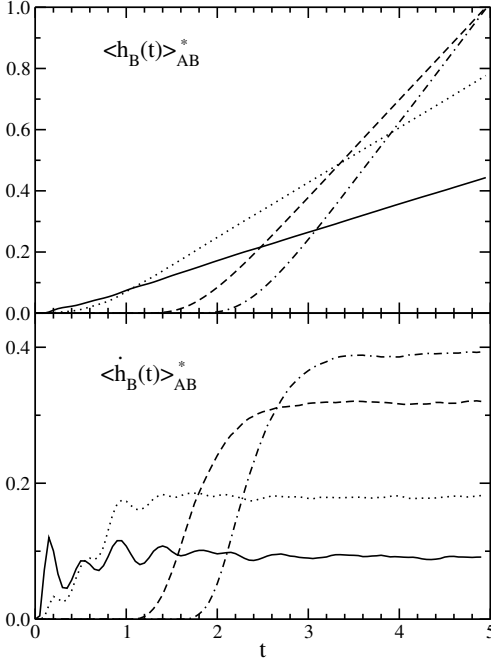
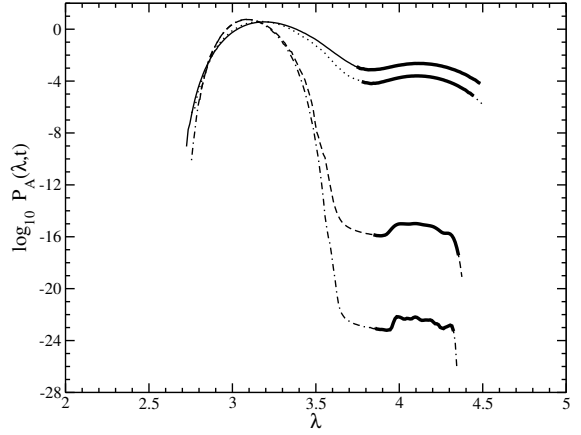


Figure 2.7: Path sampling simulations. Top: $\langle h_B(t) \rangle_{AB}^*$. Bottom: the time derivative. Solid: $E/N = -2.072$, Dotted: $E/N = -2.100$, Dashed: $E/N = -2.200$, Dot-dashed: $E/N = -2.205$.

Figure 2.8: Probability distribution $P_A(\lambda, t)$ of order parameter $\lambda = I_2$ at time $t = T$ for paths starting in A , Eq. (2.35). The factor $C(t' = T)$ is given by the integral over the thicker regions. Line styles are defined as in Fig. 2.7.



E/N	$\langle \dot{h}_B(T) \rangle_{AB}^*$	$\langle h_B(T) \rangle_{AB}^*$	$C(T)$	k_{AB}	k_{AB}^{-1}
-2.072	0.09	0.44	$8.99 \cdot 10^{-4}$	$1.85 \cdot 10^{-4}$	11.6 ns
-2.100	0.18	0.78	$8.70 \cdot 10^{-5}$	$2.04 \cdot 10^{-5}$	106 ns
-2.200	0.32	1.00	$2.69 \cdot 10^{-16}$	$8.62 \cdot 10^{-17}$	6.95 h
-2.205	0.39	1.00	$1.40 \cdot 10^{-23}$	$5.54 \cdot 10^{-24}$	12.3 ky

Table 2.4: Resume of rate constant calculations for different energies. The last columns gives the inverse rates in real units for Argon.

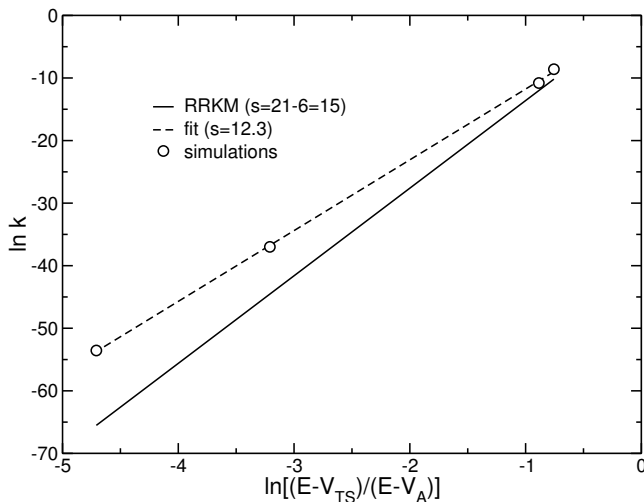


Figure 2.9: Comparison between TPS and the microcanonical TST called RRKM.

microcanonical rate constant is

$$k_{AB}(E) = \left(\frac{E - \mathcal{U}_{TS}}{E - \mathcal{U}_A} \right)^{s-1} \frac{\prod_{i=1}^s \nu_i^A}{\prod_{i=1}^{s-1} \nu_i^{TS}} \quad (2.55)$$

where in our case \mathcal{U}_A is the potential energy of the pentagonal bipyramid, and the number of degrees of freedom is $s = 3 \cdot 7 - 6 = 15$. The ν_i 's are the vibrational frequencies calculated from $\nu_i = \sqrt{\mu_i}/2\pi$, where μ_i are the positive eigenvalues of the Hessian matrix at the stationary points. Accordingly the saddle has one positive eigenvalue less. We report in Fig. 2.9 the comparison between TPS and RRKM. It appears that TPS values are better fitted using a different value for s , namely an effective $s = 12.3$ instead of the full 15. The result is consistent with another study of $(LJ)_7$ in two dimensions [90].

Concluding, we have successfully tested TPS on an isomerization reaction in a simple 7-particle LJ system. Without the knowledge of the PES, we have been able to compute accurate transition rates, even to very large timescales.

2.5 Finding the right reaction coordinate

At the end of the previous chapter, sec. 1.7, we remarked that the efficiency of the TST-BC procedure strongly depends on the choice of a reaction coordinate. It is implicitly assumed that one can guess a proper parametrization $\lambda(r)$ such that the advance and the features of the transition are well described. This is certainly not obvious in complex systems, and for this reason Transition Path Sampling was created, which does not rely on a reaction coordinate, but on an order parameter (see sec. 2.1.2). The primary requirement of an order parameter is to be able to distinguish

between the stable regions, but not more. This requirement is enough to implement algorithms able to sample the path ensemble (2.6), and hence compute dynamical properties, such as the rate constant.

The next question is what we can learn from all the sampled paths. Can we use them to find the reaction coordinate and describe the transition? Preconceived reaction coordinates might be wrong. One could use the PES to find out a RC, but in complex systems this becomes too complicated. Also, in general it is not possible to recognize the reaction coordinate by simply visualizing trajectories, e.g. using computer graphics. Indeed, we need some more statistically reliable analysis of the generated path ensemble.

This section is dedicated to this goal. Although we do not have the final answer to the problem of finding the right reaction coordinate, we review several ways of extracting from the path ensemble some interesting information.

2.5.1 Commitment probabilities

Given the path ensemble (2.5), define the commitment probability or *committor*³, as

$$p_B(r) \equiv \frac{\int \mathcal{D}x(\bar{t}) \mathcal{P}[x(\bar{t})] \delta(r_0 - r) h_B(x_{\bar{t}})}{\int \mathcal{D}x(\bar{t}) \mathcal{P}[x(\bar{t})] \delta(r_0 - r)} \quad (2.56)$$

where \bar{t} is the first hitting time to A or B . The committor $p_B(r)$ is the probability that a trajectory started from configuration r ends in state B . It indicates the commitment of r to the basin of attraction of B . The probabilistic feature is retained also for deterministic dynamics because of the random choice of momenta. The same quantity can be defined for state A , and if there are no other attractive basins

$$p_A(r) + p_B(r) = 1, \quad (2.57)$$

since the system will not stay in the buffer region forever.

Although the committor can be calculated for every phasepoint, the procedure to compute committors is implemented here by restricting (2.56) to the interesting paths, the transition path ensemble (2.6). For each slice of every path we generated, we draw random initial momenta and start a trajectory. We follow the trajectory x_t until it reaches A or B , and repeat for N_{tr} trajectories. The estimator

$$\frac{1}{N_{tr}} \sum_{i=1}^{N_{tr}} h_B(x_{\bar{t}_i}) \quad (2.58)$$

converges to $p_B(r)$ as $N_{tr} \rightarrow \infty$. Note that the momentum generation must satisfy the constraints of the distribution, as in the case of the shooting move (sec. 2.2.1). The procedure is summarized in Fig. 2.10. Since h_B is either 0 or 1, the error on the estimation is given by

$$\sigma = \sqrt{\frac{p_B(1 - p_B)}{N_{tr}}} \quad (2.59)$$

³The word 'committor' was conceived as a practical replacement for commitment probability. It becomes more easy then to speak of committor probability distributions instead of 'probability distributions of commitment probability'.

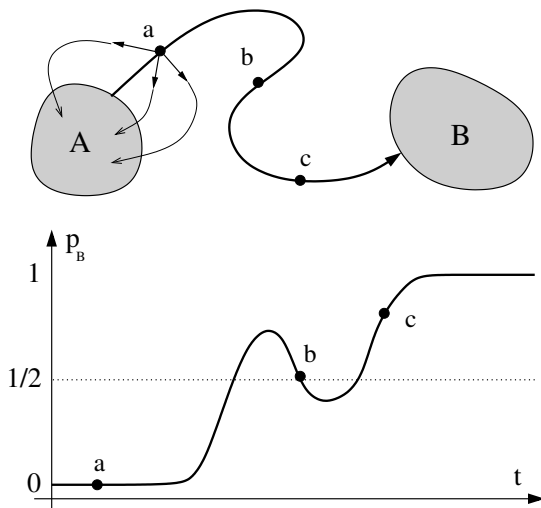


Figure 2.10: Computation of committors along a path. For each slice trajectories are shot and the p_B computed from the fraction of those that reach B . So for the point a close to A , it is almost 0, while for c close to B it is almost 1. Points for which $p_B = 1/2$ define the separatrix.

where we assume independent, uncorrelated trajectories. Given a certain level of accuracy we can stop shooting off trajectories when σ is below that level. However, we must sample at least N_{min} trajectories to ensure the estimation (2.58) of p_B we use in (2.59) is reliable. Usually $N_{min} \sim 10$ is considered sufficient [35]. The error (2.59) vanishes for $p_B = 0, 1$ and reaches a maximum $1/2\sqrt{N_{tr}}$ at $p_B = 1/2$. Given a path we can apply the committor procedure on the part of the path outside the stable states, the rest of the states being committed already. The end and the beginning of the path are close to the stable states, so p_B is close to 0 or 1, and only a few trajectories are needed. For the slices in the middle with $p_B = 0.5$, up to $N_{tr} = 100$ trajectories are needed to reach an error of 5%.

The committor can be used to regroup points in configuration space. Defining the sets

$$S_\alpha = \{r : p_B(r) = \alpha\} \quad (2.60)$$

we know that S_0 is the start of the reaction, and S_1 is the end. The committor describes the true advancement of the reaction, and in this sense it is the perfect reaction coordinate. Particularly interesting is the set $S_{1/2}$, called the *Transition State Ensemble* (TSE) or separatrix, consisting of configurations on the paths that have equal probability of ending in A or B . It is usually thought of as a surface, having in mind the TST best dividing surface of sec. 1.4.1. However, no proof exists that $S_{1/2}$ is an $n - 1$ manifold immersed in \mathbb{R}^n . It could be a closed disconnected set.

We remark that the committor computation is a very CPU time-expensive procedure. Consider the portion of a path of length \mathcal{T} close to the separatrix. Suppose it is made of $a\mathcal{T}$ slices. If t_c is the mean commitment time, the committor computation for one of these slices requires the generation of a complete set of trajectories N_{tr} of length t_c . Then for a path the committors require an effort $a\mathcal{T}N_{tr}t_c$. This is on top of the transition path sampling, which requires itself an effort $\mathcal{T}N_{path}$. Assuming $t_c \propto \mathcal{T}$ we can say the total computation time is proportional to $\mathcal{T}^3N_{tr}N_{path}$. This

is the price we have to pay to obtain the information (2.60).

2.5.2 Committor distributions

Unfortunately, there is no way to know committors a priori, in order to use them as reaction coordinate in TST for example. They give a posteriori a partition of the configuration space according to the advance of the reaction. However, the problem remains of how to characterize these sets, i.e. the large amount of configurations corresponding to a value of p_B . There is no answer to this problem. What we can do is to guess from the committor partitioning a probable good descriptive variable, and test its validity using the committor itself.

Given a parametrization $\lambda(r)$, suppose to compute the free energy $F(\lambda)$, Eq. (1.34). Then, for a fixed λ^* define the committor distribution

$$P_{\lambda^*}(p_B) = \frac{\langle \delta[p_B - p_B(r)] \delta(\lambda(r) - \lambda^*) \rangle}{\langle \delta(\lambda(r) - \lambda^*) \rangle} \quad (2.61)$$

If $\lambda(r)$ were the right reaction coordinate one would expect the configurations r for which $\lambda(r) = \lambda^*$, with λ^* the maximum of F , to coincide with the separatrix. As a consequence $P_{\lambda^*}(p_B)$ would be sharply peaked at $p_B = 1/2$. If this is not the case then we know there is a problem with our choice of $\lambda(r)$. Three possible situations are depicted in Fig. 2.11. Expression (2.61) can be used for the Geissler test, a powerful means to ascertain the validity of a chosen reaction coordinate. The test consists of two steps. First, configurations must be generated from the constrained ensemble $\lambda = \lambda^*$, using methods such as blue-moon sampling [16] or umbrella sampling [14]. Second, trajectories must be started for each configuration, and the committor computed, as explained in the previous section. Finally, the histogram of p_B is constructed. From the result one can judge the quality of a RC.

If committors from the path ensemble have been computed already, we can define a restricted path committor probability

$$\bar{P}_{\lambda^*}(p_B) \equiv \frac{\int \mathcal{D}x(T) \mathcal{P}_{AB}[x(T)] \int dt \delta[\lambda^* - \lambda(r_t)] \delta[p_B - p_B(r_t)]}{\int \mathcal{D}x(T) \mathcal{P}_{AB}[x(T)] \int dt \delta[\lambda^* - \lambda(r_t)]} \quad (2.62)$$

In practice, this is an histogram of p_B for all the slices of all the paths we have, for which $\lambda = \lambda^*$. It is just a way of rearranging the information we already have, no further computation is needed. However if $\bar{P}(p_B)$ is sharply peaked at $1/2$ it does not imply that $P(p_B)$ also is, as is clear from Fig. 2.11. But it is a necessary condition, a good reaction parameter should at least have a sharpened $\bar{P}(p_B)$. If this is not the case, then one can conclude that $\lambda(r)$ is not a good RC, even before doing the Geissler test.

2.6 Summary

TPS was developed by Chandler and collaborators [80,75,90,79,91]. The idea of a statistical description of pathways has its origins in a work by Pratt [92], where the idea of committors and separatrix was also pioneered. The TPS method has been combined with parallel tempering [81], and stochastic dynamics has been used for the case

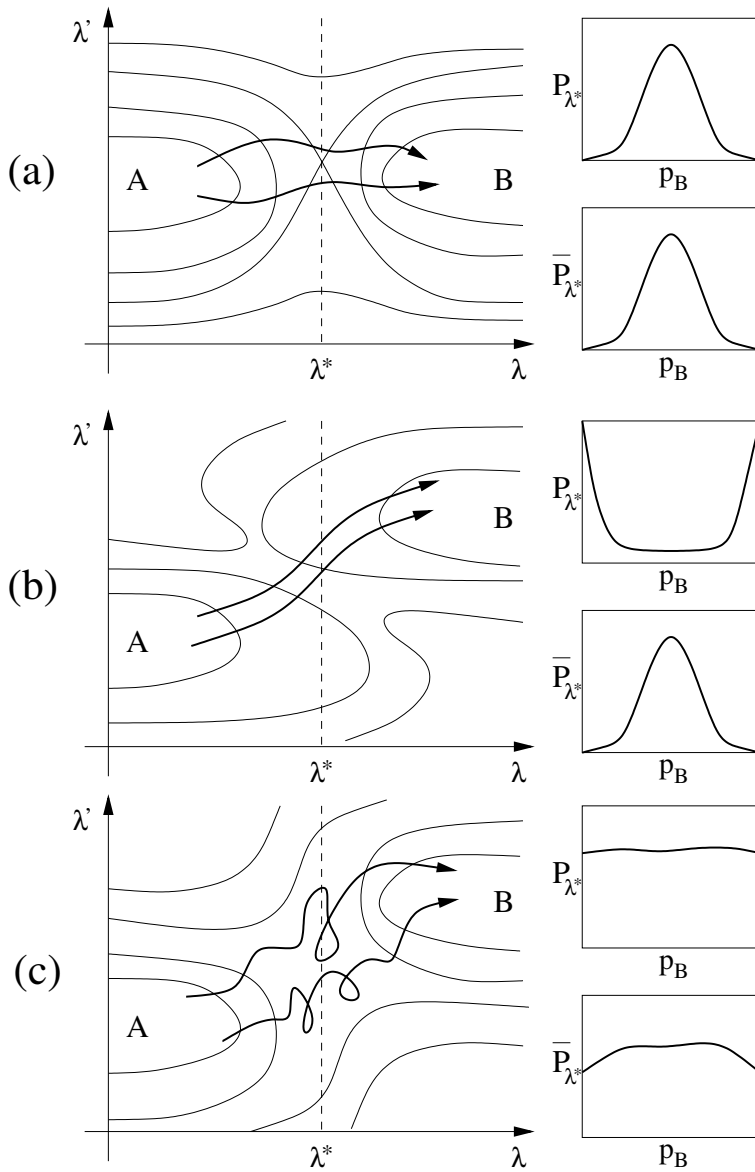


Figure 2.11: Committor distribution and Geissler test. We show a free energy contour plot in the chosen order parameter λ and an additional coordinate λ' . And we show the committor distributions Eq. (2.61) and (2.62). (a) Order parameter λ describes the reaction well, and the committor distribution $P(p_B)$ computed at the maximum λ^* in the free energy $F(\lambda)$ is sharply peaked at $p_B = 1/2$. The same for $\bar{P}(p_B)$. (b) The additional λ' plays a role, and $P(p_B)$ is bimodal. The restricted path $\bar{P}(p_B)$ cannot detect this. (c) Now $P(p_B)$ is flat, suggesting diffusive behavior. As a result also $\bar{P}(p_B)$ is flat.

of diffusive barriers [93,94]. Successful applications of TPS have been hydrogen-bond breaking in water [95], ion pair dissociation in water [96,97,98], alanine dipeptide in vacuum and in aqueous solution [99], neutral [100] and protonated [78,101] water clusters, also in combination with ab-initio simulations [102], autoionization and solvation in water [103,104], the folding of a polypeptide [3,4], a study on cavitation [105], and zeolites [106]. For a more detailed review on TPS see Refs. [107,35,108].

However, some space has been left for improvements. We have seen throughout the chapter that potential problems arise with TPS

1. Rates are computed using $C(t)$. We showed in sec. 1.5.2 that this correlation function converges to the correct result because of a cancellation of positive and negative fluxes. It can be improved using the effective positive flux.
2. Paths have a fixed length. As a result they might spend time in the stable states. This time is wasted as far as the rate constant is concerned, because only the first passage time counts.
3. An initial path must be generated before you start the path sampling, see sec. 2.2.4.

All these issues lead us to develop an improvement, which we called Interface Sampling. This is the subject of the following chapters.

3 Transition Interface Sampling

*eh...Sorry, ...maybe I missed it,
...but what does this contribute to
the unification theory?*

Titus S. van Erp

We derive an efficient scheme for the computation of rate constants, using the framework of TPS and the definition of a rate as inverse mean first passage time, see appendix A. The method is based on directly and simultaneously measuring the fluxes through many phase space interfaces. As a result the efficiency increases of at least a factor two with respect to existing transition path sampling rate constant algorithms. We introduce the theory, and present the algorithm implementation. We illustrate then the method on a diatomic molecule in a fluid of repulsive particles and make a quantitative comparison to the original TPS rate constant calculation.

3.1 Theory

The formalism of Interface Sampling has appeared in [21, 109, 20]. We try to unify it here in a compact form adapting to the line and conventions of this thesis. The definitions of sec 1.3 and 2.1 hold.

3.1.1 Interfaces and overall states

Given an order parameter $\lambda(x)$, see sec. 2.1.2, we define the interfaces $i = 0 \dots n_I$ as the hypersurfaces $\{x : \lambda(x) = \lambda_i\}$, where $\lambda_i \in \mathbb{R}$. We assume that the interfaces do not intersect, that $\lambda_{i-1} < \lambda_i$, and we describe the boundaries of state A and B by $\lambda_0 \equiv \lambda_A$ and $\lambda_{n_I} \equiv \lambda_B$ respectively, see Fig. 3.1. Usually the function $\lambda(x)$ is defined in configuration space only, but sometimes momenta might be involved, see sec. 3.3.4. It could be a simple interparticle distance, but also a quite complicated function, defined only operatively, such as the nucleation order parameter, see sec. 7.2.3. Generally defined, interfaces are $2n - 1$ manifolds immersed in $\mathbb{R}^n \times \mathbb{R}^n$ and labeled in ascending ordered according to their distance from A .

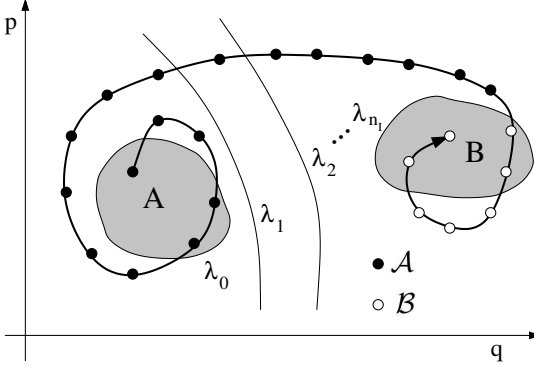


Figure 3.1: Representation of the phase space, with interfaces and overall states. A and B are the two stable states. The dots on the shown trajectory indicate the positions of the system at successive time steps. The overall state \mathcal{A} and \mathcal{B} are indicated by black and white dots respectively. The system changes from state \mathcal{A} into \mathcal{B} when it enters region B for the first time. It can leave stable region B shortly thereafter, but never go back to A in a short time. The stable regions have to be chosen to fulfill that condition.

For each phase point x and each interface i , we define a backward time $t_i^b(x)$ and a forward time $t_i^f(x)$:

$$\begin{aligned} t_i^b(x_0) &\equiv -\max[\{t | \lambda(x_t) = \lambda_i \wedge t < 0\}] \\ t_i^f(x_0) &\equiv +\min[\{t | \lambda(x_t) = \lambda_i \wedge t > 0\}], \end{aligned} \quad (3.1)$$

which mark the points of first crossing with interface i on a backward (forward) trajectory starting in x_0 . Note that t_i^b and t_i^f defined in this way always have positive values. We introduce then two-fold characteristic functions that depend on two interfaces $i \neq j$,

$$\begin{aligned} h_{i,j}^b(x) &= \begin{cases} 1 & \text{if } t_i^b(x) < t_j^b(x) \\ 0 & \text{otherwise} \end{cases} \\ h_{i,j}^f(x) &= \begin{cases} 1 & \text{if } t_i^f(x) < t_j^f(x) \\ 0 & \text{otherwise} \end{cases} \end{aligned} \quad (3.2)$$

which measure whether the backward (forward) time evolution of x will reach interface i before j or not. However, as the interfaces do not intersect, the time evolution has to be evaluated only for those phase points x that are in between the two interfaces i and j . In case $i < j$, we know in advance that $t_i^{b,f}(x) < t_j^{b,f}(x)$ if $\lambda(x) < \lambda_i$ and $t_i^{b,f}(x) > t_j^{b,f}(x)$ if $\lambda(x) > \lambda_j$. When the system is ergodic, both interfaces i and j will be crossed in finite time and thus $h_{i,j}^b(x) + h_{j,i}^b(x) = h_{i,j}^f(x) + h_{j,i}^f(x) = 1$.

The two backward characteristic functions define the TIS *overall* states \mathcal{A} and \mathcal{B} :

$$h_{\mathcal{A}}(x) = h_{0,n_I}^b(x) \quad h_{\mathcal{B}}(x) = h_{n_I,0}^b(x). \quad (3.3)$$

Overall state \mathcal{A} contains not only the points defining A but also the points of the region between A and B from which a trajectory evolves backward in time to A . The states (3.3) together span the complete phase space, as the system can never stay in

the intermediate region between A and B forever. Both overall states are illustrated in Fig. 3.1. The overall states \mathcal{A} and \mathcal{B} do not sensitively depend on the definition of stable state A and B as long as it is reasonable. Of course, the stable regions should not overlap, each trajectory between the stable states must be a true rare event for the reaction we are interested in. In addition, the probability that after this event the reverse reaction occurs shortly thereafter must be as unlikely as an entirely new event. In other words, the system must be committed to the stable states. Therefore, a reasonable definition of A and B requires that they should lie completely inside the basin of attraction of the respective two states, just as in case of TPS. Special care has to be taken with this condition for processes which show many recrossings between state A and B before settling down. Such processes can occur in solution or in dilute gasses. For instance, for organic reactions in aqueous solution, a rare specific hydrogen bonded network can lower the bond-breaking barrier and initiate the reaction. If the lifetime of those rare solvation structures is high, a sudden reverse reaction can occur as the barrier for the backward reaction is also lowered by the same amount [110,111]. A similar phenomenon can happen in dilute gasses for which rare spontaneous fluctuations in the kinetic energy are the main driving force. A particle moving from one state to another due to a very high kinetic energy as result of sequence of collisions can cross the potential energy barrier several times before it will dissipate its energy by a new collision and relax into one of the stable states (see e.g. Refs. [112,113]). These problems can in principle be solved by an adequate choice of the stable state definitions. For instance, the definition can depend explicitly on the presence of certain hydrogen bonds or on kinetic energy terms.

3.1.2 Rate constant

In sec. 1.4 we showed how TST can be used to compute the mean residence time in a state. However, a better definition of the rate constant is based on the mean first passage time (see appendix chapter A). The introduction of overall states allows the transformation of the TST computation of the former into the TIS computation of the latter definition.

The basic idea is depicted in Fig. 3.2. Combining expression (1.29) with Eq. (1.30) and (1.27), we can write for the mean residence time in A

$$\begin{aligned} t_A^{mr} &= \frac{\lim_{\mathcal{T} \rightarrow \infty} \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} h_A(t) dt}{\lim_{\mathcal{T} \rightarrow \infty} \frac{1}{2\mathcal{T}} \int_0^{\mathcal{T}} |\dot{h}_A(t)| dt} \\ &= \frac{\lim_{\mathcal{T} \rightarrow \infty} \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} h_A(t) dt}{\lim_{\mathcal{T} \rightarrow \infty} \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} [-\dot{h}_A(t)\theta(-\dot{h}_A(t))] dt} \end{aligned} \quad (3.4)$$

The second expression counts only the times when the trajectory leaves A , producing a negative delta-function in $\dot{h}_A(t)$ (see Fig. 1.5). We could have chosen the positive contributions, but in this way it is easier to generalize. Next, we substitute h_A with

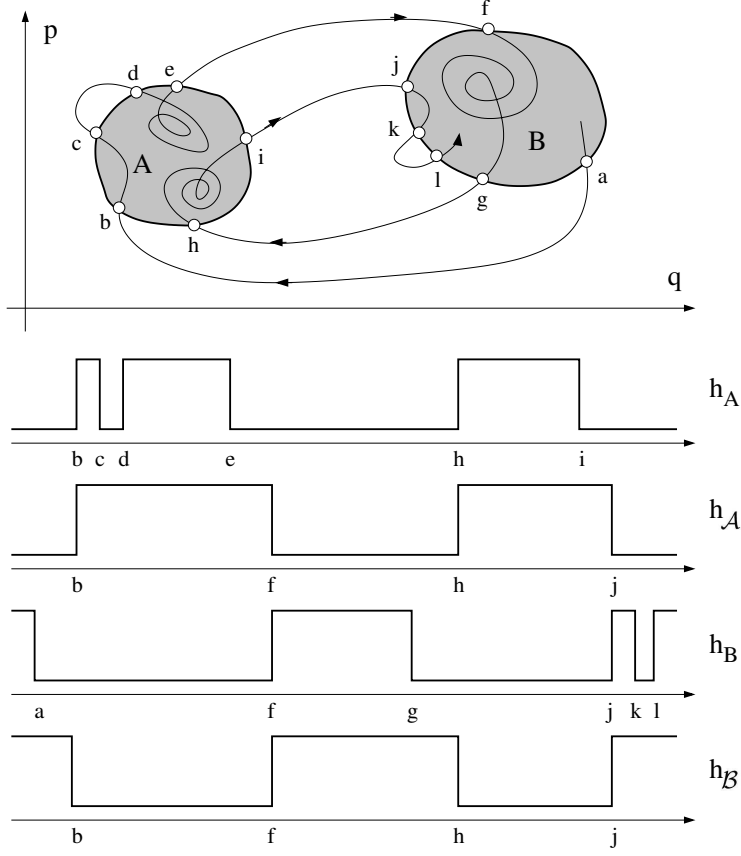


Figure 3.2: Representation of phase space and a long trajectory with 2 transitions $A \rightarrow B$ and two $B \rightarrow A$. In the bottom we show the time behavior of h_A, h_B, h_A, h_B . Note that $h_A(t) + h_B(t) \simeq 1$, because it misses the transition parts in the buffer region, while $h_A(t) + h_B(t) = 1$ for all t .

h_A in the last expression, thus replacing state A with overall state \mathcal{A} .

$$\begin{aligned}
 t_A^{mfp} &= \frac{\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T h_A(t) dt}{\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T [-\dot{h}_A(t) \theta(-\dot{h}_A(t))] dt} \\
 &= \frac{\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T h_A(t) dt}{\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T [\delta(\lambda_{n_I} - \lambda_t) (+\dot{\lambda}_t) h_{0,n_I}^b(x_t)] dt} \quad (3.5)
 \end{aligned}$$

The derivative of $\dot{h}_A(t)$ produces a delta-function weighted with the velocity, cf Eqs. (1.30), (1.32). The negative delta appears, however, only at the moments when the trajectory hits for the first time B , i.e. reaches B coming directly from A , thus giving rise to the term $h_{0,n_I}^b(x_t)$. As a result the mean residence time becomes the mean

first passage time. Using ergodicity and the definition (1.15) for the rate constant, we obtain the TIS rate expression

$$k_{AB} = (t_A^{mfp})^{-1} = \frac{\langle h_{0,n_I}^b(x) \dot{\lambda}(x) \delta(\lambda(x) - \lambda_{n_I}) \rangle}{\langle h_A \rangle} \quad (3.6)$$

In principle, this formula is an operational way to compute the rate: start an infinite long trajectory and count the number of effective positive crossings, i.e. the crossings of ∂B when coming directly from A . Naturally, this is not doable in practice because the transition is a rare event and we need a way to enhance the probability of this event. We will deal with this in the next section.

Let us conclude showing the similarity of Eq. (3.6) with other expressions from traditional TST-BC. First, noting that $h_{0,n_I}^b(x) = 0$ if $\dot{\lambda}(x) < 0$, we can insert $\theta(\dot{\lambda}(x))$ in Eq. (3.6), and using (3.3), write

$$k_{AB} = \frac{\langle h_A(x) \dot{\lambda}(x) \theta(\dot{\lambda}(x)) \delta(\lambda(x) - \lambda_{n_I}) \rangle}{\langle h_A \rangle} \quad (3.7)$$

which resembles the TST expression Eq. (1.33). Then, noting that the negative delta's in (3.5) appear at points when just after $h_B = 1$, we can write $-\dot{h}_A(x) \theta(-\dot{h}_A(x)) = -\dot{h}_A(x) h_B(0^+)$. Using time translation invariance to bring the time derivative on the second term, see Eq. (1.41), we can write

$$\begin{aligned} k_{AB} &= \frac{\langle h_A(0) \dot{h}_B(0^+) \rangle}{\langle h_A \rangle} \\ &= \frac{d}{dt} \frac{\langle h_A(0) h_B(t) \rangle}{\langle h_A \rangle} \Big|_{0^+} \\ &\equiv \dot{C}(0^+) \end{aligned} \quad (3.8)$$

where in the last line we define the TIS correlation function

$$C(t) = \frac{\langle h_A(0) h_B(t) \rangle}{\langle h_A \rangle} \quad (3.9)$$

which resembles the BC expression Eq. (1.38). However, this function shows a linear regime for $0 < t < \tau_{stable}$, instead of only for $\tau_{trans} < t < \tau_{stable}$ like in BC theory. Eventually $C(t)$ will converge to $\langle h_B \rangle \simeq \langle h_B \rangle$ as in the traditional case. But to compute the rate we can take the derivative immediately at time $t = 0^+$ and obtain the value of the plateau.

3.1.3 Effective positive flux

In order to make (3.6) computationally manageable, we have to introduce two fundamental quantities. The first is the effective positive flux. Given two interfaces i, j define

$$\phi_{ij}(x) \equiv h_{j,i}^b(x) |\dot{\lambda}(x)| \delta(\lambda(x) - \lambda_i) \quad (3.10)$$

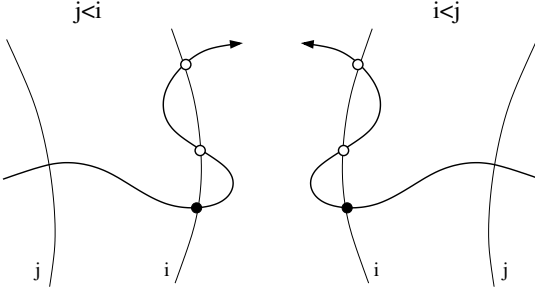


Figure 3.3: Effective positive flux. For each trajectory only one point (full circle) contributes to the flux across i , the first one coming directly from j . Directly means with no recrossing of i . The other two recrossings (open circles) we show would cancel each other in the flux and we do not count them at all. Case $j < i$, $\phi_{ij}(x) = +h_{j,i}^b(x)\dot{\lambda}(x)\delta(\lambda(x) - \lambda_i)$. Case $i < j$ $\phi_{ij}(x) = -h_{j,i}^b(x)\dot{\lambda}(x)\delta(\lambda(x) - \lambda_i)$. By definition, in any case the flux is positive.

With reference to Fig. 3.3, it can be explicitly written

$$\begin{aligned}
 \phi_{ij}(x) &= +h_{j,i}^b(x)\dot{\lambda}(x)\delta(\lambda(x) - \lambda_i) \quad \text{if } j < i \\
 &= -h_{j,i}^b(x)\dot{\lambda}(x)\delta(\lambda(x) - \lambda_i) \quad \text{if } i < j \\
 &= h_{j,i}^b(x) \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \theta(\Delta t - t_i^f(x))
 \end{aligned} \tag{3.11}$$

The last equality is an operational definition. An MD trajectory might cross interface λ_i , but consists of discrete time slices that are never exactly on the surface. However, $\phi_{ij}(x)$ can still be defined for the discrete MD set of time-slices by taking Δt equal to the molecular time-step. In words, $\phi_{ij}(x)$ equals $1/\Delta t$ if the forward trajectory crosses λ_i in one single Δt time-step and the backward trajectory crosses λ_j before λ_i . Otherwise $\phi_{ij}(x)$ vanishes. Note that independently of the relative position of i and j , the flux is always positive. With this definition Eq. (3.6) is immediately rewritten as

$$k_{AB} = \langle \phi_{n_I,0} \rangle / \langle h_A \rangle \tag{3.12}$$

The second step is to define a conditional *crossing probability* that depends on the location of any four interfaces i, j, l, m :

$$P_m^{(l|i)} \equiv \langle \phi_{ij}(x) h_{l,m}^f(x) \rangle / \langle \phi_{ij}(x) \rangle = \langle h_{l,m}^f(x) \rangle_{\phi_{ij}} \tag{3.13}$$

where we introduced the weighted average $\langle f(x) \rangle_w = \langle f(x)w(x) \rangle / \langle w(x) \rangle$. In words, this is the probability for the system to reach interface l before m under the condition that it crosses at $t = 0$ interface i , while coming directly from interface j in the past. See Fig. 3.4 for an illustration. Alternatively stated, in the ensemble ϕ_{ij} of trajectories crossing i and coming directly from j , $P_m^{(l|i)}$ is the probability of reaching l before m .

The link between the flux (3.10) and the probability (3.13) is a fundamental relation between fluxes at different interfaces. For three interfaces $i < j < k$ it holds

$$\langle \phi_{ki} \rangle = P_i^{(k|j)} \langle \phi_{ji} \rangle \tag{3.14}$$

In words, the direct flux from i through k is given by the direct flux from i through $j < k$ times the conditional probability of reaching k before i after crossing j while coming directly from i (see Fig. 3.5 (a)). A proof of this relation is given in appendix D. Consequently, we can derive the following probabilistic relation for four interfaces $i < j < k < l$ (see Fig. 3.5 (b) for a graphical explanation):

$$P_{(i|i)}^{(l|j)} = P_{(i|i)}^{(l|k)} P_{(i|i)}^{(k|j)} \quad (3.15)$$

Using this notation the left hand-side can be seen as a matrix contraction of the right-hand side. We stress that both Eq. (3.14) and (3.15) are exact and there is no underlying Markovian assumption. The probabilities retain the dependence on the past through the index i .

Now we have all the ingredients to write the TIS rate constant in terms that can be determined in a computer simulation.

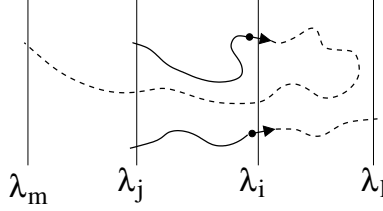


Figure 3.4: The conditional crossing probability $P_{(i|i)}^{(l|j)}$ for a certain configuration of interfaces $\lambda_i, \lambda_j, \lambda_l$, and λ_m . The condition $|_j^i$ is depicted by the arrow and the solid line for two phase points (the dots): from this phase point one should cross λ_i in one single Δt time-step in the forward direction, and, besides, its backward trajectory should cross λ_j before λ_i . Two possible forward trajectories are given by the dashed line. The upper crosses λ_m before λ_l , the lower crosses λ_l as first. The fraction whose forward trajectories behave like the last case equals $P_{(i|i)}^{(l|j)}$.

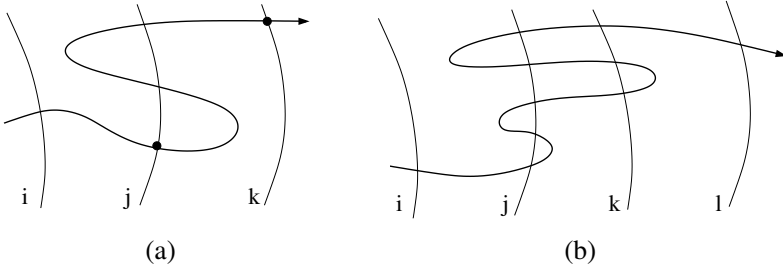


Figure 3.5: (a) Flux theorem. The flux through k can be written as the flux through $j < k$ times the probability $j \rightarrow k$. This is true if the trajectory is coming directly from i , i.e. it must not recross j in the meantime. (b) Probability theorem. The probability $j \rightarrow l$ can be factorized as Prob. $j \rightarrow k$ times Prob. $k \rightarrow l$. This is true if the trajectory is coming directly from i , i.e. it must not recross j in the meantime.

3.1.4 TIS rate expression

Given the probabilities in Eq. (3.13), the cases $m = j = 0$ and $m = j = n_I$ are of special interest for TIS and will be annotated as follows

$$P_A(j|i) \equiv P_{(0|0)}^{(j|i)}, \quad P_B(j|i) \equiv P_{(n_I|n_I)}^{(j|i)} \quad (3.16)$$

Making use of (3.14) we can write

$$\langle \phi_{n_I,0} \rangle = \langle \phi_{1,0} \rangle P_A(n_I|1), \quad (3.17)$$

thus relating the flux through ∂B to the flux through an interface closer to A . Using then (3.15) we can write

$$P_A(n_I|1) = \prod_{i=1}^{n_I-1} P_A(i+1|i) \quad (3.18)$$

which factorizes the global (very small) probability $P_A(n_I|1)$ into local (bigger) probabilities between neighboring interfaces. As a result we can rewrite (3.12) as

$$k_{AB} = \frac{\langle \phi_{1,0} \rangle}{\langle h_A \rangle} P_A(n_I|1) = \frac{\langle \phi_{1,0} \rangle}{\langle h_A \rangle} \prod_{i=1}^{n_I-1} P_A(i+1|i) \quad (3.19a)$$

$$k_{BA} = \frac{\langle \phi_{n_I-1,n_I} \rangle}{\langle h_B \rangle} P_B(0|n_I-1) = \frac{\langle \phi_{n_I-1,n_I} \rangle}{\langle h_B \rangle} \prod_{i=1}^{n_I-1} P_B(i-1|i) \quad (3.19b)$$

where for completeness we have shown the expression for the reversed rate k_{BA} as well. In (3.19a) the first factor $\frac{\langle \phi_{1,0} \rangle}{\langle h_A \rangle}$ is a flux and can be calculated by straightforward MD if interface 1 is close enough to A . The second factor is the crossing probability and is calculated with an algorithm based on the TPS shooting move, sec. 2.2.1. Just as in the previous section, we remark that even though the notation might appear Markovian, Eqs. (3.19a), (3.19b) do not involve a Markovian assumption. The subscript A is there to indicate that the probabilities are also dependent on the past history of the path, all the way from where it left A .

3.2 TIS algorithm

We describe here how to implement formula (3.19a) in a computer simulation, following [20].

3.2.1 The flux calculation

The flux factor $\frac{\langle \phi_{1,0} \rangle}{\langle h_A \rangle}$ is the effective flux through λ_1 of the trajectories coming from λ_0 , i.e. from A . If λ_1 is chosen close enough to A the flux factor can be obtained by straightforward MD initialized in A and counting the positive crossings through λ_1 during the simulation run

$$\frac{\langle \phi_{1,0} \rangle}{\langle h_A \rangle} = \frac{1}{\Delta t} \frac{N_c^+}{N_{\text{MD}}} \quad (3.20)$$

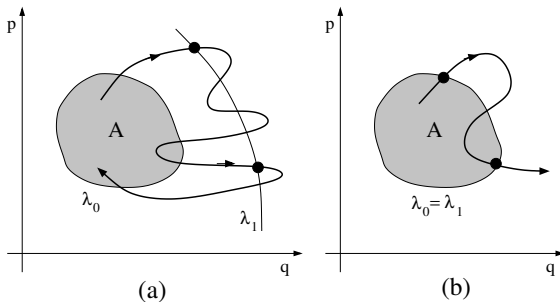


Figure 3.6: Flux calculation. (a) $\lambda_1 \neq \lambda_0$. The number of effective in-out crossings (full circles) is $N_c^+ = 2$, not all the in-out crossings with 1 must be counted. (b) $\lambda_1 = \lambda_0 + \epsilon$. All the in-out crossings are effective crossings.

with Δt the MD time step and N_{MD} the number of MD steps, so that the denominator is the total length of the MD trajectory. Then N_c^+ the number of counted effective positive crossings, which means that one only has to count the crossings when coming directly from A . An illustration is given in Fig. 3.6 (a). Actually, the flux factor is most conveniently computed with the first two interfaces identical. Although $\frac{\langle \phi_{1,0} \rangle}{\langle h_A \rangle}$ is not well defined for $\lambda_1 = \lambda_0$, we can think that $\lambda_1 = \lambda_0 + \epsilon$ in the limit $\epsilon \rightarrow 0$. In this way, the effective positive flux will be equal to the simple positive flux through λ_1 . Trajectories cannot recross without re-entering A , hence, all crossings are counted. See also Fig. 3.6 (b). Similarly, for the reverse rate k_{BA} we can set $\lambda_{n_I-1} = \lambda_{n_I} - \epsilon$.

To calculate the rate at constant temperature instead of constant energy, one can apply a Nosé-Hoover [114, 115, 116, 76] or Andersen [77] thermostat. However, one should be aware that these thermostats do give the correct canonical distribution at a given temperature, but modify the dynamics in an unphysical way. Hence, static averages $\langle A(x) \rangle$ will be correct, but time correlation functions $\langle A(x_0)B(x_t) \rangle$ most likely not. As $N_c^+ \sim \langle \theta(\lambda_1 - \lambda(x_0))\theta(\lambda(x_{\Delta t}) - \lambda_1) \rangle$ is actually a correlation function over a very short time, this effect will be small. However, if necessary one can easily correct for this by explicitly counting only phase points x that in absence of the thermostat will cross λ_1 in one Δt time-step. Applying this correction is computationally cheap as it does not require any additional force calculations. Alternatively, one can perform an NVT simulation in the same way as TPS does an NVT path sampling, see 2.2.1. Generate initial conditions from a canonical distribution using MC and compute the flux for an NVE run for each configuration. The canonical average of microcanonical flux results in the canonical flux.

3.2.2 The path sampling

To calculate the conditional probabilities in TIS we use a path sampling algorithm, see sec. 2.2.

For the MC sampling of trajectory space, we borrow from TPS the shooting move and the time-reversal move. Since in TIS the path length is variable, the acceptance criterion for the shooting move has to be slightly modified. The applicability of the path-reversal move is quite limited in TPS because B acts as an attractor and even if the paths leave B , it is most likely that they stay in the neighborhood of B . In TIS it is a very useful tool. The probability of reaching B is computed as the ratio

of the paths $A \rightarrow B$ to the total of the paths starting in A which also includes paths $A \rightarrow A$. In the sampling of the latter ones time-reversal leaves the $A \rightarrow A$ property unchanged, and at no computational cost we get another path. Shifting moves that enhanced statistics in TPS are not needed and even useless in TIS. Paths never spend time in the stable states, but go from the surface ∂A to the surface ∂B , with their length consequently tuned. The shifting move cannot be applied. Once the paths reach the boundaries they are supposed to relax into the stable basin. There is no need to check their relaxed behavior as far as the transition is concerned. The portions of TPS paths spent in the stable states are hence a waste of computational time.

The quantity of interest in TIS is the crossing probability $P_A(i+1|i)$, or $P_B(i-1|i)$ for the reverse rate constant k_{BA} . Define the λ_i ensemble as all paths that starts in A (at λ_0), cross the interface λ_i at least once, and finally end by either crossing λ_0 or λ_{i+1} . Suppose to have an initial path belonging to the ensemble. The phase space point x_0 is then defined as the first crossing point of this path with interface λ_i . Since it is uniquely defined, time shifting of the path is pointless. It is convenient to use a discrete time index $\tau = \text{int}(t/\Delta t)$, and let $\tau^b \equiv \text{int}(t_0^b(x_0)/\Delta t)$ and $\tau^f \equiv \text{int}(\min[t_0^f(x_0), t_{i+1}^f(x_0)]/\Delta t)$ be the backward and forward terminal time slice indices, respectively. Including x_0 , the initial path then consists of $N^{(o)} = \tau^b + \tau^f + 1$ time slices.

Similar to in sec. 2.2.4 we use a probability $0 < \alpha < 1$ to choose between the two possible sampling algorithms. In addition we fix a Gaussian width σ_w to be used in the momentum displacement in the shooting move. Then the TIS path sampling algorithm reads:

Main loop

1. Take a uniform random number y_1 in the interval $[0 : 1]$. If $y_1 < \alpha$ perform a time-reversal move. Otherwise, perform a shooting move.
2. If the trial path generated by either the time-reversal or shooting move is a proper path in the λ_i ensemble accept the move and replace the old path by the new one, otherwise keep the old path.
3. Update averages on the path
4. Repeat from step 1.

Time-reversal move

1. If the current path ends at λ_{i+1} reject the time-reversal move and return to the main loop.
2. If the current path starts and ends at λ_0 , reverse the momenta and the reading order of time-slices. On this reverse path, x_0 is the new first crossing point with λ_i . Return to the main loop.

Shooting move

1. On the current path with length $N^{(o)}$ choose a random time slice τ' , with $-\tau^b \leq \tau' \leq \tau^f$.
2. Change all momenta of the particles at time-slice τ by adding small randomized displacements $\delta p = \delta w \sqrt{m}$ with δw taken from a Gaussian distribution with width σ_w and m the mass of the particle.
3. In case of constant temperature (NVT) path sampling simulations accept the new momenta with a probability

$$\min \left[1, \exp \left(\beta (E(x_{\tau' \Delta t}^{(o)}) - E(x_{\tau' \Delta t}^{(n)})) \right) \right].$$

Here, $E(x)$ is the total energy of the system at phase space point x . In case of constant energy (NVE) simulations, in which possibly also total linear- or angular momentum should be conserved, rescale all the momenta of the system according to the procedure described in sec 2.2.1 and accept or reject the new rescaled momenta.

If the new momenta are accepted continue with step 4, else reject the whole shooting move and return to the main loop.

4. Take a uniform random number y_2 in the interval $[0 : 1]$ and determine a maximum allowed path length for the trial move by:

$$N_{\max}^{(n)} = \text{int}(N^{(o)}/y_2).$$

5. Integrate equations of motion backward in time by reversing the momenta at time slice τ' , until reaching either λ_0 , λ_{i+1} or exceeding the maximum path length $N_{\max}^{(n)}$. If the backward trajectory did not reach λ_0 reject and go back the main loop. Otherwise continue with step 6.
6. Integrate from time slice τ' forward until reaching either λ_0 , λ_{i+1} or exceeding the maximum path length $N_{\max}^{(n)}$. Reject and go back to the main loop if the maximum path length is exceeded or if the entire trial path has no crossing with interface λ_i . Otherwise continue with the next step.
7. Accept the new path, reassign x_0 to be the first crossing point with λ_i and return to the main loop.

3.2.3 Considerations

Here, we make some remarks that help to better understand and apply the algorithm, and also give tips we found useful from personal experience. An algorithm in pseudo C, based on this section, can be found in appendix K.

Zone system

For a λ_i ensemble, the probability $P_A(i+1|i)$ follows from the ratio of sampled paths that connect λ_0 with λ_{i+1} to the total number of sampled paths. In analogy to umbrella sampling (see sec. 2.3.1), to enhance statistics we would like to assign to each λ_i ensemble a corresponding TIS window. In each window, between the interface positions λ_i, λ_{i+1} , we define a finer grid of sub-interfaces to obtain the crossing probability function $P_A(\lambda|i)$, as a function of a continuous¹ parameter λ . Because of the probability definition, the function starts from $P_A(i|i) = 1$ and is monotonically decreasing to the value $P_A(i+1|i)$. The rematching to $P_A(\lambda|1)$ is done by simply multiplying the functions in each window by constant factors. Naturally, techniques commonly applied in umbrella sampling such as polynomial fits and overlapping windows between two successive ensemble averages can also be employed here.

A practical way to do this is to use a zone system. Starting with zone 0 identical to state A , label the phase space region between subsequent sub-interfaces with consecutive integers. Given a window λ_i, λ_{i+1} , there will be a first zone z_f for $\lambda(x) < \lambda_i$ and a last zone z_l for $\lambda(x) > \lambda_{i+1}$. Each path is a set of discretized time-slices and each time-slice belongs to a zone. Paths must start in zone 0 and show a crossing $z_f \rightarrow z_f + 1$. Then they evolve either to z_l or back to 0. If λ_{max} is the highest value of the zone number, then it must be $z_f + 1 \leq \lambda_{max} \leq z_l$. To construct $P_A(\lambda|i)$ fill the bins of a histogram from z_f up to $\lambda_{max} - 1$. For instance, if a path goes all the way to z_l , you fill up to $z_l - 1$. When going to the next window the old $z_l - 1$ zone will be the new z_f zone. Keeping the consecutive numeration one builds up histograms that can be easily rematched afterwards. The procedure is graphically explained in Fig. 3.7. Note that the built histograms are cumulative histograms. In resemblance with a calculation of an integral they are expected to be less noisy. The zone system is particular useful when interfaces are defined using more than one order parameter λ , as they make the bookkeeping in the programming more manageable.

Path representation

Paths can be stored in memory as arrays of timeslices. Each timeslice contains the positions, velocities, zone number, and whatever one thinks necessary to save for later use. Since for a large system this can require lot of memory, it is more convenient to save not every time step but only a subset of the whole trajectory. More care must be taken in case of very large memory requirements, typical of Car-Parrinello simulations [35].

The shooting move can be accomplished with the use of temporary scratch space where the new path is created while keeping the old one in another part of memory. If the new path is rejected you keep the old path as the current path. If the new path is accepted, you can swap the current and scratch path spaces. In the case of a time reversal move, the scratch space is not needed. After inverting all the velocities one can just read the path in a reversed way, starting from the end. All these memory operations are performed better using of pointers. Since in TIS paths have variable

¹ We will use in the following both the continuous notation $P_A(\lambda|i)$ and the discrete one $P_A(j|i)$ to identify an interface. We will be more explicit in case of confusion.

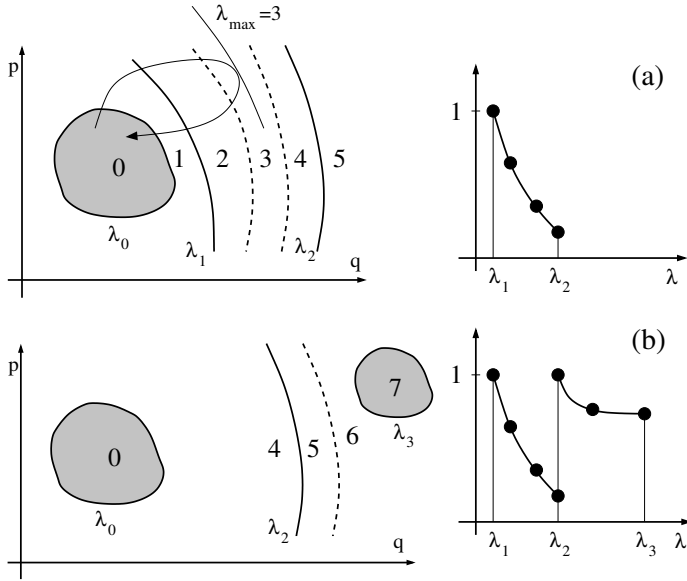


Figure 3.7: TIS path sampling and zone system. 4 Interfaces, 8 zones, two windows (a),(b). The function $P_A(\lambda|1)$ is obtained by rematching.

length, the required memory can be dynamically reallocated. However, there is no way of predicting the maximum amount of memory needed and trial runs might have to be used to avoid memory overflow.

Choice of parameters

Time reversal moves do not require any force calculations. On the other hand, two subsequent time reversals will just result in the same path. Therefore, we usually take $\alpha = 0.5$ giving shooting and time reversal move an equal probability. However, for windows close to B the number of paths returning to A decreases and the rejection rate of the time reversal move increases. As paths get also longer, the path sampling becomes more computationally demanding.

Similar reasoning is applied to the choice of σ_w . If σ_w is large, many trial moves will fail to create a proper path. On the other hand, a too small value of σ_w will result in slow decorrelation between accepted paths. Practice has shown that an optimal value of σ_w is established when approximately 40% of the paths is accepted, see sec 2.2.1. This will usually imply that σ_w will be larger for the interfaces λ_i close to A than the ones closer to B .

Notes on the TIS shooting move

In principle, NVT path simulations do not require rescaling of the momenta. The integration of the equations of motion at step 5 and 6 of the shooting move is performed by constant energy MD simulations without using a thermostat, to describe

the actual dynamics as realistic as possible. The temperature only appears at the acceptance criterion at step 3. Momentum rescaling is not needed and δp can be taken from any symmetric distribution. In the TIS algorithm we go from one phase point $x_0^{(o)}$ to a new one $x_0^{(n)}$ by means of many MD steps. Therefore, it has a strong similarity with hybrid MC [117].

In general however, generalized Hamiltonian dynamics can also be used, and proper rescaling must be applied, according to the detailed balance rules explained in sec. 2.2.1. The mass weighted momenta change at step 2 of the shooting algorithm is also chosen such that the velocity rescaling at step 3 maintains detailed balance.

The maximum allowed path length $N_{\max}^{(n)}$ in step 4 is introduced to maintain detailed balance when sampling paths of different length and to avoid having to reject very long trial paths afterward.

Initial path

To start a simulation in window i one needs an initial path that starts in A (at λ_0), crosses the interface λ_i at least once, and finally ends by either crossing λ_0 or λ_{i+1} . In TIS we can take simply a successful path from the previous λ_{i-1} interface ensemble that reached λ_i , and we can complete its evolution until reaching either A or λ_{i+1} . For the first window, just use the flux calculation and extract the part of the MD trajectory that showed a crossing with λ_1 .

Note that in contrast to TPS, TIS is an iterative procedure that slowly builds up all pathways starting from A . Since you expect to find a transition there should always, sooner or later, be a path that reaches the next interface.

Defining the interfaces

As in TPS, the order parameter λ does not have to correspond to a reaction coordinate that captures the essence of the reaction mechanism. The only requirement is that λ can distinguish between the two basins of attraction.

In TIS this occurs via the two outer interfaces λ_0 and λ_{n_I} that define state A and B . The definitions of A and B are more strict than in TPS and we outlined them in sec. 3.1.1. The boundaries λ_0 and λ_{n_I} should be defined such that

- each trajectory between the stable states is a rare event for the reaction we are interested in
- the probability that after this event the reverse reaction occurs shortly thereafter must be as unlikely as an entirely new event. In other words, a trajectory that starts in A and ends in B is allowed to leave region B shortly thereafter, but the chance that it re-enters region A in a short time must be highly unlikely

Sometimes it is not sufficient for a proper definition of the boundaries λ_0 and λ_{n_I} to only use configuration space, and an additional kinetic energy constraint must be introduced to ensure the stability of state A and B , see sec. 3.3.4. After the rematching the overall function $P_A(\lambda|1)$ is expected to show a plateau, expressing the commitment to stable state B . Paths that cross a certain interface always end up in B . The beginning of this plateau could be chosen a posteriori as the last interface

defining B . However, it is good to actually compute the plateau in order to check the consistency of the calculations.

The intermediate interfaces can be chosen freely and should be placed to optimize the efficiency. This is, of course, system dependent, but reasonable estimates can be made a priori. We defer the calculations to sec. 6.1, and we give here the rule of thumb that the interface separation should be chosen so that $P_A(i+1|i) \approx 0.2$ for all i . In the case of diffusive systems it becomes $\simeq 0.47$ (see also sec. 6.1).

3.3 Numerical results

We test here the TIS theory and algorithm and we compare it with TPS. The system is a simple diatomic bistable molecule immersed in a fluid of purely repulsive particles. It has previously been used in illustrating TPS rate constant calculations [79] and is therefore a good example for a comparison between the two methods. This section is based on [21].

3.3.1 The model

The system consists of N particles in 2 dimensions with interactions given by a pairwise Lennard-Jones (LJ) potential truncated and shifted at the minimum, often referred to as the Weeks-Chandler-Andersen (WCA) potential [118]

$$U_{WCA}(r) = \begin{cases} 4\epsilon[(r/\sigma)^{-12} - (r/\sigma)^{-6}] + \epsilon & \text{if } r \leq r_0 \\ 0 & \text{if } r > r_0, \end{cases} \quad (3.21)$$

where r is the interatomic distance, and $r_0 \equiv 2^{1/6}\sigma$. Throughout this section reduced units are used so that ϵ and σ , respectively the LJ energy and length parameters, as well as the mass of the particles are equal to unity. The LJ unit of time $(m\sigma^2/\epsilon)^{1/2}$ is therefore also unity. In addition, two of the N particles are interacting through a double well potential

$$U_{dw}(r_d) = h \left[1 - \frac{(r_d - r_0 - w)^2}{w^2} \right]^2. \quad (3.22)$$

where r_d is now the interdimer distance. This function has two minima separated by a barrier of height h corresponding to the two stable states of the molecule: a compact state for $r_d = r_0$ and extended state for $r_d = r_0 + 2w$. For a high enough barrier, transitions between the states become rare and the rate constant is well defined. Hence, this system provides a useful test case for the TIS algorithm and also to compare it with TPS.

The system is simulated at a constant energy E in a simulation square box with periodic boundary conditions. The total linear momentum is conserved and is set zero for all trajectories. The equations of motion are integrated using the velocity Verlet algorithm with a time step $\Delta t = 0.002$. As in Ref. [79] we focus here on the computation of the rate constant for the isomerization reaction of the dimer from the compact state to the extended state. In the following section we describe general

simulation details. In Section 3.3.3 we discuss the results for a system with a high enough barrier to avoid recrossings. Subsequently, we present results for a system with a low energy barrier, which shows recrossings, and we discuss the consequences for TPS and TIS.

3.3.2 Methodology

The TPS rate constant calculation evaluates the two factors in Eq. (2.43) separately as explained in sec. 2.3.2. In all our simulations we set the percentages for shooting, shifting and diffusion to 5%, 10% and 85%, respectively. The parameters involved are always gaged such that the acceptance ratio is around 40% for shooting and shifting moves, ensuring an optimum efficiency of the sampling [79]. The TIS method involves a direct determination of the flux, as explained in sec. 3.2.1, and the calculation of the crossing probability functions $P(i|i-1)$ between a series of successive interfaces as given by Eq. (3.19a), sec. 3.2.2. As in the TPS calculation we adjusted the momentum displacement for the shooting move to give an acceptance of about 40%.

Many parameters are involved in the two methods and to compare the relative efficiency we measured the CPU-time required for an arbitrary fixed error of 2.5% for each step in both the TPS and TIS calculations under the same computational conditions (1Ghz AMD Athlon). In both methods the final rate constant consists of a product of factors which have to be calculated independently. For TIS it is evident from (3.19a). Instead Eq. (2.43) for TPS requires a bit of explanation. The umbrella-sampling factor $C(t')$ comes from a rematch of windows. It can be shown that the outcome can be rewritten as a product of probabilities in each window, and that this window probability is just the ratio of the value of the histogram at the left window boundary to the value of the histogram at the right window boundary. So both TIS and TPS rate constants are effectively a product of factors. For each factor we performed N_{bl} simulation blocks of N_{cy} Monte Carlo cycles and adjusted N_{cy} such that after N_{bl} block averages the relative standard deviation of each term in Eq. (2.43) and (3.19a) was 2.5%. The total CPU-time is given by summing the individual 2.5% error CPU-times for each factor. The final error in the rate constants is obtained by the standard propagation rules using all simulation results (i.e. not only the ones for the 2.5% error CPU time calculation).

3.3.3 System with High Energy Barrier

This system consists of total number of particles $N = 25$, with a total energy $E = 25$. The square simulation box was adjusted to give a number density of 0.7. The barrier height was $h = 15$ and the width-parameter $w = 0.5$, so that the minima of $V_{dw}(r_d)$ were located at $r_d \simeq 1.12$ and $r_d \simeq 2.12$ while the top of the barrier was at $r_d \simeq 1.62$ (see also Fig. 3.10). In the TPS rate calculation we defined stable states A and B as $r_d < r_d^A = 1.5$ and $r_d > r_d^B = 1.74$, respectively. We computed the correlation function $\langle h_B(t) \rangle_{AB}^*$ using TPS with a fixed path length $\mathcal{T} = 2.0$. The correlation function is shown in Fig. 3.8 together with its time derivative, the reactive flux. The latter function clearly displays a plateau. Next, we chose four different $t' = 0.1, 0.3, 1.0, 2.0$ and performed umbrella sampling simulations using 8 windows to

Figure 3.8: TPS correlation function $\langle h_B(t) \rangle_{AB}^*$ (top) and its time derivative (bottom) for the system with high energy barrier. The error is comparable to line thickness.

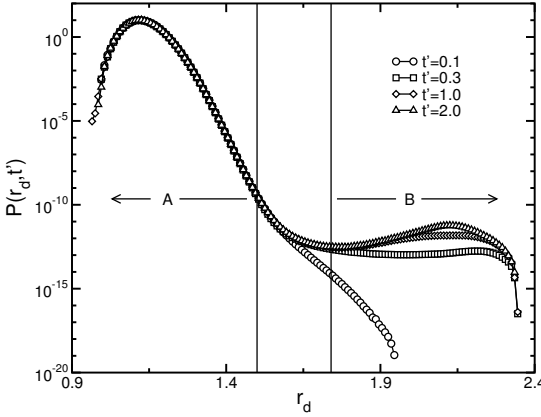
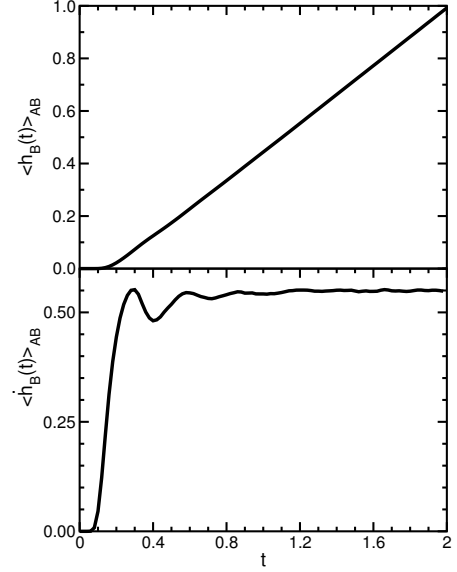


Figure 3.9: TPS probability distributions $P_A(r_d, t')$ for four $t' = 0.1, 0.3, 1.0, 2.0$ for the high energy barrier. The probability $P_A(r_d, t')$ is the chance that a path of length t' and starting in A will have the end point conformation with a diatomic distance r_d . The graph is the result of the matching of eight window calculations. These eight windows are defined as $r_d < 1.19$, $1.18 < r_d < 1.28$, $1.27 < r_d < 1.35$, $1.34 < r_d < 1.40$, $1.39 < r_d < 1.47$, $1.46 < r_d < 1.54$, $1.53 < r_d < 1.75$, $r > 1.74$. The errors on the histogram points are within the symbol size.

calculate $C(t')$. In each window we measured the probability to find the path's end point $r_d(x(t'))$ at a certain value of r_d . These probability histograms were rematched and normalized. The final probability functions are shown in Fig. 3.9. Integration of the area under the histogram belonging to region B leads to $C(t')$ and finally to the rate constant. In Table 3.1 we give the values of the different contributions to the rate constant given by Eq. (2.43), together with the rate constant. We report the average relative computation time needed to reach the 2.5% error (see Sec. 3.3.2) in Table 3.2.

For the TIS calculations we use the same order parameter r_d and the same definition for region B, i.e. interface λ_B is set at $r_d = 1.74$. Stable state A was defined by setting

TPS			
t'	$\frac{\langle h_B(T) \rangle_{AB}^*}{\langle h_B(t') \rangle_{AB}^*}$	$C(t')/10^{-13}$	$k_{AB}/10^{-13}$
0.1	3300 ± 100	0.0018 ± 0.0001	6.0 ± 0.5
0.3	7.54 ± 0.03	0.76 ± 0.02	5.8 ± 0.1
1.0	1.236 ± 0.005	4.8 ± 0.3	5.9 ± 0.4
2.0	0.553 ± 0.002	11.4 ± 0.9	6.3 ± 0.5

TIS			
$\langle \phi_{0,1} \rangle / \langle h_A \rangle$	$P_A(B 1)/10^{-13}$	$k_{AB}/10^{-13}$	
0.1196 ± 0.0005	49 ± 1	5.9 ± 0.2	

Table 3.1: Comparison of rate constants for the high energy barrier, computed with TPS at different t' and TIS. Contributing factors from Eq. (2.43) and Eq. (3.19a) are also given. The forward rate constant for the isomerization corresponds to an average transition time $k_{AB}^{-1} = (3.6 \pm 0.1)s$ in real units for argon, which is indeed many orders of magnitude beyond the MD time-step $\sim 4fs$.

TPS										
t'	$\frac{\langle h_B(T) \rangle_{AB}^*}{\langle h_B(t') \rangle_{AB}^*}$	W1	W2	W3	W4	W5	W6	W7	W8	Total
0.1	11.0	0.01	0.05	0.1	0.04	0.23	0.27	1.3	0.01	13.01
0.3	0.2	0.01	0.14	0.28	0.13	0.58	0.43	0.19	0.02	1.98
1.0	0.1	1.7	1.7	0.9	0.6	3.0	2.6	6.4	0.2	17.2
2.0	0.1	0.03	1.8	4.5	4.4	15.3	8.0	20.3	0.6	55.03

TIS							
$\langle \phi_{0,1} \rangle / \langle h_A \rangle$	Int λ_1	Int λ_2	Int λ_3	Int λ_4	Int λ_5	Total time	
0.07	0.265	0.09	0.15	0.21	0.215	1	

Table 3.2: Comparison of CPU-times required for the 2.5% error at each stage for the system with high energy barrier. The times are renormalized to the TIS total computation time. W1 to W8 denote the different windows used in the calculation, Int λ_1 to Int λ_5 denote the interface ensemble calculations.

$\lambda_A = \lambda_1$ at $r_d = 1.24$. This interface is closer to the basin of attraction than the TPS stable state definition but yields a higher flux term $\langle \phi_{0,1} \rangle / \langle h_A \rangle$ and gives better statistics. Note that the different definition of stable state A does not change the final rate constant, as the overall state \mathcal{A} does not sensitively depends on this definition. The flux term is calculated by straightforward NVE MD. As λ_A is equal to λ_1 every positive crossing of this interface is counted in the flux because all trajectories must by default come directly from A . The conditional crossing probabilities $P_A(i+1|i)$ in Eq. (3.19a) are calculated for $n_I = 5$ interfaces between the stable states (see fig. 3.10). Between these interfaces we imposed a finer grid to obtain the entire crossing probability function. The results for each stage and the final rate constant are shown in Table 3.1. The rate constants of both methods agree within the statistical accuracy, showing that the TIS method is correct. In Table 3.2 we give the relative computation time to reach the 2.5% error for each term.

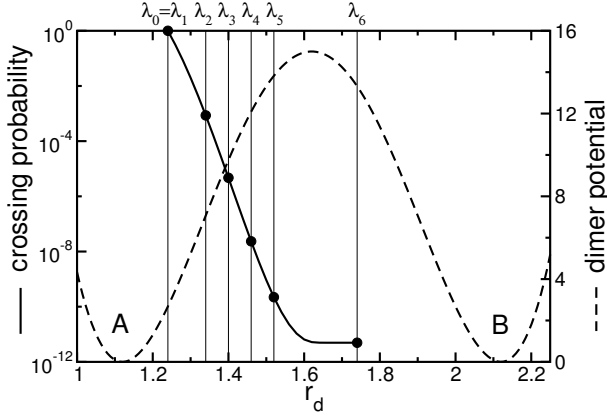


Figure 3.10: System with high energy barrier: the dimer inter-particle potential (dashed line) and the TIS crossing probability $P_A(\lambda|1)$ (solid line) as function of the order parameter $\lambda = r_d$, the dimer inter-particle separation. State A at the first minimum of the double-well corresponds to a compact state of the molecule and state B at second minimum to an extended state. The crossing probability function is computed by matching five interface ensemble calculations. These interfaces were chosen at: $\lambda_1 = 1.24$, $\lambda_2 = 1.34$, $\lambda_3 = 1.40$, $\lambda_4 = 1.46$ and $\lambda_5 = 1.52$, and are indicated by vertical lines. The error on the points is within symbol size. The smooth line joining the points was created using a finer grid of interfaces. The horizontal plateau when approaching state B at $\lambda_{n=6} = 1.74$ is an expression of the commitment of the trajectories to the final stable state. Paths that cross $r_d \simeq 1.7$ always reach eventually the final interface without going back to A. The value of the plateau equals $P_A(n_I|1)$.

In comparing both methods we have to realize that the efficiency of TPS depends strongly on the choice of t' . On the one hand the umbrella calculation of $C(t')$ is much faster for low values of t' . But on the other hand the error in the correction term $\langle h_B(t') \rangle_{AB}^*$ increases for lower t' . As a result there is an optimum t' for the error/CPU-time ratio, in this case approximately at $t' = 0.3$. Even for this optimized situation the TIS calculation is about two times faster. One could object that the correlation function in Fig. 3.8 has reached a plateau for $t = 1.5$ already, reducing the TPS computation time by a factor 3/4. But the choice for a path length $\mathcal{T} = 1.5$ can not be taken without a-priori knowledge. The first term in Eq. (2.43) implicitly depends on the path length \mathcal{T} . Changing \mathcal{T} would alter the ensemble and might result in a different shape of the flux correlation function. We did not check this in detail, but we believe that \mathcal{T} cannot be chosen much smaller without introducing systematic errors. Furthermore, we emphasize here that we put much more effort in optimizing the TPS algorithm by tuning t' , the windows, the ratio between shooting, shifting and diffusion moves than we did for TIS.

Figure 3.11 shows the histograms of path lengths for each TIS ensemble calculation and shows why TIS is faster. Sampling paths of fixed length with TPS results in spending unnecessary computation time inside the initial and final stable regions A and B. In the TIS algorithm instead every path is adapted to its minimum length. Bringing the interface in closer to A reduces these transition times. TIS optimizes itself during the simulation.

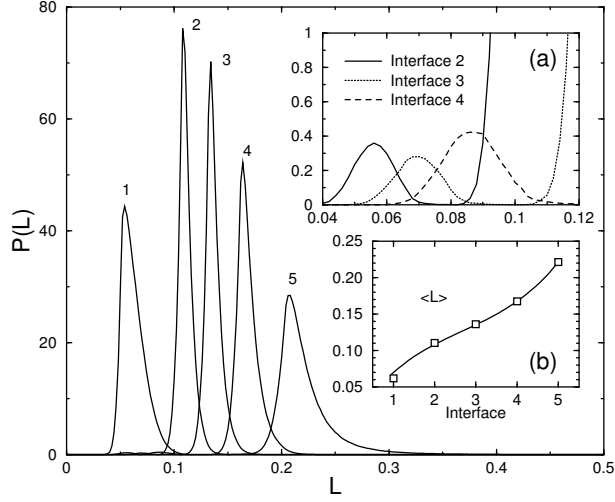


Figure 3.11: Histograms $P(L)$ of path length L for each ensemble, computed for the system with the high energy barrier. Inset (a) is an enlargement of the bottom left area, where windows 2,3,4 display a second peak. They represent that small fraction of paths that are able to cross all the interfaces up to the rightmost interface and do not have to return to A (cf. the trajectories with the white circle in Fig.3). Inset (b): average path length in each window. At variance with TPS the TIS algorithm adapts the path length to the ensemble. In going from interface 5 to interface 1 one gets closer to state A and the path length shortens accordingly.

3.3.4 System with Low Energy Barrier

In order to compare with previous results, we adopted the parameters from Ref. [79]. The total number of particles was $N = 9$, the total energy was $E = 9$ and the square simulation box was adjusted for a number density of 0.6. The barrier height is $h = 6$ and the width-parameter is $w = 0.25$. Minima are at $r_d \simeq 1.12$ and $r_d \simeq 1.62$, while the top of the barrier is at $r_d \simeq 1.37$. This barrier is much lower than in the previous section resulting in more frequent transitions. An approximate rate constant could even be achieved by straightforward MD simulations.

For the TPS calculations we defined the stable states A and B by $r_d < r_d^A = 1.30$ and $r_d > r_d^B = 1.45$, respectively [79]. Using standard TPS simulations we computed the correlation function $\langle h_B(t) \rangle_{AB}^*$ with a total path length $\mathcal{T} = 2$ (shown in Fig. 3.12). Next, we measured the probability histograms to find the paths end point at a certain order parameter value r_d for four different times $t' = 0.1, 0.4, 0.8, 2.0$, using five windows [79], see Fig. 3.13. As described in the previous section, matching the probability histograms and subsequent integration leads to $C(t')$. The resulting final rate constants, shown in Table 3.3, are comparable with the results of Ref. [79], but more accurate. We will discuss these values after giving the results of TIS.

Figure 3.14 shows that fast recrossings can occur for a low barrier, implying that

Figure 3.12: TPS correlation function $\langle h_B(t) \rangle_{AB}^*$ (top) and its time derivative (bottom) for the system with low energy barrier. The error is comparable to line thickness.

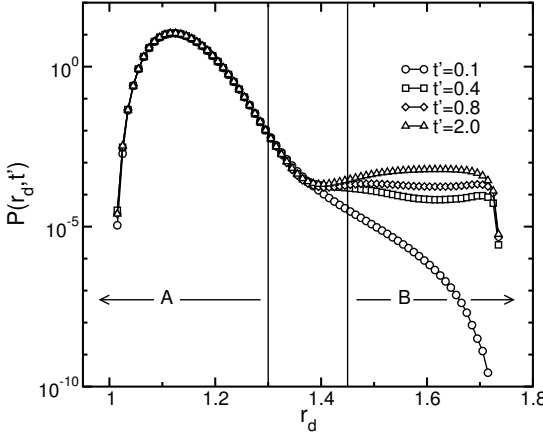
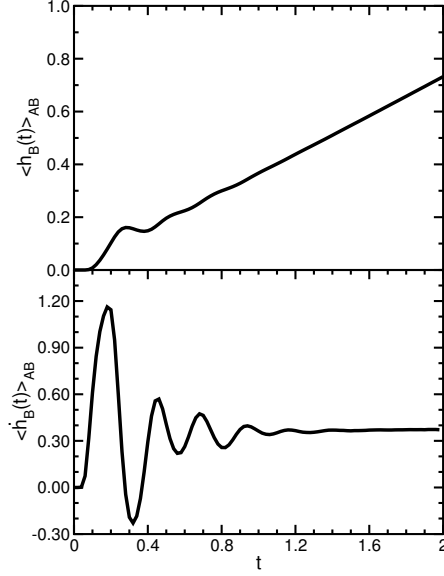


Figure 3.13: TPS probability distributions $P_A(r_d, t')$ for four $t' = 0.1, 0.4, 0.8, 2.0$ for the system with low energy barrier. $P(r_d, t')$ is defined as in Fig. 3.9. The graph is the result of the matching of five window calculations. These five window calculations are defined as $r_d < 1.22$, $1.21 < r_d < 1.26$, $1.25 < r_d < 1.30$, $1.29 < r_d < 1.46$, $r > 1.45$. The errors on the histogram points are within the symbol size.

r_d alone is not sufficient as an order parameter to define the stable states in the simulations. Apparently, this does not effect the TPS results much, but it is very important for TIS because of the assumption that stable region B is really stable and recrossings do not take place. To ensure the stability of the TIS stable states we chose a new order parameter that not only depends on the inter-atomic distance r_d in the dimer but also on a kinetic term, given by \dot{r}_d . The stable states can then be defined by

$$\begin{aligned}
 E_d(r_d, \dot{r}_d) &\equiv \frac{\dot{r}_d^2}{4} + U_{dw}(r_d) \\
 x \in A &\text{ if } r_d < 1.37 \text{ and } E_d(r_d, \dot{r}_d) \leq 1.5 \\
 x \in B &\text{ if } r_d > 1.37 \text{ and } E_d(r_d, \dot{r}_d) \leq 1.5,
 \end{aligned} \tag{3.23}$$

where E_d is the sum of the kinetic and potential energy of the dimer that has a

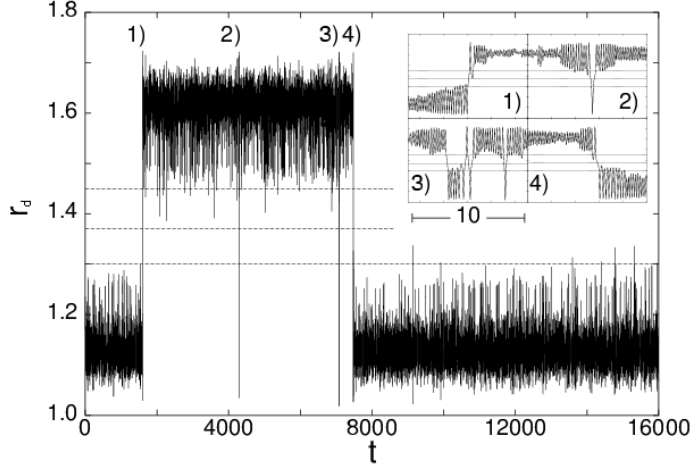


Figure 3.14: Intra-molecular distance of the dimer as function of time from a straight-forward MD simulation for the system with the low energy barrier. Horizontal dashed line at 1.37 corresponds to the top of the potential barrier. Horizontal dashed lines at 1.3 and 1.45 correspond to the TPS state definitions of Ref. [79]. Insets are enlargements of four typical events on a scale of 10. 1) and 4) correspond to true reactive events, $A \rightarrow B$ and $B \rightarrow A$ respectively while 2) and 3) are non-true, fast recrossing events. In particular, event 3) shows capricious behavior with many crossings of the barrier. The figure shows a clear separation of timescales, $\tau_{trans} \sim 1$ and $\tau_{stable} \sim 1000$.

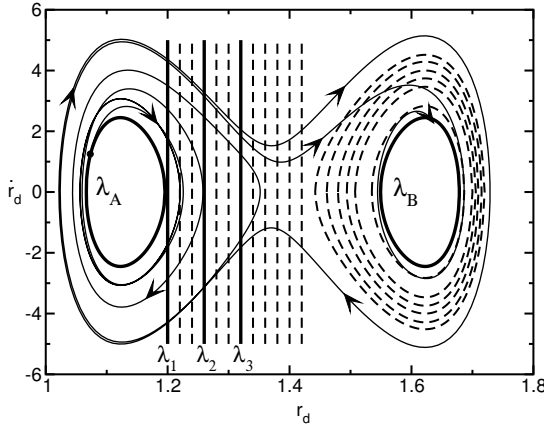


Figure 3.15: One calculated path of the low energy barrier system shown in the $\{r_d, \dot{r}_d\}$ plane. The vertical solid lines are the interface λ_1, λ_2 and λ_3 . The curves λ_A and λ_B are the boundaries of the TIS stable states. The dashed lines are the sub-interfaces. The path starts at the dot on λ_A and crosses the barrier three times before dissipating its energy and relaxing into state B .

reduced mass of $1/2$. In the $\{r_d, \dot{r}_d\}$ plane these stable states form a D-shape and an inverse D-shape regions for A and B respectively, see Fig. 3.15. Crossing the interface λ_A or λ_B implies that the vibrational energy is decreased below the threshold, $E_d = 1.5$. This threshold is made low enough to make fast recrossings to the other state unlikely. However, if we would have chosen it too low the paths would have become very long. We evaluated the crossing probability function in Eq. (3.19a) for $n_I = 3$ interfaces. The entire crossing probability function was obtained by partitioning the

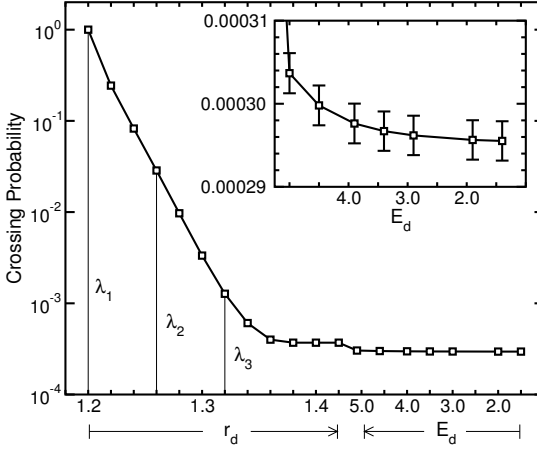


Figure 3.16: The crossing probability $P_A(\lambda|1)$ for the system with the low energy barrier. The function is computed by matching ensemble calculations with interfaces λ_1 at $r_d = 1.20$, λ_2 at $r_d = 1.26$ and λ_3 at $r_d = 1.32$. The inset is an enlargement of the final part. The function is converging to a plateau but has not yet reached it. The different values of the last points are due to the presence of fast recrossings. The error is inside the symbol size.

phase space in sub-interfaces of the form $r_d = \text{const}$ and $E_d(r_d, \dot{r}_d) = \text{const}$ as shown in Fig. 3.15. Note that in TIS multidimensional or multiple order parameters can be used in one simulation without a problem. This is more difficult in TPS, where a proper mapping of the complete phase space is required. Figure 3.16 shows the final rematched crossing probability. The monotonically decreasing function tends to reach a plateau on approaching the last interface. The last two values are not exactly equal but differ by 0.03%, indicating that a small fraction of the paths crossing the last but one sub-interface still succeed to return to A without crossing λ_B . This difference is comparable with the chance of a new independent transition (given by the rate constant). Note that without the kinetic energy definition for the stable states Eq. (3.23), the final crossing probability and thus the rate constant would have been overestimated by a factor 5/4.

For the effective flux $\langle \phi_{0,1} \rangle / \langle h_A \rangle$ calculation we performed MD simulations as described in Sec 3.3.2. In contrast to the high barrier case, λ_1 is not equal λ_A , and not all positive crossings with λ_1 are effective crossings. We counted only the first crossing when the system left region A and waited until the system fell back to region A before counting a new crossing. As the MD trajectory sometimes displayed a spontaneous transition to region B , we stopped the simulation and started again by replacing the system in a randomized configuration of A . Table 3.3 shows the final values and the corresponding errors of these calculations. The relative computation time for each term is detailed in table 3.4.

If we compare the final results of table 3.3 we see that the efficiency of TIS is more than nine times better than the TPS efficiency for $t' = 2$, and more than two times better than TPS value for $t' = 0.8$. But the TPS $t' = 0.1$ and $t' = 0.4$ efficiencies are about 20% better than TIS. When we compare the rate constants, however, we notice that the TPS results for different t' do not agree. Among the TPS rate constants only the $t' = 2$ case is consistent with the TIS result. We believe that the $t' = 0.1$ and $t' = 0.4$ results suffer from systematic errors. For instance, for the shorter paths the TPS simulations might not be completely ergodic. Another explanation might be that

TPS			
t'	$\frac{\langle \dot{h}_B(T) \rangle_{AB}^*}{\langle \dot{h}_B(t') \rangle_{AB}^*}$	$C(t')/10^{-5}$	$k_{AB}/10^{-5}$
0.1	47.3 ± 0.2	1.408 ± 0.007	6.67 ± 0.04
0.4	2.505 ± 0.007	2.67 ± 0.01	6.68 ± 0.03
0.8	1.240 ± 0.003	5.42 ± 0.05	6.72 ± 0.07
2.0	0.507 ± 0.001	13.9 ± 0.2	7.03 ± 0.09

TIS			
$\langle \phi_{0,1} \rangle / \langle h_A \rangle$	$P_A(B 1)/10^{-5}$	$k_{AB}/10^{-5}$	
0.2334 ± 0.0003	29.6 ± 0.2	6.90 ± 0.06	

Table 3.3: Comparison of rate constants for the low energy barrier computed with TPS at different t' and with TIS, including the contributing factors from Eq. (2.43) and Eq. (3.19a), respectively. Computation times are reported in units of the TIS CPU-time.

TPS							
t'	$\frac{\langle \dot{h}_B(T) \rangle_{AB}^*}{\langle \dot{h}_B(t') \rangle_{AB}^*}$	W1	W2	W3	W4	W5	Total
0.1	0.68	0.03	0.009	0.01	0.1	0.001	0.83
0.4	0.4	0.09	0.03	0.04	0.25	0.01	0.82
0.8	0.28	0.21	0.07	0.11	1.5	0.04	2.21
2.0	0.35	0.28	0.38	0.93	7.27	0.14	9.35

TIS					
$\langle \phi_{0,1} \rangle / \langle h_A \rangle$	Int λ_1	Int λ_2	Int λ_3	Total	
0.015	0.085	0.45	0.45	1	

Table 3.4: Comparison of CPU-times required for the 2.5% error at each stage for the system with the low energy barrier. The times are renormalized to the TIS total computation time.

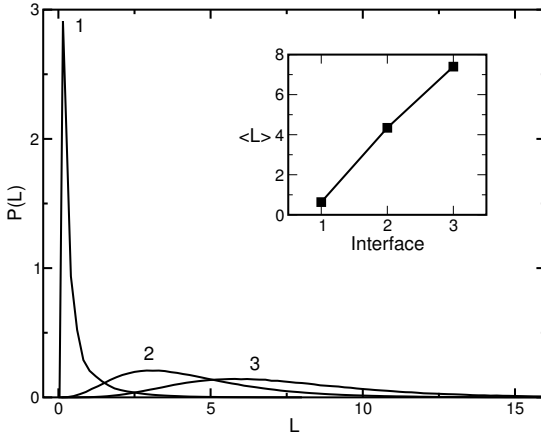


Figure 3.17: Path length distribution $P(L)$ for each interface ensemble in the low energy barrier system. The inset shows the average path length in each ensemble.

a path length of $\mathcal{T} = 2$ is too short to allow convergence of the reactive flux. In the TIS calculation the average path length in the three interface simulations, from the closest to B to the closest to A , is, respectively, 7.4, 4.3, and 0.63; much longer than the TPS path length, see Fig.3.17. It is therefore surprising that the TPS approach with the simple stable state definition and very short paths still gives approximately the right rate constant. And indeed, when we computed the TPS correlation function with the TIS state definitions Eq. (3.23), we found that the path length had to be at least $\mathcal{T} = 20$ to see a plateau. We think that TPS works even with the simple state definitions and the short paths because both positive flux and negative flux terms contribute to Eq. (2.43). The TPS algorithm collects many paths of which some are not real transitions, but fast recrossings. The cancellation of positive and negative terms of these fast recrossing paths ensure the (almost) correct final outcome. In TIS each path must be true a transition event and contributes as a positive term in the rate equation (3.19a), enhancing the convergence. This explains that the CPU time for the TIS calculation despite the much longer paths is still comparable with TPS one for low t' . We note that the path ensemble using the more strict stable state definition is, of course, more useful in the analysis of the reaction mechanism.

For a more accurate comparison of the computation time we must keep the systematic errors lower than the statistical errors. In other words, we have to make sure that the results are converged. To test the convergence of the flux correlation function in TPS we can derive the following equality from Eq. (3.19a):

$$\frac{\langle h_B(t') \rangle_{A, H_B(\mathcal{T})}}{\langle h_B(t'') \rangle_{A, H_B(\mathcal{T})}} = \frac{C(t')}{C(t'')}. \quad (3.24)$$

This equation is valid for any $t', t'' < \mathcal{T}$ if \mathcal{T} is large enough. We found that the equality does not hold for the system with the low barrier, indicating that \mathcal{T} is too low in the TPS calculation. Further examination of the flux correlation function $\langle h_B(t') \rangle_{A, H_B(\mathcal{T})}$ reveals that the apparent plateau has in fact a small positive slope. Calculations for higher values of \mathcal{T} suggest that one has to increase the path length at least to $\mathcal{T} = 8$ to convergence to a plateau. With this in mind we think that the TIS computation is about a factor five more efficient than the TPS algorithm for the model system with the low barrier.

3.4 Summary

Driven by the desire to improve the rate constant calculation in TPS, we have developed a novel method, named transition interface sampling. By retaining the path sampling idea, and adapting the computational algorithms, TIS achieves the following improvements with respect to TPS:

1. TPS computes the rate using the correlation function $C(t)$, Eq. (2.40). The time derivative $\dot{C}(t)$ converges to a plateau equal to the rate k_{AB} because is equivalent to a flux calculation. However when recrossings appear in the buffer region, the convergence is due to cancellation of positive and negative terms. In contrast TIS computes the rate using the idea of effective positive flux (see Sec. 1.5.2 and 3.1.3), and only positive terms contribute.

2. At the basis of TIS is the definition of a rate as inverse mean first passage time. As a consequence the TIS algorithm stops the integration when the stable state B is reached. In TPS, umbrella-sampling windows are used to compute $C(t')$ at a fixed t' variable parameter. In TIS windows are also used, but the TIS algorithm stops integration when the window boundaries are reached. In this way the TIS algorithm adapts itself to the optimal path length. One does not have to optimize the new method as much as TPS, where one has to find the optimal t' value and a proper balance between shooting and shifting. Indeed in TIS the shifting move, sec. 2.2.2, is not needed.
3. In TPS the generation of an initial path requires a separate set of simulations, see sec. 2.2.4. In TIS it is naturally self-implemented. Paths for the next TIS window are generated in a recursive way from the previous one.

The concept of calculating a flux comes natural with the rate constant definition, and implementation of the algorithm is hence simpler. Simple theoretical reasoning can show that TIS is at least a factor 2 faster than TPS (see sec. 6.1), but in the illustrative example we showed here that we can obtain an increase in efficiency of at least a factor of two to five with respect to the TPS method used in Ref. [79].

The analysis of reaction mechanism is the same in TIS and TPS. The generation of full transitions takes place in TIS when we reach the last interface, and at that point is basically equal to the path sampling part of TPS. The analysis proceeds using the committer distribution methods of sec. 2.5. We will show an application of this in the study of nucleation, chap. 7.

The TIS method has been successfully applied to two realistic cases, the folding of a polypeptide [3] and hydration of ethylene [119]. In this last case the method was combined with quantum ab-initio MD simulations.

In the next chapter, we present a variation of the TIS method for diffusive systems that exploits very efficiently the loss of long time scale correlation by using a recursive reformulation of the crossing probability and the sampling of much shorter paths. This is the subject of the following chapter.

4

Partial Path TIS

*All those moments will be lost
in time, like tears in rain*

Roy Batty

In chapter 3 we introduced the theory and algorithms of the Transition Interface Sampling method for the computation of rate constants in complex systems. In the case of highly diffusive systems, the efficiency of TIS can be improved by exploiting the loss of memory along the paths and thus sampling only parts of complete transition trajectories confined within a certain region. This Partial Path TIS (PPTIS) method is the subject of this chapter. We present the theory of PPTIS using the Interface Sampling framework developed in sec. 3.1, and explain the computational implementation. We compare then PPTIS and TIS for the diatomic system introduced in the previous chapter and we discuss the validity of the memory loss assumption. We also present a variation of PPTIS that, together with the rate constant, allows a computation of the free energy profile along the order parameter. Finally, we discuss an algorithm that can alleviate problems of bad path sampling. This chapter is based on [109, 20].

4.1 Theory

In this chapter we focus on transitions with a highly diffusive character, or in the regime of high solvent friction. Examples are the folding and unfolding of a protein in water, charge transfer, fragmentation reactions, diffusion of a molecule through a membrane, and nucleation processes. These types of processes have to overcome a relatively flat and wide, but still rough free energy barrier. When applying the TPS or TIS shooting algorithm, sec.2.2.1 and 3.2.2, to such a transition, the Lyapunov instability causes the paths to diverge before the basins of attraction have the chance to guide the paths to the proper stable state. Pathways will then become very long and, moreover, the acceptance ratio of shooting will be low. Hence, the shooting algorithm will be very inefficient, resulting in bad sampling.

Here, we will introduce an efficient method to calculate the rate constant for such

barriers. To do so, we make use of the TIS effective flux relation, see sec 3.1.3 and assume that the diffusivity eliminates any memory effects over a distance more than the separation between two interfaces. The rate constant can then be recast in a recursive relation for the hopping transition rates between interfaces. These hopping transition rates can be computed by sampling short trajectories connecting just three successive interfaces. If the assumption of memory loss is valid, this partial path transition interface sampling (PPTIS) procedure correctly collects the contributions of all possible paths to the rate constant, in principle, even those with infinite lengths.

We first illustrate the PPTIS concept for a simple one dimensional array of well defined metastable states. We generalize then the idea using the theory of Interface Sampling.

4.1.1 Illustration of the PPTIS concept

Before embarking on the general case of diffusive barriers, we will first consider a simple one dimensional system that serves as an illustrative example. This system exhibits a barrier consisting of a series of metastable states as is illustrated in Fig. 4.1.

The overall barrier is high compared to those between metastable states. We therefore assume that the time to relax from the barrier into a stable state is much shorter than the total reaction time. This separation of timescales gives rise to two state kinetics and well-defined overall rate constant k_{AB} . Furthermore, we assume that the system can hop from one metastable state to a neighboring one after which it will fully relax.

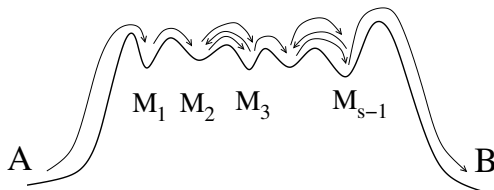


Figure 4.1: Illustration of a barrier consisting of a series of metastable states. One possible trajectory connecting A and B is shown.

Consequently, the probability to hop to left or right does not depend on the history of the path, and hence the system is Markovian. For this type of system, we might write down a master equation and solve for all the population densities in each state on the barrier as a function of time [7]. However, if we assume steady state behavior, and take into account the fact that the population on the barrier is low, the overall rate constant is only determined by the hopping probabilities. We will denote the probabilities to transfer from site i to the right or left metastable state by $\tau_{i,i+1}$ and $\tau_{i,i-1}$, respectively, which are related by $\tau_{i,i+1} + \tau_{i,i-1} = 1$. For a system with $s - 1$ metastable states M_1, M_2, \dots, M_{s-1} and the stable states $M_0 = A$ and $M_s = B$, the reaction rate k_{AB} and its reverse k_{BA} can be expressed as:

$$\begin{aligned} k_{AB} &= k_{0,1} T[1 \rightarrow_s^s], \\ k_{BA} &= k_{s,s-1} T[s-1 \rightarrow_0^0], \end{aligned} \quad (4.1)$$

with $T[i \rightarrow_m^j]$ the probability to go via an arbitrary number of hops from metastable state i to metastable state j before visiting metastable state m . The computation of the rate constants only requires the determination of the nearest neighbor hopping probabilities $\tau_{i,i+1}$ and the first hopping rates $k_{0,1}$ and $k_{s,s-1}$. The long distance

hopping probabilities $\{T[1 \rightarrow_0^j], T[j-1 \rightarrow_j^0]\}$ can be obtained via following recursive relations (see Appendix E.1):

$$T[1 \rightarrow_0^j] = \frac{\tau_{j-1,j} T[1 \rightarrow_0^{j-1}]}{\tau_{j-1,j} + \tau_{j-1,j-2} T[j-2 \rightarrow_{j-1}^0]} \quad (4.2a)$$

$$T[j-1 \rightarrow_j^0] = \frac{\tau_{j-1,j-2} T[j-2 \rightarrow_{j-1}^0]}{\tau_{j-1,j} + \tau_{j-1,j-2} T[j-2 \rightarrow_{j-1}^0]} \quad (4.2b)$$

Starting with $T[1 \rightarrow_0^1] = T[0 \rightarrow_1^0] = 1$, we can iteratively solve Eqs. (4.2) for $j = 2, 3 \dots s$. In this way we collect analytically the statistics of all possible pathways. This procedure accounts for the straightforward barrier crossings, but also accounts for the contributions to the rate of an infinite number of different pathways that lead from A to B in an infinite number of hops. Although the probability of a single pathway decreases with its length, the total contribution of the very long pathways becomes more important when s is increased. In fact, the average path length scales as $\sim s^2$. In case of uniform symmetric hopping ($\tau_{i,i+1} = \tau_{i,i-1} = \frac{1}{2}$ for all i), it is shown in appendix E.4 that $k_{AB} = \frac{1}{s} k_{0,1}$, whereas if we would only account the fastest pathway ($M_0 \rightarrow M_1 \rightarrow M_2 \dots \rightarrow M_s$) it would be much lower, $(\frac{1}{2})^s k_{0,1}$.

At first sight, it seems a bit surprising that the residence time in each metastable state and the absolute intra-barrier rates $k_{i,i\pm 1}$ have no influence on the final total rate expression. Only the relative rates are important as they determine the nearest neighbor hopping probabilities by $\tau_{i,i\pm 1} = k_{i,i\pm 1} / (k_{i,i+1} + k_{i,i-1})$. We can understand this by again invoking the separation of timescales argument. The time τ_{trans} the system spends on the barrier ($\sim s^2$) must be much smaller than the state lifetime $\tau_{stable} = 1/(k_{AB} + k_{BA}) \sim 1/(k_{0,1} + k_{s,s-1})$. This will be true when $k_{0,1}, k_{s,s-1} \ll k/s^2$ with $k \sim k_{i,i\pm 1}$ the average hopping rate on the barrier for $i = 1 \dots s-1$. Of course, when we start with a system out of equilibrium and calculate the relaxation time from A to B for a system that is initially completely in A , the intrabARRIER rates $k_{i,i\pm 1}$ will be dominant factors.

Our treatment of the model in this section can be related to the solution of the onedimensional model of sec. 1.5.2, a flat high barrier of length b . Eq. (1.56) gives for the rate constant $k_{AB} = (D/b) \exp(-\beta h)$ where h is the barrier height and D the diffusion constant. The connection becomes clear when one realizes $k_{0,1}/k_{1,0} = \exp(-\beta h)$ and $D/b = k/s$, with $k \sim k_{1,0}$ the hopping rate, and s the number of hops on the barrier. Hence, $k_{AB} = \frac{1}{s} k_{0,1}$, just as found above for the symmetric uniform hopping model. A more formal treatment of general diffusive Markov processes can be found in e.g. Ref. [7].

The model described above is of limited importance due to its highly symmetric and onedimensional character. Some processes, however, such as the diffusion of particles through a onedimensional crystal (e.g alkanes through zeolites) can be described by this uniform symmetric hopping model. More complex behavior such as diffusion on surfaces, through multidimensional crystals, or in (biological) networks usually has to be studied by means of Monte Carlo (MC) algorithms to solve the master equation, often called kinetic MC methods [120, 121, 52]. Still, the example given here is illustrative for the more complex PPTIS method advocated in this paper. The PPTIS method combines the iterative solution of Eq. (4.2) for the overall rate

constant with the TIS algorithm (see sec.3.2). This approach will enable treatment of a much wider variety of systems with a diffusive character, but not with such a rigid structure as the onedimensional Markov chain.

4.1.2 PPTIS formalism

As in sec. 3.1 we define a set of n_I non-intersecting multidimensional interfaces $\{0, 1 \dots n_I\}$ described by an order parameter $\lambda(x)$ which is a function of the phase space point x . We choose λ_i , $i = 0 \dots n_I$ such that $\lambda_{i-1} < \lambda_i$, and that the boundaries of state A and B are described by λ_0 and λ_{n_I} , respectively.

The starting point to develop PPTIS is the 4-interface probability $P_{(m|j)}^l$, Eq. (3.13). We remind that this is the probability for the system to reach interface l before m under the condition that it crosses at $t = 0$ interface i , while coming directly from interface j in the past (see Fig. 3.4). In sec.3.1.4, the derivation of Eqs. (3.19a), (3.19b), showed that the rate constants can be written in terms of the probabilities $P_A(n_I|1) = P_{(0|0)}^{n_I}$ and $P_B(0|n_I - 1) = P_{(n_I|n_I-1)}^0$. Here, we want to find an approximate relation for these probabilities in the spirit of the onedimensional model of the previous section. For this purpose we define the one-interface crossing probabilities

$$\begin{aligned} p_i^{\pm} &\equiv P_{(i-1|i-1)}^{(i+1|i)}, & p_i^{\mp} &\equiv P_{(i+1|i+1)}^{(i-1|i)}, & i &= 1 \dots n_I - 1 \\ p_i^{\bar{\pm}} &\equiv P_{(i+1|i-1)}^{(i-1|i)}, & p_i^{\dagger} &\equiv P_{(i-1|i+1)}^{(i+1|i)}, \end{aligned} \quad (4.3)$$

which fulfill the following relations:

$$p_i^{\pm} + p_i^{\bar{\pm}} = p_i^{\mp} + p_i^{\dagger} = 1. \quad (4.4)$$

A schematic visualization of these probabilities is given in Fig. 4.2. We define then long-distance crossing probabilities P_i^+ and P_i^- , similar to those in Sec. 4.1.1

$$P_i^+ \equiv P_{(0|0)}^{(i|1)}, \quad P_i^- \equiv P_{(i|i-1)}^{(0|1)}. \quad (4.5)$$

The main assumption in PPTIS is that trajectories lose their memory, over a short time, and hence over a short “distance”, as measured by λ . We require that the

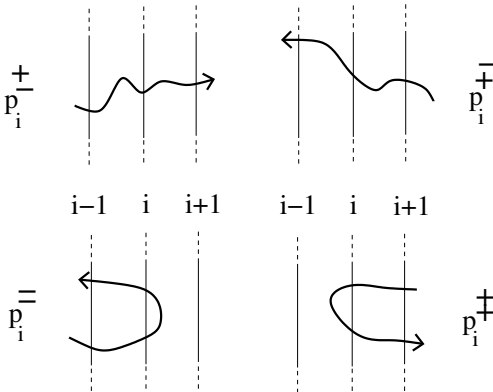


Figure 4.2: Visualization of the one-interface crossing probabilities $(p_i^{\pm}, p_i^{\bar{\pm}}, p_i^{\mp}, p_i^{\dagger})$. Possible trajectories that correspond to a positive contribution of these probabilities are shown.

interfaces are set such that no memory effects are present over more than the distance between two interfaces or, equivalently, that the following relation is obeyed:

$$\langle g(x) \rangle_{\phi_{i,i\pm q}} \approx \langle g(x) \rangle_{\phi_{i,i\pm 1}}, \quad (4.6)$$

with q an integer larger than one and $g(x)$ any observable corresponding to the actual state x or any future state. With this assumption we can derive recursive relations for the long-distance crossing probabilities using the PPTIS concept introduced in Sec. 4.1.1 (see Appendix E.2):

$$\begin{aligned} P_j^+ &= \frac{p_{j-1}^\pm P_{j-1}^+}{p_{j-1}^\pm + p_{j-1}^\mp P_{j-1}^-} & j = 2 \dots n_I \\ P_j^- &= \frac{p_{j-1}^\mp P_{j-1}^-}{p_{j-1}^\pm + p_{j-1}^\mp P_{j-1}^-} \end{aligned} \quad (4.7)$$

To solve these recursive expressions we start with $P_1^+ = P_1^- = 1$, after which we iteratively determine (P_j^+, P_j^-) for $j = 2, \dots$ until $j = n_I$. Substitution of the long distance crossing probabilities into Eqs. (3.19) results in

$$k_{AB} = \frac{\langle \phi_{1,0} \rangle}{\langle h_A \rangle} P_{n_I}^+, \quad k_{BA} = \frac{\langle \phi_{n_I-1, n_I} \rangle}{\langle h_B \rangle} P_{n_I}^-. \quad (4.8)$$

The factor $\langle \phi_{1,0} \rangle / \langle h_A \rangle$ is identical to the TIS flux factor, whereas to obtain the reverse rate k_{BA} only a single extra factor $\langle \phi_{n-1,n} \rangle / \langle h_B \rangle$ is needed. As a result we obtain the reverse rate and the equilibrium constant $C = k_{AB}/k_{BA}$ without any significant extra costs, while in TPS or TIS, as we saw in previous chapters, the calculation of the reverse rate would require another comparable computational effort. The PPTIS formalism basically transforms the process of interest into a Markovian sequence of hopping events. Yet, if the dynamics is diffusive and the interfaces are sufficiently far apart the rate formalism (4.7), (4.8) will be a good approximation.

Before describing the implementation of PPTIS, we should stress that the long-interface probabilities in PPTIS, Eq. (4.5), and in TIS, Eq. (3.16), coincide for the full transition

$$P_A(n_I|1) = P(0|0) = P_{n_I}^+, \quad P_B(0|n_I-1) = P(0|n_I-1) = P_{n_I}^- \quad (4.9)$$

so that relations (4.8) are completely equivalent to (3.19) and exact. The PPTIS approximation arises from the calculation of $P_{n_I}^+, P_{n_I}^-$ through the recursive relations (4.7). Furthermore, by definition $P_i^+ = P_A(i|1)$ for all $i = 1 \dots n_I$. On the contrary, P_i^- is not directly comparable with $P_B(i|n_I-1)$. However, because of theorem (3.15), the exact relation $P_B(0|n_I-1) = P_B(i|n_I-1)P_B(0|i)$ holds and we can write

$$P_{i+1}^- \simeq P_B(0|i) = P_B(0|n_I-1)/P_B(i|n_I-1) \quad (4.10)$$

where we used the approximate relation $P_{i+1}^- \simeq P_B(0|i)$ with $i = 0 \dots n_I - 1$. The term $P_B(0|i)$ considers paths coming from B that cross λ_i and reach A , while the term P_{i+1}^- takes into account paths that do the same but come from λ_{i+1} . The difference is given by those paths that, integrating backwards in time, after crossing λ_{i+1} do not go to B . However, they become less abundant as we shift interfaces towards B , so that the relation becomes more accurate as $i \rightarrow n_I - 1$ and is exact for $i = n_I - 1$.

4.2 PPTIS algorithm

The PPTIS method requires the determination of the $p_i^\pm, p_i^\mp, p_i^\mp$, and p_i^\dagger probabilities. However, p_i^\pm and p_i^\mp are defined in a different ensemble than p_i^\mp and p_i^\dagger . In most cases, it will be convenient to calculate the four probabilities simultaneously. To do so, we define an ensemble that includes both ensembles via the weight function $\phi_{i\pm}(x)$:

$$\phi_{i\pm}(x) \equiv \phi_{i,i-1}(x) + \phi_{i,i+1}(x) \quad (4.11)$$

In this ensemble, p_i^\pm and p_i^\mp equal

$$p_i^\pm = \frac{\langle \phi_{i,i-1} h_{i+1,i-1}^f \rangle_{\phi_{i\pm}}}{\langle \phi_{i,i-1} \rangle_{\phi_{i\pm}}} \quad (4.12)$$

$$p_i^\mp = \frac{\langle \phi_{i,i+1} h_{i-1,i+1}^f \rangle_{\phi_{i\pm}}}{\langle \phi_{i,i+1} \rangle_{\phi_{i\pm}}} \quad (4.13)$$

and p_i^\mp and p_i^\dagger follow from Eq. (4.4).

For a correct sampling of this ensemble, we generate all possible paths starting from interface $i-1$ or $i+1$ and ending either at $i-1$ or $i+1$ with at least one crossing with i . The sampling is performed using the shooting move along the same lines of sec. 3.2.2. Suppose you have an initial path belonging to the ensemble. Define the phase space point x_0 as the first crossing point of this path with interface λ_i . We use a discrete time index $\tau = \text{int}(t/\Delta t)$, and let $\tau^b \equiv \text{int}(\min[t_{i-1}^b(x_0), t_{i+1}^b(x_0)]/\Delta t)$ and $\tau^f \equiv \text{int}(\min[t_{i-1}^f(x_0), t_{i+1}^f(x_0)]/\Delta t)$, where we used the definitions (3.1). In words τ^b is the number of slices from x_0 to the beginning of the path, and τ^f is the number of slices from x_0 to the end of the path. Including x_0 , the initial path then consists of $N^{(o)} = \tau^b + \tau^f + 1$ time slices. The path sampling loop is then identical to the TIS one, on page 63 except that λ_{i-1} is used instead of λ_0 , time reversal moves are always accepted and the backward integrating at step 5 of the shooting move is not rejected when reaching λ_{i+1} as paths may start from both sides.

The one-interface crossing probabilities are then given by

$$\begin{aligned} p_i^\pm &= \frac{N_p(i-1 \rightarrow i+1)}{N_p(i-1 \rightarrow i+1) + N_p(i-1 \rightarrow i-1)} \\ p_i^\mp &= \frac{N_p(i+1 \rightarrow i-1)}{N_p(i+1 \rightarrow i-1) + N_p(i+1 \rightarrow i+1)} \\ p_i^\mp &= 1 - p_i^\pm, \quad p_i^\dagger = 1 - p_i^\mp \end{aligned} \quad (4.14)$$

where $N_p(i \rightarrow j)$ is the number of paths going from interface i to interface j .

An algorithm based on the considerations of this section can be found in appendix K. The zone system of sec. 3.2.3 can be used also for PPTIS. It is convenient to classify paths according to their begin and endpoints as paths of type $(-+)$, $(+-)$, $(--)$, $(++)$ (see Fig. 4.2). Here $-$ denotes $i-1$ and $+$ denotes $i+1$, so that a path of type $(-+)$ goes from $i-1$ to $i+1$. Then separate counters N_p can be used to compute (4.14). Similar to the TIS case one can compute for each path the values

λ_{max} , λ_{min} to check that the ensemble is being sampled correctly. For example $(-+)$ paths should have $\lambda_{max} < \lambda_{i+1}$.

In the PPTIS algorithm, we usually applied shooting and path reversal moves with equal probability. The choice of momentum displacement depends on which part of the transition one is sampling. If the window $i-1, i, i+1$ is in a position where there is an uphill free energy barrier, PPTIS is quite similar to TIS and the 40% acceptance rule of sec. 3.2.3 can be applied. In the diffusive regime, where there is supposed to be a flat free energy barrier, the shooting move in PPTIS is rejected only if the paths do not cross interface i or get too long. The acceptance can be much higher, 70 or 80% and the 40% rule does not need to be satisfied.

At variance with TIS, we have no strict theoretical rules for the position of the interfaces, even though some qualitative remarks will be given in sec. 6.1. Since TIS is quite similar to PPTIS the same setting can be applied for the uphill energy part. For the diffusive part, instead, the separation of interfaces has a lower bound dictated by the memory loss requirement, as will be discussed in the next section.

4.2.1 Memory loss assumption

Contrary to the TIS technique, where the interfaces should be close to obtain good statistics, the interfaces should be sufficiently apart in the PPTIS method to ensure complete loss of memory. A simple test for Eq. (4.6) would be to measure $\langle g(x) \rangle_{\phi_{i,i-1}}$ for different separations between λ_i and λ_{i-1} . The velocity $\dot{\lambda}$ at the crossing point through λ_i could be a good candidate for the function g ¹. Substituting $\dot{\lambda}(x)$ into Eq. (4.6) gives

$$\left\langle \dot{\lambda}(x_0) \right\rangle_{\phi_{i+1,i}} \approx \left\langle \dot{\lambda}(x_0) \right\rangle_{\phi_{i+1,i-1}} \quad (4.15)$$

This relation can be rewritten in the ensemble of $\phi_{i\pm}$:

$$\frac{\left\langle \dot{\lambda}(x_F) h_{i+1,i-1}^f(x_0) \right\rangle_{\phi_{i\pm}}}{\left\langle h_{i+1,i-1}^f(x_0) \right\rangle_{\phi_{i\pm}}} \approx \frac{\left\langle \dot{\lambda}(x_F) \phi_{i,i-1}(x_0) h_{i+1,i-1}^f(x_0) \right\rangle_{\phi_{i\pm}}}{\left\langle \phi_{i,i-1}(x_0) h_{i+1,i-1}^f(x_0) \right\rangle_{\phi_{i\pm}}} \quad (4.16)$$

where $x_F \equiv f(x_0, \min[t_{i-1}^f(x_0), t_{i+1}^f(x_0)])$ is the path endpoint and $\dot{\lambda}(x_F)$ its velocity. Only paths of type $(-+)$ contribute to the right-hand side of Eq. (4.16), while the left-hand side also includes the $(++)$ paths from $i+1$ to $i-1$ via i . A similar expression can be derived for the reverse direction. The endpoint velocity $\dot{\lambda}(x_F)$ is indicative for the path's likelihood to progress along the order parameter λ . Therefore, we can reasonably expect that if Eq. (4.16) is true for all interfaces λ_i , the systematic error in the overall crossing probability P_n^+ due to the memory loss assumption will be small. Criterion (4.16) is obeyed if the endpoint velocities of the $(-+)$ and $(++)$ paths are the same, which is the case on a relatively flat part of the barrier, provided

¹As not only the average velocity should be the same $\left\langle \dot{\lambda}(x) \right\rangle_{\phi_{i,i-q}} = \left\langle \dot{\lambda}(x) \right\rangle_{\phi_{i,i-1}}$, but the whole distribution of velocities at λ_i , we used in Sec. 4.3 the velocity distribution overlap as measure of the memory loss.

the interfaces are sufficiently far apart. Eq.(4.16) also holds if there are no $(++)$ paths present at all, which is the case for the uphill part of the free energy barrier. The memory loss requirement will be most difficult to fulfill for a system with a rough, descending free energy profile, and a dynamics without sufficient collisions to dissipate the high kinetic energy of the $(-+)$ paths. In that case, the slower $(++)$ paths might get trapped more easily in some local minimum than the $(-+)$ paths, thus requiring a careful examination of both the order parameter and the interface positions. On the downhill part of the free energy barrier inside the basin of attraction of state B any difference between the endpoint velocities of the $(++)$ and $(-+)$ paths is of minor importance, as almost all paths will reach B . As a result, the crossing probability shows a plateau after which we can neglect condition (4.16).

A quantitative indication of the fulfillment of the memory loss criterion can be obtained by defining a memory loss function (MLF), for instance the ratio of the two terms at both sides of the equality in Eq. (4.16). However, evaluation of this function is time consuming if it has to be applied for all possible interface separations. Fortunately, one can estimate the MLF for interface separations smaller than the chosen one during the rate constant calculation. If we use a fine grid of n_{sub} sub-interfaces between λ_{i-1} and λ_{i+1} (See Fig. 4.3), we can measure the MLF with a resolution of $\delta\lambda = \Delta\lambda/n_{sub}$ with $\Delta\lambda \equiv \lambda_i - \lambda_{i-1} = \lambda_{i+1} - \lambda_i$ ². The function $MLF_i(j\delta\lambda)$ with $j = 1 \dots n_{sub}$ can be calculated in the $\phi_{i\pm}$ ensemble during a PPTIS simulation. To do this, some care is required in case the path has multiple recrossings with interface i . Take for example the situation depicted in Fig. 4.3. Only one phase point (1) on the path between λ_{i-1} and λ_{i+1} belongs to the ensemble $\phi_{\lambda_i \pm \Delta\lambda} \equiv \phi_{i\pm}$. However, in the ensemble defined by the two most inner sub-interfaces $\phi_{\lambda_i \pm \delta\lambda}$ three points belong to the ensemble (1,2 and 3). All three of them can be used to measure $MLF(j\delta\lambda)$ for $j\delta\lambda < \Delta\lambda$. For every path in the $\phi_{i\pm}$ ensemble, loop over all sub-interfaces j . For each j ,

1. collect all the phase points that belong to the ensemble of $\phi_{\lambda_i \pm j\delta\lambda}$.
2. sample the MLF function consecutively for all the n points $\{x_0^{(1)}, x_0^{(2)}, \dots, x_0^{(n)}\}$ for which $\phi_{\lambda_i \pm j\delta\lambda}(x_0) \neq 0$.
3. continue the loop over j until $j = n_{sub}$.

Finally, generate a new path, and repeat the whole procedure.

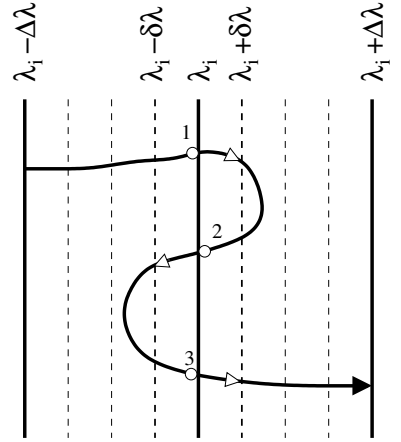


Figure 4.3: Calculation of the MLF on the grid of sub-interfaces. One possible path is shown confined between λ_{i-1} and λ_{i+1} .

²To simplify notation we assume here an equidistant interface separation for all interfaces. One is, however, by no means restricted to do so and one can place each interface at an optimum position concerning efficiency, memory loss and ergodic sampling.

The above algorithm does not obey complete detailed balance between phasepoints, which in principle can be achieved with the use of an additional MC move [109]. This however does not influence the result, and of course, detailed balance between the paths remains. Similar procedure of simultaneously sampling phasepoints of a path has also been applied in the free energy calculations of chapter 5.

4.3 Numerical Results

In sec. 3.3, the TPS and TIS methods were tested on a bistable diatomic molecule immersed in a fluid of purely repulsive particles. Here, we use the same system but with a longer barrier, to test the PPTIS method and the memory loss assumption, and to compare PPTIS and TIS.

4.3.1 The model

The system consists of N two-dimensional particles interacting via the Weeks-Chandler-Andersen (WCA) potential, Eq. (3.21), which we rewrite here

$$U_{WCA}(r) = \begin{cases} 4\epsilon[(r/\sigma)^{-12} - (r/\sigma)^{-6}] + \epsilon & \text{if } r \leq r_0 \\ 0 & \text{if } r > r_0, \end{cases} \quad (4.17)$$

where r is the interatomic distance, and $r_0 \equiv 2^{1/6}\sigma$. In the following we will use reduced units so that the energy and length parameters ϵ and σ , the mass of the particles and the unit of time $(m\sigma^2/\epsilon)^{1/2}$ are all equal to unity. In addition, two of the N particles are interacting through a diffusive double well potential

$$U_{dw}(r_d) = \begin{cases} U_{dw}(r_d) & \text{if } r_d < r_0 + w \\ h & \text{if } r_0 + w < r_d < r_0 + w + b, \\ U_{dw}(r_d - b) & \text{if } r_d > r_0 + w + b \end{cases} \quad (4.18)$$

where

$$U_{dw}(r_d) = h[1 - (r_d - r_0 - w)^2/w^2]^2. \quad (4.19)$$

and r_d specifies the dimer interparticle separation. This potential and its first derivative are continuous and the forces are therefore well defined. It has two minima at $r_d = r_0$, the compact state or state *A*, and at $r_d = r_0 + 2w + b$, the extended state or state *B*. The minima are separated by a total barrier of length $b + 2w$ and height h . For sufficiently large values of h , transitions between the states become rare and the rate constants are well defined. For sufficiently large values of b , trajectories on the barrier plateau become diffusive. See a trajectory at high energy in Fig. 4.4. We therefore expect this system to be a good test case for the new PPTIS method.

We simulate the system at constant energy $E/N = 1.0$ in a square box with periodic boundary conditions. The number density is fixed at 0.7, by adjusting the size of the box. The barrier length should always be less than half the box's edge, implying the number of particles N to increase accordingly with the value of the barrier length b . The remaining barrier parameters are set to $h = 15$ and $w = 0.5$. The total linear

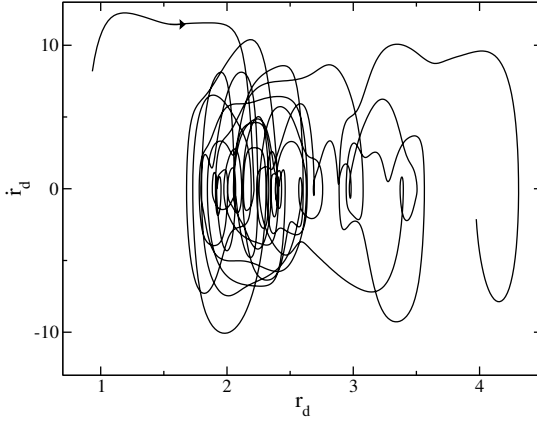


Figure 4.4: A trajectory in the r_d, \dot{r}_d plane for the dimer model. This is the outcome of an MD simulation at total energy $E/N = 10$. The system starts in A , overcomes the barrier, wanders considerably in the region of flat barrier and then ends in B .

momentum is conserved and is set to zero. The equations of motion are integrated using the velocity Verlet algorithm with a time step $\Delta t = 0.002$. The Monte Carlo path sampling is carried out both in PPTIS and TIS by means of the shooting move and the path-reversal move, as explained in Sec. 4.2 and sec. 3.2. The two moves were performed with an equal probability of 50%. The intermolecular distance r_d is a suitable order parameter λ to define the interfaces.

4.3.2 The Simulation

We simulated a system of $N = 100$ WCA particles with a barrier length $b = 2$. The minima of $U_{ddw}(r_d)$ are located at $r_d \simeq 1.12$ and $r_d \simeq 4.12$, and the diffusive plateau extends from $r_d \simeq 1.62$ to $r_d \simeq 3.62$. State A is defined by interface λ_0 as $r_d < 1.22$ and state B by interface λ_{17} as $r_d > 4.02$. In the intermediate regime 16 interfaces were chosen at $r_d = 1.24, 1.34, 1.40, 1.46, 1.52, 1.62, 2.02, 2.42, 2.82, 3.22, 3.62, 3.72, 3.78, 3.84, 3.90$, and 4.00 .

First, we ran straightforward MD simulations in state A and B to compute the fluxes that appear in both Eq. (4.8) and (3.19) by counting the number of positive crossings through interfaces λ_1 and λ_{16} , respectively, see sec. 3.2.1. We obtained the values $\langle \phi_{1,0} \rangle / \langle h_A \rangle = 0.1160 \pm 0.0008$ and $\langle \phi_{16,17} \rangle / \langle h_B \rangle = 0.117 \pm 0.001$. Subsequently, we calculated the conditional probabilities (4.3). For PPTIS we calculated the one-interface crossing probabilities for all the 16 interfaces on the barrier, while TIS simulations show convergence after 11 windows for both the forward and the backward reaction path. In Fig. 4.5 we report the one-interface crossing probabilities p_i^\pm, p_i^\mp and the long-distance crossing probability P_i^\pm, P_i^\mp . The long-distance crossing probabilities appearing in the rate constant Eq. (4.8) for $n_I = 17$ are $P_{n_I}^+ = (2.37 \pm 0.06)10^{-9}$ and $P_{n_I}^- = (1.67 \pm 0.03)10^{-9}$. These values can be compared with their TIS counterparts $P_A(n_I|1) = (2.4 \pm 0.2)10^{-9}$ and $P_B(0|n_I - 1) = (1.74 \pm 0.05)10^{-9}$. We note that because for the first 5 interfaces $i = 1 \dots 5$, p_i^\pm equals unity, P_i^- is constant up to $i = 6$. Similarly, for $i = 11 \dots 16$, p_i^\pm is unity and P_i^+ shows a plateau starting at $i = 12$. This means that in the PPTIS methods, although for the equilibrium constant C all the windows are necessary, the separate computation of k_{AB} and k_{BA} requires fewer windows. The result is consistent with what we found in TIS. We report in table

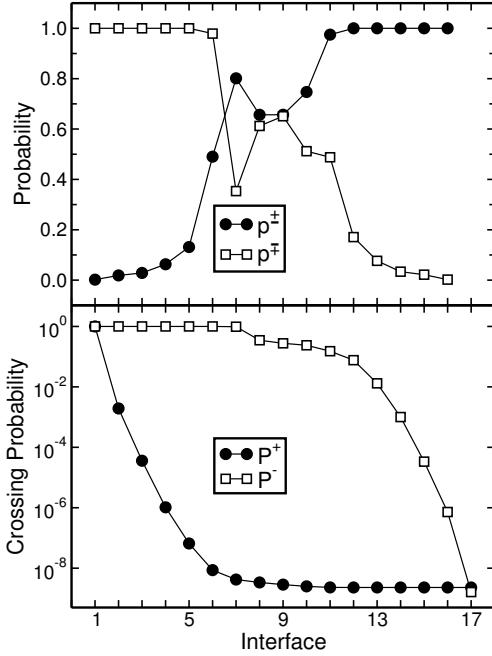


Figure 4.5: Top: PPTIS one-interface crossing probabilities p^\pm , p^\mp , see Eq. (4.3). The p^\pm , p^\mp probabilities follow directly from Eq. (4.4). Bottom: PPTIS long-distance crossing probabilities P_i^+ , P_i^- , see Eq. (4.5). The last points contribute to the rate constants as in Eq. (4.8). In both graphs the error is within symbol size.

	$k_{AB}/10^{-10}$	$k_{BA}/10^{-10}$	C
PPTIS	2.75 ± 0.07	1.95 ± 0.04	1.41 ± 0.05
TIS	2.8 ± 0.2	2.03 ± 0.06	1.4 ± 0.1

Table 4.1: Comparison of PPTIS and TIS. Forward and backward rate constants as well as the equilibrium constant are reported for the system with short energy barrier. The rate constant of $2.8 \cdot 10^{-10}$ corresponds in real units for Argon to 7.7ms.

4.1 the final rate and equilibrium constants. They all coincide within the statistical error.

The equilibrium constant C can also be derived by a free energy computation using the relation $C = \exp(\beta\Delta F)$ where ΔF is the free energy difference between states A and B . We postpone this additional test to chapter 5. Besides computing the free energy with traditional methods, we will show that the PPTIS method, created to compute dynamical quantities like the rate constants, can also be adapted to provide an equilibrium information such as the free energy.

4.3.3 Comparing TIS and PPTIS

In order to make an efficiency comparison between the two methods, we chose to estimate the computational effort for a certain fixed error. We rather calculate the error in the equilibrium constant $C = k_{AB}/k_{BA}$ instead of in the rate k_{AB} itself because the expression of C in terms of the averages, that have to be calculated

separately, is much simpler than the recursive expression (4.7) of k_{AB} . Hence, the error propagation from the error in the individual terms is simpler and yields a more transparent comparison with TIS. Because of relations (4.8) and (E.9) the equilibrium constant C can be written as:

$$C_{\text{PPTIS}} = \frac{[\langle \phi_{1,0} / \langle h_A \rangle]}{[\langle \phi_{n_I-1, n_I} \rangle / \langle h_B \rangle]} \left[\frac{p_{n_I-1}^{\pm}}{p_{n_I-1}^{\mp}} \right] \cdots \left[\frac{p_1^{\pm}}{p_1^{\mp}} \right] \quad (4.20)$$

Each term within brackets [...] is calculated separately together with its error. The error propagation of the total $n_I + 1$ terms determines the final overall error. Similarly, using (3.19) in TIS the expression for C can be written as:

$$C_{\text{TIS}} = \frac{[\langle \phi_{1,0} / \langle h_A \rangle]}{[\langle \phi_{n_I-1, n_I} \rangle / \langle h_B \rangle]} \frac{[P_A(n_I | n_I - 1)] \cdots [P_A(2 | 1)]}{[P_B(0 | 1)] \cdots [P_B(n_I - 2 | n_I - 1)]} \quad (4.21)$$

Here, in total $2n_I$ simulations have to be performed, each on a different ensemble. In practice, however, not all the interface ensembles are needed, as $P_A(i | i - 1)$ and $P_B(i | i + 1)$ will converge to unity in the limit $i \rightarrow n_I$ and $i \rightarrow 0$, respectively.

In both Eqs. (4.20) and (4.21) the PPTIS and TIS the final equilibrium constant is a product of factors. We determined each factor independently by performing N_{bl} simulation blocks of N_{cy} Monte Carlo cycles. We adjusted N_{cy} so that the relative standard deviation of each term after N_{bl} block averages was an arbitrary value of 3%. We measured, under the same computational conditions (1.4 GHz AMD Athlon), the CPU-time required and summed up all the times to get the relative efficiency. The final errors on the rate constants given above were obtained by standard propagation rules using *all* the available blocks of simulations. We computed the computation times to reach the prefixed 3% error for each factor in Eqs. (4.20) and (4.21) and found that for the simple dimer system the efficiency of PPTIS is a factor 2 higher than TIS.

In figure 4.6 we plot the average path-length in each window for the two methods. The direct comparison shows that on the barrier PPTIS keeps the path length constant while the TIS path length increases. This is expected but it does not directly imply a gain in efficiency. In Eq. (4.20) for PPTIS the error in the terms $[p_i^{\pm} / p_i^{\mp}]$ will be more or less the same for all i on the barrier. In Eq. (4.21) For TIS, however, the error in $[P_{A,B}(i | i \pm 1)]$ will decrease when its value gets closer to unity. The final efficiency ratio is a balance between pathlength and the errors in the terms. The analysis of the efficiency and its scaling with the diffusive barrier length requires some care and we defer it to the more general section 6.1.

4.3.4 Validity of the memory loss assumption

We computed the memory loss function $\text{MLF}(j\delta\lambda)$ as defined in section 4.2.1. We used a central interface at $r_d = 2.62$ and $\delta\lambda = 0.01$ and j ranging from 1 to 100, corresponding to the entire length of the barrier plateau. Since not only the mean value of the endpoint velocity $\dot{\lambda}$ but its complete probability distribution $f(\dot{\lambda})$ should be equal for paths of the ensemble $(-+)$ and $(++)$ we computed the overlap

$$\int_{-\infty}^{+\infty} \sqrt{f^{\pm}(\dot{\lambda}) f^{\mp}(\dot{\lambda})} d\dot{\lambda}. \quad (4.22)$$

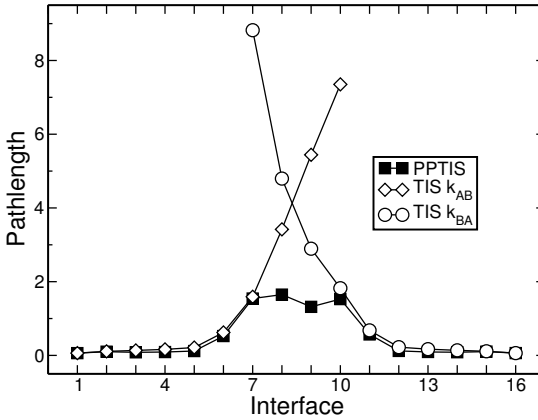


Figure 4.6: Comparison of pathlengths for PPTIS and the TIS simulations for the calculation of the forward and backward rate constant. Because of the diffusive character of the system, the TIS path-lengths keep growing as the interface moves further from the initial stable state. The PPTIS path-lengths on the contrary stay constant. The errors are within the symbol size.

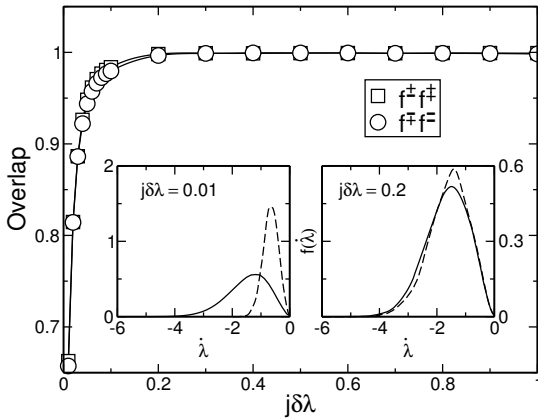


Figure 4.7: Memory loss function computed using the overlap of the distributions of the endpoint velocity $\dot{\lambda}$, see Sec. 4.3.4. In the insets we plot the distributions for paths of the $(+-)$ ensemble (solid line) and the $(--)$ one (dashed line), for two different window sizes $j\delta\lambda = 0.01$ and $j\delta\lambda = 0.2$. The first two distributions are different, and the second ones are almost overlapping.

Similar expression was used for the paths $(+-)$ and $(--)$. The results are reported in Fig. 4.7. It can be seen that for $j\delta\lambda \geq 0.2$ the memory loss assumption is satisfied. Consequently an interface separation of 0.2 is what we used for the simulation of sec. 4.3.2.

4.4 Simple PPTIS

In PPTIS the transition event is approximated by a sequence of hopping events between subsequent interfaces, see sec. 4.1. The hopping probabilities are computed in a path sampling simulation involving three subsequent interfaces, see sec. 4.2. We show here that the method still works when only two interfaces are employed, leading to a simplified version of PPTIS. We give the theoretical derivations and test them on a one particle system. We conclude discussing the applicability of this simplified PPTIS method.

4.4.1 Theory

As usual in interface sampling methods, we partition the phase space by means of $n_I + 1$ interfaces $0 \dots n_I$ defined by the hypersurfaces $\{x : \lambda(x) = \lambda_i\}$, see sec. 3.1.1

and Fig. 3.1. Interface λ_0 is the border of state A , λ_{n_I} is the border of state B , x is the system phase space point and $\lambda(x)$ the order parameter, see sec. 2.1.2.

A PPTIS window consists of three interfaces $i - 1$, i , $i + 1$ and simulations are carried out in the ensemble of all the paths starting and ending at the outer interfaces $i - 1$ or $i + 1$ and crossing the central one i at least once. Here, the idea is to use just *two* interfaces, by taking the limits $i \rightarrow (i - 1)^+$, or $i \rightarrow (i + 1)^-$. We call the two resulting interfaces l , r . The path ensemble consists of all paths that either cross interface left in the positive direction or interface right in the negative direction. In this way, it still makes sense to talk of paths of type $(-+)$, $(+-)$, $(--)$, $(++)$, see sec.4.2 and the whole PPTIS machinery can be readjusted in the limit accordingly.

The starting point is again the probability $P_m^{(l|i_j)}$, eq. (3.13), which is the probability for the system to reach interface l before m under the condition that it crosses at $t = 0$ interface i , while coming directly from interface j in the past. Instead of the one-hop probabilities Eq. (4.3) we define the local, short-distance probabilities as

$$\begin{aligned} p_{lr}^{\pm} &\equiv P(l^{\pm}|l^{\pm}), & p_{lr}^{\mp} &\equiv P(l^{\mp}|r^{\pm}) \\ p_{lr}^{\pm\pm} &\equiv P(l^{\pm}|l^{\pm}), & p_{lr}^{\mp\mp} &\equiv P(l^{\mp}|r^{\mp}), \end{aligned} \quad (4.23)$$

where r^+ means right of r , l^- left of l , and $\lambda_0 \leq \lambda_l < \lambda_r \leq \lambda_{n_I}$. In this notation the subscripts l, r are general integers, they do not have to be consecutive, and therefore the probabilities (4.23) are in general not necessarily *one-hop* probabilities. The global, long-distance probabilities (4.5) become

$$P_j^+ \equiv P(j|_0^0), \quad P_j^- \equiv P(j|_j^j). \quad (4.24)$$

For instance, $P_{n_I}^+$ is the probability of reaching n_I before 0 provided that one crosses interface 0 in the positive direction. $P_{n_I}^+$ is the same as taking $\lambda_1 = \lambda_0 + \epsilon$ in the flux calculation of TIS or PPTIS (see sec. 3.2.1). We show in appendix E.3 that using the assumption of memory loss the PPTIS recursive relations (4.7) still hold, and become

$$\begin{aligned} P_r^+ &\approx \frac{p_{lr}^{\pm} P_l^+}{p_{lr}^{\pm} + p_{lr}^{\mp} P_l^-} \\ P_r^- &\approx \frac{p_{lr}^{\mp} P_l^-}{p_{lr}^{\pm} + p_{lr}^{\mp} P_l^-}, \end{aligned} \quad (4.25)$$

with $P_0^+ = P_0^- = 1$.

Implementation of formulas (4.23), (4.25) can be done along the same lines of their PPTIS counterparts, see sec. 4.2. We just add a remark about the memory loss assumption. In the simple PPTIS method subinterfaces can be used to devise a simpler test of the assumption than the computation of the memory loss function of sec. 4.2.1. This is done as follows. Take a first window between $l = l_1 = 0$ and $r = r_1$. In between define a finer grid of subinterfaces $l \leq s \leq r$, and sample p_{ls}^{\pm} and p_{ls}^{\mp} . The sampling can be done simultaneously for all of the subinterfaces. Similar to TIS, one can use one path for the sampling of many interfaces. The p_{ls}^{\pm} is straightforward, because one just has to find the λ_{max} of the path, and fill in an histogram up to

that value, as explained in sec. 3.2.3. The p_{ls}^\mp requires more care, but is doable³. In this first window, Eqs. (4.25) say that $P_s^+ = p_{0s}^\pm$, $P_s^- = p_{0s}^\mp$. Then take a second window, that overlaps with the first, from $l = l_2$ with $\lambda_{l_1} < \lambda_{l_2} < \lambda_{r_1}$, to $r = r_2$ with $\lambda_{r_2} > \lambda_{r_1}$, and compute $p_{l_2,s}^\pm$ and $p_{l_2,s}^\mp$. Using (4.25) with $P_{l_2}^+$, $P_{l_2}^-$ fixed, given from the previous window, we get P_r^+ , P_r^- up to $r = r_2$. In the overlapping region, between r_1 and l_2 , one should get the same curve, if memory loss is satisfied. This procedure can be repeated for all pairs of windows, and gives a simple test of the validity of the memory loss within the limits of the windows.

Finally, in the simple PPTIS method the calculation of the free energy becomes trivial. Windows are now unbiased ensembles and the loop-boundary mechanism of chap. 5 is not needed. One simply has to histogram λ , and then rematch the histograms of each window, as in normal umbrella sampling, see sec. 2.3.1.

4.4.2 Numerical results

We test the simple PPTIS method on a system of one particle in a two dimensional potential. Although extremely simple, this system shows some features of interest of its own.

The model system

We consider the system proposed in [80], sec. V.A, and [11], case study 24, p. 456. One particle experiences the two-dimensional potential

$$V(x, y) = \frac{1}{6} [4(1-x^2-y^2)^2 + 2(x^2-2)^2 + ((x+y)^2-1)^2 + ((x-y)^2-1)^2 - 2] \quad (4.26)$$

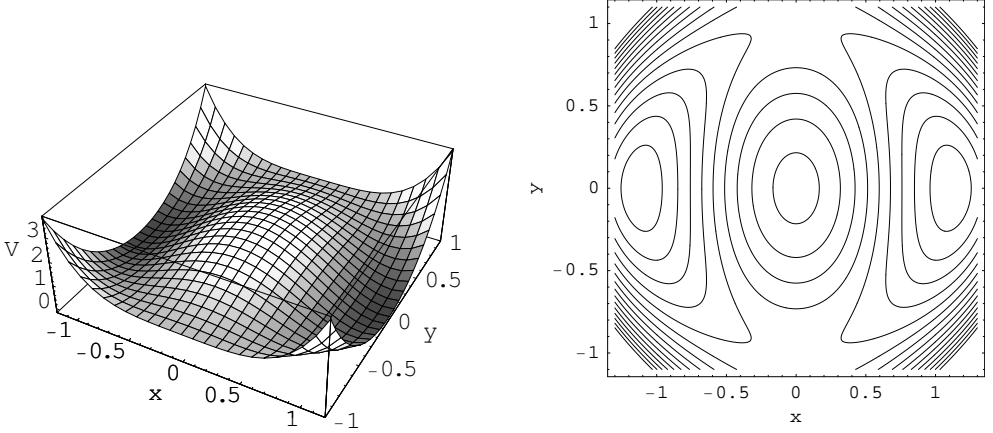
We report in fig. 4.8 a 3-d and a contour plot of the surface. There are two minima at $V(\pm\sqrt{5}/2, 0) = -1/12$, a maximum at $V(0, 0) = 2$, and two saddle points at $V(0, \pm 1) = 1$. The saddles are degenerate, the hessian has eigenvalues (0, 8), and thus the saddles are almost flat.

We consider the system in the canonical ensemble for a temperature $\beta^{-1} = T = 0.1$. Some results can be obtained analytically. We define the order parameter $\lambda(x, y) \equiv x$ and consider the free energy

$$\begin{aligned} \exp(-\beta F(\lambda)) \equiv P(\lambda) &= \langle \delta(\lambda(x) - \lambda) \rangle \\ &= \frac{1}{Z} \int dx dy e^{-\beta V(x, y)} \delta(x - \lambda) \\ &= \frac{1}{Z} \int dy e^{-\beta V(\lambda, y)} \end{aligned} \quad (4.27)$$

By numerical integration of Eq. (4.27), we obtained a theoretical free energy curve. We define state A using interface $\lambda_0 = -0.85$ and we take $\lambda_1 = \lambda_0$. State B is symmetrically defined, taking $\lambda_{n_I} = 0.85$ and $\lambda_{n_I-1} = \lambda_{n_I}$. The flux $\langle \phi_{1,0} \rangle / \langle h_A \rangle$ can

³ p_{ls}^\mp is not really the symmetric counterpart of p_{ls}^\pm , which would use λ_{min} . A (+-) path or a (-) that crosses s both contribute, but the number of crossing $s^+ \rightarrow s$ must also be counted

Figure 4.8: $V(x, y)$

λ_1	$P(\lambda_1)$	$P(\lambda_1)_{x \in \mathcal{A}}$	$\langle \theta(\dot{\lambda}) \dot{\lambda} \rangle_{\lambda_1}$	$\langle \phi_{1,0} \rangle / \langle h_{\mathcal{A}} \rangle$	$\beta F(\lambda_1) = -\ln P(\lambda_1)$
-0.85	0.102	0.204	0.126	0.0258	2.28

Table 4.2: Some theoretical results

also be obtained theoretically. Using the definition of effective positive flux Eq. (3.10) we can rewrite

$$\begin{aligned}
 \frac{\langle \phi_{1,0} \rangle}{\langle h_{\mathcal{A}} \rangle} &= \frac{\langle h_{0,1}^b | \dot{\lambda} | \delta(\lambda - \lambda_1) \rangle}{\delta(\lambda - \lambda_1)} \cdot \frac{\delta(\lambda - \lambda_1)}{\langle h_{\mathcal{A}} \rangle} \\
 &= \langle \dot{\lambda} \theta(\dot{\lambda}) \rangle_{\lambda=\lambda_1} P(\lambda_1)_{x \in \mathcal{A}}
 \end{aligned} \tag{4.28}$$

where because of the presence of $h_{0,1}^b$ only positive velocities $\dot{\lambda}$ are selected. Because of general symmetry reasons $h_{\mathcal{A}} = 1/2$ and as a consequence $P(\lambda) = P(\lambda)_{x \in \mathcal{A}}/2$. Moreover, the equilibrium constant equals $C = k_{AB}/k_{BA} = 1$. The other term in Eq. (4.28) is simply $\langle \theta(\dot{\lambda}) \dot{\lambda} \rangle_{\lambda_1} = 1/\sqrt{2\pi\beta}$, see sec. 1.4.3. We summarize the theoretical results in table 4.2.

Path Sampling

We defined the interfaces as in table 4.3. We simulated the system in the canonical ensemble at $T = 0.1$, using both PPTIS and simple PPTIS. The path sampling for both methods is performed using constant energy NVE dynamics and initial conditions extracted from a canonical distribution, as explained in sec. 4.2 and 3.2. Moreover, we performed canonical path sampling, by employing the shooting move with a Nosé-Hoover thermostat [73], as explained in sec. 2.2.1.

All dynamical schemes were integrated using a velocity-Verlet integrator [11] with time-step $\Delta t = 0.01$. The free energy histogram had a bin width $\delta\lambda = 0.001$. We used the shooting and reversal move with equal probability, 50% and harvested 10^7 paths

$\lambda_0 = \lambda_1$	-0.85	λ_5	-0.53	λ_9	0.01	λ_{13}	0.69
λ_2	-0.77	λ_6	-0.45	λ_{10}	0.45	λ_{14}	0.77
λ_3	-0.69	λ_7	-0.01	λ_{11}	0.53	$\lambda_{15} = \lambda_{16}$	0.85
λ_4	-0.61	λ_8	0	λ_{12}	0.61		

Table 4.3: Choice of interfaces for the 2d potential. They are more dense on the shoulders of the free energy curve and wider in the middle, see 4.11. The small window around 0 is to avoid metastable states (see text).

per window, after an equilibration of 10^5 paths. For normal PPTIS, the acceptance for the shooting move was around 70%. For simple PPTIS, NVE dynamics, acceptance for the shooting move was around 80%. For simple PPTIS, Nosé-Hoover dynamics, acceptance for the shooting move was around 90% because there is no canonical MC acceptance involved.

Two peculiarities were encountered in this system. First, since $V(x, y)$ has two saddle points at $(0, \pm 1)$, the particle can go from A to B over the barrier following two channels. During the transition the trajectories, that are confined in windows defined by $\lambda(x, y) = x$, choose only one of the two, as can be seen from the average y coordinate in the window, plotted in fig. 4.9. Because of the symmetry of the system, this does not seem to influence much the result, but indicates bad path sampling. In sec. 4.5 we give a possible solution to this problem. Second, since the saddles are almost flat, the system tends to be trapped in quasi-periodic orbits around $x = 0$, of the form displayed in fig. 4.10. We have found cyclic x -oscillations up to the range $-0.1, 0.1$. As a result the paths get too long and slow down the simulation considerably. Sometimes they become even too long to be stored in memory. In fact, in such a system with low number of degrees of freedom a non ergodicity can be expected. In order to prevent this, we put a smaller window around 0, making the

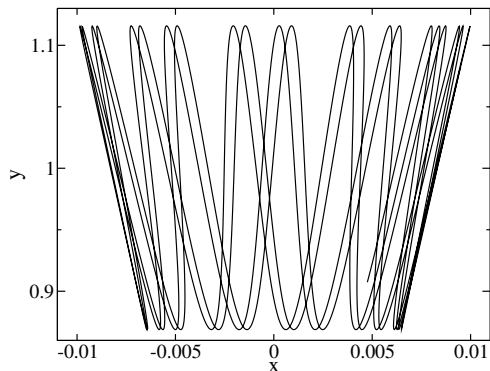
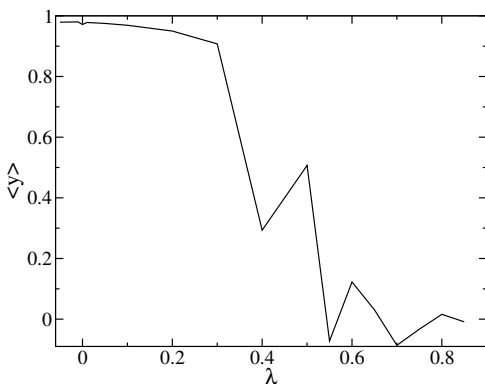


Figure 4.9: Choosing one way. From right to left: first $\langle y \rangle$ is around 0, then it chooses the upper saddle point at $(0, 1)$. **Figure 4.10:** A metastable state around the saddle point at $(0, 1)$.

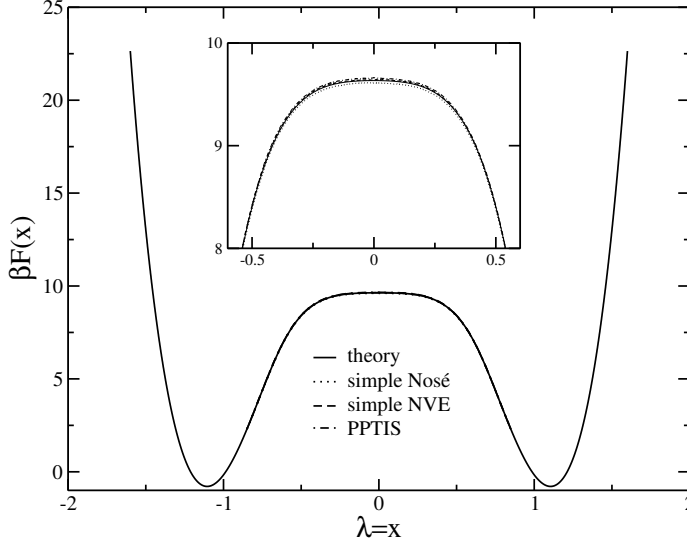


Figure 4.11: Comparing the three methods with theory. The first point at -0.85 is taken from theory. In the inset an enlargement of the middle part. All methods agree quite well.

method	$P_A(n_I 1)/10^{-4}$	$P_B(0 n_I - 1)/10^{-4}$
PPTIS	6.16	6.18
simple NVE	6.13	6.21
simple Nosé	6.40	6.38

Table 4.4: The long-distance probabilities for all the methods employed.

system escape it in a reasonable time.

We report in fig. 4.11 the free energy results and in table 4.4 the long-distance probabilities for all the methods we used. Taking the average of all the probabilities and the flux at -0.85 from the theory, see table 4.2, we get for the rate $k_{AB} = k_{BA} = (1.61 \pm 0.02)10^{-5}$. In [11] a value of $8.0 \cdot 10^{-6}$ was found. We believe the discrepancy is due to a failure of the memory loss assumption. Indeed, when computing the long-distance probabilities P_r^+, P_r^- , Eq. (4.25), with the use of subinterfaces, we found a little difference in the rematching of different windows, as explained in sec. 4.4.1. This could have been expected, as this simple potential probably shows not much dissipation nor memory loss.

4.4.3 Resume

Basically we have modified PPTIS into an hybrid MC scheme. The implementation is simpler and allows a straightforward calculation of the free energy profile along the order parameter. Almost every path created with a shooting or reversal move is

accepted. Paths are rejected only if they become too long, or due to the Boltzmann weight in case of NVT simulations. The implementation of subinterfaces allows for a simpler test of the memory loss, using the window rematching on the crossing probability, besides the free energy. We think that such an additional test could be devised for standard PPTIS as well.

However, there is a problem with simple PPTIS. Although we have not tried a rigorous test, a preliminary run of simple PPTIS on a real complex system failed to compute a rate constant or a free energy. We applied the method to study crystal nucleation of a Lennard-Jones system, see chap. 7. For a given window when the system was still in the undercooled phase the paths tended to become very short and cluster around the left interface boundary. Because of the uphill free energy barrier we ended up with the shooting move generating always a path made of 1 timeslice. Such a bad sampling is due to the lack of the constraint to cross the middle interface, that prevents a minimum amount of timeslices big enough to allow the shooting move a proper sampling of the path space.

We have presented the method for completeness and pedagogical reasons, but we have to stress the use of simple PPTIS in real applications might encounter problems. Although tempting for its simplicity, it might end up in being too simple.

4.5 Parallel path swapping

A good sampling of the transition path ensemble relies on the ability to generate a relevant subset of all the possible transitions, in the same way that in general a good sampling of the possible states of a system relies on the ability to explore all the relevant zones of the phase space. This can be problematic in the presence for example of multiple reaction channels, as we saw in the simple system of sec. 4.4.2.

Parallel tempering techniques (also known as Replica Exchange methods) can facilitate the sampling [81], but require a rather large computational effort and cannot be applied at constant energy. Here, we propose a less expensive parallel method especially tailored for PPTIS. This parallel path swapping (PPS) technique is based on the exchange of paths between two subsequent interface ensembles. Fig. 4.12 shows one path in the λ_i ensemble, consisting of all possible paths crossing λ_i while starting and ending at either λ_{i-1} or λ_{i+1} , and one in the λ_{i+1} ensemble consisting of all paths crossing λ_{i+1} at least once, while starting and ending at either λ_i or λ_{i+2} . The second half of the path in the λ_i ensemble is automatically a path for the λ_{i+1} ensemble. Using the notation of sec. 4.2, when the second half is moved to the λ_{i+1} ensemble, one just needs to elongate it forward in time to find out if it is a $(-+)$ or a $(--)$ path. In the case in figure it is a $(-+)$ path in the λ_{i+1} ensemble. Similarly, the first half of the path in the λ_{i+1} ensemble is automatically a path for the λ_i ensemble. When moved to the λ_i ensemble, one has to elongate it backward in time to find out if it is a $(-+)$ path or a $(++)$ path. In the case in figure it is a $(++)$ path.

The idea depicted in Fig. 4.12 can be transposed into a new MC move that attempts swapping the current path of the λ_i ensemble with that of the λ_{i+1} ensemble. This path swapping move requires some care to satisfy overall detailed balance. In the case in figure for example we have to pay attention when attempting the reverse move,

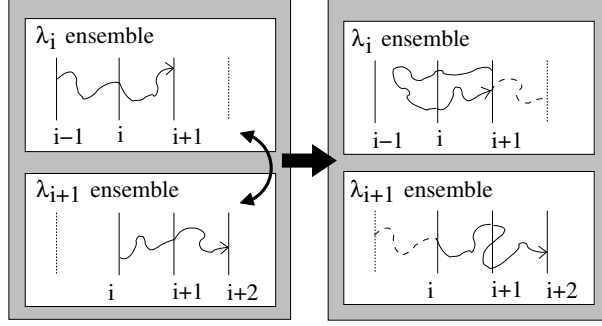


Figure 4.12: Path swapping move for PPTIS. The last half of the path in the λ_i ensemble and the first half of the path in the λ_{i+1} are swapped to the λ_{i+1} and λ_i ensembles, respectively.

because the path of ensemble λ_i has two possible contributions for the λ_{i+1} ensemble. Consider the zone common to the ensembles, between interfaces i and $i + 1$. Define for any path of window i or $i + 1$ the quantity

$$N_{piece} \equiv \text{Number of parts that directly connect } i \text{ and } i + 1 \quad (4.29)$$

This quantity can be either 0, 1 or 2. For paths in the λ_i ensemble is 0 for $(--)$ paths, is 1 for $(-+)$ and $(+-)$ paths, and is 2 for $(++)$ paths. For paths in the λ_{i+1} ensemble is 0 for $(++)$ paths, is 1 for $(-+)$ and $(+-)$ paths, and is 2 for $(--)$ paths. When $N_{piece} = 0$ the path cannot be swapped, when $N_{piece} = 2$ one of the possible parts is chosen with probability $1/2$. However, to satisfy detailed balance on the reverse move we have to introduce an additional acceptance probability. Consider moving a path from the λ_i ensemble to the λ_{i+1} ensemble, the acceptance probability reads

$$\min \left[1, \frac{N_{piece}^{(o,i)}}{N_{piece}^{(n,i+1)}} \right] \quad (4.30)$$

where in the superscripts 'o' and 'n' stand for old and new. In practice this means that when a path can be swapped, i.e. $N_{piece}^{(o)} \neq 0$, if N_{piece} does not increase on going from old to new ($1 \rightarrow 1, 2 \rightarrow 1$), the swap move is automatically accepted, otherwise if N_{piece} increases ($1 \rightarrow 2$) an additional acceptance probability of $1/2$ is required. Note that in order to calculate 4.30 for the new path, we must first complete the evolution in the λ_{i+1} ensemble, integrating forward in time. Similar reasoning holds when moving the other path from the λ_{i+1} ensemble to the λ_i ensemble.

Resuming a path sampling algorithm with a path swapping move is as follows. Suppose there are N_W windows. Naturally the swapping can occur only between neighboring windows, and there are $N_W - 1$ possibilities. Choose a number α between 0 and 1 and start the following loop

1. Draw a random number $y \in [0, 1]$. If $y < \alpha$ choose a random window $1 \dots N_W$ and perform a normal path sampling move. If $y > \alpha$ choose one of the $N_W - 1$ pairs of neighboring windows and perform a path swap move.

2. Compute $N_{piece}^{(o,i)}$ and $N_{piece}^{(o,i+1)}$. If one of the two is 0, the move is rejected, go back to step 1
3. If one of the above N_{piece} is 2, choose one of the possible pieces with probability 1/2. Swap the part of the paths belonging to the new ensemble and compute $N_{piece}^{(n,i)}$ and $N_{piece}^{(n,i+1)}$ by integrating forward and backward in time.
4. Accept the swap move with probability

$$\min \left[1, \frac{N_{piece}^{(o,i)}}{N_{piece}^{(n,i+1)}} \right] \min \left[1, \frac{N_{piece}^{(o,i+1)}}{N_{piece}^{(n,i)}} \right]$$

5. Sample path averages and repeat from step 1

PPS was introduced in [20]. As noticed by the authors, another advantage of PPS is that it allows to go beyond the pseudo-Markovian description of PPTIS. Fig. 4.12 shows that the paths at the right hand side, if we include the dashed trajectory part, can connect four interfaces instead of only three. This extension allows for a long range verification of the memory loss assumption. Moreover, while PPS can be very effective when the confinement of short paths in PPTIS can cause sampling problems, even TIS and TPS algorithms might benefit from path swapping when multiple reaction channels exist.

4.6 Summary

In this chapter we have adapted the theory and algorithms of TIS, chapter 3, for the efficient calculation of rate constants of two state activated processes with a diffusive barrier. Like in TIS the phase space is divided in regions separated by interfaces. TIS computes directly the overall transition rate k_{AB} from the first interface, corresponding to stable state A , to the last interface, corresponding to stable state B . PPTIS instead obtains the rate by means of an iterative scheme based on the hopping probabilities from one interface to the next. Using either the iterative scheme given here or for more general hopping networks the method of kinetic Monte Carlo, one can solve the master equation and obtain the final forward and backward rate constants. In deriving this algorithm we assumed complete memory loss between interface, such that the system becomes essentially Markovian, thus validating the use of kinetic Monte Carlo and similar algorithms. We showed that for a relatively simple system, the diatomic molecule, the memory loss assumption (loss of correlation) holds over the entire barrier. We expect that for more complex systems this memory loss requirement will certainly be fulfilled, provided that the dynamics has a stochastic character and the interfaces are placed sufficiently far apart. However, the choice of order parameter requires still some caution, possibly more than in TIS, in order to satisfy the memory loss requirement. For the simple dimer system, we showed that PPTIS is already twice as fast as TIS.

In sec. 4.4 we presented a variation of PPTIS that is able to simultaneously compute the kinetics and the free energy of the rare events. The simplicity of the methods

unfortunately hinders its application in real systems. However since the free energy is an important and useful additional information we adapted the idea to the standard PPTIS. Before switching in the next chapter to the problem of free energy computations in PPTIS, we add some remarks concerning the literature.

The method advocated here to tackle diffusive barriers in complex systems is not the first one that has been proposed in the literature. Several techniques have been put forward in the last decade, for instance the diffusive barrier algorithm by Ruiz-Montero et al. [17], briefly reviewed in sec. 1.5.3, and the coarse grained MD method by Hummer and Kevrekidis [45]. The latter technique uses short trajectories to calculate the average force projected on an order parameter space. They use that force to integrate a stochastic equation of motion and explore the free energy landscape in that way. Rate constants can then in principle be obtained from the dynamics on this coarse grained surface.

A similar method to PPTIS is the *milestoning* method of Faradjian and Elber [71]. The two methods are very similar, but differ on two crucial points. First, the milestoning method assumes a complete loss of memory at each interface, as in simple PPTIS. At each interface the system can hop either to the right or to the left with a certain probability and these probabilities do not depend on the history of the path. This amounts to assuming that the coordinates orthogonal to λ are at each interface distributed according to the equilibrium distribution. PPTIS instead takes a stronger history dependence into account. At each interface memory effects may persist but not much longer than the time needed to travel from one interface to the other. As mentioned in sec. 4.4, this stronger history dependence of full PPTIS was found necessary to investigate a case of crystal nucleation in a Lennard-Jones system (see chapter 7). Second, the milestoning approach puts, on the other hand, more effort in describing the time evolution on the barrier by using time dependent hopping probabilities. These are required if one wants to study, for instance, the decay of a distribution that is initially out of equilibrium, or the diffusion behavior on the barrier. This time aspect introduces another history dependence [71], which is absent in PPTIS where the final crossing probability is a quantity independent of time. This is justified by the fact that PPTIS always assumes that the barrier is low populated. Hence, the time that the system spends on the barrier can be long from a computational perspective, but is still negligible compared to the expected time the system needs to enter the barrier plateau region from one of the stable states. In principle, this condition should always be satisfied for a system that shows exponential decay and, hence, has a well defined rate, but, of course, systems that do not obey these conditions can still be interesting to study. To summarize, both methods are very similar, but each one is more accurate to one of the points described above. However, the two aspects do not exclude each other and could easily be merged into a single algorithm if needed.

We conclude stressing that there is a large difference between the reactive flux method based on transition state theory and the PPTIS technique. Although we use hyper-surfaces to divide the phase-space we do not rely on a global large transmission coefficient. Instead, we calculate local transmission coefficients and use those as hopping probabilities.

5

Interface Sampling and Free Energies

Things should be made as simple as possible, but not any simpler.

Albert Einstein

We introduce a method to evaluate simultaneously the reaction rate constant and the free energy profile of a process in complex environment. The method employs the partial path transition interface sampling technique we presented in chapter 4 for the calculation of rate constants in diffusive systems. We illustrate the applicability of the technique by studying the same system of sec. 4.3, a simple dimer in a repulsive fluid, and we show that the free energy can be obtained at no additional computational cost.

5.1 Theory

The method is based on the theory of (PP)TIS, already presented in sec. 3.1 and 4.1, but we briefly review it here.

Consider a complex system which undergoes a transition between two stable states A and B , separated by a high diffusive barrier. As the population on the barrier is very small, the system shows exponential two state behavior and has a well defined rate constant. The PPTIS method requires a set of $n_I + 1$ non-intersecting multi-dimensional interfaces $\{0, 1 \dots n_I\}$ described by an order parameter $\lambda(x)$, where x denotes the phase space point. We stress that the order parameter does not have to correspond to the real reaction coordinate. We choose λ_i , $i = 0 \dots n_I$ such that $\lambda_{i-1} < \lambda_i$, and that the boundaries of state A and B are described by λ_0 and λ_{n_I} , respectively, see Fig. 3.1.

The rate constants were derived in Eq. (4.8), which we rewrite here

$$k_{AB} = \frac{\langle \phi_{1,0} \rangle}{\langle h_A \rangle} P_{n_I}^+ \quad k_{BA} = \frac{\langle \phi_{n_I-1,n_I} \rangle}{\langle h_B \rangle} P_{n_I}^- \quad (5.1)$$

The fractions in the expression are the flux terms and can be calculated by counting

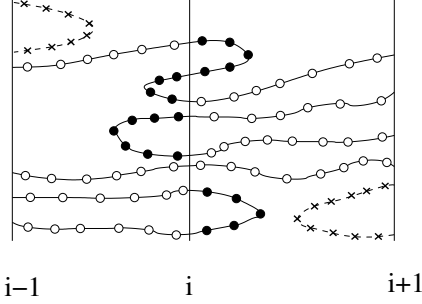


Figure 5.1: Illustration of *loop* and *boundary* points. The open circles denote *boundary* points with $b_i(x) = 1$, time-slices that hit a boundary in one time direction and reach λ_i in the opposite time direction. The *loop* points (full circles) with $l_i(x) = 1$ meet first the middle interface in both directions. For both *loop* and *boundary* points $w_i(x) = 1$. The dashed lines are paths that do not belong to the ensemble as they do not cross λ_i . Hence, the corresponding time-slices (crosses) are not part of the subset of phase points counted in the λ_i ensemble (thus $w_i(x) = 0$), but are *loop*-points of the neighboring interfaces.

the frequency of leaving A and B using straightforward MD, see sec. 3.2.1. The other factors $P_{n_I}^+$, and $P_{n_I}^-$ are the long distance probabilities defined in Eq. (4.5). In words, $P_{n_I}^+$ is the probability of leaving A and reach B before going back. $P_{n_I}^+$ is similarly defined for the reversed reaction $B \rightarrow A$. These terms can be calculated in a TIS simulation, see sec. 3.1.4. However, in case there is memory loss between the interfaces, PPTIS can approximate the long distance crossing probabilities by a recursive relation involving the one-hop probabilities (4.3) between neighboring interfaces, see Fig. 4.2. The algorithm then proceeds as follows. A PPTIS window is defined as three consecutive interfaces $i-1, i, i+1$. A window ensemble consists of all paths starting and ending from the outer interfaces and crossing at least once the middle one. The ensemble is sampled through a path sampling algorithm, and the one-hop probabilities are computed, see sec. 4.2. After repeating for all windows $i = 1 \dots n_I - 1$, the long distance probabilities are derived using Eq. (4.7).

Here, we want to show that besides the rate constant it is also possible to obtain the equilibrium free energy profiles along λ . The free energy in the canonical ensemble is defined in Eq. (1.34) as $\beta F(\lambda) \equiv -\ln P(\lambda)$, where β is the inverse temperature, and $P(\lambda) \equiv \langle \delta(\lambda(x) - \lambda) \rangle$ is the probability to find the system at a certain value of λ . In a generic ensemble the definition of free energy is simply generalized as $-\ln P(\lambda)$.

As shown in sec. 4.4, we could perform the PPTIS path sampling between two interfaces instead of three, and allow the path to be completely free, but stop integrating when an interface is hit. We would then essentially perform umbrella sampling (see sec. 2.3.1) between the interfaces using hybrid MC [117]. The PPTIS interfaces act as hard window boundaries. In that case, simply measuring the probability along the path to be at a value of λ and joining all histograms would suffice to obtain the entire free energy.

The problem in sampling $P(\lambda)$ in PPTIS is the fact that the PPTIS method introduces a bias, by restricting all paths in the ensemble to cross the middle interface. As we said before, the λ_i path-ensemble in the PPTIS formalism consists of all possible paths that start and end either at λ_{i-1} or λ_{i+1} and have at least one crossing with λ_i . The collection of all time-slices of these paths is a subset of the phase space points confined between λ_{i-1} and λ_{i+1} because we are missing the points around the outer interfaces, as shown in Fig. 5.1. We can correct for this by comparing neighboring interface ensembles. For this purpose, it is convenient to categorize the time-slices

into the *loop* type and *boundary* type of phase points as illustrated in Fig. 5.1. Here, we need the functions $h_{i,j}^{f(b)}(x)$, defined in Eq. (3.2), which are 1 if the forward (backward) deterministic trajectory starting from phase space point x meets λ_i before λ_j and 0 otherwise. We can now define the *loop*- and *boundary*-characteristic functions as:

$$l_i(x) = h_{i,i-1}^f(x)h_{i,i+1}^f(x)h_{i,i-1}^b(x)h_{i,i+1}^b(x) \quad (5.2a)$$

$$b_i(x) = h_{i-1,i}^f(x)h_{i,i-1}^b(x) + h_{i-1,i}^b(x)h_{i,i-1}^f(x) + h_{i+1,i}^f(x)h_{i,i+1}^b(x) + h_{i+1,i}^b(x)h_{i,i+1}^f(x). \quad (5.2b)$$

Again, these functions are either one or zero depending on whether x belongs to its *loop* (l) or *boundary* (b) category. Similarly, the complete set of time slices in the PPTIS ensemble can be characterized by

$$w_i(x) = l_i(x) + b_i(x) \quad (5.3)$$

which is either 1 or 0. As b_i and l_i vanish whenever w_i is zero,

$$b_i(x)w_i(x) = b_i(x), \quad (5.4)$$

and

$$l_i(x)w_i(x) = l_i(x), \quad (5.5)$$

for any phase point x . Moreover, for a phase point with $\lambda_{i-1} < \lambda(x) < \lambda_i$ it also holds

$$b_{i-1}(x) = b_i(x) \quad (5.6)$$

and

$$\begin{aligned} w_{i-1}(x) + l_i(x) &= l_{i-1}(x) + b_{i-1}(x) + l_i(x) \\ &= l_{i-1}(x) + b_i(x) + l_i(x) \\ &= l_{i-1}(x) + w_i(x) = 1 \end{aligned} \quad (5.7)$$

Using these Eqs. for any $\lambda' : \lambda_{i-1} < \lambda' < \lambda_i$:

$$\begin{aligned} P(\lambda') &= \langle \delta(\lambda(x) - \lambda') \rangle \\ &= \langle \delta(\lambda(x) - \lambda')(w_{i-1}(x) + l_i(x)) \rangle \\ &= \langle \delta(\lambda(x) - \lambda')w_{i-1}(x) \rangle + \langle \delta(\lambda(x) - \lambda')l_i(x) \rangle \\ &= \langle w_{i-1}(x) \rangle \langle \delta(\lambda(x) - \lambda') \rangle_{w_{i-1}} \\ &\quad + \langle w_i(x) \rangle \langle \delta(\lambda(x) - \lambda')l_i(x) \rangle_{w_i}, \end{aligned} \quad (5.8)$$

where $\langle A(x) \rangle_\rho \equiv \langle A(x)\rho(x) \rangle / \langle \rho(x) \rangle$ denotes a conditional ensemble average with weight-function $\rho(x)$. The term $\langle \delta(\lambda(x) - \lambda') \rangle_{w_{i-1}}$ can be calculated by histogramming all timeslices of the paths generated by the PPTIS algorithm in the interface

$i - 1$ ensemble. Similarly, $\langle \delta(\lambda(x) - \lambda') l_i \rangle_{w_i}$ can be obtained by histogramming the loop points of the trajectories in the ensemble of interface i . The remaining terms are computed by matching different histograms using scaling factors obtained from the overlapping regions between two windows. These scaling factors are here defined as $s_i \equiv \langle w_i(x) \rangle / \langle w_{i-1}(x) \rangle$ and follow, using Eqns. (5.5) and (5.7), from

$$s_i \equiv \frac{\langle w_i(x) \rangle}{\langle w_{i-1}(x) \rangle} = \frac{\langle \delta(\lambda(x) - \lambda') b_{i-1} \rangle_{w_{i-1}}}{\langle \delta(\lambda(x) - \lambda') b_i \rangle_{w_i}}, \quad (5.9)$$

for any $\lambda' : \lambda_{i-1} < \lambda' < \lambda_i$. Hence, one can integrate over λ' to obtain the most accurate value of s_i .

Using these scaling factors (5.9) and Eq. (5.8) one can derive the following relation for the relative probability of the order parameter values a and b with $\lambda_{i-1} < a < \lambda_i$ and $\lambda_i < b < \lambda_{i+1}$:

$$\frac{P(a)}{P(b)} = \frac{s_i^{-1} \langle \delta(\lambda(x) - a) \rangle_{w_{i-1}} + \langle \delta(\lambda(x) - a) l_i \rangle_{w_i}}{\langle \delta(\lambda(x) - b) \rangle_{w_i} + s_{i+1} \langle \delta(\lambda(x) - b) l_{i+1} \rangle_{w_{i+1}}}. \quad (5.10)$$

Hence, when all scaling factors s_i are known the total histogram $P(\lambda)$ can be computed by joining all probabilities from Eq. (5.10).

5.2 Algorithm

Based on the theoretical derivation of the previous section, we outline here the steps of a practical implementation, including some hints from personal experience.

Given the interfaces $0 \dots n_I$, the PPTIS windows are defined as a set of three consecutive interfaces $i - 1, i, i + 1$ for $i = 1 \dots n_I - 1$. In each window a path sampling simulation is carried out and at the end the results are rematched together to obtain the rate constant. In addition one can compute the free energy profile along λ without any expensive computations, but by just histogramming the loop and boundary points as defined in Eq. (5.2) and Fig. 5.1. The procedure is as follows.

Choice of histogram bin width $\delta\lambda$: If $\Delta\lambda_i = \lambda_{i+1} - \lambda_i$ one should choose the bin width $\delta\lambda$ such that $\Delta\lambda_i/\delta\lambda$ is an integer for all $i = 0 \dots n_I - 1$. The first bin should be then the interval $[\lambda_0, \lambda_0 + \delta\lambda]$. As a consequence all the interfaces are boundaries of bin intervals, and *no interface should fall in the middle of a bin*. This is to allow the proper counting of phase points. In a PPTIS window the integration is stopped when $i - 1$ or $i + 1$ are crossed. If a path has N_s slices, the part of the path to be used for sampling consists only of the timeslices $1 \dots N_s - 1$, because the first and last fall outside the window, and do not belong to the ensemble. If a bin is centered for example on $i - 1$, the histogram at that point will be over- or under-estimated. Of course, one could use a histogram with bins of variable width, fitting the bin boundaries to the interfaces, but this is much more complicated.

Histogramming loop and boundary points: Given a path in the ensemble of window $i - 1, i, i + 1$ it is simple to distinguish loop and boundary points. One

has to detect the first and last crossing of the path with the middle interface i . Then loop points are all those in between the two crossing points, the boundary ones are the rest. The crossing itself is actually two points, one on the left of i and one on the right, or the other way around. The point closer to the outer interfaces is a boundary point, the other is a loop one.

Window rematch Denote with $L_i(j\delta\lambda)$ and $B_i(j\delta\lambda)$ the loop and boundary histograms in window i . To maintain the proper relative weight, these histograms should *not* be normalized independently, and one could in fact consider the unnormalized raw data. Let

$$N_i = \sum_{\lambda_{i-1} < j\delta\lambda < \lambda_{i+1}} L_i(j\delta\lambda) + B_i(j\delta\lambda) \quad (5.11)$$

be the total number of points histogrammed in window i , let the boundary areas between the interfaces $i-1, i$ and $i, i+1$ be respectively

$$A_i^L = \sum_{\lambda_{i-1} < j\delta\lambda < \lambda_i} B_i(j\delta\lambda) \quad (5.12)$$

$$A_i^R = \sum_{\lambda_i < j\delta\lambda < \lambda_{i+1}} B_i(j\delta\lambda) \quad (5.13)$$

The scaling factors of Eq. (5.9) are given by the ratio of the areas in the overlapping region between i and $i+1$

$$s_i = A_i^R / A_{i+1}^L \quad (5.14)$$

Next we rematch windows i and $i+1$ in the overlapping region. First, the histograms of the boundary parts are supposed to be equal after rescaling by s_i . If this is not the case, there has been some sampling problem and the simulation must be repeated. If they are almost equal we could just use one of the two, but it is better to take a weighted average. By defining the weights $f_L = N_i / (N_i + N_{i+1})$ and $f_R = 1 - f_L$. The rematched boundary histogram then is given by

$$B_i^R(j\delta\lambda) = B_i(j\delta\lambda) * f_L + B_{i+1}(j\delta\lambda) * s_i * f_R \quad (5.15)$$

Second, we add the loop histograms, so that the final rematched histogram is

$$P_i(j\delta\lambda) = B_i^R(j\delta\lambda) + L_i(j\delta\lambda) + L_{i+1}(j\delta\lambda) * s_i \quad (5.16)$$

which is valid for $\lambda_i < j\delta\lambda < \lambda_{i+1}$. Finally the rematched window must be rescaled to the predecesing histogram between $i-1$ and i , for $i > 1$. The whole histogram $P_i(j\delta\lambda)$ must be multiplied by

$$\sum_{k=1}^{i-1} s_k \quad (5.17)$$

All the $P_i(j\delta\lambda)$, $i = 1 \dots n_I - 2$ form the probability distribution $P(\lambda)$ for $\lambda_1 < \lambda < \lambda_{n_I-1}$. The distribution can be completed for $\lambda < \lambda_1$ and $\lambda > \lambda_{n_I-1}$ using two standard MC or MD simulation in the stable states.

This procedure is in effect equivalent to umbrella sampling (see sec. 2.3.1) with the use of overlapping windows. The implementation of the procedure is sketched in Fig. 5.2. It requires some care in bookkeeping for array indices, but results in the free energy $-\ln P(\lambda)$ together with the rate constants.

5.3 Numerical results

We test the validity of the method on the same dimer system of sec. 4.3. We consider $N = 100$ particles at fixed density in dimension $d = 2$, interacting through a purely repulsive Weeks-Chandler-Andersen (WCA) potential. In addition, two of the particles experience a double well potential $U_{ddw}(r_d)$, Eq. (4.18), with r_d the dimer interparticle distance, so that its two minima correspond to a compact and an extended configuration. The barrier height is chosen such that the states are stable, transitions between them are rare, and the rate constants are well defined. We chose 18 interfaces defined by the order parameter $\lambda = r_d$. State A is defined by $r_d < \lambda_0$ and state B by $r_d > \lambda_{17}$. For all the details of the remaining numerical parameters we refer to sec. 4.3.

We considered two cases, one at constant temperature, and the other at constant energy. We first calculated the canonical free energy $\beta F(r_d) = -\ln P(r_d)$, at $T = 1/\beta = 0.755$, chosen to match the average kinetic energy of our previous microcanonical simulation, sec. 4.3.2. In Fig. 5.2 we report the histograms of loop and boundary points for two consecutive PPTIS windows, together with the rescaling and rematching procedure. By iterating the procedure on all the windows, we computed the free energy between λ_1 and λ_{16} . The free energy in the stable regions A and B was obtained by directly histogramming $P(r_d)$ by means of two standard MC simulations. To check the PPTIS result we performed an independent MC free energy calculation. By applying a biasing potential of exactly $-U_{ddw}(r_d)$ to the dimer system, one can simulate a system of pure WCA particles, and obtain the free energy from the probability of finding any two particles at distance r_d . See appendix F for the details. The resulting free energy agrees very well with our PPTIS results, as shown in Fig. 5.3.

Path sampling simulations are often performed at constant energy. We therefore also performed a PPTIS simulation on the same system, at constant energy $E/N = 1.0$. The simulation is the same of sec. 4.3.2, but now we also compute the NVE free energy. Again, we checked the result by an independent free energy calculation. As explained in appendix F, using the constant temperature biased MC simulation described before, it is possible to reweight each canonical configuration to its proper microcanonical probability for the unbiased system. We report in Fig. 5.3 the NVE free energy profile obtained from the reweighted biased MC together with the PPTIS one.

Finally, the forward and backward rate constants follow from a PPTIS simulation together with the free energy once the fluxes in (5.1) are known. For both the canonical and microcanonical cases we computed the fluxes using MD trajectories with initial points in the stable states and sampled from the appropriate corresponding distribution, see sec. 3.2.1. In table 5.1 we report the final rates. The constant energy results compare well with our previous calculations, see sec. 4.3.2.

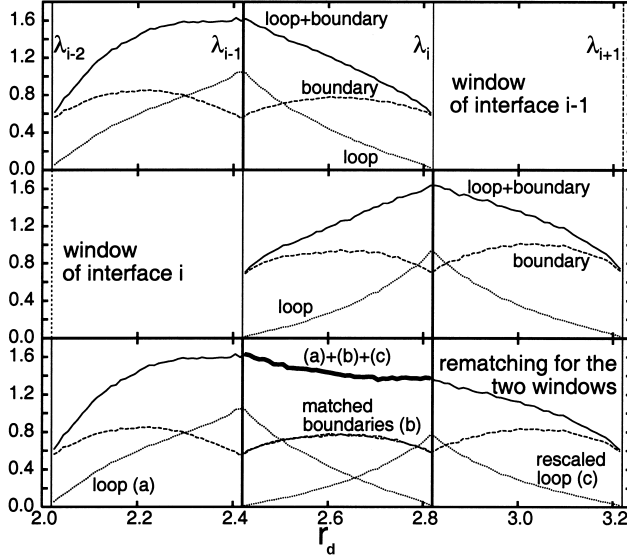


Figure 5.2: The window rematching procedure. Top and middle panel: loop and boundary points histograms for two consecutive windows of the λ_{i-1} ensemble (with $\lambda_{i-2} = 2.02 < r_d < 2.82 = \lambda_i$) and the λ_i ensemble (with $\lambda_{i-1} = 2.42 < r_d < 3.22 = \lambda_{i+1}$). Bottom: construction of the corrected histogram $P(r)$ (thick solid line) between λ_{i-1} and λ_i . Repeating this procedure with windows λ_{i-2} and λ_{i+1} results in the probability histogram over the entire range $\lambda_{i-2} = 2.02 < r_d < 3.22 = \lambda_{i+1}$.

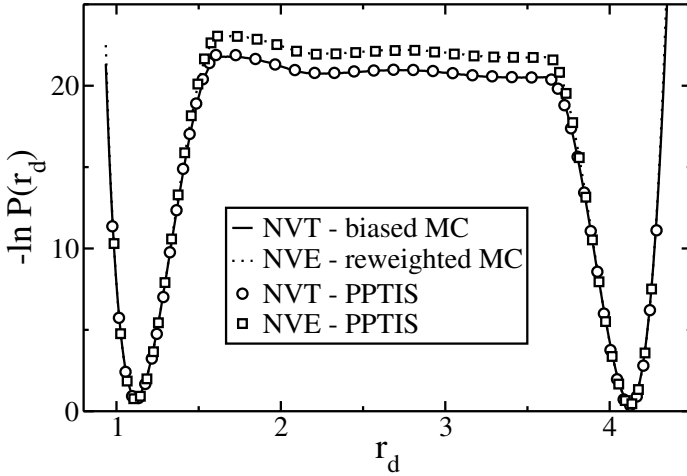


Figure 5.3: Canonical and microcanonical free energies obtained from PPTIS and MC simulations. The errors are within symbol size. The temperature and the energy of the respective NVT and NVE simulations were chosen to give the same average kinetic energy. Still, the free energy profile on top of the barrier is significantly different for the two ensembles.

	$k_{AB}/10^{-10}$	$k_{BA}/10^{-10}$	C	C_F	C_{MC}
NVT	10 ± 4	6 ± 2	1.5 ± 0.8	1.5 ± 0.2	1.419 ± 0.003
NVE	2.9 ± 0.5	1.9 ± 0.2	1.5 ± 0.3	1.39 ± 0.07	1.423 ± 0.002

Table 5.1: PPTIS forward and backward rate constants k_{AB} and k_{BA} , as well as the equilibrium constant $C = k_{AB}/k_{BA}$. Moreover, integrating $P(r_d)$ from the free energy curves over the stable regions we can obtain their relative probabilities F_A , F_B and the ratio $C_F = \exp(\beta(F_A - F_B))$, which is another expression of the equilibrium rate. We also report C_{MC} obtained using the free energies from the biased MC simulations. The results are all consistent with each other.

5.4 Free energy as function of another order parameter

So far we explained how in a PPTIS simulation, together with the rate constants, we can obtain the free energy profile along the order parameter $\lambda(x)$. In this section we show how we can also compute in the same simulation the free energy as function of another variable $\eta(x)$.

Consider the normal umbrella sampling method, briefly explained in sec. 2.3.1. In Eq. (2.38) the definition of the windows W_i is in principle independent of the histogrammed variable. So given a system with phase space point x , we partition the phase space in windows defined by $W_i = \{x : \lambda_L^{(i)} < \lambda(x) < \lambda_R^{(i)}\}$ and we compute in each simulation

$$P_{W_i}(\eta) = \frac{\int dx \rho(x) h_{W_i}(x) \delta[\eta - \eta(x)]}{\int dx \rho(x) h_{W_i}(x)} = \langle \delta[\eta - \eta(x)] \rangle_{W_i} \quad (5.18)$$

Take two neighboring windows W_i and W_{i+1} , which overlap in λ . The two corresponding histograms $P_{W_i}(\eta)$ and $P_{W_{i+1}}(\eta)$ will be different from zero in a common range of η . However, if η is not a unique function of λ , i.e. the relation is not one-to-one, contributions to the common η -zone might also come from phase points not belonging to the common λ -zone. Therefore the relative weight of $P_{W_i}(\eta)$ and $P_{W_{i+1}}(\eta)$ cannot be simply determined by the ratios of the overlapping areas. Fortunately there is a way to take this into account. We can split the histogram (5.18) into two histograms, one given by points x for which $\lambda(x) \in W_i \cap W_{i+1}$ and another for the remaining points. If we do the same in window $i+1$ naturally the histogram probability in the intersection is the same. We can compute then the relative weight for the whole $P_{W_i}(\eta)$ with respect to $P_{W_{i+1}}(\eta)$ by taking the ratio of the areas of the histograms in the intersection. Since we know which points belong to the common λ -zone, we can avoid counting them twice.

We can now apply this to PPTIS. We review the algorithm implementation of sec. 5.2 adding the new free energy histogramming. We use the definitions $l_i(x)$ and $b_i(x)$ in Eq. (5.2) for loop and boundary points. Besides the loop and boundary λ -histograms

$$L_i(\lambda) = \langle \delta(\lambda - \lambda(x)) l_i(x) \rangle_{w_i} \quad (5.19)$$

$$B_i(\lambda) = \langle \delta(\lambda - \lambda(x)) b_i(x) \rangle_{w_i} \quad (5.20)$$

we define three new η -histograms.

$$L_{\eta,i}(\eta) = \langle \delta(\eta - \eta(x)) l_i(x) \rangle_{w_i} \quad (5.21)$$

$$B_{\eta,i}^L(\eta) = \langle \delta(\eta - \eta(x)) b_i(x) \theta(\lambda_i - \lambda(x)) \rangle_{w_i} \quad (5.22)$$

$$B_{\eta,i}^R(\eta) = \langle \delta(\lambda - \lambda(x)) b_i(x) \theta(\lambda(x) - \lambda_i) \rangle_{w_i} \quad (5.23)$$

The definition of loop or boundary point x uses $\lambda(x)$, but in the histogram we use $\eta(x)$. The last two expressions distinguish between boundary points left and right of interface λ_i because these are the common areas $w_i \cap w_{i+1}$. As noticed in sec. 5.2 all these histograms should not be normalized separately, and it is in practice better to leave the bin counting unnormalized. After all the simulations have been done, the rematching is as follows. Compute the rescaling factor s_i as in Eq. (5.14), from the ratio of the areas of boundary histograms $B_i(\lambda)$ and $B_{i+1}(\lambda)$ in the common zone between interfaces i and $i+1$. Denote with N_i the total number of points in the λ histograms of window i (which should be equal to the total number of points in the η histograms). Let $f_L = N_i/(N_i + N_{i+1})$ and $f_R = 1 - f_L$ be the relative weights of the histograms. Then the final rematch of the η histograms in two windows is

$$P(\eta) = L_{\eta,i}(\eta) + L_{\eta,i+1}(\eta) * s_i + B_{\eta,i}^R(\eta) * f_L + B_{\eta,i+1}^L(\eta) * s_i * f_R \quad (5.24)$$

Equation (5.24) is in principle correct and allows the computation of a free energy as function of η even if the order parameter used in the PPTIS simulation is a different one λ . However, the bias of the interface that allows to visit the less-probable regions in the λ domain, is not necessarily a good bias for the η domain. Therefore, if η and λ are very different, this procedure might not accurately describe the $P(\eta)$ profile.

5.5 Summary

At the heart of the separation of timescales, Eq. (1.1), and thus of the rare event is a free energy barrier. Reducing the myriad degrees of freedom to a proper set of variables λ , the function $F(\lambda) = -\ln P(\lambda)$ shows two minima separated by a transition state (see Fig. 1.6).

In the approach of traditional methods, such as the Bennett-Chandler procedure, the free energy computation along a reaction coordinate is one of the basic steps to calculate the rate constant. Instead TIS and PPTIS no longer use neither the reaction coordinate nor the free energy, but the crossing probability function whose calculation is much less sensitive to the problem of the right reaction coordinate. However, for the analysis of complex and diffusive processes, e.g. conformational changes of biomolecules, it could be useful to have, besides the crossing probabilities and the rate constants, also the free energy profile along order parameters, for instance to identify metastable states and bottlenecks in the mechanism. Here, we have shown that a method developed for rate computations, also gives this free energy as a side product, with no additional computational effort.

Finally, we remark that even though the derivation of this chapter is based on PPTIS, the free energy computation can be done also in TIS. It requires however both the simulations for the forward k_{AB} and reversed rate k_{BA} .

6 Efficiency of Interface Sampling

*[...] the physics tends to be obscured
in the profusion of mathematics.*

Herbert Goldstein [88]

Throughout chapters 3 to 5 we have presented the theory and methods of Interface Sampling for the study of rare events in complex systems, as well as applications to simple systems. In this chapter instead we concentrate on pure theoretical derivations, and estimates concerning the efficiency of TPS, TIS, and PPTIS.

6.1 Scaling

In this section we present some theoretical estimates for the scaling of the CPU time required for TPS, TIS and PPTIS. In all three methods, the rate constant is obtained by rematching a series of specific path sampling simulations, in which paths are restricted to different zones of the phase space, called windows. Therefore all methods scale with the number of windows N_W . The basic expression for the CPU time we are going to use is

$$t_{CPU} \propto \sum_{i=1}^{N_W} \mathcal{T}_i \mathcal{N}_i \quad (6.1)$$

where N_W is the number of windows, and \mathcal{T}_i , \mathcal{N}_i are respectively the average path-length and the number of paths we sample in window i . The proportionality constant is the CPU time required to generate a path of unit length, which is system and computer dependent.

In the following, we will compare Eq. (6.1) for different methods. To obtain a measure of their relative efficiency, we will optimize the simulation parameters, e.g. choice of the interfaces, and we will study the scaling of CPU time as function of important qualitative properties of the system, such as the free energy barrier length. Although based on approximations and qualitative reasoning, the results can be useful as guidelines in real applications.

6.1.1 TPS and TIS

Consider a set of interfaces $0 \dots n_I$, similar to the ones in Fig. 3.1. To calculate the TIS probabilities $P_A(i+1|i)$ for $i = 1 \dots n_I - 1$, the path sampling involves $N_W = n_I - 1$ window simulations. To make a proper comparison apply the same setup to TPS. The umbrella sampling of sec. 2.3.1 requires $N_W = n_I$ windows to compute the histograms $P_{AW_i}(\lambda, t)$, because one more simulation is required in the stable state B ¹. In the TIS and TPS method we can disregard the initial simulation in state A to calculate the flux in TIS, and to complete the histogram $P_A(\lambda, t)$ in TPS. In both methods, the computational time of these steps is negligible compared to the other part, involving N_W windows.

TPS umbrella sampling simulations use paths of fixed length t' for all windows. Assuming also a fixed amount of sampled paths $\mathcal{N}_i = \mathcal{N}$ for all i we can write for the TPS simulation time

$$t_{CPU}^{TPS} \propto \mathcal{N} N_W^{TPS} t' = \mathcal{N} n_I t' \quad (6.2)$$

In TIS the pathlength is variable, and cannot be predicted a priori. However, in the simulation of the dimer system of sec. 3.3 we found a linear behavior between window 1 and N_W , see Fig. 3.11. We can assume then $\mathcal{T}_i = i/N_W t'$. In the last window the TIS pathlength becomes equal to the average transition time, the same quantity t' which is recommended in TPS umbrella sampling [35]. Assuming again the same amount of paths per window as in TPS, i.e. $\mathcal{N}_i = \mathcal{N}$ for all i , we obtain

$$t_{CPU}^{TIS} \propto \mathcal{N} \sum_{i=1}^{N_W^{TIS}} \frac{i}{N_W^{TIS}} t' = \mathcal{N} t' \frac{N_W^{TIS} + 1}{2} = \mathcal{N} t' \frac{n_I}{2} \quad (6.3)$$

For simulations carried on the same system and same computer the proportionality constant is the same and we can thus write for the ratio of computational times

$$\frac{t_{CPU}^{TPS}}{t_{CPU}^{TIS}} = \frac{\mathcal{N} n_I t'}{\mathcal{N} t' n_I / 2} = 2 \quad (6.4)$$

This result is valid for a system with an underlying uphill free energy barrier, but not for diffusive systems. Also, we assumed the same window definition in TPS in TIS and the same amount of paths per window in the two methods, which could not be the optimal setup.

This rather qualitative reasoning compares well with the results of sec. 3.3.3, see table 3.2, but we have to remark that the system considered there was quite an optimal situation for TPS. Because of the high energy barrier separating the states A and B , all transition paths have approximately the same length and once ∂B is crossed, it is absorbed in B and does not recross ∂B . Then the average TIS pathlength in the last window is equal to the fixed TPS pathlength. However, when there is a broad distribution of pathways, the TPS pathlength should, in fact, be larger than the largest path in this distribution to ensure a true plateau, while TIS will, on average,

¹As explained in sec. 3.3.2, one computes from $P_{AW_i}(\lambda, t)$ the ratios of the value of the histogram at the left and right boundary of the window. This ratio is the TPS analog of the TIS probability $P_A(i+1|i)$.

have the mean path length of this distribution. Moreover, once the interface ∂B is crossed, there might be some correlated crossings through ∂B just thereafter. Even if the path is committed and remains at the right side of the barrier, this effect also enlarges the necessary minimum pathlength for TPS. It is due to this phenomenon that in sec 3.3.4, in the system with the low energy barrier, the TPS length was about three times as large as the average TIS length for the final window (20 vs 7.4). So the factor 2 of Eq. (6.4) can be considered a lower limit, and the efficiency will in general be higher.

6.1.2 Error analysis

In order to optimize Eq. (6.1) as function of the system parameters, and in order to find the scaling of computational time as function of, for instance, the free energy barrier length, we have to fix an accuracy level for the computation. This can be done by imposing a certain relative error on the total outcome of the simulation, i.e. the rate constant.

In each window, we consider an observable O_i , we denote the average value with $p_i = \langle O_i \rangle$, the variance with σ_i^2 , and we assume for the final rate k

$$k = \prod_{i=1}^{N_W} p_i. \quad (6.5)$$

This is a general notation meant to be applied to TIS and PPTIS. In case of TIS, $O_i = P_A(i+1|i)$ and k corresponds to k_{AB} . In case of PPTIS, $O_i = p_i^{\mp}/p_i^{\pm}$ and k corresponds to C , the equilibrium constant. Note that with this definition, in TIS $0 < p_i \leq 1$, while in PPTIS simply $p_i > 0$. From Eq. (6.5) and assuming uncorrelated different windows, we can write for the total relative error on k

$$\epsilon_{tot}^2 = \sum_{i=1}^{N_W} \frac{\sigma_i^2}{p_i^2} \frac{1}{\mathcal{N}_i} \quad (6.6)$$

Here we also assumed in each window that the different estimation measures of O_i are uncorrelated. Hence, the error on the estimate of p_i scales as the inverse number of simulation paths $1/\mathcal{N}_i$. In the propagation formula (6.6), we can obtain a fixed total tolerance error ϵ_{tot}^2 on k , if we simulate in each window an amount of paths $\mathcal{N}_i = \sigma_i^2 N_W / p_i^2 \epsilon_{tot}^2$. Inserting into Eq. (6.1), the final formula we need is

$$t_{CPU} \propto \sum_{i=1}^{N_W} \mathcal{T}_i \mathcal{N}_i = \sum_{i=1}^{N_W} \mathcal{T}_i \frac{\sigma_i^2}{p_i^2} \frac{N_W}{\epsilon_{tot}^2}. \quad (6.7)$$

In reality, paths are correlated and the scaling $1/\mathcal{N}_i$ for the relative error of the single window estimate is not correct. If you consider M_i simulation blocks of m_i paths, with m_i greater than the MC correlation length, then the scaling is effectively replaced by $m_i/\mathcal{N}_i = 1/M_i$. Assuming the same correlation length for all the windows, formula (6.7) is still valid. Since we found that in the model systems of previous chapters the assumption of uncorrelated paths is approximately satisfied, we will base the following derivations on Eq. (6.7).

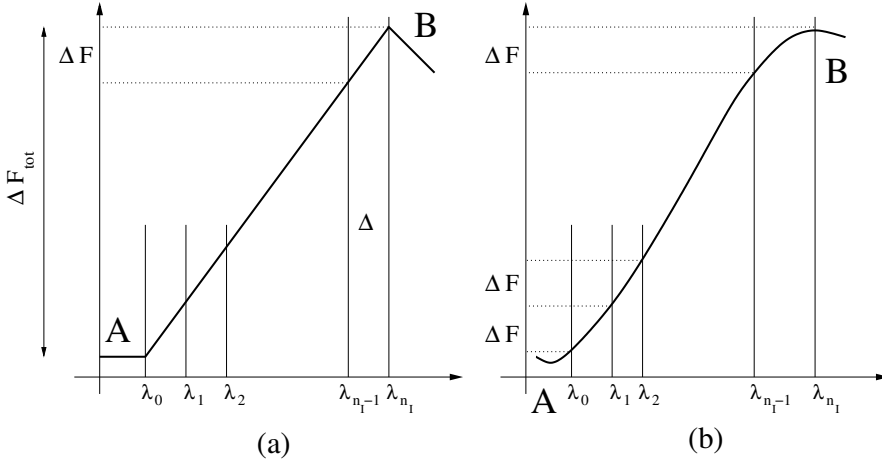


Figure 6.1: TIS setup for exponential barrier. (a) Linear free energy. (b) General free energy

6.1.3 TIS - Exponential barrier

TIS windows are defined as the set of three interfaces $0, i-1, i$. The path ensemble is defined by paths that leave interface 0, and cross interface $i-1$. Since a path that crosses $i-1$ then either reaches i or not, $O_i = P_A(i|i-1)$ is a boolean variable, and its square relative standard deviation reads

$$\frac{\sigma_i^2}{p_i^2} = \frac{1-p_i}{p_i} \quad (6.8)$$

Consider then the situation in Fig. 6.1 (a), which shows a linearly increasing free energy barrier $F = \alpha\lambda$, with $n_I + 1$ interfaces $\lambda_0 \dots \lambda_{n_I}$ equally spaced by Δ , so that the barrier length is $b \equiv \lambda_{n_I} - \lambda_0 = n_I\Delta$, and $\Delta F_{tot} \equiv F(\lambda_{n_I}) - F(\lambda_0) = n_I\Delta F$. The probability $P_A(i|i-1) = p_i = p$ is then independent of i and Eq. (6.5) becomes

$$k = p^{N_W} \Leftrightarrow N_W = \ln k / \ln p. \quad (6.9)$$

In the following we can simply assume $N_W = n_I - 1 \simeq n_I$. For such a barrier the rate decreases exponentially with barrier height, such that

$$-\ln p \sim \Delta F = \alpha\Delta \Leftrightarrow \Delta = -\frac{\ln p}{\alpha} \quad (6.10)$$

As in sec. 6.1.1, we assume the average pathlength is a linear function of the λ distance travelled by the paths, from A to window i , i.e.

$$\mathcal{T}_i \propto i\Delta = -\frac{i \ln p}{\alpha}. \quad (6.11)$$

Then, using (6.8), and substituting (6.11) into (6.7)

$$\begin{aligned}
 t_{CPU} &= \sum_{i=1}^{N_W} -\frac{i \ln p}{\alpha} \frac{1-p}{p} \frac{N_W}{\epsilon_{tot}^2} \\
 &= \frac{-\ln k}{\alpha \epsilon_{tot}^2} \frac{1-p}{p} \sum_{i=1}^{N_W} i \simeq \frac{-\ln k}{\alpha \epsilon_{tot}^2} \frac{1-p}{p} \frac{N_W^2}{2} \\
 &= \frac{b}{\epsilon_{tot}^2} \frac{1-p}{p} \frac{\Delta F_{tot}^2}{2 \ln^2 p} = \frac{1-p}{p \ln^2 p} \frac{\Delta F_{tot}}{\epsilon_{tot}^2} \frac{b \Delta F_{tot}}{2}
 \end{aligned} \tag{6.12}$$

In the last but one passage we used $b = -\ln k/\alpha$ and $N_W = (-\ln k)/(-\ln p) = \Delta F_{tot}/(-\ln p)$. We plot the first factor of the final expression in Fig. 6.2. It has a minimum around $p \simeq 0.2$, which tells us that the optimal choice of interface separation is around $P_A(i|i-1) = 0.2$. The other factors give the scaling of the CPU time and show that it is proportional to the barrier height times the area under the barrier.

This result is valid for linearly increasing free energy barriers, where the hypothesis (6.11) is more or less satisfied (see Fig. 3.11). However, this optimization can be made more general. If instead of Eq. (6.11) we have

$$\mathcal{T}_i = c_g (-i \ln p)^g \tag{6.13}$$

with $g \in \mathbb{N}$ and c_g a coefficient depending only on g , Eq. (6.12) becomes

$$\begin{aligned}
 t_{CPU} &= \sum_{i=1}^{N_W} c_g (-i \ln p)^g \frac{1-p}{p} \frac{N_W}{\epsilon_{tot}^2} \\
 &\simeq c_g \frac{1-p}{p} \frac{N_W}{\epsilon_{tot}^2} (-\ln p)^g \frac{N_W^{g+1}}{g+1} \\
 &= \frac{1-p}{p \ln^2 p} \frac{c_g}{\epsilon_{tot}^2} \frac{(-\ln k)^{g+2}}{g+1}
 \end{aligned} \tag{6.14}$$

and the minimization result $p \simeq 0.2$ is unchanged and independent of g . Consequently, if the situation is such as in Fig. 6.1 (b), we can impose again for all windows $p_i = p$

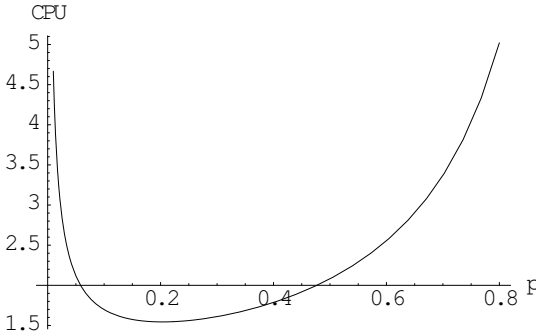


Figure 6.2: CPU time as function of TIS window crossing probability. Exponential barrier. Minimum at $p \simeq 0.2$.

and assume

$$\mathcal{T}_i = \bar{f}(\lambda_i) = \bar{f}(F^{-1}(i\Delta F)) \equiv f(-i \ln p) \quad (6.15)$$

where f is a general function which can be Taylor-expanded. Then, applying the minimization procedure to each term of the power series, the result $p \simeq 0.2$ remains valid for a general form of the free energy.

We remark however that this result assumes a fixed window crossing probability p . We have not tried to optimize TIS in the most general interface setup, and there are indications that the use of windows with variable crossing probability might perform better [122].

6.1.4 TIS - Diffusive barrier

Let us assume now a flat diffusive barrier. To help us in the theoretical derivations, we can view the barrier as a set of b discrete states, and use the discrete model of sec. 4.1.1 in the case of symmetrical hopping probabilities $\tau_{i,i+1} = \tau_{i,i-1} = 1/2$. The basic expression we need is

$$T[i \rightarrow_0^j] = i/j = 1 - T[i \rightarrow_j^0] \quad (6.16)$$

where $T[i \rightarrow_0^j]$ is the probability of being in i and reach $j > i$ before 0. This relation is proven in appendix E.4. By definition, for a diffusive barrier \mathcal{T}_i scales quadratically with the λ distance travelled. We distinguish two situations. First, we assume a fixed window width, and second a fixed crossing probability in the windows.

Fixed window width

From the discrete set of states $1, 2 \dots b$ let us build the windows $w_i = 1 + \Delta(i-1)$ with $i = 1 \dots N_W$ and $b = 1 + N_W \Delta \simeq N_W \Delta$, see Fig. 6.3(a). Then

$$p_i = T[w_i \rightarrow_0^{w_{i+1}}] = \frac{w_i}{w_{i+1}} = \frac{1 + \Delta(i-1)}{1 + \Delta i} \quad (6.17)$$

from which, using Eq. (6.8)

$$\frac{\sigma_i^2}{p_i^2} = \frac{\Delta}{1 + \Delta(i-1)}. \quad (6.18)$$

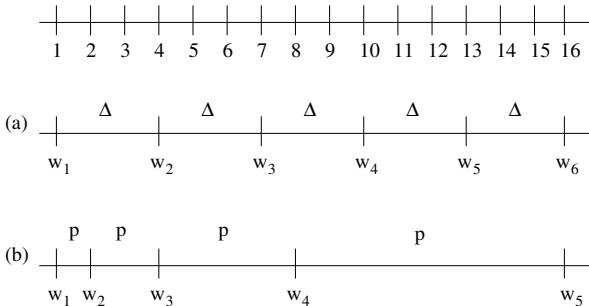


Figure 6.3: Window choice for TIS on diffusive barrier of length $b = 16$. (a) fixed window width (b) fixed crossing probability

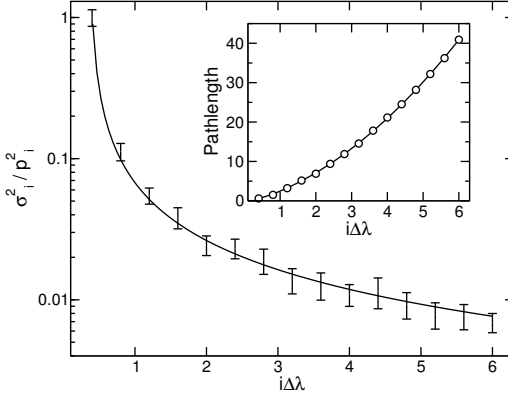


Figure 6.4: The square relative standard deviation σ_i^2/p_i^2 for the TIS crossing probability $P_A(i+1|i)$ plotted as function of the barrier length for a dimer system (sec. 4.3.1) with total barrier length $b = 6$. The values σ_i^2/p_i^2 have been rescaled to the one of the first interface measured from the start of the plateau, and fitted with an inverse linear function. Inset: the average path length for these simulations as function of the barrier length. The error bar is within symbol size. The solid line is a second-order polynomial fit.

For the mean pathlength it holds $\mathcal{T}_i = (i\Delta)^2$ because the system is diffusive. We have checked this behavior for σ_i^2/p_i^2 and \mathcal{T}_i on the dimer system in sec. 4.3.1, using a barrier length $b = 6$. As shown in Fig. 6.4 the findings based on the onedimensional model can be verified also in a more complex system.

Eq. (6.7) becomes then

$$\begin{aligned} t_{CPU} &= \sum_{i=1}^{N_W} (i\Delta)^2 \frac{\Delta}{1 + \Delta(i-1)} \frac{N_W}{\epsilon_{tot}^2} \\ &= \frac{b^3}{\epsilon_{tot}^2 N_W^2} \sum_{i=1}^{N_W} \frac{i^2}{1 + (i-1)b/N_W}. \end{aligned} \quad (6.19)$$

We have studied this function numerically using *Mathematica* [123]. The function shows a minimum for $N_W = N_W^*$, defining the optimum number of interfaces. This minimum N_W^* has a behavior

$$N_W^* \simeq cb^f \quad \ln c \simeq 0.5, \quad f \simeq 2/7 \quad (6.20)$$

which inserted into Eq. (6.19) yields, after an asymptotic expansion,

$$t_{CPU} \sim b^{2+f} \quad (6.21)$$

Fixed crossing probability

Consider the setup of Fig. 6.3(b). Now $p_i = p \forall i$, Eq. (6.9) holds and the windows w_i are defined by

$$p_i = T[w_i \rightarrow_0^{w_{i+1}}] = \frac{w_i}{w_{i+1}} = p \quad (6.22)$$

the recursive relation $w_{i+1} = w_i/p$ is easily solved by $w_i = 1/p^{i-1}$, and since $\mathcal{T}_i \propto w_{i+1}^2$ we obtain

$$\begin{aligned}
 t_{CPU} &= \sum_{i=0}^{N_W} p^{-2i} \frac{1-p}{p} \frac{N_W}{\epsilon_{tot}^2} \\
 &= \frac{-\ln k}{\epsilon_{tot}^2} \frac{p-1}{p \ln p} \sum_{i=0}^{N_W} \left(\frac{1}{p^2} \right)^i \\
 &= \frac{-\ln k}{\epsilon_{tot}^2} \frac{p-1}{p \ln p} \frac{1-p^{-2(N_W+1)}}{1-p^{-2}} \\
 &= \frac{-\ln k}{\epsilon_{tot}^2} \frac{p-1}{p \ln p} \frac{1-p^{-2}k^{-2}}{1-p^{-2}} \\
 &= \frac{-\ln k}{\epsilon_{tot}^2} \frac{p-1}{p \ln p} \frac{p^2-k^{-2}}{p^2-1} \\
 &= \frac{-\ln k}{\epsilon_{tot}^2} \frac{p^2-k^{-2}}{p(p+1) \ln p} \tag{6.23}
 \end{aligned}$$

A numerical study of this function showed a minimum for $p \simeq 0.47$ which is not sensitive to k for $k^{-2} > 10$.

The scaling with b follows from Eq. (6.23)

$$\begin{aligned}
 t_{CPU} &= \frac{-\ln k}{\epsilon_{tot}^2} \frac{p^2-k^{-2}}{p(p+1) \ln p} \\
 &= \frac{N_W}{\epsilon_{tot}^2} \frac{k^{-2}-p^2}{p(p+1)} \sim N_W k^{-2} \sim b^2 \ln b \tag{6.24}
 \end{aligned}$$

where in the last step we used $k \sim 1/b$ and $n = \ln k / \ln p = \ln b / \ln p^{-1} \sim \ln b$. Comparing with Eq. (6.21) we see that the setup with fixed crossing probability yields a more efficient scaling for large b .

6.1.5 PPTIS

As in the previous section we make use of the onedimensional model of sec. 4.1.1. Consider a diffusive barrier of length b and a fixed interface separation $\Delta = b/n$. Then it follows that $\mathcal{T}_i \propto (2\Delta)^2$. Using as observable $O_i = p_i^\mp / p_i^\pm$, we can write

$$\frac{\sigma_i^2}{p_i^2} = \frac{\sigma_{p_i^\mp}^2}{(p_i^\mp)^2} + \frac{\sigma_{p_i^\pm}^2}{(p_i^\pm)^2} = \frac{1-p_i^\mp}{p_i^\mp} + \frac{1-p_i^\pm}{p_i^\pm}. \tag{6.25}$$

where the first equality follows from $\text{cov}(p_i^\mp, p_i^\pm) = 0 \forall i$, which can be easily proven, and the second is an approximation which assumes the ensembles $(\mp, =)$ and (\pm, \mp) to be independent. However, for a flat barrier

$$p_i^\pm = T[\Delta \rightarrow_0^{2\Delta}] = 0.5 = T[\Delta \rightarrow_{2\Delta}^0] = p_i^\mp \tag{6.26}$$

Then, we simply have $\sigma_i^2/p_i^2 = 2$, and

$$t_{CPU} \propto \sum_{i=1}^{N_W} \Delta^2 \frac{N_W}{\epsilon_{tot}^2} \sim N_W^2 \Delta^2 = b^2 \quad (6.27)$$

is independent of Δ .

This means that the choice of the windows does not matter as far as the total relative error on the equilibrium constant is concerned. This seemingly crude result is in accordance with a recent efficiency analysis of the regular Umbrella Sampling method [124]. Some textbooks [11,125] have stated that the efficiency of a free energy calculation on a relative flat landscape increases linearly with the number of windows. This suggest that, apart from sampling problems, an infinite number of infinite small windows would be the best possible choice. However, as correctly noticed in [124], the gain of diffusing through smaller window is exactly cancelled by the error propagation which was neglected in Refs. [11,125]. Hence, it was found that the efficiency scales as b^2 and is independent of the number of windows used.

In this section we have given scaling results for the rate constant computation on diffusive barriers using TIS and PPTIS. In the next section we present scaling considerations for the TST approach, based on the calculation of the transmission coefficient. Finally, we will compare and summarize the different approaches.

6.2 Transmission coefficients revisited

In sec. 1.5.2 (page 19) we mentioned some alternative expressions for the TST transmission coefficient κ . These expressions try to improve the efficiency of the counting of trajectories crossing the TST dividing surface defined at $\lambda = \lambda^*$, see Fig. 1.10. We also introduced Eq. (1.65), based on the effective positive flux, whose definition was given in sec. 3.1.3. Now, we are in a position able to derive a formal complete expression for κ^{TIS} and to compare it with the some of the other approaches.

6.2.1 Transmission coefficient based on effective positive flux

Consider the TIS formula for the rate constant, Eq. (3.12). With the help of the flux relation Eq. (D.2) we can write

$$k_{AB} = \frac{\langle \phi_{n_I,0} \rangle}{\langle h_A \rangle} = \frac{\langle \phi_{\lambda^*,0} h_{n_I,0}^f \rangle}{\langle h_A \rangle} \quad (6.28)$$

where as usual interface 0 is the border of state A , interface n_I the border of state B , and we introduced an intermediate interface at the TST dividing surface. Using the

definition (3.10) of the effective positive flux for $\phi_{\lambda^*,0}$ we can proceed rewriting

$$\begin{aligned}
 k_{AB} &= \frac{\langle \phi_{\lambda^*,0} h_{n_I,0}^f \rangle}{h_A} \\
 &= \frac{\langle h_{0,\lambda^*}^b |\dot{\lambda}| \delta(\lambda - \lambda^*) h_{n_I,0}^f \rangle}{\langle h_A \rangle} \\
 &= \frac{\langle h_{0,\lambda^*}^b |\dot{\lambda}| \delta(\lambda - \lambda^*) h_{n_I,0}^f \rangle}{\langle \delta(\lambda - \lambda^*) \rangle} \cdot \frac{\langle \delta(\lambda - \lambda^*) \rangle}{\langle \dot{\lambda} \theta(\dot{\lambda}) \delta(\lambda - \lambda^*) \rangle} \cdot \frac{\langle \dot{\lambda} \theta(\dot{\lambda}) \delta(\lambda - \lambda^*) \rangle}{\langle h_A \rangle} \cdot \frac{\langle h_A \rangle}{\langle h_A \rangle} \\
 &\simeq \frac{\langle h_{0,\lambda^*}^b |\dot{\lambda}| \delta(\lambda - \lambda^*) h_{n_I,0}^f \rangle_{\lambda=\lambda^*}}{\langle \dot{\lambda} \theta(\dot{\lambda}) \rangle_{\lambda=\lambda^*}} \cdot k_{AB}^{TST} \quad (6.29)
 \end{aligned}$$

where in the last passage we used the definition of k_{AB}^{TST} , Eq. (1.33), and we approximated $\langle h_A \rangle / \langle h_A \rangle \simeq 1$. Because of the definition of the transmission coefficient, k_{AB}/k_{AB}^{TST} , we can write

$$\kappa^{TIS} = \frac{\langle h_{0,\lambda^*}^b \dot{\lambda} \theta(\dot{\lambda}) \delta(\lambda - \lambda^*) h_{n_I,0}^f \rangle_{\lambda=\lambda^*}}{\langle \dot{\lambda} \theta(\dot{\lambda}) \rangle_{\lambda=\lambda^*}} \quad (6.30)$$

where we used the fact that only positive velocities contribute inside the average in the numerator, due to the h^b, h^f functions, and hence $|\dot{\lambda}|$ can be substituted with $\dot{\lambda} \theta(\dot{\lambda})$. Comparing with Eq. (1.65) gives

$$\chi_{AB}^{epf}(x_0) = h_{0,\lambda^*}^b(x_0) h_{n_I,0}^f(x_0) \quad (6.31)$$

The expression (6.30) for κ^{TIS} can be found in [20] and was first derived by van Erp in [119]. The calculation is performed by taking a set of independent states x_0 on the surface λ^* and initiating from each state a series of trajectories. Only if $\dot{\lambda}(x_0) > 0$, one has to integrate the equations of motion backward in time. Because of the term h_{0,λ^*}^b , one can stop the integration if the evolution hits again the surface λ^* . If instead one reaches state A , we start again from x_0 and integrate forward in time to check if we reach state B or not. As a result, this procedure only counts positive flux terms (see Fig. 6.5).

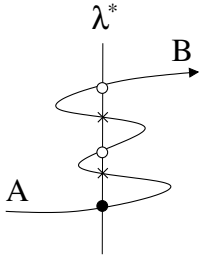


Figure 6.5: Calculation of transmission coefficient κ with the effective positive flux idea. When initial points x_0 are sampled with negative velocity $\dot{\lambda} < 0$ (crosses) they do not contribute, since to hit A on the backward evolution they must first cross λ^* , and hence $h_{0,\lambda^*}^b(x_0)$ is automatically zero. When points are sampled with positive velocity $\dot{\lambda} > 0$ but on the backward evolution they hit λ^* before state A (open circles), they also do not contribute. In the example in figure the path eventually ends in B , so it must be considered as one reaction. This is properly counted by the first point from bottom. It is effectively the only one that contributes because the other points, even if different, simply belong to the *same* reaction path.

6.2.2 Comparison with other methods

Using the error analysis ideas of this chapter and the results of sec 1.5.3, we can now give some comparative scaling considerations for different approaches to the calculation of the rate constant in a diffusive regime. Namely, we consider

1. the standard TST-BC procedure, Eq. (1.48),
2. the Ruiz-Montero method, Eq. (1.62),
3. the effective positive positive flux approach of this section, Eq. (6.29),
4. TIS,
5. PPTIS.

The cases of TIS and PPTIS were discussed in sec. 6.1.4 and 6.1.5 respectively, so let us concentrate on methods 1,2, and 3. They can all be considered as combinations of a free energy calculation followed by the computation of a transmission coefficient. As mentioned in sec. 6.1.5, the CPU time t_{CPU}^{US} for the free energy calculation with the use of umbrella sampling scales as b^2 , the square of the barrier length. To compute the scaling of the transmission coefficient κ we use the formula

$$t_{CPU}^{\kappa} \propto \frac{\sigma_{\kappa}^2}{\kappa^2} \mathcal{T} \quad (6.32)$$

where \mathcal{T} is the average length of the trajectories used in the calculation. Formula (6.32) can be derived similarly to Eq. (6.7), and we assumed $\epsilon_{tot}^2 = 1$.

The scalings of $\sigma_{\kappa}^2/\kappa^2$ for the methods 1,2, and 3 have been derived in sec. 1.5.2, respectively in Eqs. (1.54),(1.64), and (1.67). Since here we are interested in the scaling of κ and not of its estimator we can drop the dependence on N_{tr} .

The estimate of \mathcal{T} needs some considerations. Method 1 requires the trajectories to be committed to the stable states in order to reach a plateau in the time-dependent transmission coefficient $\kappa(t)$. Therefore paths must be generated on the whole length of the barrier, and because of diffusion their time-length scales as $\mathcal{T} \sim b^2$. Method 2 instead is based on the computation of the integral of the velocity autocorrelation function, Eq. (1.63). It therefore requires integration of trajectories only for a time equal to the decay time of the correlation function. In the case of a flat square barrier, where estimation (1.64) holds, this decay time is constant, and thus $\mathcal{T} \sim const$. For method 3 we can estimate \mathcal{T} as follows, also assuming a flat square barrier separating the states. Suppose to generate initial points x_0 for the trajectories on the surface λ^* (see Fig. 6.5), which we can put in the middle of the barrier between A and B . Assuming a symmetric velocity distribution, for half of the points $\dot{\lambda}(x_0) < 0$ holds, which implies $h_{0,\lambda^*}^b(x_0) = 0$ and no integration of dynamics is needed. The other half, for which $\dot{\lambda}(x_0) > 0$, we divide into two sets

$$S_1 = \{x_0 | \dot{\lambda}(x_0) > 0 \wedge h_{0,\lambda^*}^b(x_0) = 0\} \quad (6.33)$$

$$S_2 = \{x_0 | \dot{\lambda}(x_0) > 0 \wedge h_{0,\lambda^*}^b(x_0) \neq 0\} \quad (6.34)$$

and we denote with $(1 - \alpha)/2$ and $\alpha/2$ the fraction of points in each set respectively. Necessary condition for a point to be counted is that its evolution backward in time reaches A before λ^* . The set S_1 are points not satisfying this condition. In this case trajectories are integrated only the time needed to hit λ^* , and this time scales on average linearly² with b . The set S_2 contains the only points that might contribute to the transmission coefficient if $h_{n_I,0}^f(x_0) \neq 0$, i.e. if the forward evolution reaches B before A . But in either case, whether it reaches B or A first, the integration must be carried for the full length of the barrier and thus \mathcal{T} scales as b^2 . We still need to estimate α , but this is easily done since the set S_2 contains the reactive paths and their amount is just the definition of κ . Since the barrier is flat we assume the other points also contribute in similar amount and we estimate $\alpha \sim 2\kappa$. Indeed, as shown in Fig. 6.5, only one point belongs to S_2 out of all the crossing points with λ^* , and we know from sec.1.5.2 that the inverse number of recrossings is proportional to κ . Resuming, we can estimate the average pathlength for the effective positive flux as

$$\begin{aligned} \mathcal{T} &\sim \frac{1}{2} \cdot 0 + \frac{1}{2} [(1 - \alpha)b + \alpha b^2] \\ &\sim \frac{1}{2} [(1 - 2\kappa)b + 2\kappa b^2] \sim \frac{1}{2} [b + 2b] \sim b \end{aligned} \quad (6.35)$$

where we used the result $\kappa \sim 1/b$ for diffusive barriers (see Eq. (1.58) and sec. 4.1.1).

Using Eq. (6.32) we can derive the scaling of the CPU time t_{CPU}^κ for the computation of the transmission coefficient κ , and the total CPU time is then $t_{CPU}^{tot} = t_{CPU}^S + t_{CPU}^\kappa$. We report the final results for methods 1-3 in table 6.1 together with the TIS and PPTIS results. We also converted the scaling as function of the barrier length b into the scaling as function κ , using again $\kappa \sim 1/b$.

The results are quite interesting, indicating that PPTIS and the effective positive flux approach are comparable to the Ruiz-Montero, and, surprisingly, that TIS still performs quite well compared to PPTIS. However, we must remark that the comparison presented here is based on models and approximations, most important that of a flat square barrier separating the stable states. The qualitative findings of table 6.1 are just an indicative starting point and could change on a more careful theoretical analysis. Moreover, we have not taken into consideration the other methods named in sec. 1.5.3, giving different expressions for the calculation of the transmission coefficient. It would be interesting then to perform a more comprehensive study, also testing the methods on realistic cases. Here, we conclude noticing that a comparative study of ion channel diffusion [24] showed that the algorithm based on the effective positive flux expression of Anderson [26] was superior to the other transmission rate expressions. Moreover, it was found as efficient as an optimized version of the more complicated Ruiz-Montero method.

² This is a problem of first exit time through a specific end of an interval [6]. Suppose to have a onedimensional diffusion in the interval $[0, L]$ where the boundaries are absorbing. We want to know the mean first exit time $t(x)$ through boundary 0 when starting in x . Proceeding similarly to sec. A.2, a backward Fokker-Planck equation can be written for $t(x)$. Since the barrier is assumed flat, only diffusion is considered, and the equation contains no drift term. The solution displays two limiting behaviors. Let $\epsilon \ll 1$ and D the diffusion coefficient. For $x = L - \epsilon$, one obtains $t \sim L^2/D$ and the usual diffusive behavior is recovered. For $x = \epsilon$, one obtains $t \sim L\epsilon/D$, which is a linear function of the interval width.

method	t_{CPU}^{US}	σ_κ^2/κ^2	\mathcal{T}	t_{CPU}^{tot}	scaling
BC	b^2	$1/\kappa^2$	b^2	$b^2 + b^4 \sim b^4$	$1/\kappa^4$
RM	b^2	1	$const$	$b^2 + const \sim b^2$	$1/\kappa^2$
epf	b^2	$1/\kappa$	b	$b^2 + b^2 \sim b^2$	$1/\kappa^2$
TIS	n/a	n/a	b^2	$b^2 \ln b$	$-\ln \kappa/\kappa^2$
PPTIS	n/a	n/a	$const$	b^2	$1/\kappa^2$

Table 6.1: Scaling of total CPU time for different methods in the case of diffusive barriers. BC=Bennett-Chandler, RM=Ruiz-Montero et al., epf=effective positive flux. Those methods are based on a procedure that requires first a free energy calculation, which we assume here performed with Umbrella Sampling (US), and then the calculation of a transmission coefficient κ . TIS and PPTIS are based on a different approach and this procedure is not applicable (n/a).

6.3 The acceptance ratio for TIS on flat diffusive barriers

In TIS simulations of diffusive systems with a barrier length b , we found that the acceptance ratio α for the shooting move plateaus to a fixed value as the windows approach the end of the barrier. This phenomenon happens for sufficiently large b and is approximately independent of the momentum displacement in the shooting move. In this section we try to give a qualitative explanation for it.

We refer again to the onedimensional model of sec. 4.1.1, see Fig. 4.1: $s - 1$ metastable states separate state 0 and state $s \gg 1$. Consider a TIS path ensemble close to the end of the interface, e.g. for simplicity paths that start from 0, and cross $s - 1$. Most of the paths in this ensemble will end up in s and we neglect those that go back to state 0. The paths then start in 0, diffuse across the barrier and reach s . Suppose to perform a shooting move taking a slice of the path which lies in state i . The shooting will be accepted if integrating backward in time we reach 0 before s and integrating forward in time we reach s before 0. We neglect for the moment the acceptance probability caused by the variable pathlength, see sec. 3.2.2. Because of Eq. (E.19) the acceptance probability is written as

$$p(i) = T[i \rightarrow_s^0] T[i \rightarrow_0^s] = \left(1 - \frac{i}{s}\right) \frac{i}{s} \quad (6.36)$$

The overall acceptance probability is obtained summing Eq. (6.36) on all timeslices. However Eq. (6.36) is given as function of the state i . Hence we need the distribution $\rho(i)$ given by the (normalized) number of time slices in state i . A moment reflection shows that $\rho(i) \propto p(i)$ because the probability of a timeslice to be part of the path is exactly the probability of acceptance of a new path generated by the shooting move. The proportionality constant is given by the constraint $\sum_{i=0}^s \rho(i) = 1$. Using the continuous approximation

$$\sum_{i=0}^s i^g \simeq \frac{s^{g+1}}{g+1} \quad g \in \mathbb{R} \quad (6.37)$$

in the limit $s \gg 1$ we obtain $\rho(i) = 6p(i)/s$. Consequently we can write for the overall acceptance ratio

$$\begin{aligned}
 \alpha &= \sum_{i=0}^s \rho(i)p(i) \\
 &= \frac{6}{s} \sum_{i=0}^s p^2(i) \\
 &= \frac{6}{s^3} \sum_{i=0}^s i^2 + \frac{i^4}{s^2} - \frac{2i^3}{s} \\
 &\simeq \frac{1}{5} = 0.2
 \end{aligned} \tag{6.38}$$

This result disregards the acceptance probability given by the ratio of old and new pathlengths, which lowers the value of α . Indeed when we performed simulations of this simple onedimensional model we found that the acceptance ratio plateaus to a value $\alpha = 0.135$. This effect can be taken into account if we know the distribution of pathlengths, i.e. in TIS the distribution of first passage times. In general it is system dependent, because is related to the diffusion constant on top of the barrier, so that the result $\alpha = 0.2$ can be considered as an upper bound to the acceptance ratio of the shooting move for general systems. Indeed, in a study of a 247-atom peptide in water it was found a much lower acceptance of the order of a few percent [3].

6.4 Summary

This chapter concludes the presentation of the theory and algorithms of Interface Sampling. Before switching to a real application with the study of crystal nucleation, we want to summarize here some of the literature.

Most of what we presented in chapters 3-5 has appeared in [21,126,109,20,127]. In [20] more details can be found on the TIS shooting acceptance criterion for stochastic dynamics. Also in [20] one can find more techniques of Interface Sampling that we have not presented here, such as the use of CBMC based shooting moves, an algorithm for the calculation of activation energies, different algorithms for the flux computation, and the use of time as an order parameter. Up to now the TIS method has been successfully applied to two realistic cases, the folding of a polypeptide [3], and hydration of ethylene [119]. These results show that TIS is capable of studying rare events processes in complex systems efficiently and should encourage even more challenging applications.

7 Crystal Nucleation of LJ particles

In the conventional Becker-Döring nucleation theory, [...] all properties of the cluster are uniquely fixed by their size, fluctuations in the cluster properties to be described by other coordinates being neglected.

Kurt Binder [128]

In chapters 3,4, and 5, we have developed the interface sampling methods and implemented them on model systems. Here, we want to apply Interface Sampling to a more realistic case, the solid-liquid nucleation of a system of Lennard-Jones particles. A supercooled liquid is in a metastable state because solidification is not initiated until clusters of a critical size are formed (nucleated). Due to the presence of a free energy barrier, this nucleation process is a rare event. Though the system is fairly simple, the process is definitively not trivial.

7.1 The problem of nucleation

Liquids can be cooled significantly below the freezing temperature and kept in this state without crystallizing. The phenomenon is known as undercooling, and its experimental observations on water date back to the beginning of the 18th century ¹. An undercooled system is thermodynamically metastable, being by definition in a region of the phase diagram where the corresponding solid phase is more stable. In most experimental situations the freezing is triggered by some disturbances, such as dust, shocks, or the effect of the container walls. These heterogeneities help to initiate the crystallization, which would otherwise occur only because of spontaneous fluctuations. We restrict ourselves to this latter case, called homogeneous nucleation. As an activated process, it has difficulties to start spontaneously, but proceeds very easily once started.

¹D.B. Fahrenheit, Phil. Trans. Roy. Soc. **39**, 78 (1724)

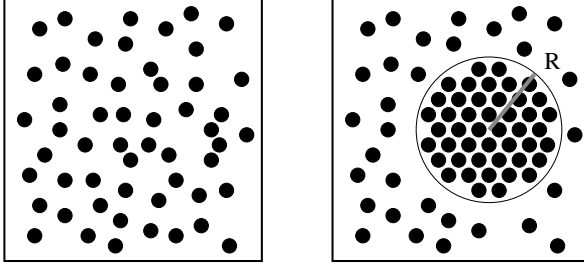


Figure 7.1: Left: undercooled liquid. Right: undercooled liquid plus a spherical droplet of bulk solid phase

The reason for this phenomenon is that the transition to the solid phase proceeds by formation of small nuclei, and can be qualitatively explained using classical nucleation theory (CNT). Nuclei are assumed to be spherical regions of solid phase immersed in the undercooled liquid. Consider a system I of pure liquid, and a system II where a spherical droplet of radius R is present (see fig. 7.1). The Gibbs free energy difference between the two is written as

$$\Delta G = G^{\text{II}} - G^{\text{I}} = 4\pi R^2 \gamma + \frac{4}{3}\pi R^3 \rho_s \Delta\mu \quad (7.1)$$

where γ is the liquid-solid surface free energy density, ρ_s is the number density of the bulk solid, and $\Delta\mu = \mu_{\text{sol}} - \mu_{\text{liq}}$ is the difference in Gibbs free energy per particle between the liquid and the solid. The first term is a surface free energy term and is always positive because of the work that must be done to create an interface. The second term is a bulk term, and since we are in an undercooled state $\Delta\mu < 0$ because the solid is more stable than the liquid. As a result of the competition between the two terms, the free energy as function of R displays a maximum at a critical nucleus size R^* (see Fig. 7.2). Nuclei of radius smaller than R^* tend to shrink, but when sizes larger than the critical size are attained, the solid nuclei can grow indefinitely. From Eq. (7.1) we can derive an expression for the barrier height $\Delta G^* = \Delta G(R^*)$

$$\Delta G^* = \frac{16\pi\gamma^3}{3\rho_s^2\Delta\mu^2} \quad (7.2)$$

which is inversely proportional to $\Delta\mu^2$. If we heat the liquid, it becomes less undercooled, and we approach the melting temperature T_m from below. Then $\Delta\mu$ vanishes and in the thermodynamic limit the barrier becomes infinitely high, so that the two system can coexist at $T = T_m$.

As shown in appendix G, the nucleation rate per volume, usually denoted with I , can be written in classical nucleation theory as

$$I = \rho_l \frac{24D_S(n^*)^{2/3}}{\lambda_D^2} \sqrt{\frac{|\Delta\mu|}{6\pi k_B T n^*}} e^{-\beta\Delta G^*} \quad (7.3)$$

where ρ_l is the density of the liquid, n^* is the number of particles in the critical nucleus, D_S is a self-diffusion coefficient, and λ_D is a typical diffusion distance for particles to attach to the critical nucleus. Expression (7.3) is in qualitative agreement with experiments [129, 130], but in most cases neither λ_D nor γ are accurately known,

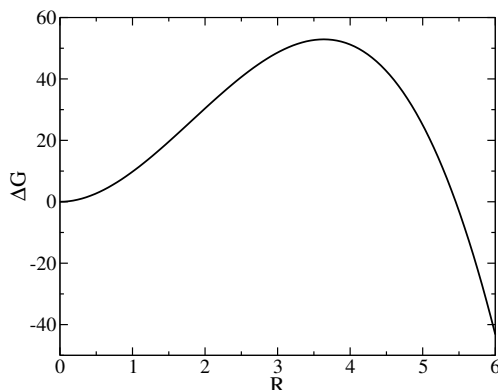


Figure 7.2: The Gibbs free energy of a spherical nucleus as function of its radius R as given by classical nucleation theory, Eq. (7.1).

and they are usually used as free parameters to fit the experimental data. Though theoretical refinements as well as more accurate experimental methods have been developed [131], CNT is too crude an approximation in real cases and a microscopic picture of the nucleation process is still lacking. Computer simulations are then a natural tool to help the understanding at fundamental level.

7.1.1 Rare event simulations of nucleation

Close to coexistence we can approximate [129] $\Delta\mu \approx \Delta h \Delta T$, where Δh is the enthalpy change per particle at coexistence and

$$\Delta T \equiv (T_m - T)/T_m \quad (7.4)$$

is the *degree of undercooling*. From Eqs. (7.3) and (7.2) we see that when ΔT decreases, the rate decreases exponentially as $\exp(1/T(\Delta T)^2)$. Hence, when ΔT is low, the inverse nucleation rate may exceed the accessible computer time. Given the limits of current computers on the size of the simulation box and the time length of the simulations, one has to impose very large undercoolings of the order of 40% in order to observe spontaneous nucleation [131, 132]. The results for such cases are not reliably applicable to experimental cases at realistic undercoolings closer to coexistence.

In [131, 133, 134, 135] ten Wolde, Ruiz-Montero and Frenkel studied a Lennard-Jones system at a moderate undercooling of 20%, and treated the nucleation as a rare event using the Bennett-Chandler procedure (see sec. 1.5). Considering NPT simulations at two different pressures, they computed the nucleation free-energy barrier as function of a structural order parameter, the global Q_6 (see later, sec. 7.2.3), by means of Umbrella Sampling, and calculated the transmission coefficient using the Bluemoon ensemble technique. They could thus obtain the nucleation rate and investigate the nucleation mechanism. In the precritical phase, nuclei are found to be mainly body-centered-cubic (bcc) ordered. This preference for bcc is in accordance with a scenario proposed by Alexander and McTague [136], who on the basis of general symmetry considerations, concluded that in three dimensions the first nucleated phase for simple fluids is bcc. As the nuclei grow to the critical size, the bcc structure becomes confined to the surface, while the core develops a face-centered-cubic (fcc)

phase. Critical and postcritical nuclei retain a bcc shell of approximately constant width, but the interface between the crystal nuclei and the surrounding liquid is quite diffuse. Both the density and the structural order parameter decay smoothly to a liquidlike value. The calculation of the transmission coefficient showed the nucleation process to be distinctly diffusive. When the system is near the top of the barrier, it remains close to the top and the size of the largest cluster fluctuates around the critical size. The overall transition rate was found to be two or three orders of magnitude larger than the one predicted by CNT or found in later calculations [137]. Moreover, the mechanism analysis was performed using the equilibrium free energy calculations along a predetermined reaction coordinate, a procedure that might hinder kinetic effects (see sec. 1.7).

Summarizing, straightforward MD is not realistic, and the TST-BC investigation might be missing some aspects of the process. Here, we do not want to rely on Eq. (7.3) and we avoid using the TST-BC procedure, because we do not want to assume a specific way of how nucleation proceeds by choosing a reaction coordinate. For an investigation of the nucleation mechanism, in line with a first principle study, possibly free of biases, nucleation is then a natural application for Interface Sampling.

7.2 Methodology

Here we describe the system and the MD scheme used in the TIS and PPTIS simulations. We discuss then the preparation of the system in the initial undercooled liquid state, and the choice of an order parameter to characterize the nucleation towards the final solid phase.

7.2.1 The system

We consider a system of N particles in three dimensions interacting through a pairwise Lennard-Jones potential, Eq. (2.51). All the particles have equal mass m , and we use reduced units, so that the LJ unit of energy ϵ , the LJ unit of length σ and m are unity. The LJ unit of time $(m\sigma^2/\epsilon)^{1/2}$ is therefore also unity.

Phase transitions are usually studied in the isobaric-isothermal ensemble performing separate simulations of the two phases at the coexistence point [11]. In our case however, we investigate the growth of one phase inside the other. In reality, when a liquid solidifies, for example at room temperature, the latent heat of solidification is dissipated into the environment, acting as a reservoir. Instead in simulations the system temperature is kept constant using an artificial thermostat, like stochastic noise or additional variables coupled to the system, but is never a real bath of surrounding particles. The same reasoning applies to a variation in (specific) volume, which is dissipated by the environment for example at atmospheric pressure. To avoid artificial biases, we should use in principle the constant energy microcanonical ensemble and simulate a very big system comprising also a sufficiently big environment. Since this is not feasible, we chose for a compromise, and simulated the system in the isobaric-isenthalpic ensemble (NPH). We keep the pressure constant, but we let the system develop latent heat as the nucleation proceeds.

To simulate the NPH ensemble we applied an extended hamiltonian method by

Andersen [77] using a Trotter factorization of the associated Liouville operator [76] with a time step $\Delta t = 0.01$. The mass of the piston was $W = 0.0027$. We used a cut and shifted LJ potential with a cutoff radius set at 2.5. Long-range corrections to pressure and potential energy were applied on the fly, directly in the force subroutine at every time step. In order to speed up the simulation we used a Verlet neighbor list and a linked list to update the neighbor list [138]. Further details on the NPH dynamics and the tail corrections can be found in appendix H.

To perform equilibration runs we also employed NPT simulations using a Berendsen thermostat [139]. In equilibration and test simulations the number of particles ranged ² from $N = 1000$ to $N = 4000$. The production runs were all performed in the NPH ensemble with $N = 10648$.

7.2.2 LJ phase diagram

We benefitted from the extensive literature on the Lennard-Jones system to check the outputs of our MD code and to locate the desired undercooling point in the phase diagram.

Since the time the potential was introduced by Lennard-Jones [140], numerous theoretical and numerical studies have been performed. The coexistence curves were investigated by Hansen and Verlet in 1969 [141]. The following studies on the gas-liquid coexistence were reviewed in 1993 by Johnson, Zollweg and Gubbins [142], who fitted the simulation data to an accurate equation of state for the fluid phase. In the middle of 1990's the full LJ phase diagram was derived by Kofke [143] and Agrawal and Kofke [144] using the Gibbs-Duhem integration. Recently, the coexistence curves have been computed again by means of absolute free energy calculations. Van der Hoef derived in [145] an equation of state for the solid phase and theoretical fits to the solid-liquid coexistence curves. In [146] the same author extended the fits to the gas-liquid coexistence. One of the last computations of the complete LJ phase diagram was done by Barroso and Ferreira in [147]. We summarize the results of the cited literature in Fig. 7.3. The LJ solid state is face-centered-cubic (fcc), and argon and the other rare gases with the exception of helium are known to crystallize in that structure [148]. One might ask whether a bcc solid phase exists in some regions of the phase space. It is known that hard-core potentials tend to favor close-packed structures, while soft-core potentials prefer body-centered-cubic (bcc) [149,150]. Even though the bcc structure has been found mechanically unstable for LJ systems at room temperature [151], there might be the possibility of a stable bcc phase at high temperatures and pressures, as suggested for other more realistic potentials for rare gases [152,153], but to our knowledge no fcc-bcc transition has been found for LJ systems. Anyway, we are not concerned with these special cases, and we consider the solid LJ phase to have a close-packed fcc structure.

In our simulations the pressure is constant and we chose the value $P = 5.68$, used also by ten Wolde [131]. For this pressure the theoretical melting temperature is $T_m = 1.10676$. To prepare the system in the undercooled state we first melted a simple cubic lattice at a temperature $T = 2.0$, and then using NPT simulations we followed

²In the case of lattice simulations we took care of choosing an N such that the number of atoms per unit cell were the characteristic one of the lattice

an isobar decreasing gradually the temperature until $T = 0.83$, corresponding to 25% degree of undercooling, Eq. (7.4). This corresponds to a nucleation event less rare than in the system of ten Wolde (see sec. 7.1.1). Since the computational procedure is hence less expensive, we chose these conditions as a first attempt to test the interface sampling. The undercooled state was then equilibrated with an NPH simulation, yielding a constant enthalpy per particle $H/N = 1.412$. The final state was used as the starting point for the Interface Sampling. The equilibration procedure is also visualized in Fig. 7.3.

7.2.3 Choice of the order parameter

In order to distinguish the solid and liquid state we used a method developed by ten Wolde [131] that is able to find the solid clusters in the system. The method is based on an algorithm that is able to identify if a particle i is solid or liquid by looking at the orientational order around the particle. The algorithm makes use of the complex bond-order parameter $\bar{q}_{6m}(i)$, $m = -6 \dots 6$, defined in appendix I, and works as follows.

1. define the neighbors of a particle i as all the particles j within a cutoff radius r_q . The radius r_q can be taken from the first minimum of the $g(r)$.
2. compute for each particle the bond-order parameter \bar{q}_{6m} , Eq. (I.1)
3. for a fixed particle i loop over the neighbors j and compute the normalized dot-product

$$d_{ij} = \frac{\sum_{m=-6}^6 \bar{q}_{6m}(i) \bar{q}_{6m}^*(j)}{\left(\sum_{m=-6}^6 |\bar{q}_{6m}(i)|^2\right)^{1/2} \left(\sum_{m=-6}^6 |\bar{q}_{6m}(j)|^2\right)^{1/2}} \quad (7.5)$$

As shown in appendix I, $d_{ij} = d_{ji}^* = d_{ji}$. Therefore $d_{ij} \in \mathbb{R}$ and moreover $-1 \leq d_{ij} \leq 1$. The distribution of dot-products has a different behavior in solids and liquids. In particular in solids tends to be peaked towards 1, while in liquids is broader around 0. We define then particles i and j to be *connected* if d_{ij} exceeds a given threshold (in our case was 0.5).

4. in this way, for each particle one can compute the Number of Connections per Particle (NCP). Again, solids tend to have more connections per particle than liquids, and using another threshold (in our case 8.5), we define a particle to be solid if its NCP exceeds the threshold. Otherwise the particle is liquid.

Once particles are distinguished into solid and liquid ones, we apply the criterion that two solid particles belong to the same cluster if they are neighbors, i.e. if their relative distance is less than r_q . Having identified all the solid clusters in the system, we choose as an order parameter λ for the definition of the interfaces (see sec. 3.2.3) the size of the biggest solid cluster $\lambda = n_{\text{big}}$.

Instead of using a local order parameter such as n_{big} , another possibility is to use a global order parameter, depending on the structure of the whole system. Van Duijnveltdt and Frenkel [154] have shown that a set of bond-order parameters introduced

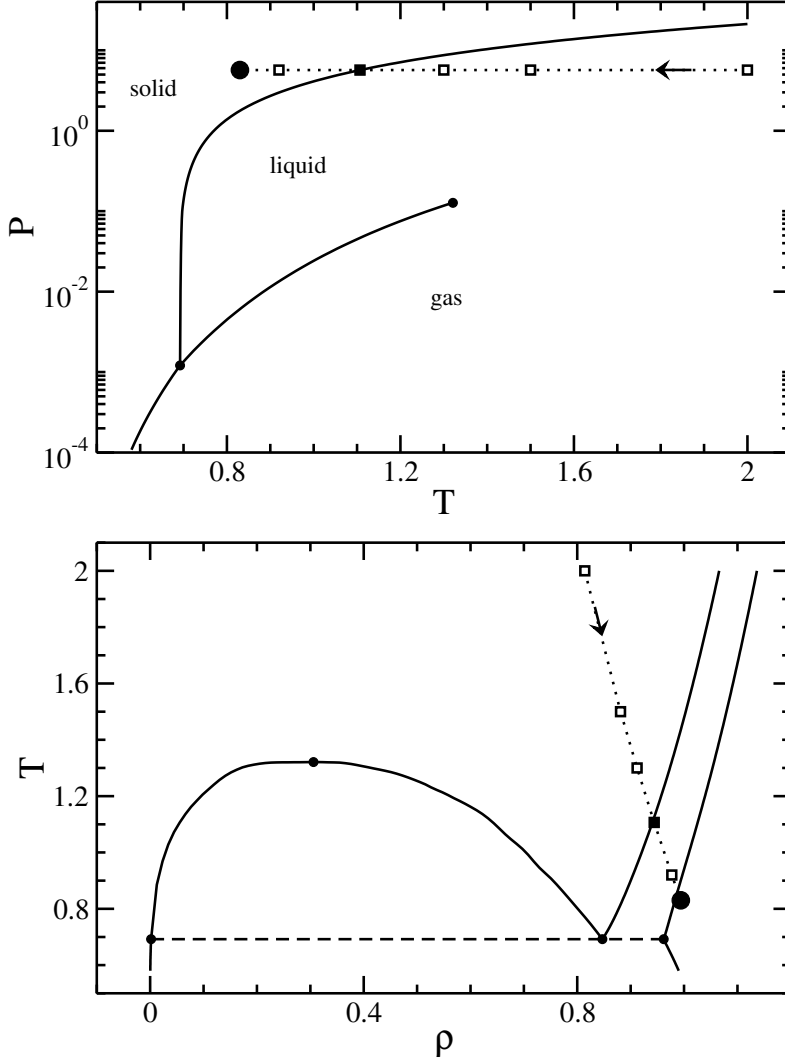


Figure 7.3: LJ phase diagram in the temperature-density plane (lower figure) and in the pressure-temperature plane (upper figure). We also indicated the solid (fcc), liquid and gas regions. For the solid-liquid coexistence curves we used the theoretical fits of [145], for the solid-gas the ones of [146], and for the solid-liquid curves we used the data of [143]. The critical point is located at $(T_c, \rho_c, P_c) = (1.321, 0.306, 0.127)$ and the triple point at $(T_{tr}, P_{tr}) = (0.692, 0.00121)$, where the three phase coexist with densities $\rho_{tr}^{gas} = 0.00178, \rho_{tr}^{liq} = 0.847, \rho_{tr}^{sol} = 0.962$. The open squares indicate the isobar which was followed at $P = 5.68$ to undercool the system. The solid square indicates the corresponding coexistence temperature $T_m = 1.10676$ and the final circle is the starting point for Interface Sampling simulations, at $T = 0.83$, corresponding to 25% undercooling.

by Steinhardt, Nelson and Ronchetti [155] might be used to investigate nucleation. These order parameters are sensitive to the degree of spatial orientational correlation of the vectors joining neighboring particles. In a liquid there is only local orientational order, the correlations decay rapidly and all bond-order parameters vanish in the thermodynamic limit. In a crystal instead, the vector orientations are correlated and the bond-order parameter are of order 1. In appendix I we define the bond-order parameters, and in table 7.1 we report their values for some typical structures. In his study of crystal nucleation [131], ten Wolde used Q_6 as a crystalline order parameter, because it vanishes in the liquid phase, while it is large for the simple crystal lattices of interest. However, he also proved that the use of global order parameters, as Q_6 , might induce artificial effects. When the total number of solid particles is small, the system tends to distribute them in many small clusters because of entropic reasons. Instead when the total number of solid particles is large, the interfacial free energy dominates and a single big cluster is more favorable. In the study of nucleation, the number of solid particles increases, and an undesirable discontinuous crossover from one behavior to the other was found, which is also in contrast with the CNT picture of one growing nucleus. This effect disappears in the thermodynamic limit, but it is not negligible in computer simulations. The use of a local order parameter such as n_{big} is therefore more recommendable. Moreover, with this choice of the order parameter the simulations and the free energy computations can be related to CNT, which is based on the picture of a growing nucleus. The bond-order parameters of table 7.1 will be used however to analyze the configurations and distinguish different crystal structures.

In order to check the proposed algorithm for the identification of the biggest solid cluster and to set the related parameters, we prepared the system in a liquid, body-centered cubic (bcc), and face-centered cubic (fcc) structures, and equilibrated them at the pressure $P = 5.68$ and a temperature $T = 1.15$, close to coexistence. These equilibration runs were followed by NPH simulations. First, we computed the radial

Geometry	Q_4	Q_6	\hat{W}_4	\hat{W}_6
hcp	0.097	0.485	0.134	-0.012
fcc	0.191	0.575	-0.159	-0.013
bcc	0.036	0.511	0.159	0.013
sc	0.764	0.354	0.159	0.013
ico	0	0.663	0	-0.170
(liq)	0	0	0	0
fcc-eq	0.080	0.388	-0.159	-0.013
bcc-eq	0.023	0.358	0.159	0.013

Table 7.1: Bond orientational order parameters for different cluster geometries. hcp: hexagonal close-packed structure, fcc: face-centered-cubic structure, bcc: body-centered-cubic structure, sc: simple cubic structure, ico: icosahedral. We also report the parameters for a bcc and fcc lattice equilibrated at the condition of our NPH simulations, $P = 5.68$, $H = 1.412$, corresponding to 25% undercooling. Note that the Q_4, Q_6 change when going from the perfect lattice to the equilibrated one, but the \hat{W}_4, \hat{W}_6 do not.

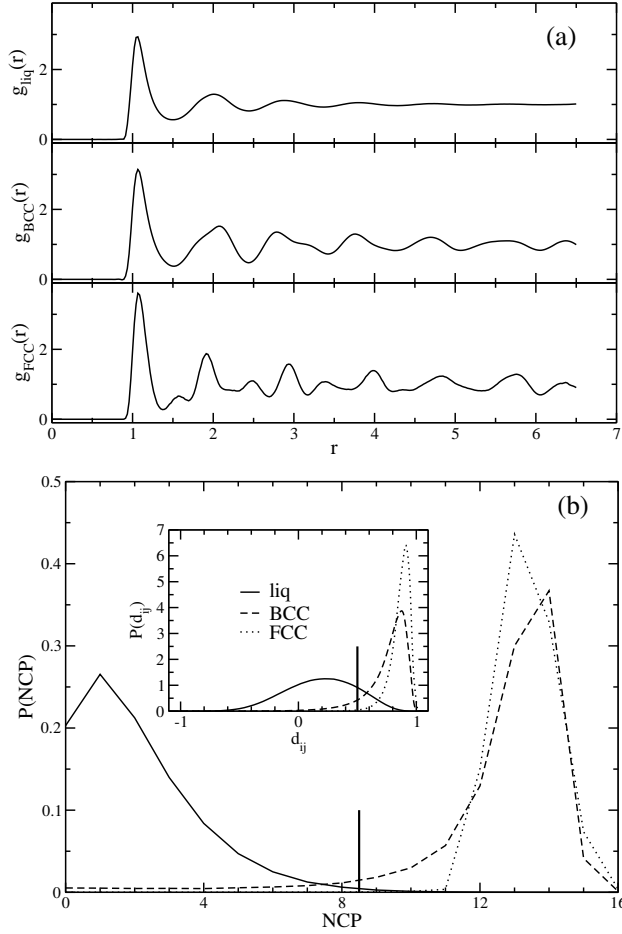


Figure 7.4: Top: radial distribution function $g(r)$. Bottom: distribution of the number of connections per particle, and in the inset distribution of the dot-product Eq. (7.5). The thresholds for the cluster-recognition algorithm are also shown. All curves were computed from simulations of liquid, bcc and fcc structures equilibrated at $P = 5.68$ and $T = 1.15$, close to coexistence.

distribution functions $g(r)$, shown in Fig. 7.4a. The $g(r)$ gives an estimate of a value for the cutoff r_q from the first minimum at 1.5. Second, similar to the simulations of ten Wolde [131], we computed the dot-product distribution, reported in the inset of Fig. 7.4. Clearly the liquid state can be distinguished from the different solid states. Following [131] we chose a threshold value of 0.5 to consider two particles connected. Third, we computed a histogram of the number of connections per particle, plotted in Fig. 7.4b. Since for fcc the minimum of the $g(r)$ is slight less than 1.5, more than one shell is taken into account by the algorithm and the NCP is peaked at 13 instead of 12. This is however not a problem, since we are only interested in distinguishing liquid from solid configurations. Taking a threshold of 8.5 we identify particles as solid if their number of connections is 9 or more. Otherwise they are considered liquid.

7.3 Rate constant

We computed the NPH rate constant k_{AB} for the nucleation transition from under-cooled liquid (state A) to solid (state B) using TIS and PPTIS.

In both methods the first stage is the calculation of the factor $\langle \phi_{1,0} \rangle / \langle h_A \rangle$, the effective flux through interface λ_1 of the trajectories coming from interface λ_0 , which defines the initial state A (see sec. 3.2.1). After some short inspection runs we decided to define $\lambda_0 = 25.5$ and $\lambda_1 = 30.5$, which allow us to compute a flux factor big enough to gather sufficient statistics in a reasonable computer time. In Fig. 7.5 we report $\lambda(t) \equiv n_{\text{big}}(t)$ on a trajectory of time length 20. Using formula (3.20) of sec. 3.2.1, we obtain the flux factor from the ratio of the number of effective positive crossings of λ_1 to the total trajectory time length (see also inset of Fig. 7.5). Averaging on a series of 30 runs, each of time length 200, we obtained the value $\langle \phi_{1,0} \rangle / \langle h_A \rangle = 1.29 \pm 0.03$.

A part of a flux trajectory that crosses λ_1 coming from A is by definition a path belonging to the first TIS window ensemble (see sec. 3.2.3). We chose one of such parts out of the flux series and initiated a path sampling simulation. Interface λ_2 was set using the rule of sec. 3.2.3, that the crossing probability $P_A(2|1)$ should approximately be 0.2. After some tests we found a convenient position at $\lambda_2 = 40.5$, and we performed a production run of 10 series of 100 paths each. The path-reversal move was applied with a probability of 10% and the momentum displacement in the shooting move was chosen to obtain an acceptance around 40% (see sec. 3.2.3). One of the paths $A \rightarrow \lambda_2$ was then used to start a simulation in the following window ensemble. We iterated the procedure for a total of 12 windows, employing the additional interfaces $\lambda_3 = 60.5, \lambda_4 = 80.5, \lambda_5 = 110.5, \lambda_6 = 130.5, \lambda_7 = 160.5, \lambda_8 = 190.5, \lambda_9 = 230.5, \lambda_{10} = 270.5, \lambda_{11} = 310.5, \lambda_{12} = 360.5$, and $\lambda_{n_I} = 410.5$ with $n_I = 13$. We also used subinterfaces to smooth the results. The rematched crossing probability function is reported in Fig. 7.6 and shows a plateau which defines the last interface and the state B . In TIS a path that reaches a nucleus of size $n_{\text{big}} = 410.5$ is committed to grow until complete solidification. The final crossing probability is $P_A(n_I|1) = (8 \pm 6)10^{-7}$, and multiplying by the flux, the rate constant is $k_{AB} = (1.0 \pm 0.8)10^{-6}$.

The first PPTIS simulation was also initiated using the trajectories from the flux calculation. The third interface was defined like in TIS, at $\lambda_2 = 40.5$, and then we simply used a fixed interface separation defining $\lambda_i = 40.5 + 20 * (i - 2)$, for $i = 2 \dots 38 \equiv n_I$. A series of 100 simulations of 200 paths each was performed in a total of 37 windows. Shooting and path-reversal moves were applied with equal probability. At the same time the free energy was computed using the loop-boundary method of chap. 5 and a bin width $\delta\lambda = 1$. We pushed PPTIS until $n_{\text{big}} = 760$ (using interface $\lambda_{38} = 760.5$) to check the attainment of a plateau and to obtain the corresponding free energy until $n_{\text{big}} = 740$. The resulting long-distance probability $P_{n_{\text{big}}}^+$ is plotted in Fig. 7.6 together with the TIS crossing probability $P_A(n_{\text{big}}|1)$. We did not perform a systematic test of the memory loss assumption using for example the MLF method of sec. 4.2.1. However, comparing TIS and PPTIS we can state a posteriori that this system is diffusive enough to satisfy memory loss within our choice of interface separation. As seen in Fig. 7.6, PPTIS reaches a plateau at a value of the final probability $P_{n_I}^+ = (1.4 \pm 0.9)10^{-6}$, which coincides with the plateau value of TIS within the error. Consequently, the final PPTIS rate constant $k_{AB} = (2 \pm 1)10^{-6}$ also coincides with the TIS rate.

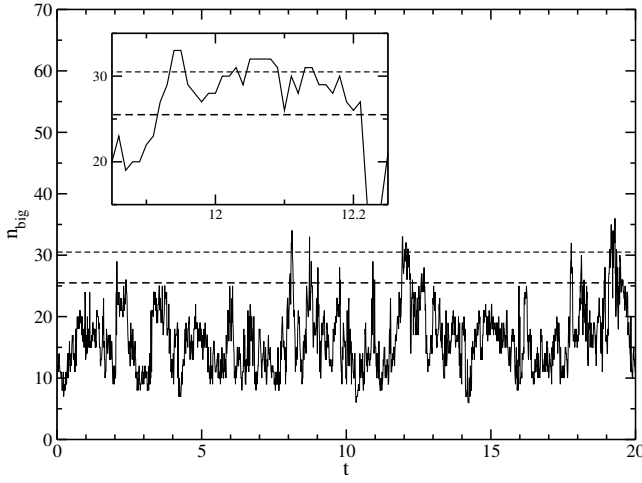


Figure 7.5: A trajectory in the undercooled state. Interfaces at $\lambda_0 = 25.5$ and $\lambda_1 = 30.5$ are shown. The flux is computed counting the number of peaks above λ_1 that come and return to A immediately before and after. In the example in the inset then only one peak is counted.

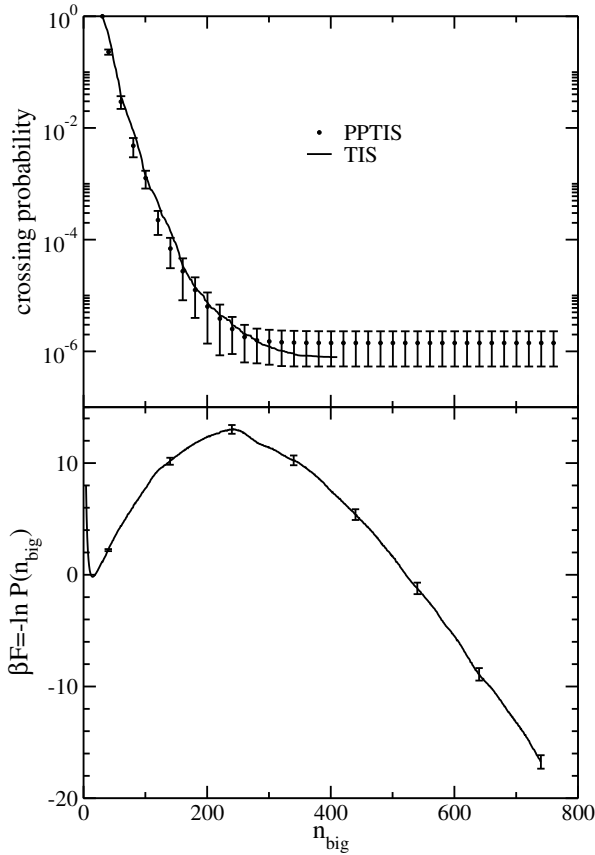


Figure 7.6: Top: TIS and PP-TIS crossing probability. The error on the TIS curve is comparable to the error of the PP-TIS one. Bottom: free energy $-\ln P(n_{\text{big}})$.

Reintroducing units, the rate k_{AB} is in units of τ^{-1} . Nucleation rates I are usually expressed as nuclei formed per unit time per volume (see sec. 7.1). We can obtain this rate just dividing k_{AB} by the volume of the simulation box, which in turn is equal to N/ρ_{liq} , where the liquid density is $\rho_{liq} = 0.994$ and the number of particles $N = 10648$. The result is $I = (9 \pm 7)10^{-11}\sigma^{-3}\tau^{-1}$. Finally, using the values of $\sigma = 0.3405nm$ and $\tau = 2.156ps$ for argon, we get $I = (1.1 \pm 0.9)10^{-6}nm^{-3}s^{-1}$. As noted by ten Wolde, on an experimental scale this means that argon would crystallize basically instantly, and this is in accordance with the known difficulty in supercooling argon [131].

The free energy $\beta F(n_{big}) = -\ln P(n_{big})$ is plotted in Fig. 7.6. At small values of n_{big} we used histogram data from the flux simulations in state A.

The free energy curve $\beta F(n_{big})$ shows an artificial minimum, while the correct free energy should be derived by the distribution of cluster sizes $P(n)$ (see appendix G.2). Only for the biased region, is $P(n)$ equal to $P(n_{big})$, as the probability to find a second large cluster is very low. We plot the correct free energy³ $\beta F(n) = -\ln P(n)$ in Fig. 7.7. It displays a maximum at a critical nucleus $n^* = 243$ for which $\beta F(n^*) = 25.2 \pm 0.7$.

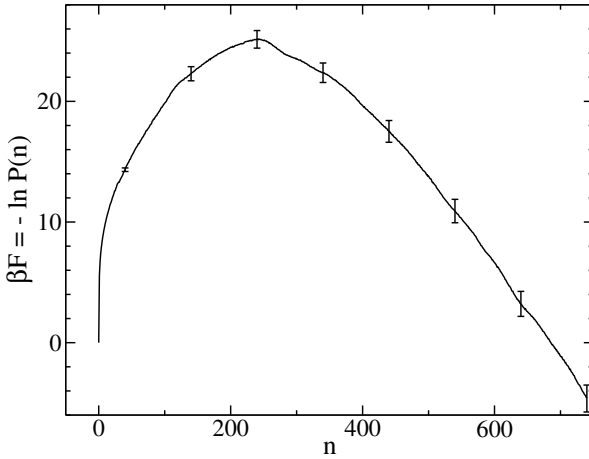


Figure 7.7: Free energy $-\ln P(n)$, as derived from $P(n_{big})$ corrected for small values of n . The correction amounts to compute $P(n)$ at low n from MD simulations in the undercooled liquid, and results in a steep initial increase of the free energy profile.

7.4 Mechanism

We investigated the transition mechanism using paths generated in the last TIS window ensemble. When a path going from A to the last interface (at $\lambda_{n_I} = 410.5$) was found, we first checked if it corresponded to a real transition by elongating it in time and testing if it reached $n_{big} > 1000$ within a maximum transition time of 300. If that was the case, the path was saved for later analysis. An average decorrelation time of 5 successful shooting moves was then waited before a new path $A \rightarrow \lambda_{n_I}$ was checked. In this way we generated a total of 84 NPH transitions going from an undercooled state A for $n_{big} \leq 25$ to a solid state for $n_{big} > 1000$.

³We remind that because of our choice of the dynamics, this is an NPH free energy, and not a Gibbs NPT free energy.

7.4.1 Path analysis

In this section we describe the procedure we used to analyze each timeslice of each generated transition pathway.

We can distinguish the chosen observables into global ones, functions of all particles' coordinates and momenta, and local ones, computed only on the particles defining the biggest solid cluster.

The first kind include, for instance, the slice time index t , the pressure P , the density ρ , the temperature T , the potential energy per particle \mathcal{U}/N , the total energy per particle E/N , and the enthalpy per particle H/N . As indicators of the global ordering in the system we used the bond-order parameters $Q_4, Q_6, \hat{W}_4, \hat{W}_6$. From the algorithm for the identification of the biggest solid cluster (see sec. 7.2.3) we also obtain the total number of solid and liquid particles, as well as the number of all solid clusters in the system. To the global observables naturally belongs the committor p_B , defined in sec. 2.5 and computed as explained in sec. 2.5.1. The minimum number of shooted trajectories N_{min} to obtain p_B ranged from 10 to 30, and the total number of trajectories was chosen to reach an error σ_{p_B} ranging from 0.05 to 0.1, fixed for every slice. Hence, because of Eq. (2.59), the slices for which $p_B \simeq 0.5$ require the biggest number of shooted paths, and are the computationally most expensive. In addition, we tested a new idea on a generalization of the committor that includes not only the trajectories initiated from the timeslice and integrated forward in time, but also the backward ones. We report the results in appendix J.

The second set of observables, computed from the cluster particles only, can be further divided into quantities determining the cluster shape and indicators of cluster structure.

In order to characterize, besides the nucleus size n_{big} , also the shape, we calculated the radius-of-gyration tensor [156]

$$\mathbf{R}_g^2 = \frac{1}{n_{big}} \sum_{i=1}^{n_{big}} \mathbf{r}_i \mathbf{r}_i \quad (7.6)$$

where the positions \mathbf{r}_i of cluster particles are referred to the cluster center of mass. The eigenvalues of the tensor ⁴ are denoted in descending order as $R_{g1}^2 \geq R_{g2}^2 \geq R_{g3}^2$ and the radius of gyration is defined as $R_g^2 = R_{g1}^2 + R_{g2}^2 + R_{g3}^2$.

In order to characterize the cluster structure, we employed again the bond-order parameters $Q_4, Q_6, \hat{W}_4, \hat{W}_6$, but now defined using only the bonds between cluster particles. To have a further indication on the formation of bcc, fcc, or liquid-like structures, we used an analysis technique developed by ten Wolde [131], and explained in detail in appendix I. This technique is based on the distributions of the local bond-order parameters $q_4(i), q_6(i), \hat{w}_6(i)$, defined for each cluster particle i . In

⁴ The tensor \mathbf{R}_g^2 defined by Eq. (7.6) is closely related to the inertia tensor \mathbf{I} , Eq. (2.53). For a cluster of N particles with equal masses m , the relation $\mathbf{I} = mN(R_g^2 \mathbf{1} - \mathbf{R}_g^2)$ holds, where $\mathbf{1}$ is the identity tensor. The physical meaning of the two tensors is however different. The moments of inertia I_i result from an average of the square particle distances *from* axis i , which is useful to characterize the rotation of the body around the axis. The radii of gyration $R_{g,i}^2$ are instead the mean square distances *along* axis i , and are indeed useful to characterize the shape of the body, since each $R_{g,i}^2$ directly corresponds to the body elongation in direction i .

separate NPH simulations we computed these distributions in reference systems of pure liquid, bcc, and fcc particles, equilibrated at the same conditions of our system ($P = 5.68$, $H/N = 1.412$). The distributions calculated for an arbitrary cluster structure can then be projected onto the distributions of these reference systems, yielding three scalars f_{liq} , f_{bcc} , f_{fcc} that quantify respectively the degree of liquid, bcc and fcc structure in the cluster. A parameter Δ^2 quantifies the contribution of the remaining components, orthogonal to the reference systems. The smaller Δ^2 , the better the quality of our projection.

Finally, in the structure analysis of a cluster, we also distinguished the cluster particles into bulk and surface particles. For a particle to be of bulk type, the number of cluster neighbors should be 12 or more. Otherwise it is considered a surface particle. We thus obtained the number of surface particles n_{surf} and the number of bulk particles $n_{\text{bulk}} = n_{\text{big}} - n_{\text{surf}}$. The above structure analysis with the use of $Q_4, Q_6, \hat{W}_4, \hat{W}_6$ and $f_{liq}, f_{bcc}, f_{fcc}, \Delta^2$ was applied both to the cluster as a whole and to the core only, with the surface particles removed.

7.4.2 Transitions

We show in Fig. 7.8 a typical transition. As the nucleation advances, and the size n_{big} of the biggest cluster increases, the potential energy decreases, and the global density increases. Since we perform NPH simulations, the temperature must also increase to maintain the total enthalpy constant. The structural bond-order parameter Q_6 also goes up, and eventually reaches the value typical of an equilibrated fcc system. The other bond-order parameters $Q_4, \hat{W}_4, \hat{W}_6$ also plateau to the corresponding fcc values. In Fig. 7.8 we have plotted a trajectory from the liquid state until full solidification of the simulation box, but as specified at the beginning of sec. 7.4, for the transition analysis we stopped the TIS path integration when $n_{\text{big}} = 1000$ is reached (in this case at a time $t = 137$). The distribution of TIS transition pathlengths is shown in Fig. 7.9 and shows the behavior typical of the distributions of first passage times [157, 158]. In Fig. 7.8 we also plot the committor $p_B(t)$ until $t = 137$, which has a distinct diffusive behavior. The trajectory crosses the transition state region $p_B = 0.5$ several times before switching to the final solid state, and during the transition time it even visits again configurations committed to the liquid state at $p_B = 0$.

Subsequently, we analyzed the cluster shape and structure during the transitions as function of the TIS order parameter, the size of the biggest solid cluster n_{big} .

First, we investigated the shape of the clusters. In Fig. 7.10(a) we plot the results of the radius of gyration analysis. For a spherical object R_g^2 scales as $n_{\text{big}}^{2/3}$, whereas for chains R_g^2 scales as n_{big}^α , with $1.2 < \alpha < 2$, depending on the stiffness of the chain [131]. Hence, $R_g^2/n_{\text{big}}^{2/3}$ should approach a constant value for spherical clusters, while it should increase with n_{big} for chain-like clusters. At the very beginning, for $n_{\text{big}} < 30$, $R_g^2/n_{\text{big}}^{2/3}$ slightly increases, and until about $n_{\text{big}} = 200$ the first eigenvalue R_{g1}^2 is significantly larger than the other two. Then the three eigenvalues get close to each other. This indicates that the clusters are first somewhat chain-like, then elongated objects, and finally spherical. A more careful analysis takes into account the dispersion of the shape observables. In Fig. 7.10(b) we plot the number of surface particles, again normalized to $n_{\text{big}}^{2/3}$ for all the clusters in all the transitions. Indeed,

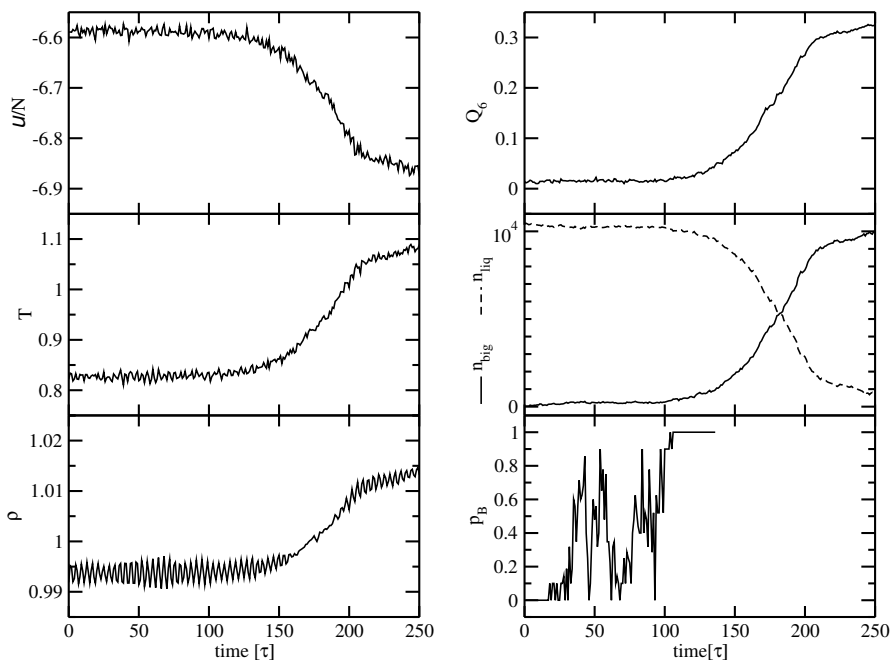


Figure 7.8: Several observables as function of time for a typical transition: the potential energy per particle U/N , the temperature T , the density ρ , the global bond-order parameter Q_6 , the order parameter n_{big} size of the biggest cluster in the system, together with the number of liquid particles. The bottom right panel shows the committor p_B , and the error on the points is ± 0.1 . The original TIS path was sampled until n_{big} reaches 1000 at $\tau = 137$. Here the path was extended until full solidification of the simulation box.

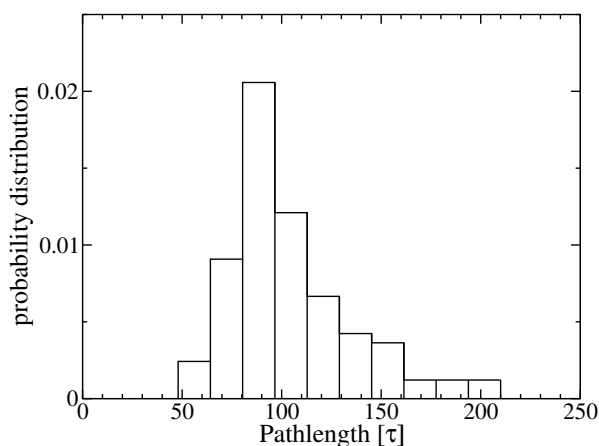


Figure 7.9: Distribution of transition pathlengths.

up to $n_{\text{big}} = 30$, $n_{\text{surf}}/n_{\text{big}}^{2/3}$ follows the curve $n_{\text{big}}^{1/3}$, corresponding to $n_{\text{bulk}} = 0$ (dashed line in figure), meaning the clusters are not entirely compact. Then the average $n_{\text{surf}}/n_{\text{big}}^{2/3}$ (solid line in figure) becomes more flat quite rapidly around $n_{\text{big}} = 100$ and finally approaches the constant value 4.8 typical of spheres⁵. However there is substantial scattering of the $n_{\text{surf}}/n_{\text{big}}^{2/3}$ points around the average, more pronounced between $n_{\text{big}} = 200$ and $n_{\text{big}} = 400$. Bringing together these results with the radius of gyration analysis, we can conclude that small initial elongated clusters become spheres, but during the transition the spherical condition is approached with a variance in the compactness. Some of the clusters grow compact, some retain a degree of elongation.

Second, we looked at the structure of the clusters. In Fig. 7.11 we plot the results of the analysis of the bond-order parameter distributions (see sec. 7.4.1). The small value of Δ^2 indicates that the analysis is sensible, and the cluster structure is reasonably well represented by the superposition of fcc, bcc, and liquid structures. As n_{big} increases, the bcc component stays almost constant, while the liquid part decreases to make space for a fraction of fcc particles. Comparing with the results of ten Wolde (see sec. 7.1.1) we can interpret this as a developing fcc core wetted by a constant bcc surface. However, at variance with the findings of ten Wolde, we do not see a transition from bcc- to fcc-dominated structure at the critical cluster size (here $n_{\text{big}}^* = 243$) [131].

Summarizing, from the analysis of the transition paths as function of the size of the biggest cluster, we get the picture of a growing object, spherical on average but with roughness and distortion, which has an increasing fcc component and a constant bcc one. Before we discuss the nuclei in more detail, we would like to address the question of what is the right reaction coordinate to describe nucleation. In path sampling methods this question can, in principle, be answered by the analysis of the committers.

7.4.3 Committer analysis

In the previous section we analyzed the transitions as function of the order parameter n_{big} , effectively considering it as a reaction coordinate. Here we want to check if n_{big} is a good reaction coordinate. For this purpose we computed the committor p_B for each timeslice of each path. This was the computationally most expensive part of our investigation and required about three months of simulation time on one hundred nodes of 1Ghz on average.

In Fig. 7.12 we plot the committor p_B as function of the order parameter n_{big} . Surprisingly, we do not find a clear monotonic curve. States at $p_B = 0$, that are committed to the liquid phase, include configurations containing the smallest clusters, but also configurations with cluster sizes up to around 300. At the opposite side, slices with $p_B = 1$, committed to end in a solid phase, do contain the biggest clusters (up to above $n_{\text{big}} = 1000$), but also clusters with sizes as small as 200. In the middle,

⁵ For a sphere of radius R containing N particles at a density ρ , the relation $N = 4\pi\rho R^3/3$ holds. The number of particles in a surface shell of width ΔR is $N_{\text{surf}} = 4\pi\rho R^2\Delta R$. Consequently $N_{\text{surf}}/N^{2/3} = 4\pi/(4\pi/3)^{2/3}\rho^{1/3}\Delta R$. Using the fact that the cluster density is about 1 (see Fig. 7.18) and for LJ particles the diameter is about 1, we get $N_{\text{surf}}/N^{2/3} \simeq 4.8$.

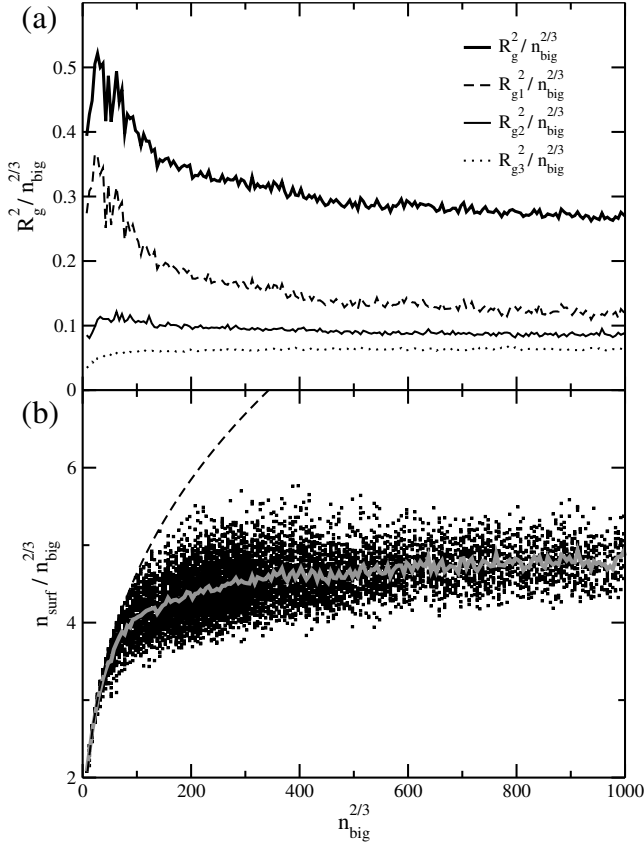
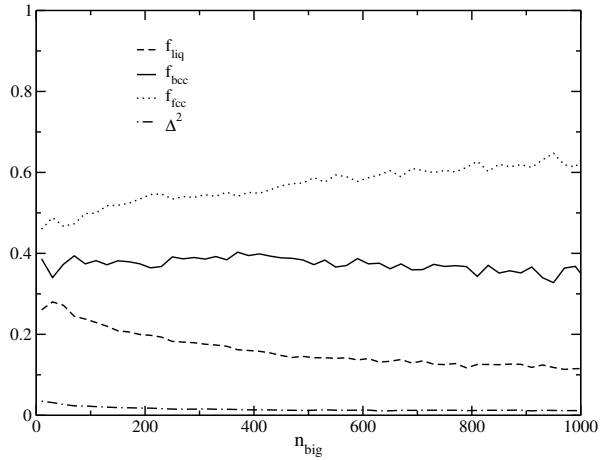


Figure 7.10: (a) Eigenvalues of the radius of gyration tensor, and radius of gyration $R_g^2 = R_{g1}^2 + R_{g2}^2 + R_{g3}^2$. (b) Number of surface particles normalized to $n_{\text{big}}^{2/3}$, for all the biggest clusters of each slice of each path. The average value is also shown. The dashed line represents the curve $n_{\text{big}}^{1/3}$, which would be followed if the cluster consisted of surface particles only ($n_{\text{surf}} = n_{\text{big}}$, no bulk particles).

Figure 7.11: Structural composition of the biggest cluster as function of the cluster size, as found by projections of the bond-order parameter distribution (see sec. 7.4.1). In order to improve the analysis, clusters with common size were regrouped before computing the distributions.



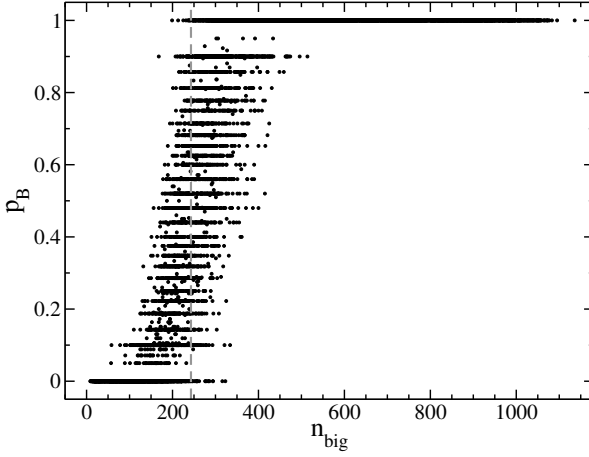


Figure 7.12: Committor p_B as function of the order parameter n_{big} . The critical size $n_{\text{big}}^* = 243$ is reported as a dashed line. The uneven distribution along the y-axis is due to the changing number of trajectories used to compute p_B with a fixed error (see Eq. (2.59)).

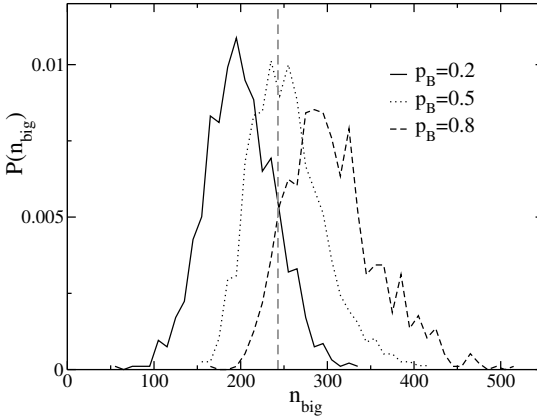


Figure 7.13: Cluster size distributions at different committor values. The dashed line corresponds to $n_{\text{big}}^* = 243$, the maximum of the free energy $\beta F(n_{\text{big}})$ (see Fig. 7.6). Critical nuclei correspond to $p_B = 0.5$. Clearly they do not have one well-defined size, but rather exhibit a broad size distribution.

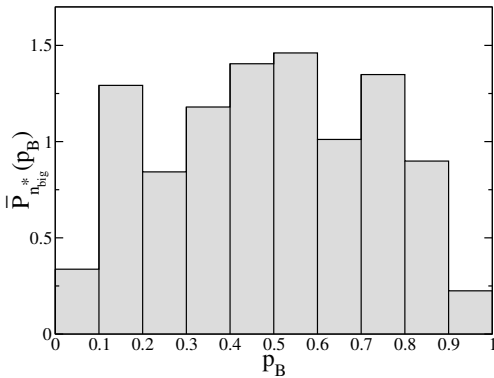


Figure 7.14: Path committor probability $\bar{P}_{n_{\text{big}}}^*(p_B)$ restricted to the transition ensemble. The curve is computed using points for which $n_{\text{big}} \in [240, 246]$. Even if transitions only are considered the distribution is substantially flat, indicating n_{big} is not a good reaction coordinate.

configurations with increasing p_B have on average an increasing n_{big} , but there is a large scattering in the data points, the dispersion being of the order of ± 100 .

In particular, the transition states ⁶ at $p_B = 0.5$ include cluster sizes with a dis-

⁶We consider here states with $p_B = 0.5$ timeslices for which $p_B \in [0.4, 0.6]$. So actually they are states with $p_B = 0.5 \pm 0.1$, because 0.1 was the error chosen for committor computation. Unless

tribution peaked around the free-energy maximum $n_{\text{big}}^* = 243$, but ranging in the interval $[151, 415]$ (see Fig. 7.6). The distribution is displayed in Fig. 7.13, together with the distributions of n_{big} at $p_B = 0.2$ and $p_B = 0.8$. These distributions overlap with each other, and the critical cluster size $n_{\text{big}}^* = 243$ falls in between. Indeed, the critical size $n_{\text{big}}^* = 243$ comprises states with all the committor values, from 0 to 1, as shown by the dashed line in Fig. 7.12. Using the points around this line we performed the Geissler test restricted to the transition path ensemble (see sec. 2.5.2). The resulting committor distribution $\bar{P}_{n_{\text{big}}^*}(p_B)$ (see Eq. (2.62)) is shown in Fig. 7.14. A necessary condition for n_{big} to be a good reaction coordinate is that this distribution is peaked around $p_B = 0.5$, but this is clearly not the case. Indeed if we perform the complete Geissler test, the committor distribution $P_{n_{\text{big}}^*}(p_B)$, computed for all phase points with $n_{\text{big}} = n_{\text{big}}^*$, might even display a worse profile than the one of $\bar{P}_{n_{\text{big}}^*}(p_B)$ (see Fig. 2.11c). The results of Figs. 7.12, 7.13, and above all Fig. 7.14 demonstrate that even if n_{big} tends to follow the progress of the transition, as given by p_B , it cannot describe the progress completely, and therefore n_{big} is *not* a (sufficiently) good reaction coordinate.

Having asserted that n_{big} is not sufficient, we searched for a better description of the transition. Similar to Fig. 7.12, we analyzed the committor p_B as function of all the observables defined in sec. 7.4.1. Structural observables as Q_4, Q_6 or n_{surf} also react to the advance of the transition, but in the best cases we were only able to reproduce a diffuse behavior similar to that of Fig. 7.12. To identify a correlation between two observables, we regrouped slices in sets with common committor in an interval of ± 0.1 and plotted the corresponding two chosen observables for all slices in this group in a two-dimensional plot. In these scatter plots we looked for the observable combinations that displayed a most significant correlation and produced the lowest overlap between groups with different committor value. We tried around 60 combinations, using the best choices from the one-dimensional correlation analysis. Often the graphs show complete overlap between sets, or do not improve the onedimensional findings. Sometimes they generate trivial correlations, as in the case of the correlation between n_{big} and the global Q_6 , which is just linear for any p_B (it can be deduced also from Fig. 7.8). We did find nevertheless a non-trivial improvement using the cluster size n_{big} and the cluster Q_6 . We plot in Fig. 7.15 the results of the analysis. The scattering is still rather diffuse, and there is overlap between different p_B sets, but this overlap is reduced with respect to the overlap of the onedimensional projections along the axes, because of the shape of the iso- p_B regions. In particular, the transition state ensemble, even if not a clear dividing surface, is at an angle with respect to the horizontal axis, allowing better distinction from the other regions. We dedicate the next section to the analysis of this transition region.

7.4.4 Transition state ensemble

We decided to study the *critical nuclei* at $p_B = 0.5$ analyzing the structural composition of the system in spherical shells around the center of mass of the cluster. Similar to ten Wolde [131], we regrouped the particles of all critical timeslices into spherical bins of radius r and then computed for each bin $f_{\text{liq}}(r)$, $f_{\text{bcc}}(r)$ and $f_{\text{fcc}}(r)$. At the

specified, same reasoning applies when we mention other values of the committor.

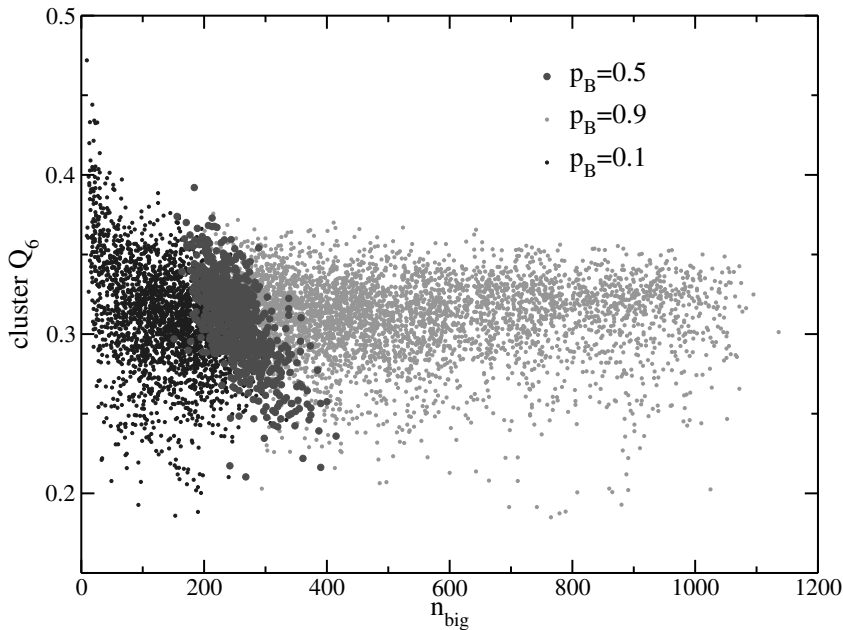


Figure 7.15: Two dimensional correlation plot of the cluster Q_6 versus the cluster size n_{big} for different committor values.

same time we computed the average cluster radial density $\rho(r)$. We remark that in this analysis we used not only the cluster particles, but all the system particles in the shells. The contribution of non-cluster particles becomes important as the surface of the cluster is approached and ensures that the computed properties reach the corresponding ones for the bulk liquid. This radial analysis is sensible if clusters have more or less the same shape, i.e. they are spheres of the same size. Since the $p_B = 0.5$ region includes clusters of different sizes, ranging from 151 to 415, we separated the set into three subsets, one before the critical size $n_{\text{big}}^* = 243$, one around, one after. We implicitly assumed the clusters are spherical, which turned out to be not entirely true for the larger critical subset (see later, Fig. 7.19). However, the position of the center of mass is hardly affected by the asymmetric shape, and given the larger size, the first spherical shells around the center of mass are still part of the cluster, validating the findings for the inner structure of the nuclei that we are going to present.

In Fig. 7.16 we plot the radial profiles. Clearly, as r increases, all the values approach the bulk liquid ones. They do it smoothly, indicating the surface is quite diffuse. More importantly, inspecting the profiles for low r , we can distinguish two different kinds of clusters. The first is represented by the top graphs in Fig. 7.16, referring to the small, but still critical clusters, and show a dense object with a large fcc component in the core. The density is approaching that of bulk fcc solid. The bcc component is stronger on the surface. With increasing critical cluster size (from top graphs to bottom graphs), the core fcc part is substituted by an increasing bcc

component and the density decreases. In the bottom graph, referring to the largest critical nuclei, we find a second kind of critical clusters. The density has a plateau at a value of 1.04, distinctly lower than $\rho_{fcc} = 1.0665$ and the cluster has a relatively large bcc component. Hence, the first, compact fcc clusters, corresponding to the top and middle graphs of Fig. 7.16, are in accordance with the results of ten Wolde [131], even though in that case the density displayed a flatter plateau at the center of the cluster. They can then be considered close to the equilibrium Umbrella Sampling results, with some kinetic distortion. The second, less dense clusters of the bottom panels, consisting of an fcc-bcc mixture are instead new, and can be considered another kind of transition state that is not occurring in equilibrium sampling, but is generated by kinetic effects.

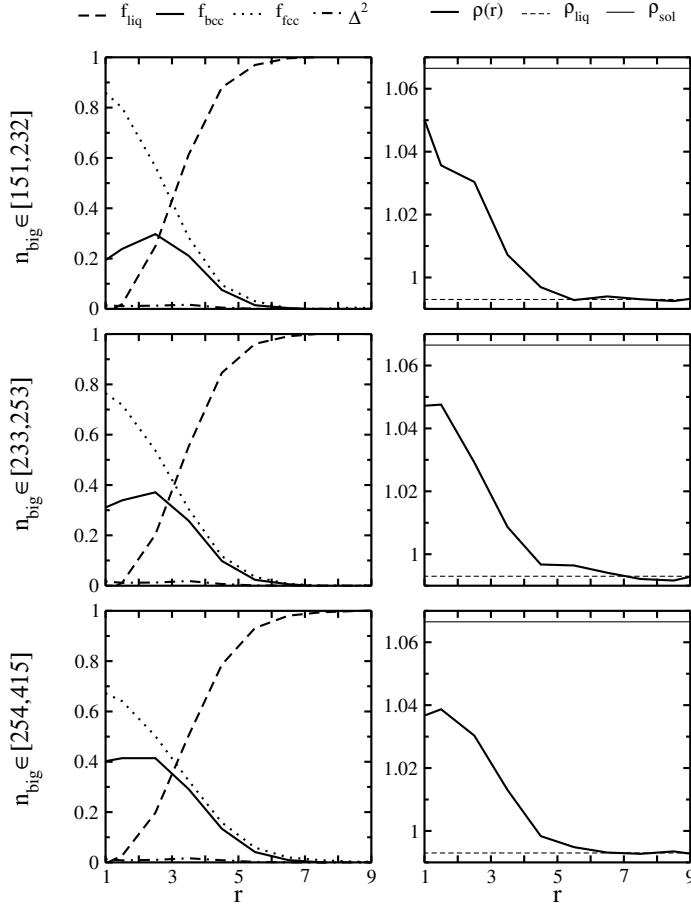


Figure 7.16: Composition of the system around the center of mass of the critical clusters at $p_B = 0.5$. All these clusters belong to the Transition State Ensemble, and are therefore critical nuclei. The top two rows correspond to nuclei close to the free-energy saddle point, while the bottom row does not (see Fig. 7.19).

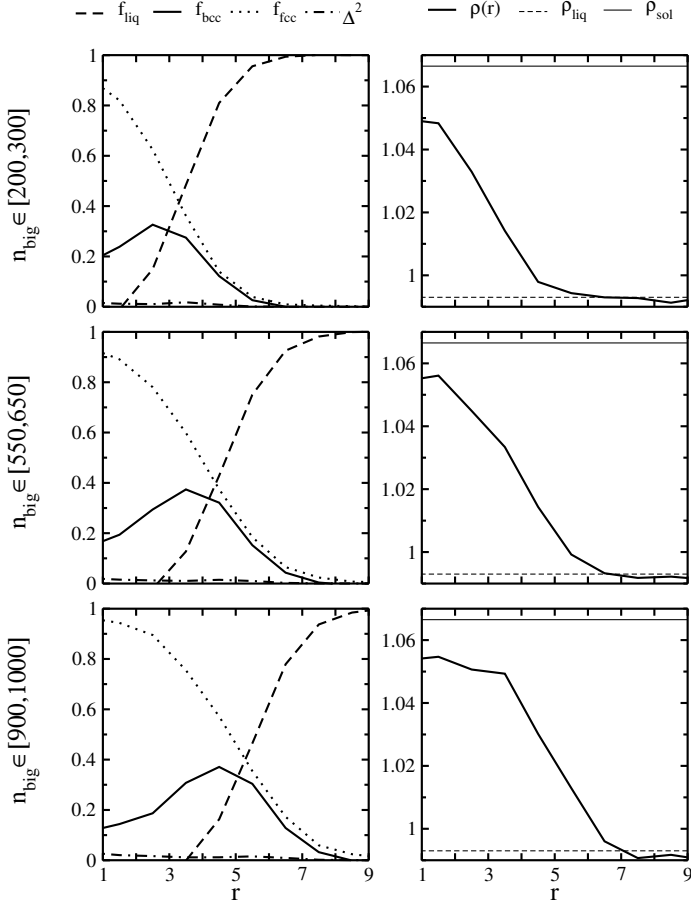


Figure 7.17: Composition of the system around the center of mass of the clusters with $p_B = 0.95 \pm 0.05$.

In order to check the progression of these critical clusters towards solidification, we applied the same radial analysis to post-critical configurations with committors in the range $p_B \in [0.9, 1.0]$. As can be seen from Fig. 7.12 they include clusters ranging in size from 200 up to 1000. The results are shown in Fig. 7.17. Interestingly, now all three size-ranges have the same behavior. They just show a scaling with the growing cluster radius. The behavior is basically that of the first kind of critical clusters: an fcc core wetted by a bcc surface. It seems then that the second kinetic transition state has equilibrated into this structure. We rearranged the information of Fig. 7.17 into Fig. 7.18 to compare the different sizes. As found by ten Wolde [131], the bcc surface stays of constant width while the fcc core grows. The density inside the cluster plateaus to a value slightly lower than that of bulk fcc, which was also found by ten Wolde ⁷.

⁷ We remark that for a proper comparison, the density $\rho_{sol} = 1.0665$ plotted in Fig. 7.16, 7.17, and 7.18 is not at the same NPH conditions ($P = 5.68, H = 1.412$), but at the same NPT conditions ($P = 5.68, T = 0.83$, 25% undercooling). Because of the constraint of fixed total enthalpy, a complete

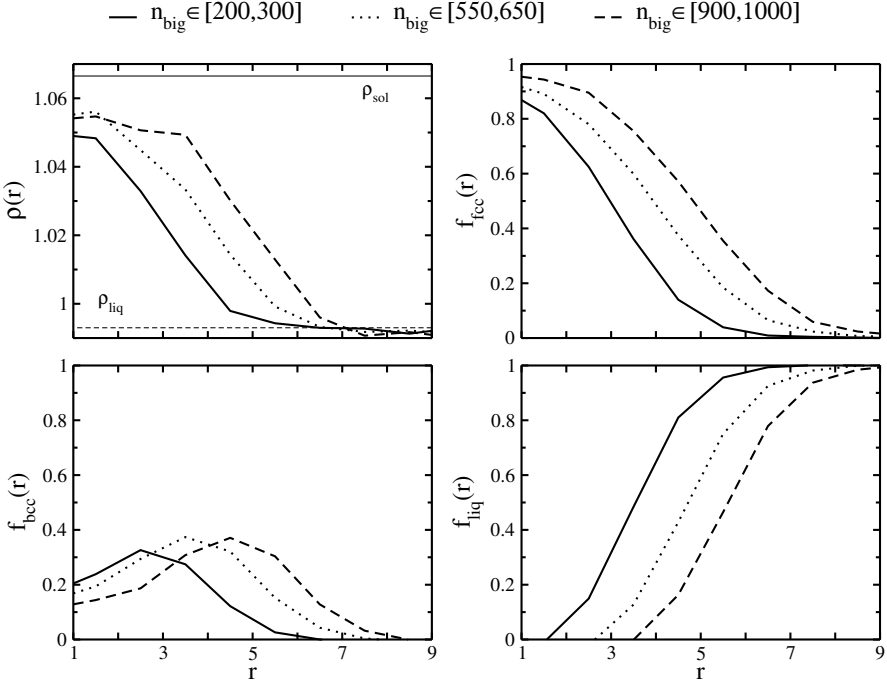


Figure 7.18: Composition of the system around the center of mass of the clusters with $p_B = 0.95 \pm 0.05$. The information of Fig. 7.17 is rearranged to compare different clusters with different sizes. A bcc surface shell of constant width surrounding a growing fcc core can be recognized.

Summarizing, using path sampling we were able to shed more light on the nucleation process than the analysis of ten Wolde [131] based on equilibrium free energy calculations with n_{big} as reaction coordinate. In sec. 7.4.3, we also found that a better (though not completely satisfactory) description of the nucleation event can be made using n_{big} in combination with the bond-order parameter Q_6 of the cluster (see Fig. 7.15). In order to check if this description could be found also with free energy calculations, we calculated the two dimensional free energy $\beta F(n_{\text{big}}, Q_6) = -\ln P(n_{\text{big}}, Q_6)$. The results are shown in Fig. 7.19, together with the results of the committor analysis as from Fig. 7.15. Interestingly, the transition state region is not perpendicular to the minimum free energy path in the (n_{big}, Q_6) plane. Even though the transition states at $p_B = 0.5$ are concentrated at the saddle point $n_{\text{big}} = 243, Q_6 = 0.3$, the transition state ensemble exhibits configurations with large Q_6 and small n_{big} and viceversa (whose snapshots are also in figure). Their existence is due to dynamical contributions, and is a new finding of the TIS path sampling.

fcc solid would have a much higher temperature $T = 1.137$ corresponding to a density $\rho = 1.020$ (see Fig. 7.8).

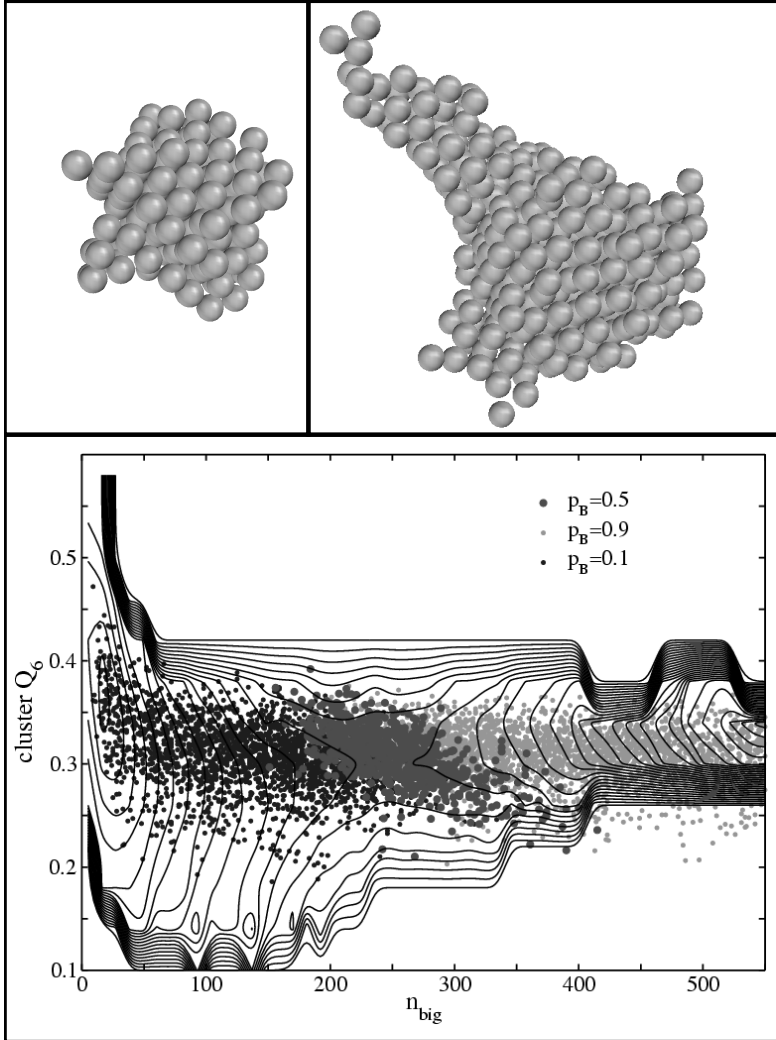


Figure 7.19: Contour plot of the two dimensional free energy as function of the size n_{big} and the bond-order parameter Q_6 of the biggest cluster. The contour lines are separated by $1k_B T$. A minimum is present at about $(n_{\text{big}}, Q_6) = (16, 0.37)$, and a saddle point at $(243, 0.3)$. Transition states at $p_B = 0.5$ are also shown together with pre-critical ($p_B = 0.1$) and post-critical ($p_B = 0.9$) configurations. We also show the two kinds of transition states we found. In the top left panel compact dense fcc objects, corresponding to small n_{big} and large Q_6 . In the top right panel we show a critical cluster with opposite structure, large n_{big} and small Q_6 , and a more mixed structure of fcc and bcc particles.

7.5 Summary

In this chapter we have investigated the crystal nucleation in a Lennard-Jones system from an undercooled liquid phase at $P = 5.68, T = 0.83$, corresponding to 25% undercooling. Under these conditions the nucleation is a rare event and we therefore applied the Interface Sampling methods described in the rest of the thesis. We calculated the rate constant using both TIS and PPTIS. The methods agree with each other, showing the system is diffusive and the memory loss assumption is satisfied. We also studied the mechanism of crystallization by generating about a hundred transition trajectories, and applying the committor analysis.

Using the committor analysis we were able to prove that the size of the biggest cluster n_{big} is not a sufficiently good reaction coordinate, as it does not clearly distinguish configurations with different values of the committor p_B . In particular, the transition states are a mixture of clusters with sizes ranging from 150 up to 420. We could improve the identification of a reaction coordinate combining n_{big} with a structural order parameter, the bond-order parameter Q_6 of the cluster.

Our results on the nucleation mechanism are mostly in accordance with those of ten Wolde (see sec. 7.1.1). We do not find a sharp transition from initial bcc- to fcc-dominated nuclei, but we do confirm the picture of a nucleus that is spherical on average, and with a growing fcc core surrounded by a bcc surface shell. However, we also discovered, by the analysis of the transition state ensemble, a second kind of critical cluster, less spherical, and with a more bcc-like inner structure. This kind of critical cluster equilibrates to the first kind of cluster structure, i.e. with an fcc core, as the reaction proceeds. The newly found critical structure cannot be explained with free energy calculations, even taking into account n_{big} and Q_6 . It is a kinetic effect that could only be discovered using path sampling.

Conclusion

*What you will see, if you leave the
Mirror free to work, I cannot tell.
For it shows things that were, and
things that are, and things that yet
may be.*

Galadriel, Lady of Lórien

Rare events are transitions between stable states separated by a high free energy barrier. Throughout this thesis, we have tried to outline how such events can be investigated. The assumption of traditional methods as the TST-BC procedure, is that the free energy profile contains all the relevant information: from the free energy the rate can be computed, and the mechanism deduced. We have proved that this is not always true. First, the rate computation must be completed with the calculation of the transmission coefficient, which is far from trivial. Second, the choice of the reaction coordinate, along which the free energy is computed, inevitably biases the way we look at the mechanism. Moreover, the dynamical information neglected by a static free energy calculation can be important in many cases.

For these reasons Transition Path Sampling was created. TPS extracts the information from the transitions themselves, i.e. from the interesting parts of all the possible trajectories of the system. The reaction coordinate is replaced by an order parameter, which is an algorithmic parameter, and does not, in principle, influence the sampling of the trajectories. Based on TPS, we have developed the Transition Interface Sampling methods. TIS methods improve the sampling efficiency of the transition path ensemble and thus speed up the rate computation. Although very similar in practice to TPS, TIS is not just a technical refinement, it has a different spirit, because of the idea of the effective positive flux. The rate is computed measuring fluxes through interfaces dividing the states, avoiding cancellations from recrossings and false transitions, as was in TST-BC and TPS. In the case of diffusive systems, Partial Path TIS efficiently exploits the loss of memory along the diffusive trajectories to further improve the efficiency. We have also shown that the free energy can also be computed, but now as a side product, and not a necessary step to the rate computation.

The analysis of the mechanism and the identification of the reaction progress is performed by computing committor distributions. The committor $p_B(r)$ is the com-

mittment probability of configuration r to the final stable state B . When $p_B = 0$ the reaction is at the beginning, in state A . When $p_B = 1$ the reaction is at the end, in state B . The configurations for which $p_B = 0.5$ define the transition states. For this reason, p_B is the perfect reaction coordinate. Naturally, such an important indicator is not cheap to obtain. For every configuration of all the sampled transition paths, a series of trajectories must be initiated with random momenta and integrated until reaching A or B . The whole process requires considerable time and computer power. However, when committors have been finally computed, precious information is available. First, by using the Geissler test, a chosen reaction coordinate can be identified as good or bad. Second, the regrouping of configurations as a function of their p_B shows the advance of the reaction. The mechanism can then be elucidated using the chemical, physical, or biological intuition, depending on the context. All additional information, including free energy profiles, can be used to help.

We have shown an application of the above methodology for the case of the crystal nucleation of a Lennard-Jones system. The size of the growing solid nucleus from an undercooled liquid was usually thought to be the proper reaction coordinate. Solid clusters smaller than the critical size, at the free energy maximum, shrink back to the liquid state. Solid clusters larger than the critical size grow to full solidification. However, when we computed the committor distribution for clusters of critical size, we found an almost flat uniform curve. Indeed the transition states at $p_B = 0.5$ include clusters of different sizes dispersed around the critical size. Some are compact spherical objects with a fcccore, some have more oblong shapes with a mixed bccand fccstructure. As the reaction proceeds (p_B increases), the dispersion equilibrates towards spherical clusters with a growing fcccore wetted by a bccsurface of constant width. The behavior at high p_B is in agreement with previous TST-BC studies, but the identification of the transition states is a new dynamical finding.

In summary, the TIS interface fluxes for the rate computation and the TIS path sampling, in combination with the committor analysis for the mechanism comprise a complete method for the study of rare events. The whole procedure is expensive, and in this sense it is a brute force approach, as TPS itself. However, it is, in principle, very powerful. Using TIS path sampling in place of the TST-BC procedure, not only the correct rate can be efficiently computed, but also the final mechanism properly elucidated.

A

Rates as first passage times

In this section we want to show that real systems can be approximated with a random telegraph if the jump probabilities are defined as the inverse mean first passage times. Otherwise stated, we want to prove that the sensible definition of a rate constant is Eq. (1.15). Our treatment follows refs [6, 10]. It becomes more complicated, and we will try not to go too deep into mathematical details. And the end of the section we add the study of a simple onedimensional model to clarify the general results.

Consider the Langevin dynamics (1.22) in the high friction limit $\gamma \gg 1$, where it reduces to a first order differential equation involving only the coordinates r . The corresponding Itô stochastic differential equation is

$$\gamma dr = -\nabla_r \mathcal{U} dt + \sqrt{2\gamma\beta^{-1}} d\eta \quad (\text{A.1})$$

where $d\eta$ is a Wiener process [6]. The analysis of the Langevin dynamics could be done in the full phase space (r, v) , but while it does not change the essence of the results, it complicates the treatment considerably. Measuring time in units of γ we can take $\gamma = 1$. In this appendix, since functions of more than one spatial variable appear, we always specify the variable on which the differential operators act, as in ∇_r .

To the stochastic process defined by (A.1) is associated the probability (see also (1.3))

$$\begin{aligned} p(r, t|r', 0) &= \text{Probability of } r(t) \in [r, r + dr] \text{ at time } t \\ &\text{provided that } r(t = 0) = r' \end{aligned} \quad (\text{A.2})$$

This probability satisfies the forward and backward Fokker-Planck equations, respectively

$$\partial_t p = \nabla_r \cdot (\nabla_r \mathcal{U}(r)p + \beta^{-1} \nabla_r p) \quad (\text{A.3a})$$

$$\partial_t p = -\nabla_{r'} \mathcal{U}(r') \cdot \nabla_{r'} p + \beta^{-1} \Delta_{r'} p \quad (\text{A.3b})$$

Defining the current $J = -\nabla_r \mathcal{U}(r)p - \beta^{-1} \nabla_r p$ the first takes the form of a continuity equation.

We will consider (A.3) for two classes of boundary conditions, namely

1. forward equation and reflecting boundaries at infinity: $J = 0$. The time-independent stationary solution $p_{st}(r)$ is then given by the Gibbs distribution (1.16)

$$p_{st}(r) = Z_r^{-1} e^{-\beta \mathcal{U}(r)} = \rho(r) \quad (\text{A.4})$$

2. backward equation and absorbing boundary on a surface S . The probability of being on the boundary is zero, i.e. $p(r, t) = 0 \forall r \in S$. In this case $p_{st}(r) \equiv 0$. This boundary condition is useful to solve first passage time problems, and we will face them in sec. A.2.

A.1 Eigenvalue problem for the Fokker-Planck operator

Basically, all the properties of p , Eq. (A.2) can be deduced from the eigenvalue problem associated to (A.3). Similar to the treatment of the Schrödinger equation in quantum mechanics, we seek solutions for (A.3a) and (A.3b), respectively, in the form

$$\psi(r, t) = \psi_\lambda(r) e^{-\lambda t} \quad (\text{A.5})$$

$$\varphi(r, t) = \varphi_\lambda(r) e^{-\lambda t} \quad (\text{A.6})$$

Then to the functions $\psi_\lambda(r)$, $\varphi_\lambda(r)$ are associated the right and left time-independent eigenfunction equations

$$L_r \psi_\lambda \equiv \nabla_r \cdot (\nabla_r \mathcal{U} \psi_\lambda + \beta^{-1} \nabla_r \psi_\lambda) = -\lambda \psi_\lambda \quad (\text{A.7})$$

$$L_r^* \varphi_\lambda \equiv -\nabla_r \mathcal{U} \cdot \nabla_r \varphi_\lambda + \beta^{-1} \Delta \varphi_\lambda = -\lambda \varphi_\lambda \quad (\text{A.8})$$

where we have introduced the forward and backward Fokker-Planck operators L_r , L_r^* . It can be proven [6] that

1. The eigenfunctions form a biorthogonal system

$$\int dr \psi_\lambda(r) \varphi_{\lambda'}(r) = \delta_{\lambda\lambda'} \quad (\text{A.9})$$

2. Since the Gibbs distribution is the stationary time-independent solution, see Eq. (A.4), the probability (A.2) satisfies the detailed balance [6] equation

$$p(r', t|r, 0) p_{st}(r) = p(r, t|r', 0) p_{st}(r') \quad (\text{A.10})$$

connecting the transition $r \rightarrow r'$ and the reversed one $r' \rightarrow r$. As a consequence for any solution $\psi(r, t)$ of the forward equation (A.3a),

$$\varphi(r, t) = \psi(r, t) / p_{st}(r) \quad (\text{A.11})$$

is a solution of the backward equation (A.3b). Which also implies $\varphi_\lambda(r) = \psi_\lambda(r) / p_{st}(r)$.

3. In a suitably defined Hilbert space L_r is self-adjoint and negative semi-definite, which implies the eigenvalues are real and

$$\lambda \geq 0 \quad (\text{A.12})$$

4. Assuming completeness, the solution of (A.3) can then be written as

$$p(r, t|r', 0) = \sum_{\lambda} \psi_{\lambda}(r) \varphi_{\lambda}(r') e^{-\lambda t} \quad (\text{A.13})$$

satisfying the initial condition

$$\delta(r - r') = p(r, 0|r', 0) = \sum_{\lambda} \psi_{\lambda}(r) \varphi_{\lambda}(r') \quad (\text{A.14})$$

5. When (A.3) is considered with reflecting boundary conditions, there always exist a null-eigenvalue $\lambda_0 = 0$, so that the spectrum is

$$\lambda_0 = 0 < \lambda_1 \leq \lambda_2 \leq \dots \quad (\text{A.15})$$

Then (A.13) admits a stationary solution $p_{st}(r) \neq 0$ for $t \rightarrow \infty$. Using then (A.13) and (A.11) one can derive that $\varphi_0(r)$ is a constant, and can be chosen equal to 1. Then from (A.9) follows $\psi_0(r) = p_{st}(r)$.

6. When (A.3) is considered with absorbing boundary conditions, see next section, the spectrum (A.15) misses the null-eigenvalue λ_0 and accordingly the stationary solution is $p_{st}(r) = 0$

Let us now assume that there is a *spectral gap*

$$\lambda_1/\lambda_2 \ll 1 \quad (\text{A.16})$$

i.e. the first non-zero eigenvalue is much smaller than the rest. There exists then a range of times such that $\lambda_1 t \simeq 0$ and $\lambda_i t \gg 0 \forall i > 1$. This implies that expansion (A.13) can be approximately truncated at λ_1 . And $p(r, t|r', 0)$ can be expressed in terms of the first eigenfunctions only. We distinguish two cases. In the case of reflecting boundary conditions, we said already that $\psi_0(r) \varphi_0(r') = p_{st}(r)$. The eigenvalue λ_1 being small, it has a behavior similar to λ_0 . Indeed $\psi_1(r)$ is the restriction of $p_{st}(r)$ to the sets A, B with opposite sign, and $\varphi_1(r)$ is the characteristic function of the set. In the case of absorbing boundary conditions on the surface ∂B , the functions are defined on $B^c = \mathbb{R}^n \setminus B$ and with the same reasoning $\varphi_1(r)$ is a constant, and can be taken equal to 1, and $\psi_1(r)$ is the restriction of $p_{st}(r)$ to A . Opposite situation holds in the case of absorbing surface ∂A . This behavior is summarized in Fig. A.1.

Because of the spectral gap assumption we can prove that each λ_i^{-1} is associated to the time to span (in ergodic sense) the region associated with the support of the corresponding eigenfunctions. See for example [159] or, for a list of references to more detailed explanations at various levels, the Ref. 3 of [70]. Then λ_1^{-1} is associated to the lifetime τ_{stable} of stable states A and B (see next section) while λ_2^{-1} is associated to the transition time τ_{trans} . As a consequence relation (A.16) is equivalent to relation (1.1), the separation of timescales.

A.2 Mean first passage time

Suppose to start in A and to compute the time needed to hit B . One has to consider the situation of Fig. A.1 (b), absorbing boundary condition on ∂B . Since the evolution

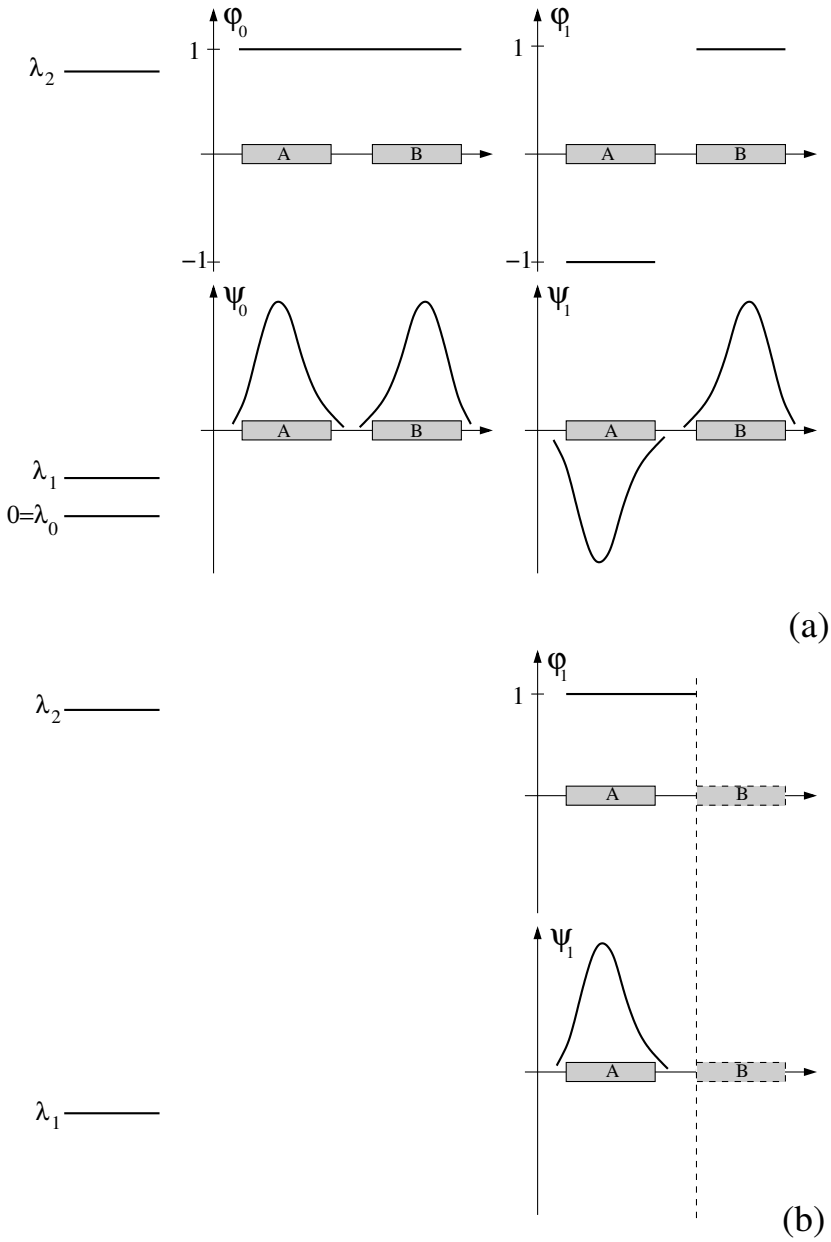


Figure A.1: Eigenfunctions for the Fokker-Planck operator in a system with spectral gap $\lambda_1 \ll \lambda_2$. (a) reflecting boundaries (b) absorbing boundaries on $\mathbb{R}^n \setminus B$. Note that there is no null-eigenvalue and associated eigenfunctions φ_0, ψ_0 because of the absorbing boundary conditions, point 2 on page 153.

stops when it reaches the surface (gets absorbed), if the system is still in B^c at time t it means it has never left it.

Let $t^{fp}(r)$ be the first passage time, i.e.

$$t^{fp}(r) \equiv \min_t \{t : r(t) \in B^c, r(0) = r\} \quad (\text{A.17})$$

then for the probability of being still in B at time t holds

$$\begin{aligned} G(r, t) &\equiv \int_{B^c} p(r', t | r, 0) dr' \\ &= \text{Prob}\{t^{fp}(r) \geq t\} \\ &= \int_t^\infty \rho(t^{fp}(r)) d\{t^{fp}(r)\} \end{aligned} \quad (\text{A.18})$$

where $\rho(t^{fp}(r))$ is the probability distribution¹ for $t^{fp}(r)$. Integrating (A.3b) on r , and renaming $r' \rightarrow r$ for simplicity, we see that $G(r, t)$ also satisfies a backward Fokker-Planck equation

$$\partial_t G(r, t) = -\nabla_r \mathcal{U}(r) \cdot \nabla_r G(r, t) + \beta^{-1} \Delta_r G(r, t) \quad (\text{A.19})$$

with initial conditions

$$G(r, 0) = 1 \quad \forall r \in B^c \text{ and } 0 \text{ elsewhere} \quad (\text{A.20})$$

$$G(r, t) = 0 \quad \forall r \in \partial B \quad (\text{A.21})$$

Because of the definition of G the mean first passage time is

$$t^{mfp}(r) = \int_0^\infty t^{fp}(r) \rho(t^{fp}(r)) d\{t^{fp}(r)\} = - \int_0^\infty t \partial_t G(r, t) dt = \int_0^\infty G(r, t) dt \quad (\text{A.22})$$

where we used the fact that $\partial_t G(r, t) = -\rho(t)$ follows from (A.18), and we integrated by parts. We can derive an equation for $t^{mfp}(r)$ integrating (A.19) over $(0, \infty)$. Noting that

$$\int_0^\infty \partial_t G(r, t) dt = G(r, \infty) - G(r, 0) = -1 \quad (\text{A.23})$$

we get

$$-\nabla_r \mathcal{U}(r) \cdot \nabla_r t^{mfp}(r) + \beta^{-1} \Delta_r t^{mfp}(r) = -1 \quad (\text{A.24})$$

with the initial condition $t^{mfp}(r) = 0$ if $r \in \partial B$.

Let us now apply what we learnt from the eigenvalue problem. Assuming the left eigenfunctions are a complete set, we expand $t^{mfp}(r)$ as

$$t^{mfp}(r) = \sum_\lambda t_\lambda \varphi_\lambda(r) \quad (\text{A.25})$$

¹ An equation for $\rho(t^{fp}(r))$ can also be derived, but the solution is not easy, even for simple cases [157, 158]. The shape of such distribution however is approximately general and of the form pictured in Fig. 7.9.

inserting into (A.24) we get

$$-1 = \sum_{\lambda} t_{\lambda} L_r^* \varphi_{\lambda}(r) = - \sum_{\lambda} t_{\lambda} \lambda \varphi_{\lambda}(r) \quad (\text{A.26})$$

and making use of (A.14)

$$\begin{aligned} \sum_{\lambda} t_{\lambda} \lambda \varphi_{\lambda}(r) &= 1 \\ &= \int_{B^c} dr' \delta(r' - r) \\ &= \sum_{\lambda} \int_{B^c} dr' \psi_{\lambda}(r') \varphi_{\lambda}(r) \end{aligned} \quad (\text{A.27})$$

Equating each term we find t_{λ} . Inserting back into (A.25) we get

$$t^{mfp}(r) = \sum_{\lambda} \frac{\varphi_{\lambda}(r)}{\lambda} \int_{B^c} dr' \psi_{\lambda}(r') \quad (\text{A.28})$$

Finally, let us consider the separation of timescales. Suppose (A.16) holds. Then we can approximate (A.28) with its first term

$$t^{mfp}(r) \simeq \frac{1}{\lambda_1} \int_{B^c} dr' \psi_{\lambda}(r') = \frac{1}{\lambda_1} \quad (\text{A.29})$$

where we used the property of the eigenfunctions of λ_1 , namely being the characteristic function of B^c , see Fig. A.1b. Since the buffer region has a negligible probability this result is valid for $r \in A$, i.e. $\lambda_1^{-1} = t^{mfp}(A)$. Similar reasoning applies for the problem with absorbing boundary conditions at ∂A . When we consider then the full system, with reflecting boundaries at infinity, because of the linearity and the negligible weight of the buffer region, we get

$$\lambda_1 \simeq t^{mfp}(A)^{-1} + t^{mfp}(B)^{-1} \equiv k_{AB} + k_{BA} \quad (\text{A.30})$$

where we apply definition (1.15).

The connection to a random telegraph model is achieved defining

$$n_A(t|r) = \int_A dr' p(r', t|r, 0) \quad n_B(t|r) = \int_B dr' p(r', t|r, 0) \quad (\text{A.31})$$

Using the expansion (A.13) truncated at λ_1

$$n_A(t|r) = \langle h_A \rangle + e^{-\lambda_1 t} \int_A dr' \psi_1(r') \varphi_1(r) \quad (\text{A.32})$$

where we used again $\psi_0(r') \varphi_0(r) = p_{st}(r)$. We can then rewrite

$$n_A(t|r) = \langle h_A \rangle + e^{-\lambda_1 t} [n_A(0|r) - \langle h_A \rangle] \quad (\text{A.33a})$$

$$n_B(t|r) = \langle h_B \rangle + e^{-\lambda_1 t} [n_B(0|r) - \langle h_B \rangle] \quad (\text{A.33b})$$

which correspond to the (1.7) and (1.8) as soon as we identify

$$\lambda_1 = W_{AB} + W_{BA} \quad (\text{A.34})$$

$$\langle h_A \rangle = \frac{W_{BA}}{\lambda_1} \quad \langle h_B \rangle = \frac{W_{AB}}{\lambda_1} \quad (\text{A.35})$$

Otherwise stated, what we have proven in this section is

$$k_{AB} \equiv t^{mf p}(A)^{-1} = W_{AB} \quad k_{BA} \equiv t^{mf p}(B)^{-1} = W_{BA} \quad (\text{A.36})$$

i.e. a system with separation of timescales (1.1), or equivalently a spectral gap (A.16), can be approximated with a random telegraph if we take as jump probabilities the inverse mean first passage times.

This somewhat qualitative connection we presented can be refined into what is called Kramers' method, originally derived in [160] and rediscovered and reformulated several times [6]. For a rigorous modern mathematical derivation see [161].

A.3 Correlation function for reactive flux

The Bennett-Chandler (BC) procedure or reactive flux formalism of sec. 1.5 makes use of the correlation function (1.38), which we rewrite here

$$C(t) \equiv \frac{\langle h_A(0)h_B(t) \rangle}{\langle h_A \rangle} \quad (\text{A.37})$$

Using the results of the previous sections we can prove relation (1.39).

Consider the numerator:

$$\langle h_A(0)h_B(t) \rangle = \int dr dr' h_A(r') h_B(r) p(r, t; r', 0) \quad (\text{A.38})$$

where $p(r, t; r', 0)$ is the two-point probability of being in r at time t and at r' at time 0. It is related to the two-point conditional probability by $p(r, t; r', 0) = p(r, t | r', 0) p_{st}(r')$. Hence, we can rewrite

$$\begin{aligned} \langle h_A(0)h_B(t) \rangle &= \int dr dr' h_A(r') h_B(r) p(r, t | r', 0) p_{st}(r') \\ &\simeq \int dr dr' h_A(r') h_B(r) \\ &\quad [p_{st}(r') p_{st}(r) + p_{st}(r') \psi_1(r) \varphi_1(r') e^{-\lambda_1 t}] \\ &= \int dr dr' h_A(r') h_B(r) \\ &\quad [p_{st}(r') p_{st}(r) - p_{st}(r') p_{st}(r) e^{-\lambda_1 t}] \\ &= \langle h_A \rangle \langle h_B \rangle (1 - e^{-\lambda_1 t}) \end{aligned} \quad (\text{A.39})$$

where we used in order, the expansion (A.13) truncated at λ_1 , and the properties of the eigenfunctions Fig. A.1. As a result, using (A.34)

$$C(t) \simeq \langle h_B \rangle \left(1 - e^{-(k_{AB} + k_{BA})t} \right) \quad (\text{A.40})$$

which is valid for $t \gg \lambda_2^{-1}$. It can be approximated for times $\lambda_2^{-1} \ll t \ll \lambda_1^{-1}$, i.e. $\tau_{trans} \ll t \ll \tau_{stable}$, see section A.1. The result is then the relation we were looking for

$$C(t) \simeq \langle h_B \rangle \lambda_1 t = k_{AB} t \quad (\text{A.41})$$

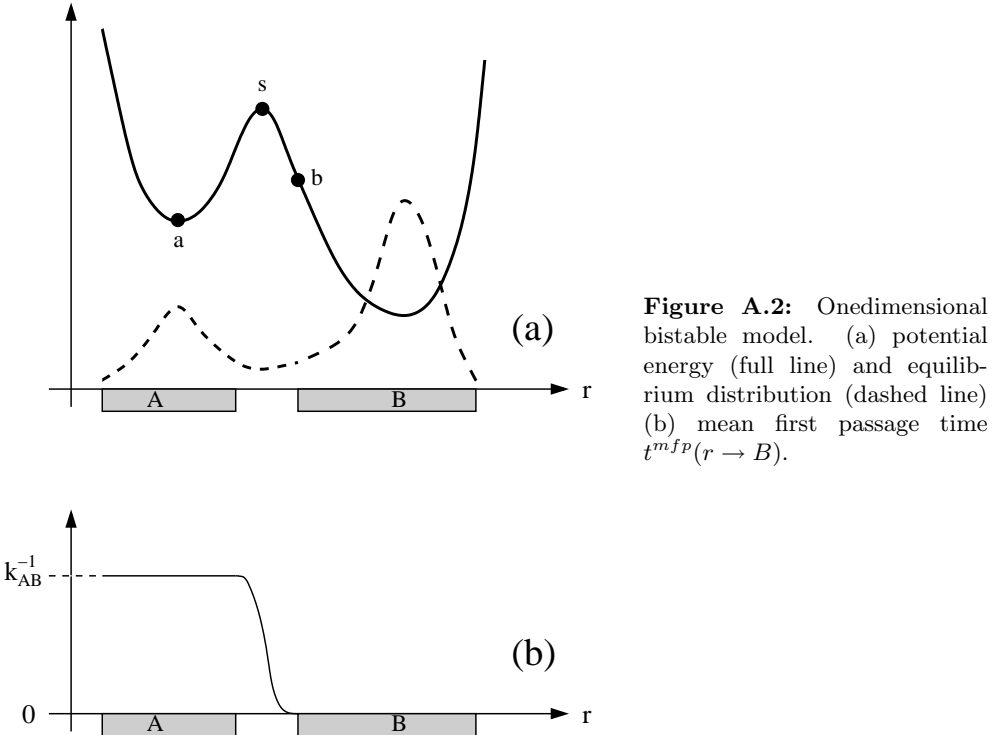
where we used (A.34), (A.35).

Originally this formula was derived using linear response on a random telegraph approximation which was assumed valid a priori as a reasonable macroscopic law [12]. Here the derivation is reversed. Thanks to the complicated but powerful spectral analysis of the Fokker-Planck operator, we have been able to derive the random telegraph approximation from basic microscopic principles, and Eq. (A.41) as a consequence.

A.4 A model

Let us analyze a bistable onedimensional model. We compute analytically the mean first passage time from one state to the other and we show that converges to a constant value, identifiable with the inverse rate constant.

Consider the potential $U(r)$ of Fig. A.2 a. The system admits the equilibrium distribution $p_{st}(r) = \exp[-\beta U(r)]$ also plotted in figure. We put an absorbing boundary at $r = b$, the border of state B. The equation (A.24) for the mean first passage times



becomes

$$-U'(r)\partial_r t^{mfp}(r) + \beta^{-1}\partial_r^2 t^{mfp}(r) = -1 \quad (\text{A.42})$$

with the initial condition $t^{mfp}(b) = 0$. Requiring an additional reflecting boundary at $-\infty$ the solution is [6]

$$t^{mfp}(r \rightarrow B) = \beta \int_r^b dy e^{\beta U(y)} \int_{-\infty}^y dz e^{-\beta U(z)} \quad (\text{A.43})$$

So far we have made no approximation. Now if the maximum at s is high and the temperature low, then around s , $\exp[\beta U(y)]$ will be sharply peaked while $\exp[-\beta U(z)]$ is very small. Therefore the inner integral is a very slowly varying function of y near $y = s$, i.e. almost constant for those value where the other exponent is significantly different from zero. Hence we can put $y = s$ in the inside integral and take it outside:

$$t^{mfp}(r \rightarrow B) \simeq \beta \int_{-\infty}^s dz e^{-\beta U(z)} \int_r^b dy e^{\beta U(y)} \quad (\text{A.44})$$

The first integral can then be approximated with a Gaussian integral at a over $(-\infty, \infty)$, while the second can be approximately related to an error function. The resulting behavior is shown in Fig A.2 b. For r in state A, in particular $r \simeq a$, there is a plateau, due to the stability of the state. Extending the integral again to $(-\infty, \infty)$ we get for the plateau value

$$k_{AB} = t^{mfp}(A \rightarrow B)^{-1} = \frac{\sqrt{U''(a)|U''(s)|}}{2\pi\gamma} e^{-\beta[U(s)-U(a)]} \quad (\text{A.45})$$

with the characteristic exponential behavior as function of barrier height, see also (1.36). We have also reintroduced the friction coefficient γ from (A.1), so that k_{AB} has the right dimension of an inverse time. Formula (A.45) is valid for $\gamma \gg 1$.

We apply then TST formula (1.35) to this simple system, with $\lambda(r) = r$ and $\lambda^* = s$. In one dimension the free energy coincides with the potential energy. In the same approximation of Gaussian integrals, we get

$$k_{AB}^{TST} = \frac{\sqrt{U''(a)}}{2\pi\sqrt{m}} e^{-\beta[U(s)-U(a)]} \quad (\text{A.46})$$

so that

$$\frac{k_{AB}}{k_{AB}^{TST}} = \frac{\sqrt{m|U''(s)|}}{\gamma} \quad (\text{A.47})$$

The meaning of the result is discussed in sec. 1.5.2.

B Alternative TST rate expression

We derive (1.68) from (1.33). Considering only the numerator:

$$\begin{aligned}
\langle \delta(\lambda(r) - \lambda^*) \dot{\lambda} \theta(\dot{\lambda}) \rangle &= \frac{1}{2} \langle \delta(\lambda(r) - \lambda^*) |\dot{\lambda}| \rangle \quad (\text{B.1}) \\
&= \frac{1}{2} \int_{\mathbb{R}^n \times \mathbb{R}^n} dr dv \rho(r, v) \delta(\lambda(r) - \lambda^*) |\nabla \lambda \cdot v| \\
&= \frac{1}{2} \int_{\mathbb{R}^n} dr \rho(r) \delta(\lambda(r) - \lambda^*) \\
&\quad Z_v^{-1} \int_{\mathbb{R}^n} dv |\nabla \lambda \cdot v| e^{-\frac{\beta}{2} v^T M v} \\
&= \frac{1}{2} \int_{\mathbb{R}^n} dr \rho(r) \delta(\lambda(r) - \lambda^*) \\
&\quad \frac{\int_{\mathbb{R}^n} dv |\nabla \lambda \cdot v| e^{-\frac{\beta}{2} v^T M v}}{\int_{\mathbb{R}^n} dv e^{-\frac{\beta}{2} v^T M v}}
\end{aligned}$$

Consider the inner fraction only. It can be easily evaluated if the exponent were rotationally invariant. We change then coordinates to $y = \sqrt{M}v$. The jacobian $|\det M|^{-1/2}$ cancels out in the ratio and we are left with

$$\frac{\int_{\mathbb{R}^n} dy |(M^{-1/2} \nabla \lambda) \cdot y| e^{-\frac{\beta}{2} y^T y}}{\int_{\mathbb{R}^n} dy e^{-\frac{\beta}{2} y^T y}} \quad (\text{B.2})$$

Now we can choose a reference frame such that the first coordinate y_1 is along the constant vector $(M^{-1/2} \nabla \lambda)$. The integral reduces then to

$$|(M^{-1/2} \nabla \lambda)| \frac{\int_{-\infty}^{+\infty} dy_1 |y_1| e^{-\frac{\beta}{2} y_1^2}}{\int_{-\infty}^{+\infty} dy_1 e^{-\frac{\beta}{2} y_1^2}} = 2 |(M^{-1/2} \nabla \lambda)| \sqrt{\frac{k_B T}{2\pi}} \quad (\text{B.3})$$

Inserting back into (B.1) we get

$$\langle \delta(\lambda(r) - \lambda^*) \dot{\lambda} \theta(\dot{\lambda}) \rangle = \sqrt{\frac{k_B T}{2\pi}} \int_{\mathbb{R}^n} dr \rho(r) \delta(\lambda(r) - \lambda^*) |(M^{-1/2} \nabla \lambda)| \quad (\text{B.4})$$

162 B. Alternative TST rate expression

We could introduce mass-weighted coordinates $r' = \sqrt{M}r$, but not to complicate things further let us assume all masses equal to m . Then $M^{-1/2}\nabla = m^{-1/2}\nabla$ and we have

$$\langle \delta(\lambda(r) - \lambda^*) \dot{\lambda} \theta(\dot{\lambda}) \rangle = \sqrt{\frac{k_B T}{2\pi m}} \int_{\mathbb{R}^n} dr \rho(r) \delta(\lambda(r) - \lambda^*) |\nabla \lambda| \quad (\text{B.5})$$

which is the definition of a surface integral on $\lambda(r) = \lambda^*$ with the surface element $d\sigma(r) = \delta(\lambda(r) - \lambda^*) |\nabla \lambda| dr$.

C

RRKM theory

We want to derive (2.55) from the general TST formula (1.33), in the microcanonical ensemble and using an harmonic approximation to the potential energy. The result has its origins in 1927, even before TST, which dates back to 1935, and is known as the RRKM theory, from the names of its creators, Rice, Ramsperger, Kassel, Marcus [162,163,164,165]. The derivation is a mathematical exercise that shows how to compute ensemble averages with constraints.

At the basis of the demonstration there is the following integral formula, the surface integration of a multidimensional sphere. Let $x \in \mathbb{R}^k$ and $C > 0$, $\alpha_i > 0, i = 1 \dots k$. Then

$$\begin{aligned} \int d^k x \delta \left[\frac{1}{2} \sum_{i=1}^k \alpha_i x_i^2 - C \right] &= 2 \int d^k x \delta \left[\sum_{i=1}^k \alpha_i x_i^2 - 2C \right] \\ &= \frac{2}{\prod_i \sqrt{\alpha_i}} \int d^k y \delta \left[\sum_{i=1}^k y_i^2 - R^2 \right] \\ &= \frac{2S_k}{\prod_i \sqrt{\alpha_i}} \int_0^\infty dy y^{k-1} \delta [y^2 - R^2] \\ &= \frac{2S_k}{\prod_i \sqrt{\alpha_i}} \int_0^\infty dy \frac{y^{k-1}}{2R} [\delta(y+R) + \delta(y-R)] \\ &= \frac{S_k}{\prod_{i=1}^k \sqrt{\alpha_i}} (2C)^{\frac{k-2}{2}} \end{aligned} \tag{C.1}$$

In the first line we used $\delta(ax) = \delta(x)/|a|$. In the second a change of variables $y_i = \sqrt{\alpha_i} x_i$, $R^2 = 2C$ to have a spherical integrand. Then we moved to polar coordinates, S_k is the k -dimensional surface area of unit-radius hypersphere. The final outcome follows from the properties of the delta function. Note that the result is proportional to the radius to the power $k-2$ and not $k-1$ as expected from geometrical intuition.

Given a system of N particles in dimension $d = 3$, let r, p be the $n = dN$ dimensional vectors of coordinates and momenta. The Hamiltonian and the microcanonical

distribution are written

$$\mathcal{H}(r, p) = \frac{p^T p}{2m} + \mathcal{U}(r) \quad (\text{C.2})$$

$$\rho(r, p) = \Omega^{-1} \delta[\mathcal{H}(r, p) - E] \delta(\mathbf{P}) \delta(\mathbf{L}) \quad (\text{C.3})$$

where Ω is the microcanonical partition function, we assume equal mass m for all particles, and \mathbf{P} , \mathbf{L} are the vectors of total linear and angular momentum.

Consider now the denominator of Eq. (1.33). The average $\langle \theta(\lambda^* - \lambda(r)) \rangle$ is the measure of stable state A , which is assumed to be the basin of a configurational energy minimum. We approximate the potential $\mathcal{U}(r)$ with a quadratic function at the minimum and write

$$\langle h_A \rangle = \Omega^{-1} \int d^n r d^n p \delta \left[\frac{p^T p}{2m} + \frac{r^T Q r}{2} + \mathcal{U}_A - E \right] \delta(\mathbf{P}) \delta(\mathbf{L}) \quad (\text{C.4})$$

Now we proceed to the integration of the constraints $\delta(\mathbf{P}) \delta(\mathbf{L})$. They are linear and represent fixed planes in the p space. Intersecting fixed planes with a sphere reduces the dimension of the sphere. Explicitly, the constraints are rewritten as

$$\delta(\mathbf{P}) = \prod_{\alpha=1}^d \delta(v_\alpha \cdot p) \quad (\text{C.5})$$

$$\delta(\mathbf{L}) = \prod_{\alpha=1}^d \delta(w_\alpha(r) \cdot p) \quad (\text{C.6})$$

Here $v_\alpha \in \mathbb{R}^n$ has component 1 in the subspace of dimension-coordinate α and zero otherwise, e.g. $v_1 = (1, 0, 0, 1, 0, 0, \dots)$. And $w_\alpha = L_r^T v_\alpha$, where $L_r \in \mathbb{R}^n \times \mathbb{R}^n$ consists of N block matrices 3×3 of angular momentum multiplication. Vectors v_α, w_α only depend on r , and are constants in the p space. Since the integrand in (C.4) is invariant under rotations in the p space we can choose a reference frame such that v_1 coincides with the n -dimensional z axis. The Jacobian is unity for orthogonal transformations and as a consequence $\delta(v_1 \cdot p) = \delta(p_n)$. We can reduce the integral to $d^{n-1} p$. Repeating for all the constraints we write the reduced integral

$$\langle h_A \rangle = \Omega^{-1} \int d^n r d^s p \delta \left[\frac{p^T p}{2m} + \frac{r^T Q r}{2} + \mathcal{U}_A - E \right] \quad (\text{C.7})$$

where $s = n - 6 = 3N - 6$, or $3N - 5$ for linear molecules. The symmetric quadratic form Q can be diagonalized with an orthogonal transformation, such that

$$Q = O^T D O \quad O^T O = 1 \quad (\text{C.8})$$

$$D = \text{diag}(\mu_1^A, \dots, \mu_s^A, 0, 0, 0, 0, 0, 0) \quad (\text{C.9})$$

and $\mu_i^A > 0 \forall i = 1 \dots s$. Applying the transformation $r \rightarrow Or$ in the integral we can write

$$\begin{aligned} \langle h_A \rangle &= \Omega^{-1} \int d^{n-s} r \int d^s r d^s p \delta \left[\frac{p^T p}{2m} + \frac{r^T D r}{2} + \mathcal{U}_A - E \right] \\ &= \frac{\Omega^{-1} V^{n-s} S_{2s}}{\prod_{i=1}^s \sqrt{\frac{\mu_i^A}{m}}} [2(E - \mathcal{U}_A)]^{s-1} \end{aligned} \quad (\text{C.10})$$

where we used Eq. (C.1), and V is the volume of the system.

Next, consider the numerator of Eq. (1.33), which requires the definition of a reaction coordinate. Since we identify the transition state as a saddle point of index one, we take precisely the coordinate corresponding to the unstable mode, which we assign to axis s

$$\lambda(r) = \hat{\mathbf{i}}_s^T O^T r \quad (\text{C.11})$$

$$\dot{\lambda}(r, p) = \hat{\mathbf{i}}_s^T O^T p/m \quad (\text{C.12})$$

Here we approximate again $\mathcal{U}(r)$ with a quadratic form at the saddle point. A diagonalization like (C.8) holds, but now $\mu_i^{TS} > 0 \forall i = 1 \dots s-1$ and $\mu_s^{TS} < 0$. The vector $\hat{\mathbf{i}}_s$ has component 1 for coordinate s and zero otherwise. The equation for $\dot{\lambda}$ follows from the Hamiltonian (C.2). The numerator is written then (using (1.37))

$$\frac{\Omega^{-1}}{2m} \int d^n r d^n p \delta \left[\frac{p^T p}{2m} + \frac{r^T Q r}{2} + \mathcal{U}_{TS} - E \right] \delta(\mathbf{P}) \delta(\mathbf{L}) \delta(\hat{\mathbf{i}}_s^T O^T r) |\hat{\mathbf{i}}_s^T O^T p| \quad (\text{C.13})$$

We want to apply the same procedure of before to eliminate the constraints. The term in square brackets is invariant under rotations in p space, but the last modulus is not. This is not a problem however, because after rotating to integrate the $\delta(\mathbf{P})\delta(\mathbf{L})$, we can rotate back to the original frame in the remaining subspace. So the constraints disappear and we are left with an integration on $d^s p$. Using now the orthogonal transformation that diagonalizes Q , on both r and p we get

$$\begin{aligned} & \frac{\Omega^{-1}}{2m} \int d^{n-s} r \int d^s r d^s p \delta \left[\frac{p^T p}{2m} + \frac{r^T D r}{2} + \mathcal{U}_{TS} - E \right] \delta(r_s) |p_s| \quad (\text{C.14}) \\ &= \frac{\Omega^{-1} V^{n-s}}{2m} \int dp_s |p_s| \int d^{s-1} r d^{s-1} p \delta \left[\frac{p^T p}{2m} + \frac{r^T D' r}{2} + \mathcal{U}_{TS} - E \right] \\ &= \frac{\Omega^{-1} V^{n-s}}{2m} \int dp_s |p_s| \frac{S_{2s-2}}{\prod_{i=1}^{s-1} \sqrt{\frac{\mu_i^{TS}}{m}}} \left[2(E - \mathcal{U}_{TS}) - \frac{p_s^2}{2m} \right]^{s-2} \\ &= \frac{\Omega^{-1} V^{n-s} S_{2s-2}}{2m \prod_{i=1}^{s-1} \sqrt{\frac{\mu_i^{TS}}{m}}} \frac{1}{m^{s-2}} \int_0^{2m(E-\mathcal{U}_{TS})} y^{s-2} dy \\ &= \frac{\Omega^{-1} V^{n-s} S_{2s-2}}{2(s-1) \prod_{i=1}^{s-1} \sqrt{\frac{\mu_i^{TS}}{m}}} [2(E - \mathcal{U}_{TS})]^{s-1} \end{aligned}$$

where $D' = \text{diag}(\mu_1^A, \dots, \mu_{s-1}^A)$, we used formula (C.1) again, and we changed variable from p_s to $y = 2(E - \mathcal{U}_{TS}) - p_s^2/2m$. The integration top limit is cut to maintain the integrand positive.

Finally the ratio of (C.10) and (C.14) is

$$\begin{aligned} k_{AB}^{RRKM}(E) &= \left(\frac{E - \mathcal{U}_{TS}}{E - \mathcal{U}_A} \right)^{s-1} \frac{\prod_{i=1}^s \sqrt{\frac{\mu_i^A}{m}}}{\prod_{i=1}^{s-1} \sqrt{\frac{\mu_i^{TS}}{m}}} \frac{S_{2s-2}}{2(s-1)S_{2s}} \\ &= \left(\frac{E - \mathcal{U}_{TS}}{E - \mathcal{U}_A} \right)^{s-1} \frac{\prod_{i=1}^s \nu_i^A}{\prod_{i=1}^{s-1} \nu_i^{TS}} \end{aligned} \quad (\text{C.15})$$

where we used $S_{2s-2}/2(s-1)S_{2s} = 1/2\pi$ and defined the vibrational frequencies

$$\nu_i = \frac{1}{2\pi} \sqrt{\frac{\mu_i}{m}}. \quad (\text{C.16})$$

This expression is usually written using the definition of geometric mean. For completeness we also mention that in distribution (C.2) we did not take into account discrete symmetries of the stable and transition state. When everything is considered the RRKM formula reads

$$k_{AB}^{RRKM}(E) = \left(\frac{E - \mathcal{U}_{TS}}{E - \mathcal{U}_A} \right)^{s-1} \frac{(\bar{\nu}^A)^s}{(\bar{\nu}^{TS})^{s-1}} \frac{h_{PG}^A}{h_{PG}^{TS}} \quad (\text{C.17})$$

where h_{PG} is the order of the prevailing point group [33].

A final note concerns systems where the masses are not all equal to m . When the system has a mass matrix M , Eq. (1.18), not proportional to the identity, we lose the rotational invariance of \mathcal{H} in the full p space (but not in the real space!). The procedure then is to apply a canonical transformation $p \rightarrow M^{-1/2}p$, $r \rightarrow M^{+1/2}r$, to mass weighted coordinates and unity mass matrix. When the new potential energy is approximated with a quadratic form, new stationary points are found and new associated eigenvalue problems are solved. The harmonic RRKM procedure is the same, and expression (C.17) still holds, but the frequencies (C.16) are defined as $\nu_i = \sqrt{\mu'_i}/2\pi$ where μ'_i are the eigenvalues of the mass-weighted quadratic potential energy $M^{-1/2}QM^{-1/2}$. All this complicated procedure has a simple geometrical explanation. As pointed out by Goldstein [88] when diagonalizing two quadratic forms simultaneously, first one has to be reduced to a sphere, to be able to rotate the other without changing the first.

D

Flux relation

Here, we prove relation (3.14). The demonstration is based on the continuity of the trajectories, regardless of deterministic or stochastic dynamics, and can be generalized also to a discrete state dynamics, if the evolution allows only jumps of one-step, e.g. ± 1 .

Given three interfaces $i < j < k$ we show how the effective flux $\phi_{i,k}$ can be related to the effective flux $\phi_{i,j}$. If at time $t = 0$ a trajectory passes interface k while having started in i some time earlier, there must always be an unique time when it passed interface j for the first time. Therefore we can write:

$$\phi_{ki}(x_0) = \phi_{ki}(x_0) \int_0^{t_{ki}^b(x_0)} dt \phi_{ji}(x_{-t}) \quad (\text{D.1})$$

We denote with $t_{ki}^b(x_0)$, $t_{ki}^f(x_0)$ the backward and forward exit times from the region between i and k . The situation is that of Fig. 3.5 (a), x_0 is the black dot on k and comes directly from i . The integral contributes only at the point on j denoted with another black dot. The contribution is 1 because of the definition of ϕ_{ji} .

Hence

$$\begin{aligned} \langle \phi_{ki}(x_0) \rangle &= \int_0^\infty dt \langle \phi_{ji}(x_{-t}) \phi_{ki}(x_0) \theta(t_{ki}^b(x_0) - t) \rangle \\ &= \int_0^\infty dt \langle \phi_{ji}(x_0) \phi_{ki}(x_t) \theta(t_{ki}^b(x_t) - t) \rangle \\ &= \left\langle \phi_{ji}(x_0) \int_0^\infty dt \phi_{ki}(x_t) \theta(t_{ki}^b(x_t) - t) \right\rangle \\ &= \left\langle \phi_{ji}(x_0) \int_0^{t_{ki}^f(x_0)} dt \phi_{ki}(x_t) \right\rangle \\ &= \langle \phi_{ji}(x_0) h_{ki}^f(x_0) \rangle. \end{aligned} \quad (\text{D.2})$$

The last but one equation needs some explanation. Denote with $t_\Omega^f(x)$, $t_\Omega^b(x)$ the forward and backward exit time from a region Ω . For each phase point x_0 and phase space region Ω it can be shown that if $t > t_\Omega^f(x_0)$ then $t_\Omega^b(x_t) \leq t$ and hence $\theta(t_\Omega^b(x_t) - t) = 0$. With reference again to Fig. 3.5 (a), now x_0 is the black dot on j .

We can have a contribution (equal to 1) from the integral only if starting from x_0 we do not touch i before k , otherwise is 0. That is precisely the function $h_{ki}^f(x_0)$.

Finally we rewrite the last expression of Eq. (D.2) as a different ensemble average:

$$\begin{aligned}
 \langle \phi_{ki}(x_0) \rangle &= \frac{\langle \phi_{ji}(x_0) h_{ki}^f(x_0) \rangle}{\langle \phi_{ji}(x_0) \rangle} \cdot \langle \phi_{ji}(x_0) \rangle \\
 &= \langle h_{ki}^f(x_0) \rangle_{\phi_{ji}} \cdot \langle \phi_{ji}(x_0) \rangle \\
 &= P_i^{(k|j)} \langle \phi_{ji} \rangle
 \end{aligned} \tag{D.3}$$

which is relation (3.14).

E Recursive relations for PPTIS

E.1 Recursive relations for the illustrative example

In this appendix we will derive the recursive relations (4.2) for the chain of metastable states. For the transfer in the positive direction we can write

$$\begin{aligned} T[1 \rightarrow_0^j] &= T[1 \rightarrow_0^{j-1}]T[j-1 \rightarrow_j^j] \\ &= T[1 \rightarrow_0^{j-1}](1 - T[j-1 \rightarrow_j^0]) \end{aligned} \quad (\text{E.1})$$

and for the reverse direction

$$\begin{aligned} T[j-1 \rightarrow_j^0] &= \tau_{j-1,j-2}T[j-2 \rightarrow_j^0] \\ &= \tau_{j-1,j-2} \left(T[j-2 \rightarrow_{j-1}^0] + \right. \\ &\quad \left. T[j-2 \rightarrow_0^{j-1}]T[j-1 \rightarrow_j^0] \right) \\ &= \tau_{j-1,j-2} \left(T[j-2 \rightarrow_{j-1}^0] + \right. \\ &\quad \left. (1 - T[j-2 \rightarrow_{j-1}^0])T[j-1 \rightarrow_j^0] \right) \end{aligned} \quad (\text{E.2})$$

Bringing the $T[j-1 \rightarrow_j^0]$ terms of Eq. (E.2) to the left side gives us:

$$T[j-1 \rightarrow_j^0] = \frac{\tau_{j-1,j-2}T[j-2 \rightarrow_{j-1}^0]}{1 - \tau_{j-1,j-2}(1 - T[j-2 \rightarrow_{j-1}^0])} \quad (\text{E.3})$$

Using $1 - \tau_{j-1,j-2} = \tau_{j-1,j}$, we see that Eq. (E.3) is equivalent to Eq. (4.2b). Eq. (4.2a) is then obtained by substitution into Eq. (E.1).

E.2 Recursive relations for a general barrier

The criterion of Eq. (4.6) gives for any positive integer $q > 0$ the following approximate relations:

$$\begin{aligned} P_m^{(l|i_{\pm q})} &\approx P_m^{(l|i_{\pm 1})} \\ P_{i-1}^{(i+q|i_{+1})} &\approx P_{i-1}^{(i+q|i_{-1})}(p_i^{\dagger}/p_i^{\pm}) \\ P_{i+1}^{(i-q|i_{-1})} &\approx P_{i+1}^{(i-q|i_{+1})}(p_i^{\bar{}}/p_i^{\mp}) \end{aligned} \quad (\text{E.4})$$

With this in mind we can start a derivation similar to Appendix E.1:

$$\begin{aligned} P_j^+ &\equiv P_{(0|0)}^{(j|1)} = P_{(0|0)}^{(j-1|1)} P_{(0|0)}^{(j|j-1)} \\ &\approx P_{(0|0)}^{(j-1|1)} P_{(0|j-2)}^{(j|j-1)} \\ &= P_{j-1}^+ \left(1 - P_{(j|j-2)}^{(0|j-1)} \right) \\ &\approx P_{j-1}^+ \left(1 - P_{(j|j)}^{(0|j-1)} \frac{p_{j-1}^{\bar{}}}{p_{j-1}^{\mp}} \right) \\ &= P_{j-1}^+ \left(1 - P_j^- \frac{p_{j-1}^{\bar{}}}{p_{j-1}^{\mp}} \right) \end{aligned} \quad (\text{E.5})$$

and for the reverse direction we can write:

$$\begin{aligned} P_j^- &= P_{(j|j)}^{(0|j-1)} = p_{j-1}^{\mp} P_{(j|j)}^{(0|j-2)} \\ &\approx p_{j-1}^{\mp} P_{(j|j-1)}^{(0|j-2)} \\ &\approx p_{j-1}^{\mp} [P_{(j-1|j-1)}^{(0|j-2)} + P_{(0|j-1)}^{(j-1|j-2)} P_{(j|j-2)}^{(0|j-1)}] \\ &= p_{j-1}^{\mp} [P_{j-1}^- + (1 - P_{j-1}^-) P_{(j|j-2)}^{(0|j-1)}] \\ &\approx p_{j-1}^{\mp} [P_{j-1}^- + (1 - P_{j-1}^-) P_{(j|j)}^{(0|j-1)} \frac{p_{j-1}^{\bar{}}}{p_{j-1}^{\mp}}] \\ &= p_{j-1}^{\mp} [P_{j-1}^- + (1 - P_{j-1}^-) P_j^- \frac{p_{j-1}^{\bar{}}}{p_{j-1}^{\mp}}] \end{aligned} \quad (\text{E.6})$$

Bringing the P_j^- terms to the left results in:

$$P_j^- = \frac{P_{j-1}^- p_{j-1}^{\mp}}{1 - (1 - P_{j-1}^-) p_{j-1}^{\bar{}}} \quad (\text{E.7})$$

With the help of Eq. (4.4) we can see that this is equivalent to expression (4.7). Substitution of this relation into Eq. (E.5) results in the expression for P_j^+ in Eq. (4.7).

The recursive relations Eq. (4.7) also admit a non-recursive solution. First, since for the ratio P_j^+/P_j^- we can write

$$\frac{P_j^+}{P_j^-} = \frac{P_{j-1}^+}{P_{j-1}^-} \cdot \frac{p_{j-1}^{\pm}}{p_{j-1}^{\mp}}, \quad (\text{E.8})$$

we can easily find a solution for the n -th term of the sequence:

$$\frac{P_n^+}{P_n^-} = \prod_{j=1}^{n-1} \frac{p_j^\pm}{p_j^\mp} \quad (\text{E.9})$$

Second, it can be checked by direct substitution that the following formulas are a solution to the recursive relations (4.7):

$$P_n^+ = D_n^{-1} \prod_{i=1}^{n-1} p_i^\pm \quad (\text{E.10})$$

$$P_n^- = D_n^{-1} \prod_{i=1}^{n-1} p_i^\mp \quad (\text{E.11})$$

$$D_n \equiv \prod_{i=2}^{n-1} p_i^\pm + \sum_{k=2}^{n-1} \left[\left(\prod_{i=1}^{k-1} p_i^\mp \right) p_k^- \left(\prod_{i=k+1}^{n-1} p_i^\pm \right) \right] \quad (\text{E.12})$$

They have indeed a structure similar to the solution of mean-first passage time problems for birth-death processes [6].

E.3 Recursive relations for simple PPTIS

Following the same derivations of sec. E.2

$$\begin{aligned} P_l^+ &\equiv P(0|0_-) = P(r|0_-)P(l|0_-) \\ &\approx P_r^+ P(0|_{r-}) \end{aligned} \quad (\text{E.13})$$

where I used the memory loss assumption. For the last quantity I can write

$$\begin{aligned} P(0|_{r-}) &= 1 - P(l|_{r-}) \\ &= 1 - P(r|_{r-})P(l|_{r+}) \\ &= 1 - P(r|_{r-})[1 - P(0|_{r+})] \\ &= 1 - P(r|_{r-})[1 - P(r|_{r+})P(0|_{r-})] \\ &= 1 - p_r^- [1 - (1 - P_r^-)P(0|_{r-})] \end{aligned} \quad (\text{E.14})$$

and solving for $P(0|_{r-})$:

$$P(0|_{r-}) = \frac{p_r^\pm}{p_r^\pm + p_r^- P_r^-} \quad (\text{E.15})$$

Putting (E.15) into (E.13) you get the first of (4.25).

Then the reverse part

$$\begin{aligned} P_l^- &\equiv P(0|l_+) = P(r|l_+)P(0|l_+) \\ &\approx p_r^\pm P(l|_{r+}) \end{aligned} \quad (\text{E.16})$$

where I used the memory loss assumption. For the last quantity I can write

$$\begin{aligned}
P(l^0|_{r+}) &= 1 - P(l^l|_{r+}) \\
&= 1 - P(l^r|_{r+})P(l^l|_{r-}) \\
&= 1 - (1 - P_r^-)[1 - P(l^0|_{r-})] \\
&= 1 - (1 - P_r^-)[1 - P(l^r|_{r-})P(l^0|_{r+})] \\
&= 1 - (1 - P_r^-)[1 - p_r^- P(l^0|_{r+})]
\end{aligned} \tag{E.17}$$

and solving for $P(l^0|_{r+})$

$$P(l^0|_{r+}) = \frac{P_r^-}{p_r^\pm + p_r^- P_r^-} \tag{E.18}$$

Putting (E.18) into (E.16) you get the second of (4.25).

E.4 Probability relation for symmetrical hopping model

Consider the discrete model of sec. 4.1.1 in the case of symmetrical hopping probabilities $\tau_{i,i+1} = \tau_{i,i-1} = 1/2$ for all i . We want to prove the relation

$$T[i \rightarrow_0^j] = \frac{i}{j} \tag{E.19}$$

where $T[i \rightarrow_0^j]$ is the probability of being in i and reach $j > i$ before 0. By definition $T[i \rightarrow_0^j] = 1 - T[i \rightarrow_j^0]$.

The proof proceeds in two steps. First we prove by induction that

$$T[j \rightarrow_0^{j+1}] = \frac{j}{j+1} \tag{E.20}$$

For $j = 1$, it is easily seen that $T[1 \rightarrow_0^2] = 1/2$. Suppose then that (E.20) holds for $j - 1$, we have to prove it holds for j . Using Eq. (E.3) in the case of symmetrical hopping probabilities, we can write

$$T[j \rightarrow_{j+1}^0] = \frac{T[j-1 \rightarrow_j^0]}{1 + T[j-1 \rightarrow_j^0]} \tag{E.21}$$

from which

$$\begin{aligned}
T[j \rightarrow_0^{j+1}] &= 1 - \frac{T[j-1 \rightarrow_j^0]}{1 + T[j-1 \rightarrow_j^0]} \\
&= \frac{1}{1 + T[j-1 \rightarrow_j^0]} \\
&= \frac{1}{2 - T[j-1 \rightarrow_0^j]} \\
&= \frac{1}{2 - (j-1)/j} = \frac{j}{j+1}
\end{aligned} \tag{E.22}$$

where in the last passage we have used the induction hypothesis.

Second we use induction again on formula (E.20), for fixed i and varying $j > i$. We have just proved that it holds for $j = i + 1$. Let us assume it holds for j and let us prove it for $j + 1$. This is immediately seen because

$$\begin{aligned} T[i \rightarrow_0^{j+1}] &= T[i \rightarrow_0^j] T[j \rightarrow_0^{j+1}] \\ &= \frac{i}{j} \frac{j}{j+1} = \frac{i}{j+1} \end{aligned} \quad (\text{E.23})$$

A consequence of expression (E.19) is that using Eq. (4.1) the rate constant k_{AB} can be written as

$$k_{AB} = k_{0,1} T[1 \rightarrow_0^s] = \frac{k_{0,1}}{s} \quad (\text{E.24})$$

F Biased and reweighted averages

In Fig. 5.3 we showed the canonical and microcanonical free energies for the model system of sec. 4.3.1. We explain here how we computed them using a single biased MC simulation.

Within the usual convention of this thesis, we denote with $r = (\mathbf{r}_1, \dots, \mathbf{r}_N)$ and $p = (\mathbf{p}_1, \dots, \mathbf{p}_N)$ the $n = dN$ dimensional vectors of coordinates and momenta, where d is the dimension of the system and N the number of particles. The Hamiltonian of the model reads

$$\begin{aligned}
 \mathcal{H}(r, p) &= \mathcal{K}(p) + \mathcal{U}(r) \\
 &= \sum_{i=1}^N \frac{\mathbf{p}_i^2}{2m} + \mathcal{U}_{WCA}(r) + U_{ddw}(r_d(r)) \\
 &= \sum_{i=1}^N \frac{\mathbf{p}_i^2}{2m} + \sum_{i < j} U_{WCA}(r_{ij}) + U_{ddw}(r_{12})
 \end{aligned} \tag{F.1}$$

where $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$ and the order parameter is $\lambda(r, p) = r_d(r) = r_{12}$, identifying particles 1 and 2 as the dimer.

For the canonical free energy we can write

$$e^{-\beta F(r_d)} \equiv \langle \delta(r_d(r) - r_d) \rangle = Z_r^{-1} \int dr \delta(r_d(r) - r_d) e^{-\beta \mathcal{U}(r)} \tag{F.2}$$

$$Z_r = \int dr e^{-\beta \mathcal{U}(r)} \tag{F.3}$$

where we consider only the configurational part because the order parameter does not depend on momenta. Expression (F.2) is easily remanipulated as

$$\begin{aligned}
 e^{-\beta F(r_d)} &= Z_r^{-1} \int dr e^{-\beta [\mathcal{U}(r) + U_{ddw}(r_d(r)) - U_{ddw}(r_d(r))]} \delta(r_d(r) - r_d) \\
 &= Z_r^{-1} \int dr e^{-\beta [\mathcal{U}(r) - U_{ddw}(r_d(r))]} e^{-\beta U_{ddw}(r_d(r))} \delta(r_d(r) - r_d) \\
 &= e^{-\beta U_{ddw}(r_d)} \frac{Z_{WCA}}{Z_r} Z_{WCA}^{-1} \int dr e^{-\beta \mathcal{U}_{WCA}(r)} \delta(r_d(r) - r_d) \\
 &= e^{-\beta U_{ddw}(r_d)} \frac{Z_{WCA}}{Z_r} \langle \delta(r_d(r) - r_d) \rangle_{WCA}
 \end{aligned} \tag{F.4}$$

where

$$Z_{WCA} = \int dr e^{-\beta \mathcal{U}_{WCA}(r)} \quad (\text{F.5})$$

Hence, in Eq. (F.4) the second term is a constant and the last average is on a system of pure WCA particles. In such a system it does not matter which particles are considered 1 and 2, and one can increase the statistics by averaging on all pairs. As a matter of fact the last term is simply the radial distribution function without the radial normalization. Using (F.4) we obtain

$$F(r_d) = U_{ddw}(r_d) - k_B T \ln \langle \delta(r_d(r) - r_d) \rangle_{WCA} \quad (\text{F.6})$$

In the same simulation we can also compute the microcanonical free energy using a reweighting of the sampled configurations. To do this, we need to know the reduced microcanonical probability density $\rho_{mic}(r)$, defined as

$$\rho_{mic}(r) = \Omega_r^{-1} \int dp \delta[\mathcal{H}(r, p) - E] \delta(\mathbf{P}) \quad (\text{F.7})$$

$$\Omega_r = \int dr dp \delta[\mathcal{H}(r, p) - E] \delta(\mathbf{P}) \quad (\text{F.8})$$

where \mathbf{P} is the d -dimensional vector of total linear momentum, which is conserved, see sec. 2.4.1. Note that the term is not present in the canonical free energy because of the separation of coordinates and momenta in the integral average. The computation of (F.7) is along the same lines of sec. C:

$$\begin{aligned} \rho_{mic}(r) &= \Omega_r^{-1} \int d^n p \delta \left[\frac{p^T p}{2m} - (E - \mathcal{U}(r)) \right] \delta(\mathbf{P}) \\ &= \Omega_r^{-1} \int d^s p \delta \left[\frac{p^T p}{2m} - (E - \mathcal{U}(r)) \right] \\ &= \Omega_r^{-1} S_s m^{s/2} [2(E - \mathcal{U}(r))]^{\frac{s-2}{2}} \end{aligned} \quad (\text{F.9})$$

In the second line we eliminated the d linear constraints $\delta(\mathbf{P})$ using the fact that the integrand is invariant for rotations in p space. The remaining degrees of freedom are $s = n - d = dN - d$, and the last line uses formula (C.1). Using the fact that in the MC simulation of a pure WCA system the configurations are sampled with weights proportional to the reduced canonical probability distribution

$$\rho_{can}(r) = Z_r^{-1} e^{-\beta \mathcal{U}_{WCA}(r)} \quad (\text{F.10})$$

we can write for the ratio of the total microcanonical and WCA canonical probability distributions

$$\frac{\rho_{mic}(r)}{\rho_{can}(r)} \propto e^{+\beta \mathcal{U}_{WCA}(r)} [E - \mathcal{U}(r)]^{(dN-d-2)/2} \quad (\text{F.11})$$

The simultaneous computation of the NVT and NVE probabilities $P_{NVT}(r_d)$, $P_{NVE}(r_d)$ proceeds as follows. Since in the pure WCA system it does not matter

which two particles are considered a dimer, define for a configuration r and an inter-particle distance r_{ij} the weight

$$w_{ij} = \begin{cases} \frac{[E - \mathcal{U}_{WCA}(r) - U_{ddw}(r_{ij})]^{(dN-d-2)/2}}{e^{-\beta \mathcal{U}_{WCA}(r)}} & \text{if } [E - \mathcal{U}_{WCA}(r) - U_{ddw}(r_{ij})] > 0 \\ 0 & \text{otherwise} \end{cases} \quad (\text{F.12})$$

derived from (F.11) with the requirement of positive kinetic energy. For each configuration r generated by the MC simulation, loop over all pairs of particles, with a separation less than half the simulation box. For each pair compute the distance r_{ij} , and histogram accordingly. For the NVT histogram add simply 1 to the bin counter, for the NVE histogram, add the weight w_{ij} . Normalization follows as usual at the end after the simulation. The NVE histogram gives $P_{NVE}(r_d)$ directly. To obtain $P_{NVT}(r_d)$ from the NVT histogram instead, one has to add $U_{ddw}(r_d)$ according to formula (F.6).

G

Classical Nucleation Theory

In a system undergoing a nucleation process a new phase is produced inside a metastable phase, called the parent phase. At the end of the 19th century, J.W. Gibbs was the first to realize that the stability of the parent phase is related to the work that has to be done to create a critical nucleus of the new phase [166]. In the 1920's and 1930's his ideas were developed into a kinetic theory of nucleation by Volmer and Weber, Faruhas, and Becker and Döring [167, 168, 169]. This theory was further developed by Zeldovich [170] and Frenkel [171] in the 1940's, and is now known as classical nucleation theory.

We introduced CNT in sec. 7.1 where we discussed solid-liquid nucleation. We present here CNT for the general nucleation of a phase α inside a phase β , and derive Eq. (7.1) for the nucleation barrier and Eq. (7.3) for the nucleation rate.

G.1 Nucleation barrier

Consider a system (I) containing the homogeneous, metastable phase β and a system (II) containing the parent phase β with a nucleus of phase α . The situation is schematized in Fig. G.1 a. We want to compute the Gibbs free energy difference, at constant temperature T and pressure p .

To do so, we consider the difference in total internal energy. Assuming that the internal energy U is a homogeneous first-order function of the extensive parameters S, V, N [172], we can write for system (I)

$$U^I = T^I S^I - p^I V^I + \mu^I N^I \quad (\text{G.1})$$

where S is the entropy, N the number of particles, V the volume, and μ the chemical potential. In system (II) a droplet of phase α is also present. We assume the temperature T^{II} is uniform throughout the system, but in general two different pressure and chemical potentials are associated to phase α and β . Taking into account the energy of the interface between the phases we can write

$$U^{\text{II}} = T^{\text{II}} S^{\text{II}} - p_{\alpha}^{\text{II}} V_{\alpha}^{\text{II}} - p_{\beta}^{\text{II}} V_{\beta}^{\text{II}} + \gamma A + \mu_{\alpha}^{\text{II}} N_{\alpha}^{\text{II}} + \mu_{\beta}^{\text{II}} N_{\beta}^{\text{II}} \quad (\text{G.2})$$

where γ is the surface free energy density, A is the area of the interface, N_{α}, N_{β} are the number of particles in each phase, and V_{α}, V_{β} are the volumes of each phase. Since the

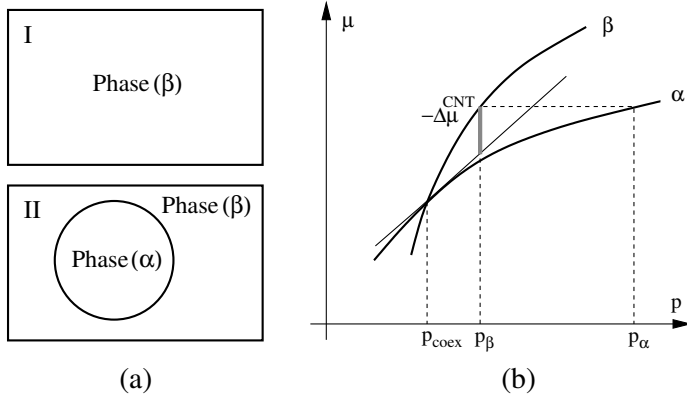


Figure G.1: (a) Formation of a nucleus of a stable phase α in a metastable phase β . (b) Gibbs free energy per particle $\mu(p)$ for the two phases in system II. The phase β is at a supersaturated pressure $p_\beta > p_{coex}$ greater than the coexistence pressure, and is metastable $\Delta\mu = \mu_\alpha(p_\beta) - \mu_\beta(p_\beta) < 0$. Here the case of a critical droplet is shown, where $\mu_\alpha(p_\alpha) = \mu_\beta(p_\beta)$, and the CNT approximation of incompressible phase α , which results in a linear approximation for the curve $\mu(\alpha)$.

total volume is $V^\text{II} = V_\alpha^\text{II} + V_\beta^\text{II}$ and the total number of particles is $N^\text{II} = N_\alpha^\text{II} + N_\beta^\text{II}$, we can rewrite Eq. (G.2) as

$$U^\text{II} = T^\text{II} S^\text{II} - p_\beta^\text{II} V^\text{II} + (p_\beta^\text{II} - p_\alpha^\text{II}) V_\alpha^\text{II} + \gamma A + \mu_\beta^\text{II} N^\text{II} + (\mu_\alpha^\text{II} - \mu_\beta^\text{II}) N_\alpha^\text{II} \quad (\text{G.3})$$

As we consider the two systems at constant number of particles, temperature and pressure, we have $N^\text{I} = N^\text{II}$, $p^\text{I} = p_\beta^\text{II} = p$, $T^\text{I} = T^\text{II} = T$. Moreover, since the parent phase is the same in the two systems $\mu^\text{I} = \mu_\beta^\text{II}$. Taking the difference of Eq. (G.3) and Eq. (G.1), and rearranging for $\Delta G = \Delta U + p\Delta V - T\Delta S$, we obtain

$$\Delta G = G^\text{II} - G^\text{I} = (p_\beta^\text{II} - p_\alpha^\text{II}) V_\alpha^\text{II} + \gamma A + (\mu_\alpha^\text{II} - \mu_\beta^\text{II}) N_\alpha^\text{II} \quad (\text{G.4})$$

This equation holds in general and no approximations have been made. As an illustrative example, think of phase α as a liquid and phase β as a supersaturated vapor, i.e. a vapor at $p_\beta > p_{coex}$. A droplet of liquid at pressure $p_\alpha > p_\beta$ is present in the vapor, and $\mu_\alpha < \mu_\beta$ because the liquid is more stable. In (G.4) this implies that the volume terms proportional to V_α and N_α give a negative contribution, while the surface term γA is always positive. As a consequence ΔG has a maximum for a critical nucleus size. The probability of growth is greater than the probability of shrinking for nuclei with size greater than the critical size. For smaller nuclei, the probability of shrinking prevails. For critical nuclei, the probability of shrinking is equal to that of growth, which implies chemical equilibrium (no net flux of matter), so that $\mu_\alpha^* = \mu_\beta^*$, where the superscript $*$ denotes conditions at the critical size. In this case the pressure difference between the phases is known as the Laplace pressure $\Delta p = p_\alpha - p_\beta$ (see also Fig. G.1 b).

In order to obtain a more useful expression for ΔG , CNT assumes:

1. phase α is incompressible, i.e. $\rho_\alpha \equiv N_\alpha/V_\alpha$ is a constant. From the thermodynamic relations $(\partial G/\partial P)_T = V$, $G = \mu N$, we know that at constant T, N

$$d\mu = dp \frac{V}{N} = \frac{dp}{\rho} \quad (\text{G.5})$$

Since in addition ρ is constant because of incompressibility, we can integrate this relation and obtain a linear relation between μ and p (see also Fig. G.1 b):

$$\mu_\alpha^{\text{II}}(p_\alpha^{\text{II}}) = \mu_\beta^{\text{II}}(p_\beta^{\text{II}}) + \frac{p_\alpha^{\text{II}} - p_\beta^{\text{II}}}{\rho_\alpha} \quad (\text{G.6})$$

Substituting this last expression into Eq. (G.4), in which $\mu_\alpha^{\text{II}} = \mu_\alpha^{\text{II}}(p_\alpha^{\text{II}})$ and $\mu_\beta^{\text{II}} = \mu_\beta^{\text{II}}(p_\beta^{\text{II}})$, we obtain

$$\Delta G = \gamma A + [\mu_\alpha^{\text{II}}(p_\beta^{\text{II}}) - \mu_\beta^{\text{II}}(p_\beta^{\text{II}})] \rho_\alpha V_\alpha^{\text{II}} \quad (\text{G.7})$$

2. the phase α is characterized by its bulk properties, in particular ρ_α is the density of bulk phase α
3. the nucleus is spherical. Then $V_\alpha = 4\pi R^3/3$ and $A = 4\pi R^2$, where R is the radius of the nucleus
4. the surface tension γ is independent of R

Putting all the assumptions together we can rewrite Eq. (G.7) as

$$\Delta G = 4\pi R^2 \gamma + \frac{4}{3} \pi R^3 \rho_\alpha \Delta \mu \quad (\text{G.8})$$

where $\Delta \mu = \mu_\alpha^{\text{II}}(p_\beta^{\text{II}}) - \mu_\beta^{\text{II}}(p_\beta^{\text{II}}) < 0$ is the difference in chemical potential between the two phases at the same supersaturation pressure p_β . Eq. (G.8) is the same of Eq. (7.1) where the phase α is a solid inside an undercooled liquid phase β .

Taking the derivative with respect to R we can find the critical radius that maximizes ΔG

$$R^* = \frac{2\gamma}{\rho_\alpha |\Delta \mu|} \quad (\text{G.9})$$

$$\Delta G^* = \frac{16\pi\gamma^3}{3\rho_\alpha^2 |\Delta \mu|^2} \quad (\text{G.10})$$

Finally, we can also rewrite Eq. (G.8) as function of the number of particles in the nucleus $n = 4\pi R^3 \rho_\alpha/3$

$$\Delta G = 4\pi\gamma \left(\frac{3n}{4\pi\rho_\alpha} \right)^{2/3} + n\Delta \mu \quad (\text{G.11})$$

which gives a critical size

$$n^* = \frac{32\pi\gamma^3}{3\rho_\alpha^2 |\Delta \mu|^3}. \quad (\text{G.12})$$

G.2 Equilibrium distribution of cluster sizes

In this section we relate the CNT free energy, Eq. (G.11), to the distribution of cluster sizes. We give here a qualitative thermodynamic description, but the final relation is exact and can be derived rigorously from statistical mechanics [131, 173].

Consider a system with homogeneous temperature T and denote with N_n the number of clusters of size n . So the system consists of N_1 monomers, N_2 dimers, \dots , N_n n -mers, in a solvent of N_β particles of phase β . We denote the total number of particles with N , and we assume $N_\beta \gg N_n \forall n$ so that $N \approx N_\beta$. Eq. (G.11) can be rewritten

$$\begin{aligned}\Delta G &= \gamma A + n[\mu_\alpha(p) - \mu_\beta(p)] \\ &= \mu_n(p) - n\mu_\beta(p)\end{aligned}\tag{G.13}$$

where we defined the chemical potential of the cluster $\mu(p) \equiv \gamma A + n\mu_\alpha(p)$, and p is the pressure of phase β . We assume that the concentration of clusters is low enough that we can ignore interactions between them and consider the system as an ideal mixture. Using Eq. (G.5) and the equation of state for ideal gases $p = \rho k_B T$ we can obtain the chemical potential of the cluster at a different pressure

$$\mu_n(p_n) = \mu_n(p) + k_B T \ln(p_n/p)\tag{G.14}$$

where we chose the other pressure as the partial pressure p_n exerted by clusters of size n . Substituting in Eq. (G.13) we obtain

$$\begin{aligned}\Delta G &= \mu_n(p_n) - k_B T \ln(p_n/p) - n\mu_\beta(p) \\ &= -k_B T \ln(p_n/p)\end{aligned}\tag{G.15}$$

In the last passage we made use of the equilibrium condition $\mu_n(p_n) = n\mu_\beta(p)$, for the substitution of n particles of phase β with a cluster of n particles of phase α . Since the ratio of pressures p_n/p is equal to the ratio $N_n/N_\beta \approx N_n/N$ (Raoult's law), we finally arrive at the expression

$$P(n) \equiv \frac{N_n}{N} = e^{-\beta \Delta G(n)}\tag{G.16}$$

which relates the cluster size distribution $P(n)$ to the corresponding CNT free energy.

In a simulation, a direct histogramming of $P(n)$ can produce the free energy $\beta \Delta G(n) = -\ln P(n)$ only for small values of n because configurations with big clusters are exponentially rare. Nevertheless special methods can be used, which rely on biasing techniques, like the umbrella sampling method discussed in sec. 2.3.1. Usually the bias is a function of the size of the biggest cluster in the system n_{big} . However, the histogram of n_{big} is approximately equal to the histogram of n . In fact, we can write

$$P(n) = \sum_i P_i(n)\tag{G.17}$$

where $P_i(n)$ is the probability of having i cluster of size n . Assuming that the formations of different clusters are uncorrelated, $P_i(n) = P_1(n)^i$, and we obtain

$$P(n) = \sum_i P_1(n)^i \simeq P_1(n) \equiv P(n_{\text{big}})\tag{G.18}$$

where we used the fact that for rare clusters $P_1(n)$ is small, so that the sum is dominated by the first term.

G.3 Nucleation rate

Here we derive an expression for the nucleation rate in the context of CNT. We assume that clusters grow or shrink via the attachment of single molecules. Consequently, the cluster distribution $N_n(t)$ satisfies a birth-death Master Equation [6, 7]

$$\frac{dN_n}{dt} = w_{n-1}^+ N_{n-1} + w_{n+1}^- N_{n+1} - (w_n^+ + w_n^-) N_n \quad (\text{G.19})$$

Defining the current $J_n = w_n^- N_n - w_{n-1}^+ N_{n-1}$, the above equation is rewritten as $dN_n/dt = J_{n+1} - J_n$, and a time-independent stationary solution N_n^{st} is found imposing $J_n = 0$, or equivalently the detailed balance condition

$$\frac{w_n^-}{w_{n-1}^+} = \frac{N_{n-1}^{st}}{N_n^{st}} \quad (\text{G.20})$$

A solution can be found by recursion and reads [6]

$$N_n^{st} = N_0^{st} \prod_{i=1}^n \frac{w_{i-1}^+}{w_i^-} \quad (\text{G.21})$$

Since we are interested in the rate constant, we follow the indications of chapter 1 and appendix A, and we apply the definition (1.15) of inverse mean first passage time. Following the solution procedure of first passage problem, see sec.A.2 and the example in sec. A.4, we put a reflecting boundary at $n = 0$ and an absorbing boundary at $n = \bar{n} > n^*$, and we consider the equation for the mean first passage time starting from $n = 0$. For onedimensional birth-death processes, the solution reads [6]

$$t_0^{mfp} = \sum_{i=0}^{\bar{n}} \sum_{j=0}^i \frac{\phi(i)}{w_j^+ \phi(j)} \quad (\text{G.22})$$

$$\phi(x) = \prod_{i=1}^x \frac{w_i^+}{w_i^-} \quad (\text{G.23})$$

Making use of Eq. (G.21) we can rearrange

$$\begin{aligned} t_0^{mfp} &= \sum_{i=0}^{\bar{n}} \sum_{j=0}^i \frac{N_{j+1}^{st} w_{j+1}^-}{t_j^+ N_{i+1}^{st} w_{i+1}^-} \\ &= \sum_{i=0}^{\bar{n}} \frac{1}{N_{i+1}^{st} w_{i+1}^-} \sum_{j=0}^i \frac{N_{j+1}^{st} w_{j+1}^-}{t_j^+} \\ &= \sum_{i=0}^{\bar{n}} \frac{1}{N_i^{st} w_i^+} \sum_{j=0}^i N_j^{st} \end{aligned} \quad (\text{G.24})$$

where we made use of the detailed balance equation (G.20).

At equilibrium N_n^{st} is given by Eq. (G.16). Since the free energy has the form (G.11) (see also Fig. 7.2), the first sum in Eq. (G.24) is dominated by the term for $i = n^*$. Correspondingly, the second sum is almost constant, the upper limit j can be replaced with n^* , and it can be evaluated as $N_0^{st} \approx N$. Finally, approximating the free energy with a quadratic form at the maximum, and replacing the first sum with a gaussian integral, we can write

$$(t_0^{mfp})^{-1} = \sqrt{\frac{|\Delta G''(n^*)|}{2\pi k_B T}} w_{n^*}^+ N e^{-\beta \Delta G(n^*)} \quad (\text{G.25})$$

$$= (NZ)(w_{n^*}^+ N) \left(\frac{e^{-\beta \Delta G(n^*)}}{N} \right) \quad (\text{G.26})$$

where we have defined the Zeldovich factor $Z = \sqrt{|\Delta G''(n^*)|/2\pi k_B T}$ [170]. In the second passage we have rewritten the rate as a product of three factors to compare it with the TST-BC expression (1.48). The last factor is the probability of being on top of the barrier when starting from pure phase β , $w_{n^*}^+ N$ is the crossing flux, and NZ is the transmission coefficient κ to take into account the recrossings.

We can also make an estimate for the attachment rate on top of the barrier $w_{n^*}^+$. We multiply the number of particles available at the surface of the nucleus, which is proportional to $(n^*)^{2/3}$, with a typical transition rate of these particles to become part of the nucleus. This transition rate is proportional to D_S/λ_D^2 where D_S is a self-diffusion coefficient and λ_D a typical diffusion distance, of the order of the particle diameter. The final result is [174]

$$w_{n^*}^+ = \frac{24D_S(n^*)^{2/3}}{\lambda_D^2} \quad (\text{G.27})$$

Finally, making use of the CNT expression for the free energy, Eq. (G.11), and dividing by the volume of the system, we can rewrite Eq. (G.25) into an expression for the rate per unit volume I

$$I = \rho_\beta \frac{24D_S(n^*)^{2/3}}{\lambda_D^2} \sqrt{\frac{|\Delta\mu|}{6\pi k_B T n^*}} e^{-\beta \Delta G(n^*)} \quad (\text{G.28})$$

which in the case of solid-liquid nucleation corresponds to Eq. (7.3).

H

NPH dynamics

Consider a system of N particles with mass m . Following Andersen [77], the Hamiltonian for the isobaric-isenthalpic (NPH) ensemble can be written as

$$\mathcal{H}(\boldsymbol{\rho}, \boldsymbol{\pi}) = \frac{1}{2mV^{2/3}} \sum_{i=1}^N \pi_i^2 + \sum_{i<j} U(V^{1/3} \rho_{ij}) + \frac{p_V^2}{2W} + P_{ext}V \quad (\text{H.1})$$

Here V is the volume of the system, $\boldsymbol{\rho} = (\boldsymbol{\rho}_1, \dots, \boldsymbol{\rho}_N)$ are the coordinates scaled with the length of the box $L = V^{1/3}$ and $\boldsymbol{\pi} = (\boldsymbol{\pi}_1, \dots, \boldsymbol{\pi}_N)$ are the conjugate momenta. So $\boldsymbol{\rho}_i = \mathbf{r}_i/L$ and $\boldsymbol{\pi}_i = \mathbf{p}_i L$, where $r = (\mathbf{r}_1, \dots, \mathbf{r}_N)$ are the unscaled coordinates and $p = (\mathbf{p}_1, \dots, \mathbf{p}_N)$ are the unscaled momenta. The pairwise interaction U depends on the modulus $\rho_{ij} = \sqrt{\boldsymbol{\rho}_{ij} \cdot \boldsymbol{\rho}_{ij}} = |\boldsymbol{\rho}_{ij}| = |\boldsymbol{\rho}_i - \boldsymbol{\rho}_j|$, p_V is the momentum conjugate to the volume, W is the mass of the piston, and P_{ext} is the external pressure we want to keep constant.

From (H.1) the equations of motion are derived

$$\dot{\boldsymbol{\rho}}_i = \frac{\partial \mathcal{H}}{\partial \boldsymbol{\pi}_i} = \frac{\boldsymbol{\pi}_i}{mV^{2/3}} \quad (\text{H.2a})$$

$$\dot{\boldsymbol{\pi}}_i = -\frac{\partial \mathcal{H}}{\partial \boldsymbol{\rho}_i} = V^{1/3} \mathbf{F}_i(\boldsymbol{\rho}, V) \quad (\text{H.2b})$$

$$\dot{V} = \frac{\partial \mathcal{H}}{\partial p_V} = \frac{p_V}{W} \quad (\text{H.2c})$$

$$\dot{p}_V = -\frac{\partial \mathcal{H}}{\partial V} = F_V(\boldsymbol{\rho}, \boldsymbol{\pi}, V) \quad (\text{H.2d})$$

We have defined the force \mathbf{F}_i acting on particle i

$$\begin{aligned} \mathbf{F}_i &= \sum_{j \neq i} \mathbf{F}_{ij}(\boldsymbol{\rho}, V) = - \sum_{j \neq i} \frac{\boldsymbol{\rho}_{ij}}{\rho_{ij}} U'(V^{1/3} \rho_{ij}) \\ &= - \sum_{j \neq i} \frac{\mathbf{r}_{ij}}{r_{ij}} U'(r_{ij}) = \sum_{j \neq i} \mathbf{F}_{ij}(\mathbf{r}_{ij}) \end{aligned} \quad (\text{H.3})$$

where \mathbf{F}_{ij} is the contribution of particle j to the force acting on particle i . As shown by Eq. (H.3) it corresponds to the force in the unscaled system. In Eq. (H.2) we have

also defined a force F_V acting on the volume V

$$F_V = P_{int}(\rho, \pi, V) - P_{ext} \quad (\text{H.4})$$

where P_{int} is the internal pressure of the system

$$\begin{aligned} P_{int} &= P_{int}(\rho, \pi, V) = \frac{1}{3V} \left[\frac{1}{mV^{2/3}} \sum_{i=1}^N \pi_i^2 - V^{1/3} \sum_{i<j} \rho_{ij} U'(V^{1/3} \rho_{ij}) \right] \\ &= \frac{1}{3V} \left[\sum_{i=1}^N \frac{\mathbf{p}_i^2}{m} + \sum_{i<j} \mathbf{r}_{ij} \cdot \mathbf{F}_{ij}(\mathbf{r}_{ij}) \right] = P_{int}(r, p, V) \end{aligned} \quad (\text{H.5})$$

In the second line P_{int} is written as the sum of a kinetic contribution and the virial term [175], and again corresponds to the internal pressure in the unscaled system.

By definition of the radial distribution function $g(r)$, Eq. (H.5) yields [175]

$$\begin{aligned} P = \langle P_{int} \rangle &= \left\langle \frac{1}{3V} \sum_{i=1}^N \frac{\mathbf{p}_i^2}{m} \right\rangle + \left\langle \frac{1}{3V} \sum_{i<j} \mathbf{r}_{ij} \cdot \mathbf{F}_{ij}(\mathbf{r}_{ij}) \right\rangle \\ &= \rho k_B T - \frac{2}{3} \pi \rho^2 \int_0^\infty g(r) U'(r) r^3 dr \end{aligned} \quad (\text{H.6})$$

where $\rho = N/V$ is the number density. Since we use a cutoff radius r_c for the interaction, in a simulation we can only compute the virial term up to $r = r_c$, and Eq. (H.6) is rewritten as

$$P = \rho k_B T + \left\langle \frac{1}{3V} \sum_{i<j} \mathbf{r}_{ij} \cdot \mathbf{F}_{ij}(\mathbf{r}_{ij}) \right\rangle - \frac{2}{3} \pi \rho^2 \int_{r_c}^\infty g(r) U'(r) r^3 dr \quad (\text{H.7})$$

During a phase transition both the density ρ and the $g(r)$ change so that the total pressure stays constant. However in a simulation we do not know the function $g(r)$ a priori and we have to make some assumptions. Usually it is assumed that $g(r) = 1$ for $r > r_c$, and in the case of a LJ interaction $U(r)$, Eq. (H.7) is rewritten [138]

$$P = \rho k_B T + \left\langle \frac{1}{3V} \sum_{i<j} \mathbf{r}_{ij} \cdot \mathbf{F}_{ij}(\mathbf{r}_{ij}) \right\rangle + \frac{16}{3} \pi \rho^2 \left[\frac{2}{3} r_c^{-9} - r_c^{-3} \right] \quad (\text{H.8})$$

where the last term are the long-range corrections. In the case of a liquid-solid transition the $g(r)$ functions look like in Fig. 7.4. Consequently, large values of the cutoff ($r_c > 6$) are usually employed, or the corrections are applied a posteriori using the $g(r)$ computed in the separate phases (see for example [147]). In our case however we directly simulate the transition and it is more convenient to have a direct estimate of the pressure. Therefore we calculated P_{int} in the integration scheme using formula Eq. (H.5) and adding the correction factor given by the last term of Eq. (H.8). In our simulations we use $r_c = 2.5$ which is not so large. We checked however that on increasing r_c the difference in pressure stays within a tolerable few percent.

In order to derive an integration algorithm for the equations of motion (H.2) we follow the procedure of [76] to construct explicit time reversible integrators. We have decomposed the Liouville operator associated to the Hamiltonian (H.1) into four parts

$$iL = iL_1 + iL_2 + iL_3 + iL_4 \quad (\text{H.9})$$

where we have chosen

$$iL_1 = \sum_{i=1}^N \frac{1}{mV^{2/3}} \boldsymbol{\pi}_i \cdot \frac{\partial}{\partial \boldsymbol{\rho}_i} \quad (\text{H.10a})$$

$$iL_2 = \frac{p_V}{W} \frac{\partial}{\partial V} \quad (\text{H.10b})$$

$$iL_3 = \sum_{i=1}^N V^{1/3} \mathbf{F}_i \cdot \frac{\partial}{\partial \boldsymbol{\pi}_i} \quad (\text{H.10c})$$

$$iL_4 = F_V \frac{\partial}{\partial p_V} \quad (\text{H.10d})$$

We have employed a Trotter factorization of the Liouville propagator that yields the following propagator accurate to $O(\Delta t^2)$

$$e^{iL_4(\Delta t/2)} e^{iL_3(\Delta t/2)} e^{iL_2(\Delta t/2)} e^{iL_1(\Delta t)} e^{iL_2(\Delta t/2)} e^{iL_3(\Delta t/2)} e^{iL_4(\Delta t/2)} \quad (\text{H.11})$$

from which we obtain the integrator

$$p_V(\Delta t/2) = p_V(0) + \frac{\Delta t}{2} F_V(0) \quad (\text{H.12a})$$

$$\boldsymbol{\pi}_i(\Delta t/2) = \boldsymbol{\pi}_i(0) + \frac{\Delta t}{2} V^{1/3} \mathbf{F}_i(0) \quad (\text{H.12b})$$

$$V(\Delta t/2) = V(0) + \frac{\Delta t}{2} \frac{p_V(\Delta t/2)}{W} \quad (\text{H.12c})$$

$$\boldsymbol{\rho}_i(\Delta t) = \boldsymbol{\rho}_i(0) + \Delta t \frac{\boldsymbol{\pi}_i(\Delta t/2)}{mV^{2/3}} \quad (\text{H.12d})$$

$$V(\Delta t) = V(\Delta t/2) + \frac{\Delta t}{2} \frac{p_V(\Delta t/2)}{W} \quad (\text{H.12e})$$

$$\boldsymbol{\pi}_i(\Delta t) = \boldsymbol{\pi}_i(\Delta t/2) + \frac{\Delta t}{2} V^{1/3} \mathbf{F}_i(\Delta t) \quad (\text{H.12f})$$

$$p_V(\Delta t) = p_V(\Delta t/2) + \frac{\Delta t}{2} F_V(\Delta t) \quad (\text{H.12g})$$

Because of the symmetric factorization of the Liouville propagator, the integrator is time-reversible. Moreover, the dynamical system defined by Eqs. (H.2) has a vanishing phase space compressibility, and thus the algorithm is area-preserving [73]. Note that since we have sandwiched the propagation of V between the propagation of $\boldsymbol{\rho}_i$ and $\boldsymbol{\pi}_i$, we have to compute the forces only once per time-step.

In practice it is more convenient to use scaled coordinates but unscaled momenta.

Inspection of Eqs. (H.12) shows that they can be rewritten as

$$p_V(\Delta t/2) = p_V(0) + \frac{\Delta t}{2} F_V(0) \quad (\text{H.13a})$$

$$\mathbf{p}_i(\Delta t/2) = \mathbf{p}_i(0) + \frac{\Delta t}{2} \mathbf{F}_i(0) \quad (\text{H.13b})$$

$$V(\Delta t/2) = V(0) + \frac{\Delta t}{2} \frac{p_V(\Delta t/2)}{W} \quad (\text{H.13c})$$

$$\boldsymbol{\rho}_i(\Delta t) = \boldsymbol{\rho}_i(0) + \frac{\Delta t}{L} \frac{\mathbf{p}(\Delta t/2)}{m} \quad (\text{H.13d})$$

$$V(\Delta t) = V(\Delta t/2) + \frac{\Delta t}{2} \frac{p_V(\Delta t/2)}{W} \quad (\text{H.13e})$$

$$\mathbf{p}_i(\Delta t) = \mathbf{p}_i(\Delta t/2) + \frac{\Delta t}{2} \mathbf{F}_i(\Delta t) \quad (\text{H.13f})$$

$$p_V(\Delta t) = p_V(\Delta t/2) + \frac{\Delta t}{2} F_V(\Delta t) \quad (\text{H.13g})$$

We remind that although the scaled coordinates are propagated, F_i and F_V correspond to quantities in the unscaled system. In this hybrid scheme scaled coordinates and unscaled momenta are used, and since the compressibility does not vanish, the integrator is no longer area preserving.

Bond order parameters

The order parameters we present here were introduced in [155] to study orientational order in liquids and glasses. They were then applied in [154] to compute nucleation free energy barriers of soft-spheres. In this thesis, the use of bond order parameters is two-fold. They are used to construct the Interface Sampling order parameter n_{big} , the size of the biggest solid cluster in the system, as they enter the algorithm for recognition of solid particles (see sec. 7.2.3). Besides, they are used to analyze the nucleation clusters found by the Path Sampling, as indicators of the type and order of the generated structures.

The bond order parameters are computed looking at the orientational order around each particle. A criterion is needed to recognize the neighbors of a particle i , and we chose to identify them as the particles j within a cutoff radius r_q , which can be derived for instance from the first minimum of the $g(r)$. Let $\mathbf{r}_i, \mathbf{r}_j$ be the position vectors of particle i and j . Define the relative vector $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, its modulus $r_{ij} = |\mathbf{r}_{ij}|$ and its orientation $\hat{\mathbf{r}}_{ij} = \mathbf{r}_{ij}/r_{ij}$. The neighbors of i are those particles j for which $r_{ij} \leq r_q$, and the corresponding $\hat{\mathbf{r}}_{ij}$ is called a bond. Other choices of the definition of neighbors can be devised, based for example on a Voronoy tassellation of the space. It has been shown however that the bond order parameters are not very sensitive to this choice [155], and we conveniently used therefore the numerically cheapest.

We call a bond-function the spherical harmonic $Y_{lm}(\hat{\mathbf{r}}_{ij})$. We consider $\hat{\mathbf{r}}_{ij}$ equivalent to $-\hat{\mathbf{r}}_{ij}$, meaning that the direction of the bond is not important. Hence we consider only spherical harmonics $Y_{lm}(\hat{\mathbf{r}}_{ij})$ with l even [155, 154], for which $Y_{lm}(\hat{\mathbf{r}}_{ij}) = Y_{lm}(-\hat{\mathbf{r}}_{ij})$. Denote then with $N_b(i)$ the number of bonds of particle i , and with N_b the total number of bonds in the system. The relation $N_b = \sum_{i=1}^N N_b(i)/2$ holds, but note that N_b is not the number of pairs of particles because it is defined on neighbors only.

Now we construct averages, denoted by a bar, of bond-functions $Y_{lm}(\hat{\mathbf{r}}_{ij})$ on

1. the neighbor bonds $j = 1 \dots N_b(i)$, to get the *local* quantity, denoted by a small letter

$$\bar{q}_{lm}(i) = \frac{1}{N_b(i)} \sum_{j=1}^{N_b(i)} Y_{lm}(\hat{\mathbf{r}}_{ij}) \quad (\text{I.1})$$

2. all the bonds, or equivalently the former averaged on all particles, to get the

global quantity, denoted by a big letter

$$\bar{Q}_{lm} = \frac{1}{N_b} \sum_{bonds} Y_{lm}(\hat{\mathbf{r}}_{ij}) = \frac{\sum_{i=1}^N N_b(i) \bar{q}_{lm}(i)}{\sum_{i=1}^N N_b(i)} \quad (\text{I.2})$$

where N is the number of particles in the system.

Then we can average on the m -component of the angular momentum to get rotationally invariant objects. Respectively

1. from the local quantities

$$q_l(i) = \left(\frac{4\pi}{2l+1} \sum_{m=-l}^l |\bar{q}_{lm}(i)|^2 \right)^{1/2} \quad (\text{I.3})$$

$$w_l(i) = \sum_{\substack{m_1, m_2, m_3 \\ m_1+m_2+m_3=0}} \begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix} \bar{q}_{lm_1}(i) \bar{q}_{lm_2}(i) \bar{q}_{lm_3}(i) \quad (\text{I.4})$$

$$\hat{w}_l(i) = w_l(i) / \left(\sum_{m=-l}^l |\bar{q}_{lm}(i)|^2 \right)^{3/2} \quad (\text{I.5})$$

2. from the global quantities

$$Q_l = \left(\frac{4\pi}{2l+1} \sum_{m=-l}^l |\bar{Q}_{lm}|^2 \right)^{1/2} \quad (\text{I.6})$$

$$W_l = \sum_{\substack{m_1, m_2, m_3 \\ m_1+m_2+m_3=0}} \begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix} \bar{Q}_{lm_1} \bar{Q}_{lm_2} \bar{Q}_{lm_3} \quad (\text{I.7})$$

$$\hat{W}_l = W_l / \left(\sum_{m=-l}^l |\bar{Q}_{lm}|^2 \right)^{3/2} \quad (\text{I.8})$$

where the terms in square brackets in (I.4), (I.7) are Wigner $3j$ -symbols [176]. The q_l, Q_l are called second-order invariants, the w_l, W_l third-order invariants and the \hat{w}_l, \hat{W}_l are reduced order parameters [154].

I.1 Use in the analysis of cluster structures

In an isotropic liquid, averages of bond-functions, like Eq. (I.2), are expected to vanish for $l \neq 0$, and consequently the rotationally invariant quantities as well. In a solid instead, the quantities Q_l, W_l, \hat{W}_l are sensitive to the specific ordering of the lattice. Naturally, the definitions of the global quantities can be also applied to a subset of particles in the system, such as a nucleating cluster appearing in an undercooled liquid. One has to compute the average (I.2) only on the bonds connecting particles

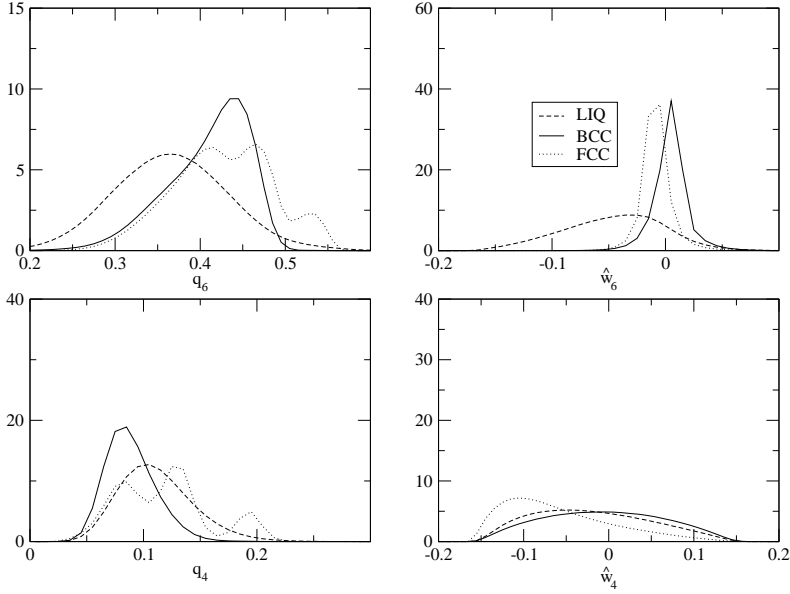


Figure I.1: Distribution functions of the local order parameters $q_4(i)$, $q_6(i)$, $\hat{w}_4(i)$, $\hat{w}_6(i)$ for a thermally equilibrated liquid, bcc, and fcc structure at $P = 5.68$ and constant enthalpy $H/N = 1.412$ corresponding to 25% undercooling.

in the cluster. As a consequence, the bond-order parameters can give an indication on the structure of the cluster. In [155] bond-order parameters up to $l = 10$ were investigated and the set $Q_4, Q_6, \hat{W}_4, \hat{W}_6$ was found particularly useful to distinguish different types of solid structures. In table 7.1 of sec. 7.4 we reported the values of this set of bond-order parameters for a number of regular cluster geometries, as well as for a bcc and fcc lattices equilibrated at the pressure and enthalpy of our NPH Interface Sampling simulations. To each geometry corresponds a different signature, which helps recognizing the cluster structures.

In Sec. 7.4 we also discuss the identification of cluster structures by means of an algorithm based on the distributions of $q_l(i)$, $\hat{w}_l(i)$. The technique uses the local bond-order parameters $q_4(i)$, $q_6(i)$, $\hat{w}_6(i)$ computed for each cluster particle i , but using the full neighbor shell, which on the surface includes also non-cluster particles. Similar to [131], we computed in separate NPH MD simulations the distribution of these local parameters on three equilibrated system, prepared in a liquid, bcc and fcc state, at the same parameters $P = 5.68$, $H/N = 1.412$ of our system. We report them in Fig. I.1, where for completeness we also show the distribution of \hat{w}_4 . We concatenated the distributions of q_4, q_6, \hat{w}_6 into a vector \hat{v}_α , $\alpha = liq, bcc, fcc$, normalized to 1, i.e. $\hat{v}_\alpha \cdot \hat{v}_\alpha = \hat{v}_\alpha^2 = 1$. The number of components of the vector is equal to the sum of the number of bins in each histogram, which in our case ¹ was $100+100+200=400$. Each

¹We used a bin width of 0.01 and the number of bins followed from the fact that $q_l \in [0, 1]$ and $\hat{w}_l \in [-1, 1]$. In practice the distributions are restricted to a smaller interval (see Fig. I.1) and the

of these vectors \hat{v}_α is a sort of signature of the corresponding structure. Then, for a given cluster to be analyzed, we compute the concatenated normalized vector \hat{v}_{cl} and we find the scalar projections f_α onto the structure vectors \hat{v}_α by minimizing

$$\Delta^2 = (\hat{v}_{cl} - \sum_{\alpha} f_{\alpha} \hat{v}_{\alpha})^2 \quad (\text{I.9})$$

The stationary condition reads

$$\frac{\partial \Delta^2}{\partial f_{\beta}} = 2(\hat{v}_{cl} - \sum_{\alpha} f_{\alpha} \hat{v}_{\alpha}) \cdot (-\hat{v}_{\beta}) = 0 \quad (\text{I.10})$$

which is a linear equation and can be written as

$$\hat{v}_{cl} \cdot \hat{v}_{\beta} = \sum_{\alpha} f_{\alpha} \hat{v}_{\alpha} \cdot \hat{v}_{\beta} \quad (\text{I.11})$$

or in matrix form

$$y = Mf \quad (\text{I.12})$$

where we defined the vector $y_{\beta} = \hat{v}_{cl} \cdot \hat{v}_{\beta}$ and the metric matrix $M_{\alpha\beta} = \hat{v}_{\alpha} \cdot \hat{v}_{\beta}$. The first is computed for each cluster to be analyzed, while the second is derived from the basis vectors only and is a fixed quantity in the simulation. If the basis vectors \hat{v}_{α} were orthogonal, then the matrix M would be the identity matrix and the minimization solution f equal to the standard scalar products y . Inverting Eq. (I.12) we can find the requested f -components, namely $f_{liq}, f_{bcc}, f_{fcc}$. Together with Δ^2 , computed at the end from Eq. (I.9), they give an indication of the relative abundances of the selected structures in the cluster from which \hat{v}_{cl} was computed. The normalization of the f -components follows from Eq. (I.9) together with the stationary solution Eq. (I.12), and reads

$$1 = \Delta^2 + f \cdot y = \Delta^2 + f^T M f \quad (\text{I.13})$$

where the role of M as a scalar product matrix is explicit.

I.2 Use in the determination of the biggest cluster

Another use of the bond-order parameters arises in the algorithm for the determination of the biggest solid cluster. In order to identify solid and liquid particles the procedure of sec. 7.2.3 has to determine if two particles i and j are connected. This is done testing if the dot-product d_{ij} , defined by Eq. (7.5), exceeds a threshold. We want to prove here that this test is well defined, demonstrating that the complex quantity d_{ij} is actually real and independent of the order of i and j , i.e. we want to demonstrate $d_{ij} = d_{ij}^* = d_{ji}$. It suffices to consider only the numerator in Eq. (7.5).

effective number of non-zero bins was around 100.

Using the definition of \bar{q}_{lm} Eq. (I.1)

$$\begin{aligned}
 \sum_{m=-l}^l \bar{q}_{lm}(i) \bar{q}_{lm}^*(j) &= \frac{1}{N_b(i)N_b(j)} \sum_{m=-l}^l \sum_{k=1}^{N_b(i)} Y_{lm}(\hat{\mathbf{r}}_{ik}) \sum_{k'=1}^{N_b(j)} Y_{lm}^*(\hat{\mathbf{r}}_{jk'}) \\
 &= \frac{1}{N_b(i)N_b(j)} \sum_{m=-l}^l \sum_{kk'} Y_{l,-m}^*(\hat{\mathbf{r}}_{ik}) Y_{l,-m}(\hat{\mathbf{r}}_{jk'}) \\
 &= \frac{1}{N_b(i)N_b(j)} \sum_{m=-l}^l \sum_{kk'} Y_{l,m}^*(\hat{\mathbf{r}}_{ik}) Y_{l,m}(\hat{\mathbf{r}}_{jk'}) \\
 &= \sum_{m=-l}^l \bar{q}_{lm}^*(i) \bar{q}_{lm}(j) \tag{I.14}
 \end{aligned}$$

where in the second line we used $Y_{l,-m}(\hat{r}) = (-1)^m Y_{lm}(\hat{r})$, and in the third the fact that the sum on m is symmetrical around $m = 0$. This proves $d_{ij} = d_{ij}^*$. By definition then d_{ij}^* is also equal to d_{ji} . Note that the demonstration holds for any value of l and not only $l = 6$ as in Eq. (7.5). Because of this property the number of operations to compute a dot-product like Eq. (I.14) can be reduced to nonnegative m 's only, i.e.

$$\begin{aligned}
 \sum_{m=-l}^l \bar{q}_{lm}(i) \bar{q}_{lm}^*(j) &= \bar{q}_{l0}(i) * \bar{q}_{l0}(j) \\
 &\quad + 2 \sum_{m=1}^l \Re(\bar{q}_{lm}(i)) \Re(\bar{q}_{lm}(j)) + \Im(\bar{q}_{lm}(i)) \Im(\bar{q}_{lm}(j)) \tag{I.15}
 \end{aligned}$$

where \Re, \Im denote the real and imaginary part. The expression applies also in the case $i = j$, to calculate the denominator in Eq. (7.5). So the spherical harmonics can be stored and evaluated for just the nonnegative values of m , bringing a speed-up in the calculation.

J

Generalized committors

Consider Eq. (2.56) defining the committor p_B . Using the definition of the path ensemble (sec. 2.1) and of the function $h_{ij}^f(x)$ (Eq. (3.2)), p_B can be rewritten as

$$p_B(r) = \frac{\langle \delta(r_0 - r) h_{n_I,0}^f(x_0) \rangle}{\langle \delta(r_0 - r) \rangle} = \langle h_{n_I,0}^f(x_0) \rangle_{r_0=r} \quad (\text{J.1})$$

where $x_0 = (r_0, p_0)$ is the phase space point consisting of coordinates r_0 and momenta p_0 of all particles at time 0. So for a given configuration r , $p_B(r)$ is 1 if trajectories started with random momenta from the ensemble distribution reach B at interface n_I before A at interface 0.

We generalize the definition to

$$p_{AB}(r) = \langle h_{0,n_I}^b(x_0) h_{n_I,0}^f(x_0) \rangle_{r_0=r} \quad (\text{J.2})$$

where the backward trajectory is also included. So $p_{AB}(r)$ is 1 if trajectories started from r come from A and end up in B . With the same reasoning we can define $p_{BA}(r)$, $p_{AA}(r)$ and $p_{BB}(r)$.

For any r , $p_{AB}(r) = p_{BA}(r)$ because of microscopic reversibility, and if there are no other attraction basins the normalization $2p_{AB}(r) + p_{AA}(r) + p_{BB}(r) = 1$ holds. The connection with the previous committor is given by the relations

$$p_B = p_{AB} + p_{BB} \quad (\text{J.3})$$

$$p_A = p_{BA} + p_{AA} \quad (\text{J.4})$$

which shows that the standard committor includes not only reactive trajectories $A \rightarrow B$, or $B \rightarrow A$ but also contributions from the $A \rightarrow A$ and $B \rightarrow B$ ones.

We chose a path out of the nucleation transitions (see sec. 7.4) and we computed in two separate simulations the committor p_B and the generalized committors p_{AB}, p_{AA}, p_{BB} . We report the results in Fig. J.1. In the inset we show that relation (J.3) is satisfied. For $p_B = 1/2$ it follows from Eq. (J.3) and the normalization that $p_{AA} = p_{BB}$. Slices around $t = 25$ satisfy this relation, and $p_{AA} = p_{BB} \simeq 0.25$, which implies $p_{AB} = p_{BA} \simeq 0.25$, i.e. equiprobability of ending in A and B , as expected from a transition state.

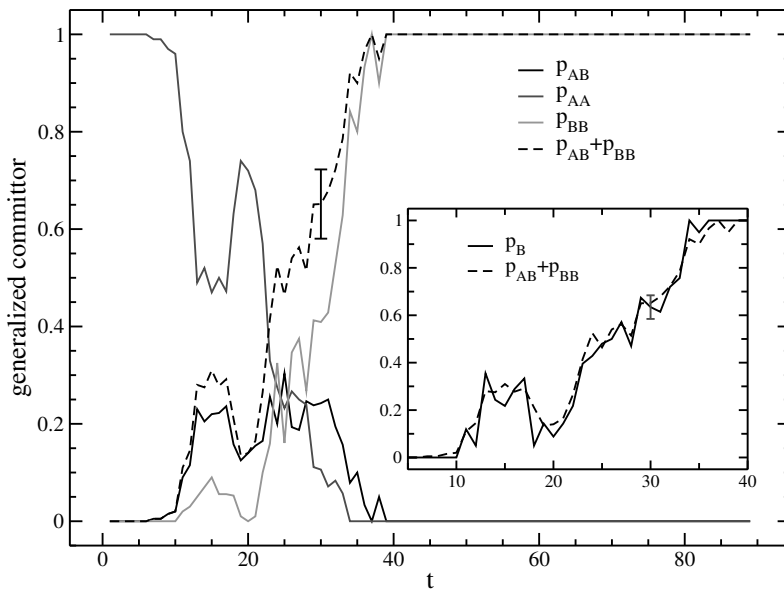


Figure J.1: Generalized committors. A typical error is shown. Inset: comparison with normal committor. Only the part different from 0 and 1 is shown.

We introduced the generalized committors for fear that strange behaviors might occur that are not highlighted by the standard committors. For example a state for which $p_{AA} = p_{BB} \simeq 1$, would still yield $p_B = 0.5$, even though reactive trajectories do not pass through it ($p_{AB} = p_{BA} \simeq 0$). The test reported here proved that this is not the case, at least for the system considered. However, the calculation of generalized committors requires the same computational programming and effort of the standard ones, which can be recovered afterwards from relations (J.3). In any case we would then advise the use of generalized committors as an additional check.

K

Algorithms

We give here an indication of how to build a TIS code and a PPTIS code, including the free energy calculation. We give examples in a pseudo code based on C, and we make extensive use of pointers to handle dynamical memory operations. This appendix is a schematic introduction, but a full working Interface Sampling code can be found at <http://www.science.uva.nl/~moroni/>.

K.1 TIS

Let us start with the variable definitions. The basic data structure we need is the definition of a timeslice

```
typedef struct {  
    double      r,v;                system phase point  
  
    double      Temp;               temperature  
    double      Energy;             conserved energy  
  
    int         nbig;               order parameter  
    int         zone;              zone identifier  
} slice;
```

which we divided into three parts. The first is the system phase point, defined as the minimum amount of information one needs to start an MD trajectory. For NVE dynamics the positions and velocities of all particles are sufficient. For generalized Hamiltonian dynamics the additional degrees of freedom must also be included. The second part of the slice data structure are properties computed from the slice phase point (in the example, temperature and conserved energy). It is not strictly necessary to store them, as they can be recomputed every time a slice is accessed. However it comes useful, and their memory storage is usually negligible compared to the one of the phasepoint, proportional to the number of particles. Moreover, in some cases it might be really necessary to store these properties, as the re-computation might be too time-consuming. The third part of a slice is constituted by the order parameter, which in turn identifies the zone. As explained in sec. 3.2.3, TIS window i is defined by the two interfaces λ_i and λ_{i+1} . We use the zone system, so the variable **zone**

can take the value 0, corresponding to state A , or range between **Fzone** and **Lzone**, defined respectively as the zone before interface λ_i and the zone after interface λ_{i+1} . The integers **Fzone** and **Lzone** are in general not consecutive because there might be sub-interfaces in between.

Using arrays of slices we can construct paths

```
typedef struct {
    slice      data[100];          the slice array

    slice      *start;             path start
    int        Nslice;             number of time slices
    int        direc;              reading direction (+1/-1)
    int        pathtype;           0:  $A \rightarrow \lambda_{i+1}$  or 1:  $A \rightarrow A$ 
} path;
```

The path data structure can be divided in two parts. The first is simply the array of all slices, and in this case we assumed a maximum of 100 slices per path. The second part are path properties that change as paths are sampled. The **start** variable is a pointer to the starting slice of the path, which is not the base address of the array, but changes dynamically. The other variables are self-explaining and their use will become evident in a moment.

The final variables we need, are those defining how the path sampling is performed, plus of course the paths themselves:

```
int        pathstep;              MD steps between slices
int        offset;               offset slice for a path
double     deltav;               velocity change in shooting
double     revprob;              prob. of a reverse move

path       mypath1;              allocated paths
path       mypath2;
path       *pathaux;             pointer to scratch path
path       *pathnow;            pointer to actual path

histogram  interface;            interface crossing prob.

MDstate_data MDstate1;           evolution state 1
MDstate_data MDstate2;           evolution state 2
```

The first variable is used to store timeslices only every **pathstep** MD steps, in order to save memory. Variable **offset** is a fixed offset used in the creation of new paths. The **mypath** variables are fixed memory allocations containing the paths, while the pointers **pathaux** and **pathnow** are dynamical, and are continually swapped by successful shooting moves. The mechanism is explained in Fig. K.1. Variable **interface**¹ is the sampled $P_A(i+1|i)$. Finally we declared two variables **MDstate**. They are the input of an external subroutine responsible for the MD evolution. The paths are constructed moving data back and forward between the slices and the **MDstate** variables.

The main algorithm looks like

¹No **histogram** data type exists in C. Here it is just a notation to distinguish it from other variables. One can just think of it as an array of integers.

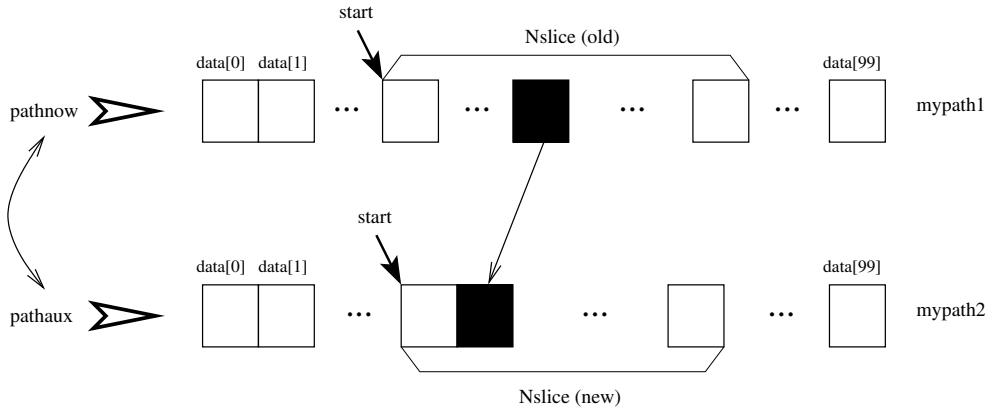


Figure K.1: Path representation and shooting algorithm. An amount of memory is allocated for the timeslices (squares). Paths consists of `Nslice` slices starting from `start`. An attempt to create a new path is done in `pathaux` using a slice (black square) from the present path `pathnow`. If successful, `pathaux` and `pathnow` are swapped.

```
int main()
{
    int Npath;           total number of paths
    int MM;              counter for MC random walk

    initialize();        set-up

    for (MM=1;MM<=Npath;MM++) {    loop on all paths
        if (random()>revprob) {    MC random walk
            shoot();
        } else {
            if (pathnow->pathtype==1) {reverse();}
        }
        sample();            sample on the generated path
    }

    results();            dismiss
}
```

The function `random()` generates a random number uniformly in $[0, 1]$. Reversal move can be applied only if the path starts in A and ends in A , which we define here of type 1. The `initialize()` subroutine is responsible for the set-up of all the variables needed for the MD evolution and the ones named before for the path-sampling. The subroutine `results()` writes out the final output. The rest is the core of path sampling and we describe it in detail.

The reverse move is as follows

```
void reverse()
```

```

{
    slice *p;
    int i;

    pathnow->start+=(pathnow->Nslice-1)*pathnow->direc;
    pathnow->direc=-pathnow->direc;

    p=pathnow->start;
    for (i=0;i<pathnow->Nslice;i++) {
        slice_invert(p); p+=pathnow->direc;
    }
}

```

where the subroutine `slice_invert()` inverts the velocities of all particles. Paths have a starting point `pathnow->start` and a reading direction `pathnow->direc`. To access for example the third slice, one must consider the pointer `pathnow->start+3*pathnow->direc`.

The shooting algorithm is as follows

```

int shoot()
{
    int M;
    int nsmax;
    int Nsliceff;
    slice *q,*p,*ps;
    int cross[100];

```

Pre-MD : set variables

```

Nsliceff=pathnow->Nslice-2;

M=(1+(int)(random()*Nsliceff))*pathnow->direc;
q=pathnow->start+M;
p=pathaux->data+offset+M;
if (pathnow->direc==-1) p+=pathnow->Nslice;
copy_slice(q,p);

slice_vchange(p,deltav);
slice_vadjust(p);
slice_recalculate(p);

copy_slice_MD(p,&MDstate1);
prepare_MD(&MDstate1);
copy_MDstate(&state1,&state2);

nsmax=ceil((double)Nsliceff/random())+2;
pathaux->Nslice=1;
pathaux->direc=pathnow->direc;
ps=p;

```

no shoot from endpoints

slice q in the old path

slice p in the new path

copy slice q into p

change velocities

adjust constraints

recalculate slice properties

copy slice to MD state

prepare the MD state

duplicate MD state

decide max pathlength

reset new number of slices

complete path info

save the shooting point

Backward shooting

```
do {
    p-=pathaux->direc;           move slice pointer backward
    pathaux->Nslice++;           increment length
    MD(&state1,pathstep,-1);     MD evolve for pathstep steps
    copy_MD_slice(&state1,p);    transfer MD state into slice
    slice_recalculate(p);        recalculate slice properties
} while ((p->zzone>0) && (p->zzone<Lzone) && (pathaux->Nslice<nsmax));

if ( (p->zzone != 0) || (pathaux->Nslice>=nsmax) ) return(0);
                                if A is not reached or pathlength
                                is exceeded, reject. Else go on.

pathaux->start=p;               set starting point of new path
```

Forward shooting

```
p=ps;                           restart from the shooting point
do {
    p+=pathaux->direc;           move slice pointer forward
    pathaux->Nslice++;           increment length
    MD(&state2,pathstep,+1);     MD evolve for pathstep steps
    copy_MD_slice(&state2,p);    transfer MD state into slice
    slice_recalculate(p);        recalculate slice properties
} while ((p->zzone>0) && (p->zzone<Lzone) && (pathaux->Nslice<nsmax));

if (pathaux->Nslice>=nsmax) return(0); if pathlength is exceeded, reject
```

Post-MD: Check crossing

```
if ( Ncrossings(pathaux,cross,Fzone)>0 ) {
    pathaux->pathtype=FindPathtype(pathaux); find type
    swap_path(&pathaux,&pathnow);          swap path pointers
    return(1);
}
else return(0);
}
```

We select only slices excluding the endpoints of the path, which means paths have a minimum of three slices. Because of the time-reversal move, paths can be read in two different directions, which is accounted for in the choice of the shooting point. To every velocity component of each particle the subroutine `slice_vchange` adds a random number extracted from a gaussian distribution of zero mean and standard deviation `deltav`. The linear constraints of vanishing linear and angular momentum are taken into account by subroutine `slice_vadjust`, as explained in sec. 2.2.1. The

subroutine `slice_recalculate` computes the second and third part of variables in the slice data structure, as defined at the beginning of this section. They include the order parameter and the zone number. Two `MDstate` variables are used, one for the backward, and one for the forward shooting. It is slightly more efficient than using just one, as the MD forces do not have to be computed again. After the MD evolution two more subroutines require some explanation. `FindPathtype` is the first and reads

```
int FindPathtype(path *pa)
{
    Finds the type looking at the last slice of the path
    0:  $A \rightarrow \lambda_{i+1}$  or 1:  $A \rightarrow A$ 

    slice *p;

    p=pa->start+(pa->Nslice-1)*pa->direc;    go to last slice

    if (p->zone==0) return(1);                if it is in A, type is 1
    else return(0);                           else type is 0
}
```

The second subroutine `Ncrossings` finds all the crossings of a path with a specified interface. It reads

```
int Ncrossings(path *pa, int *cross, int s)
{
    Finds the crossings of the path with the interface s
    interface s separates: zone s — zone s+1

    int nc;    number of crossings
    int i;
    slice *p;

    p=pa->start;
    nc=0;

    for (i=0;i<(pa->Nslice-1);i++) {    Check all slices but last
        if ( crosses(pa,p,s,1) ) {      If crosses
            cross[nc]=i;                 save the slice number
            nc++;
        }
        p+=pa->direc;
    }

    return(nc);
}
```

The crossings are stored in the array `cross` taken as input, and the number of crossing is returned. The auxiliar function `crosses` is

```
int crosses(path *pa, slice *p, int s)
{
```

```

Finds out whether slice p crosses interface s (either directions)
Interface s separates: zone s | zone s+1
A slice cross is defined: p to p+pa->direc

int sR=s+1;

if ( (p->zone<sR && (p+pa->direc)->zone>s) ||
    (p->zone>s && (p+pa->direc)->zone<sR) ) return(1);
else return(0);

}

```

which takes into account the two possible crossings left/right or right/left. In TIS it is not needed to know all the crossings of the path with λ_i , but just to know if the path has at least one. However, the subroutine **Ncrossing** will turn useful for PPTIS.

Finally, when a path has been generated, we can sample the desired crossing probability

```

void sample()
{
    int zonemax;                maximum zone reached
    int i;
    slice *p;

    p=pathnow->start;           set initial value
    zonemax=p->zone;

    for (i=1;i<pathnow->Nslice;i++) { loop on the slices
        p+=pathnow->direc;
        if (p->zone > zonemax) zonemax=p->zone;
    }

    for (i=0;i<zonemax-Fzone;i++) fill the bins up to zonemax
        interface.count[i]++;
}

```

If the zone **zonemax** has been reached, all the (sub)interfaces up to that point have been crossed, and the histogram is filled up accordingly. By definition of the path ensemble all paths must cross the first interface λ_i , so that the value of first bin at the end will be exactly **Npath**. The crossing probability $P_A(i+1|i)$ follows by normalizing the whole histogram to this value, and is performed by the subroutine **results()** after the path sampling.

K.2 PPTIS

The PPTIS algorithm is very similar. The data structure of slices and paths are the same. The types of path are now 4

```

int          pathtype;          0 :  $\pm$ , 1 :  $\mp$ , 2 :  $=$ , 3 :  $\dagger$ 

```

PPTIS window i is defined by interfaces $i-1, i, i+1$. Variable **zone** ranges now only between **Fzone**, before λ_{i-1} , and **Lzone**, after λ_{i+1} . Then **Lzone=Fzone+2** since we do not consider sub-interfaces. In the sampled properties **interface** is replaced by

```
int          Npathtype[4];          counts the type of paths
histogram    loop;                  recrossing points
histogram    bound;                 boundary points
```

containing also the free energy quantities.

The main algorithm only differs in the fact that now the reverse move can always be applied, so no check on the path type is needed

```
    if (random()>revprob) {          MC random walk
        shoot();
    } else {
        reverse();
    }
```

The reverse move is identical, while the shooting move differs in two points. First, in the condition to stop the MD integration interface 0 must be replaced by **Fzone**

```
    } while((p->zone>Fzone) && (p->zone<Lzone) && (pathaux->Nslice<nsmax));
```

Second, the FindPathtype subroutines now reads

```
int FindPathtype(path *pa)
{
    Finds the type looking at the first and last slice of the path
    0 :  $\pm$ , 1 :  $\mp$ , 2 :  $=$ , 3 :  $\ddagger$ 

    slice *p0,*pF;          pointers to first and last slice

    p0=pa->start;            first slice
    pF=p0+(pa->Nslice-1)*pa->direc;    last slice

    if (p0->zone==in->Fzone) {
        if (pF->zone==in->Lzone) pathtype=0;          type  $\pm$ 
        else if (pF->zone==in->Fzone) pathtype=2;      type  $=$ 
    }
    else if (p0->zone==in->Lzone) {
        if (pF->zone==in->Fzone) pathtype=1;          type  $\mp$ 
        else if (pF->zone==in->Lzone) pathtype=3;      type  $\ddagger$ 
    }
}
```

The main difference then is in the **sample()** subroutine, which now reads

```
void sample()
{
    int nc,cross[100];          number of crossings, the crossings with  $\lambda_i$ 
    int i;
    slice *p;
```


Crossing probability

```
Npathtype[pathnow->pathtype]++;
```

count the number of paths
distinguishing the types

Free energy

```
nc=Ncrossings(pathnow,cross,Fzone+1); Find crossings with  $\lambda_i$ 

p=pathnow->start+pathnow->direc;      Start from the second slice
for (i=1;i<pathnow->Nslice-1;i++) {    Loop, up to the last slice but one
    if (i<=cross[0] || i>cross[nc-1]) updatehisto(&bound,p->nbig);
                                     If the slice is before the first cross
                                     or after the last, update bound
    else {updatehisto(&loop,p->nbig);}  else update loop
    p+=pathnow->direc;
}
}
```

The variables `Npathtype` count the paths differentiated per type, and the crossing probabilities follow from Eq. 4.14. The free energy is computed afterwards from the rematch of boundary and loop histograms of all windows, as explained in sec. 5.2.

Bibliography

- [1] D. Turnbull, J. Chem. Phys. **20**, 411 (1952).
- [2] B. M. S. Hansen *et al.*, Astrophys. J. **574**, L155 (2002).
- [3] P. G. Bolhuis, Proc. Nat. Acad. Sci. USA **100**, 12129 (2003).
- [4] P. G. Bolhuis, Biophys. J. **88**, 50 (2005).
- [5] L. Stryer, *Biochemistry*, 4th ed. (W. H. Freeman and Company, New York, 1995).
- [6] C. W. Gardiner, *Handbook of Stochastic Methods* (Springer-Verlag, Berlin, 1983).
- [7] N. G. van Kampen, *Stochastic processes in Physics and Chemistry* (North-Holland, Amsterdam, 1981).
- [8] H. Eyring, J. Chem. Phys. **3**, 107 (1935).
- [9] E. Wigner, Trans. Faraday Soc. **34**, 29 (1938).
- [10] W. E and E. Vanden-Eijnden, in *Multiscale Modelling and Simulation*, Vol. 39 of *Lecture Notes in Computational Science and Engineering*, edited by S. Attinger and P. Koumoutsakos (Springer-Verlag, Berlin, 2004), p. 35.
- [11] D. Frenkel and B. Smit, *Understanding molecular simulation*, 2nd ed. (Academic Press, San Diego, CA, 2002).
- [12] D. Chandler, J. Chem. Phys. **68**, 2959 (1978).
- [13] C. H. Bennett, in *Algorithms for Chemical Computations*, ACS Symposium Series No. 46, edited by R. Christofferson (American Chemical Society, Washington, D.C., 1977).
- [14] G. M. Torrie and J. P. Valleau, Chem. Phys. Lett. **28**, 578 (1974).
- [15] E. A. Carter, G. Ciccotti, J. T. Hynes, and R. Kapral, Chem. Phys. Lett. **156**, 472 (1989).
- [16] M. Sprik and G. Ciccotti, J. Chem. Phys. **109**, 7737 (1998).
- [17] M. J. Ruiz-Montero, D. Frenkel, and J. J. Brey, Mol. Phys. **90**, 925 (1997).
- [18] W. H. Miller, Accounts Chem. Res. **9**, 306 (1976).
- [19] W. H. Miller, J. Chem. Phys. **61**, 1823 (1974).
- [20] T. S. van Erp and P. G. Bolhuis, J. Comp. Phys. (in press).
- [21] T. S. van Erp, D. Moroni, and P. G. Bolhuis, J. Chem. Phys. **118**, 7762 (2003).
- [22] J. P. Bergsma, J. R. Reimers, K. R. Wilson, and J. T. Hynes, J. Chem. Phys. **85**, 5625 (1986).
- [23] B. J. Berne, in *Multiple Time Scales*, edited by J. Brackhill and B. Cohen (Academic Press, Orlando, 1985), p. 419.
- [24] G. W. N. White, S. Goldman, and C. G. Gray, Mol. Phys. **98**, 1871 (2000).
- [25] G. Hummer, J. Chem. Phys. **120**, 516 (2004).
- [26] J. B. Anderson, Adv. Chem. Phys. **91**, 381 (1995).
- [27] J. C. Keck, J. Chem. Phys. **32**, 1035 (1960).
- [28] D. G. Truhlar, B. C. Garrett, and S. J. Klippenstein, J. Phys. Chem. **100**, 31 (1996).
- [29] D. E. Makarov and H. Metiu, J. Chem. Phys. **107**, 7787 (1997).
- [30] G. H. Jóhannesson and H. Jónsson, J. Chem. Phys. **115**, 9644 (2001).

- [31] J. N. Murrell and K. J. Laidler, *Trans. Faraday Soc.* **64**, 371 (1968).
- [32] G. H. Vineyard, *J. Phys. Chem. Solids* **3**, 121 (1957).
- [33] D. J. Wales *et al.*, *Adv. Chem. Phys.* **115**, 1 (2000).
- [34] D. J. Wales, *Energy Landscapes* (Cambridge University Press, Cambridge, 2003).
- [35] C. Dellago, P. G. Bolhuis, and P. L. Geissler, *Adv. Chem. Phys.* **123**, 1 (2002).
- [36] A. F. Voter, *J. Chem. Phys.* **106**, 4665 (1997).
- [37] A. F. Voter, *Phys. Rev. Lett.* **78**, 3908 (1997).
- [38] A. F. Voter, *Phys. Rev. B* **57**, R13985 (1998).
- [39] A. F. Voter and M. R. Sørensen, *Mat. Res. Soc. Symp. Proc.* **538**, 427 (1999).
- [40] M. R. Sørensen and A. F. Voter, *J. Chem. Phys.* **112**, 9599 (2000).
- [41] H. Grubmüller, *Phys. Rev. E* **52**, 2893 (1995).
- [42] A. Laio and M. Parrinello, *Proc. Natl. Acad. Sci. USA* **99**, 12562 (2002).
- [43] M. Iannuzzi, A. Laio, and M. Parrinello, *Phys. Rev. Lett.* **90**, 238302 (2003).
- [44] S. Melchionna, *Phys. Rev. E* **62**, 8762 (2000).
- [45] G. Hummer and I. G. Kevrekidis, *J. Chem. Phys.* **118**, 10762 (2003).
- [46] C. J. Cerjan and W. H. Miller, *J. Chem. Phys.* **75**, 2800 (1981).
- [47] J. P. K. Doye and D. J. Wales, *Z. Phys. D* **40**, 194 (1997).
- [48] G. T. Barkema and N. Mousseau, *Phys. Rev. Lett.* **77**, 4358 (1996).
- [49] G. Henkelmann and H. Jónsson, *J. Chem. Phys.* **111**, 7010 (1999).
- [50] D. T. Gillespie, *J. Comput. Phys.* **28**, 395 (1978).
- [51] A. F. Voter, *Phys. Rev. B* **34**, 6819 (1986).
- [52] K. A. Fichthorn and W. H. Weinberg, *J. Chem. Phys.* **95**, 1090 (1991).
- [53] D. J. Wales, *Mol. Phys.* **100**, 3285 (2002).
- [54] W. Huisinga, C. Schütte, and A. M. Stuart, *Comm. Pure and Appl. Math.* **56**, 0234 (2003).
- [55] W. Huisinga *et al.*, *J. Comp. Chem.* **20**, 1760 (1999).
- [56] G. Mills, H. Jónsson, and G. K. Schenter, *Surf. Sci.* **324**, 305 (1995).
- [57] G. Henkelman, B. P. Uberuaga, and H. Jónsson, *J. Chem. Phys.* **113**, 9901 (2000).
- [58] G. Henkelman and H. Jónsson, *J. Chem. Phys.* **113**, 9978 (2000).
- [59] H. Jónsson, G. Mills, and K. W. Jacobsen, in *Classical and Quantum Dynamics in Condensed Phase Simulations*, International School Enrico Fermi, edited by B. J. Berne, G. Ciccotti, and D. Coker (World Scientific, Singapore, 1998).
- [60] W. E, W. Ren, and E. Vanden-Eijnden, *Phys. Rev. B* **66**, 052301 (2002).
- [61] W. E, W. Ren, and E. Vanden-Eijnden, *J. Appl. Phys.* **93**, 2275 (2003).
- [62] W. E, W. Ren, and E. Vanden-Eijnden, *Communications on Pure and Applied Mathematics* **LVII**, 0637 (2004).
- [63] D. Passerone and M. Parrinello, *Phys. Rev. Lett.* **87**, 108302 (2001).
- [64] D. M. Zuckerman and T. B. Woolf, *J. Chem. Phys.* **111**, 9475 (1999).
- [65] R. Olender and R. Elber, *J. Chem. Phys.* **105**, 9299 (1996).
- [66] V. Zaloj and R. Elber, *Comp. Phys. Comm.* **128**, 118 (2000).
- [67] R. Elber, J. Meller, and R. Olender, *J. Phys. Chem. B* **103**, 899 (1999).
- [68] P. Eastman, N. Grønbech-Jensen, and S. Doniach, *J. Chem. Phys.* **114**, 3823 (2001).
- [69] S. Tănase-Nicola and J. Kurchan, *Phys. Rev. Lett.* **91**, 188302 (2003).
- [70] S. Tănase-Nicola and J. Kurchan, *J. Stat. Phys.* **116**, 1201 (2004).
- [71] A. K. Faradjian and R. Elber, *J. Chem. Phys.* **120**, 10880 (2004).
- [72] R. Car and M. Parrinello, *Phys. Rev. Lett.* **55**, 2471 (1985).
- [73] M. E. Tuckerman, Y. Liu, G. Ciccotti, and G. J. Martyna, *J. Chem. Phys.* **115**, 1678 (2001).
- [74] N. Metropolis *et al.*, *J. Chem. Phys.* **21**, 1087 (1953).
- [75] C. Dellago, P. G. Bolhuis, and D. Chandler, *J. Chem. Phys.* **108**, 9236 (1998).

- [76] G. J. Martyna, M. E. Tuckerman, D. J. Tobias, and M. L. Klein, *Molecular Physics* **87**, 1117 (1996).
- [77] H. C. Andersen, *J. Chem. Phys.* **72**, 2384 (1980).
- [78] P. L. Geissler, C. Dellago, and D. Chandler, *Phys. Chem. Chem. Phys.* **1**, 1317 (1999).
- [79] C. Dellago, P. G. Bolhuis, and D. Chandler, *J. Chem. Phys.* **110**, 6617 (1999).
- [80] C. Dellago, P. G. Bolhuis, F. S. Csajka, and D. Chandler, *J. Chem. Phys.* **108**, 1964 (1998).
- [81] T. J. H. Vlugt and B. Smit, *Phys. Chem. Comm.* **2**, 1 (2001).
- [82] J. P. K. Doye and D. J. Wales, *J. Chem. Phys.* **116**, 3777 (2002).
- [83] D. J. Wales and H. A. Scheraga, *Science* **28**, 1368 (1999).
- [84] D. J. Wales *et al.*, The Cambridge Cluster Database, <http://www-wales.ch.cam.ac.uk/CCD.html>.
- [85] J. E. Jones and A. E. Ingham, *Proc. R. Soc. A* **107**, 636 (1925).
- [86] C. J. Tsai and K. D. Jordan, *J. Phys. Chem.* **97**, 11227 (1993).
- [87] D. J. Wales and R. S. Berry, *J. Chem. Phys.* **92**, 4283 (1990).
- [88] H. Goldstein, *Classical Mechanics*, 2nd ed. (Addison-Wesley, Reading, Massachusetts, 1980).
- [89] M. A. Miller and D. J. Wales, *J. Chem. Phys.* **107**, 8568 (1997).
- [90] P. G. Bolhuis, C. Dellago, and D. Chandler, *Faraday Discuss.* **110**, 421 (1998).
- [91] D. Chandler, in *Classical and Quantum Dynamics in Condensed Phase Simulations*, International School Enrico Fermi, edited by B. J. Berne, G. Ciccotti, and D. Coker (World Scientific, Singapore, 1998), pp. 51–66, check it.
- [92] L. R. Pratt, *J. Chem. Phys.* **85**, 5045 (1986).
- [93] P. G. Bolhuis, *J. Phys. Cond. Matter* **15**, S113 (2003).
- [94] G. E. Crooks and D. Chandler, *Phys. Rev. E* **64**, 026109 (2001).
- [95] F. S. Csajka and D. Chandler, *J. Chem. Phys.* **109**, 1125 (1998).
- [96] P. L. Geissler, C. Dellago, and D. Chandler, *J. Phys. Chem. B* **103**, 3706 (1999).
- [97] J. Martí, F. S. Csajka, and D. Chandler, *Chem. Phys. Lett.* **328**, 169 (2000).
- [98] J. Martí and F. S. Csajka, *J. Chem. Phys.* **113**, 1154 (2000).
- [99] P. G. Bolhuis, C. Dellago, and D. Chandler, *Proc. Natl. Acad. Sci. USA* **97**, 5877 (2000).
- [100] J. Rodriguez, G. Moriena, and D. Laria, *Chem. Phys. Lett.* **356**, 147 (2002).
- [101] D. Laria, J. Rodriguez, C. Dellago, and D. Chandler, *J. Phys. Chem. A* **105**, 2646 (2001).
- [102] P. L. Geissler *et al.*, *Chem. Phys. Lett.* **321**, 225 (2000).
- [103] P. L. Geissler *et al.*, *Science* **291**, 2121 (2001).
- [104] P. L. Geissler and D. Chandler, *J. Chem. Phys.* **113**, 9759 (2000).
- [105] P. G. Bolhuis and D. Chandler, *J. Chem. Phys.* **113**, 8154 (2000).
- [106] T. J. H. Vlugt, C. Dellago, and B. Smit, *J. Chem. Phys.* **113**, 8791 (2000).
- [107] P. G. Bolhuis, D. Chandler, C. Dellago, and P. L. Geissler, *Ann. Rev. Phys. Chem.* **53**, 291 (2002).
- [108] P. G. Bolhuis, C. Dellago, P. L. Geissler, and D. Chandler, *J. Phys. Cond. Matter* **12**, A147 (2000).
- [109] D. Moroni, P. G. Bolhuis, and T. S. van Erp, *J. Chem. Phys.* **120**, 4055 (2004).
- [110] M. Strnad *et al.*, *J. Chem. Phys.* **106**, 3643 (1997).
- [111] Y. Okuno, *J. Chem. Phys.* **105**, 5817 (1996).
- [112] Y. J. Cho, S. R. van de Linde, L. Zhu, and W. L. Hase, *J. Chem. Phys.* **96**, 8275 (1992).
- [113] L. Sun, W. L. Hase, and K. Song, *J. Am. Chem. Soc.* **123**, 5753 (2001).
- [114] S. Nosé, *J. Chem. Phys.* **81**, 511 (1984).

- [115] S. Nosé, *Mol. Phys.* **52**, 255 (1984).
- [116] W. G. Hoover, *Phys. Rev. A* **31**, 1695 (1985).
- [117] S. Duane, A. Kennedy, B. J. Pendleton, and D. Roweth, *Phys. Lett. B* **61**, 216 (1987).
- [118] J. D. Weeks, D. Chandler, and H. C. Andersen, *J. Chem. Phys.* **54**, 5237 (1971).
- [119] T. S. van Erp, Ph.D. thesis, Universiteit van Amsterdam, 2003.
- [120] A. B. Bortz, M. H. Kalos, and J. L. Lebowitz, *J. Comput. Phys.* **17**, 10 (1975).
- [121] D. T. Gillespie, *J. Comput. Phys.* **22**, 403 (1976).
- [122] T. S. van Erp, private communications.
- [123] S. Wolfram, *The Mathematica Book*, 5th ed. (Cambridge University Press, Cambridge, 2003).
- [124] P. Virnau and M. Müller, *J. Chem. Phys.* **120**, 10925 (2004).
- [125] D. Chandler, *Introduction to Modern Statistical Mechanics* (Oxford University Press, New York, 1987).
- [126] D. Moroni, T. S. van Erp, and P. G. Bolhuis, *Physica A* **340**, 395 (2004).
- [127] D. Moroni, T. S. van Erp, and P. G. Bolhuis, *Phys. Rev. E* **71**, in press (2005).
- [128] K. Binder, *Journal de Physique* **41**, C4 (1980).
- [129] D. Turnbull and J. C. Fischer, *J. Chem. Phys.* **17**, 71 (1949).
- [130] T. Palberg, *J. Phys. Cond. Matter* **11**, 323 (1999).
- [131] P. R. ten Wolde, Ph.D. thesis, Universiteit van Amsterdam, 1998, available at <http://www.amolf.nl>.
- [132] W. C. Swope and H. C. Andersen, *Phys. Rev. B* **41**, 7042 (1990).
- [133] P. R. ten Wolde, M. J. Ruiz-Montero, and D. Frenkel, *Phys. Rev. Lett.* **75**, 2714 (1995).
- [134] P. R. ten Wolde, M. J. Ruiz-Montero, and D. Frenkel, *J. Chem. Phys.* **104**, 9932 (1996).
- [135] P. R. ten Wolde, M. J. Ruiz-Montero, and D. Frenkel, *Faraday Discuss.* **104**, 93 (1996).
- [136] S. Alexander and J. McTague, *Phys. Rev. Lett.* **41**, 702 (1978).
- [137] H. E. A. Huitema, J. P. van der Eerden, J. J. M. Janssen, and H. Human, *Phys. Rev. B* **62**, 14690 (2000).
- [138] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Oxford University Press, Oxford, 1987).
- [139] H. J. C. Berendsen *et al.*, *J. Chem. Phys.* **81**, 3684 (1984).
- [140] J. E. Lennard-Jones, *Proc. R. Soc. London, Ser. A* **106**, 463 (1924).
- [141] J.-P. Hansen and L. Verlet, *Phys. Rev.* **184**, 151 (1969).
- [142] J. K. Johnson, J. A. Zollweg, and K. E. Gubbins, *Mol. Phys.* **78**, 591 (1993).
- [143] D. A. Kofke, *J. Chem. Phys.* **98**, 4149 (1993).
- [144] R. Agrawal and D. A. Kofke, *Mol. Phys.* **85**, 43 (1995).
- [145] M. A. van der Hoef, *J. Chem. Phys.* **113**, 8142 (2000).
- [146] M. A. van der Hoef, *J. Chem. Phys.* **117**, 5092 (2002).
- [147] M. A. Barroso and A. L. Ferreira, *J. Chem. Phys.* **116**, 7145 (2002).
- [148] N. W. Ashcroft and N. D. Mermin, *Solid State Physics* (Saunders College Publishing, Orlando, 1976).
- [149] P. A. Monson and D. A. Kofke, *Adv. Chem. Phys.* **115**, 113 (2000).
- [150] P. Loubeyre and J.-P. Hansen, *Phys. Rev. B* **31**, 634 (1985).
- [151] M. Parrinello and A. Rahman, *Phys. Rev. Lett.* **45**, 1196 (1980).
- [152] D. Frenkel, *Phys. Rev. Lett.* **56**, 858 (1986).
- [153] A. B. B. R. Ahuja and B. Johansson, *Phys. Rev. Lett.* **87**, 165505 (2001).
- [154] J. S. van Duijneveldt and D. Frenkel, *J. Chem. Phys.* **96**, 4655 (1992).
- [155] P. J. Steinhardt, D. R. Nelson, and M. Ronchetti, *Phys. Rev. B* **28**, 784 (1983).
- [156] M. Fixman, *J. Chem. Phys.* **36**, 306 (1962).
- [157] M. C. Wang and G. E. Uhlenbeck, *Rev. Mod. Phys.* **17**, 323 (1945).

- [158] A. J. F. Siegert, Phys. Rev. **81**, 617 (1951).
- [159] G. Biroli and J. Kurchan, Phys. Rev. E **64**, 016101 (2001).
- [160] H. A. Kramers, Physica **7**, 284 (1940).
- [161] A. Bovier, M. Eckhoff, V. Gayrard, and M. Klein, WIAS preprints 767 (2002).
- [162] L. S. Kassel, J. Phys. Chem. **32**, 225 (1928).
- [163] O. K. Rice and H. C. Ramsperger, J. Am. Chem. Soc. **49**, 1617 (1927).
- [164] O. K. Rice and H. C. Ramsperger, J. Am. Chem. Soc. **50**, 617 (1928).
- [165] R. Marcus and O. K. Rice, J. Phys. Colloid Chem. **55**, 894 (1951).
- [166] J. W. Gibbs, *The Scientific Papers of J. Willard Gibbs* (Dover, New York, 1961).
- [167] M. Volmer and A. Weber, Z. Phys. Chem. **119**, 227 (1926).
- [168] L. Faruhas, Z. Phys. Chem. **125**, 236 (1927).
- [169] R. Becker and W. Döring, Ann. Phys. **24**, 719 (1935).
- [170] J. Zeldovich, J. Expr. Theor. Phys. (Russia) **12**, 525 (1942).
- [171] J. Frenkel, *Kinetic Theory of Liquids* (Clarendon, Oxford, 1946).
- [172] H. B. Callen, *Thermodynamics and an Introduction to Thermostatistics, 2nd ed.* (John Wiley & Sons, New York, 1985).
- [173] H. Reiss and R. K. Bowles, J. Chem. Phys. **111**, 7501 (1999).
- [174] K. F. Kelton, in *Crystal Nucleation in Liquids and Glasses*, edited by H. Ehrenreich and D. Turnbull (Academic Press, Boston, 1991), Vol. 45, pp. 75–177.
- [175] J. P. Hansen and I. R. McDonald, *Theory of Simple Liquids, 2nd ed.* (Academic Press, London, 1986).
- [176] E. W. Eisenstein, Wigner 3j-Symbol, from MathWorld, A Wolfram Web Resource, <http://mathworld.wolfram.com/Wigner3j-Symbol.html>.

Samenvatting

Dit proefschrift gaat over de theorie en simulatie van fysische of chemische *rare events*. Dit zijn zeldzame transities tussen stabiele toestanden die met zo'n lage frequentie plaatsvinden ten opzichte van de moleculaire tijdschaal, dat ze niet zo maar in een computersimulatie nagebootst kunnen worden. Onder rare events verstaan we dus niet de passage van de komeet van Halley, maar bijvoorbeeld het vouwen van eiwitten, conformatieveranderingen in moleculen, chemische reacties en nucleatie in faseovergangen. Veelgebruikte simulatiemethoden zoals *Molecular Dynamics* (MD), genereren dynamische trajectoria door middel van integratie van de bewegingsvergelijkingen voor alle atomen in het systeem. Maar omdat de moleculaire tijdstap in de orde van femtoseconden ligt, hebben zelfs moderne computers drie jaar rekentijd nodig om een enkele microseconde te simuleren van een redelijk klein atomair systeem (bijv. een eiwit). Dus gewone MD kan niet worden gebruikt voor het simuleren van rare events en speciale methoden moeten ontwikkeld worden. Dit proefschrift beschrijft een recente strategie om transities te bestuderen, de zogenaamde path sampling methode. Speciale aandacht krijgt de door ons ontwikkelde *Transition Interface Sampling* (TIS) methode en zijn toepassing op de kristalnucleatie in een vloeistof.

In hoofdstuk 1 introduceren we het probleem van rare events en de traditionele manieren om deze problemen aan te pakken. Aan de hand van een simpel stochastisch model, de willekeurige telegraaf, geven we een definitie van de reactiesnelheidsconstante: de frequentie waarmee de gebeurtenis of transitie plaatsvindt. In complexere, realistische systemen moet deze definitie gegeneraliseerd worden. De eerste historische poging is de Transition State Theory (TST). TST geeft echter alleen een benadering van de daadwerkelijke reactiesnelheidsconstante, die gecorrigeerd kan worden met behulp van zogenoemde Bennett-Chandler (BC) procedure. Hierin neemt men aan dat er een speciale variabele bestaat, genaamd de reactiecoördinaat (RC), die de voortgang van de reactie goed kan beschrijven. De BC-procedure bestaat uit twee stappen. De eerste stap is de berekening van de vrije energie barrière als functie van de RC. Deze vrije energie curve vertoont een maximum tussen de twee stabiele minima, waaruit de barrièrehoogte is af te leiden. De tweede stap is de berekening van de transmissiecoëfficiënt, door middel van het schieten van dynamische trajectoria vanaf de top van de barrière. Uit deze twee bijdragen kan de reactiesnelheidsconstante berekend worden. De wiskundige details van TST en BC, en de connectie met de theorie van stochastische processen, zijn in de bijlagen A en B beschreven. Het laatste deel van hoofdstuk 1 is een samenvatting van enkele andere methoden voor rare events die in de literatuur zijn verschenen.

Hoofdstuk 2 beschrijft de Transition Path Sampling (TPS) methode, waarop TIS geïnspireerd is. Het achterliggende idee van TPS is om alleen op de interessante stukken van al de mogelijke trajectoria zich te concentreren: de transitiepaden die over de barrière leiden. Met behulp van een Monte Carlo algoritme in de padruimte kan een reeks van trajectoria worden verzameld die de begin- en eindtoestand verbinden: het zogenaamde transitiepadensemble. We leggen uit hoe dit algoritme werkt en hoe de reactiesnelheidsconstante kan worden afgeleid. We geven ook een toepassing van TPS op structurele veranderingen in een kleine cluster van Lennard-Jones deeltjes. We vergelijken de resultaten met de RRKM theorie, een soort gemodificeerde TST (zie bijlage C). TPS heeft een belangrijk voordeel ten opzichte van de TST-BC-procedure. TST-BC is gebaseerd op voorkennis van een RC, maar deze kan erg moeilijk te vinden zijn in complexe systemen. Een slechte benadering van de RC leidt tot een grote fout in de reactiesnelheidsconstante. Daarentegen is TPS gebaseerd op het genereren van de echte dynamische transitie, zonder de noodzakelijkheid van een *a priori* keuze van de RC. Bovendien kan het transitiepadensemble worden geanalyseerd met de *committor* techniek, die in de laatste sectie van het hoofdstuk wordt beschreven. Met deze techniek kan de kwaliteit van een RC bepaald worden. De committors zijn een krachtig hulpmiddel om het reactie mechanisme uit te zoeken.

Hoofdstuk 3 beschrijft de details van de TIS methode. TIS dankt zijn naam aan het gebruik van multidimensionele oppervlakten in de faseruimte: de interfaces. Twee van zulke oppervlakken definiëren de stabiele toestanden, waartussen de andere interfaces geplaatst zijn. De reactiesnelheidsconstante kan berekend worden door middel van een efficiënte bepaling van de *effective positive flux* door deze oppervlakken. De theorie is uitgelegd in sectie 3.1 en in bijlage D. Daarna beschrijven we de praktische uitvoering van een TIS-simulatie. TIS gebruikt een Monte Carlo algoritme, vergelijkbaar met dat van TPS, maar veel sneller, zoals blijkt uit een test op een dimeer in een vloeistof.

In hoofdstuk 4 passen wij TIS aan om diffusieve systemen te kunnen bestuderen. In dat geval is de barrière breed, zodat een pad veel tijd nodig heeft om de andere stabiele toestand te bereiken vanuit de initiële toestand. De transitiepaden worden zodoende te lang, wat de TIS methode vertraagt. De oplossing die wij voorstellen, is om deze lange transitie op te delen in een aantal kortere, partiële paden. Deze procedure geeft de juiste reactiesnelheidsconstante met de aanname dat er decorrelatie (geheugenverlies) van de paden optreedt voor afstanden groter dan de partiële padlengte. De theorie van deze *Partial Path TIS (PPTIS)* wordt behandeld in sectie 4.1 (met behulp van bijlage E), vervolgd door een praktische beschrijving van een computersimulatie. We testen PPTIS op een diffusieve versie van hetzelfde dimeermodel als voor TIS werd gebruikt. TIS en PPTIS geven dezelfde reactiesnelheid, maar PPTIS is veel sneller. In de laatste sectie van hoofdstuk 4 opperen we de mogelijkheid om met behulp van parallelisatie van PPTIS een betere sampling van de padruimte te bereiken.

In hoofdstuk 5 beschrijven we hoe de PPTIS-methode, die op het genereren van dynamische trajectoria gebaseerd is, tegelijkertijd een statische evenwichts- vrije energie kan uitrekenen. Het idee is om de Umbrella Sampling methode voor de vrije energie berekening aan de PPTIS-techniek aan te passen. Uit dezelfde paden die voor de reactieconstant gegenereerd zijn, kan de vrije energie berekend worden, zonder dat dat extra computertijd kost. Achtereenvolgens beschrijven we de noodzakelijke theorie,

de uitvoering van een computersimulatie en de toepassing op hetzelfde dimeermodel zoals gebruikt in hoofdstuk 4. Het resultaat wordt vergeleken met reguliere Umbrella Sampling. Details zijn te vinden in bijlage F.

Hoofdstuk 6 beëindigt de discussie over Interface Sampling. We geven hier puur theoretische afleidingen voor de schaling van de efficiëntie van TIS en PPTIS. Ook nemen we de berekening van de transmissiecoëfficiënt nogmaals onder de loep en bespreken een nieuwe manier om deze uit te rekenen, gebaseerd op het idee van effective positive flux.

In het laatste deel van het proefschrift, hoofdstuk 7, worden de TIS-, PPTIS- en RC-analysetechnieken toegepast op het kristalnucleatieproces. Als een vloeistof onder de bevriezingstemperatuur wordt gebracht, begint de faseovergang naar de vaste stof alleen als een kristalkiem groot genoeg is. De traditionele theorie voor nucleatie, de *classical nucleation theory*, is in bijlage G beschreven. Deze theorie neemt in feite aan dat de grootte van de kiem het enige belangrijke aspect is. Ook de traditionele BC-procedure heeft deze impliciete aanname. Het is echter mogelijk dat ook andere eigenschappen belangrijk zijn tijdens het nucleatieproces. Met behulp van de pad-sampling methoden kunnen we deze belangrijke eigenschappen bepalen. In bijlagen H en I beschrijven wij de details van het systeem en de methoden. De nucleatiesnelheden die uit TIS en PPTIS volgen, komen goed met elkaar overeen. Vervolgens gebruiken wij de committors om het padensemble te analyseren. Een generalisatie van de techniek is ook in bijlage J getest. De belangrijkste conclusie is dat nucleatie kan plaatsvinden via verschillende mechanismen. Soms zijn de kritische kiemen klein en compact, een andere keer groot en minder gestructureerd. Een goede RC zou dus niet alleen de grootte van de cluster moeten bevatten, maar ook de kwaliteit van de kristalstructuur.

In de laatste bijlage K geven wij de computercode van de TIS- en PPTIS-algorithmen. De uitvoering van de methoden is in detail beschreven.

Acknowledgements

In summer 2000 I applied for a PhD position in Amsterdam and came over for an interview. Despite my incomprehensible talk, prof. Berend Smit and dr. Peter Bolhuis decided to take me in the group. I thank them for the trust they showed, and for helping during these years to redirect my rigid and formal preparation in theoretical physics towards more applied problems. I just hope they will understand if I have tried to retain a bit of my background. In this sense I thank my supervisor Peter Bolhuis for the patience with which he has faced my sometimes rebellious student ways of behaving.

I acknowledge the financial support of the university of Rome “La Sapienza” in the last months of my PhD project.

I thank Eric Vanden-Eijnden for inspiring chapter 1. I borrowed ideas from his impressive whiteboard-presentations and papers. For chapter 2 I largely used the first section of ref. [35]. I thank the authors and in particular Christoph Dellago for not complaining about that. The research of chapter 7 was done in collaboration with Pieter Rein ten Wolde, whom I also thank for letting me use his code as a basis for my own.

The rest of this thesis followed from the ideas of my friend Titus van Erp, which me and Titus worked out together under the supervision of Peter Bolhuis. I am still amazed by the straightforwardness with which he faced the problem of rare events, reformulated it in his own way and created Transition Interface Sampling. To me TIS will always stand for *Titus Interface Sampling*.

To my friends

Two persons deserve at least this first paragraph. Titus, jij bent de eerste nederlandse vriend van mij. In vier jaar tijd heb ik geleerd om jouw onafhankelijkheid van het denken, en jouw artistieke, moedige, en aardige creativiteit te waarderen. Jean Pierre, ton premiere pensée est toujours pour les autres, plutôt que pour toi. Tu a eu la force de persévérer dans les moments difficiles. Avec ton exemple et aide, moi aussi. Merci pour ça. Titus, Jean Pierre, even if sometimes you behave a bit clumsy, this is completely negligible with respect to the rest. Your sons will be proud of you. I already am.

Ai miei genitori devo l'inizio e la fine di tutto. Questa tesi è dedicata a loro. Se sono arrivato fin qui lo devo anche agli altri miei parenti, ed Enrica e Tommaso. In particolare lo devo a mio zio Roberto, la sua passione per la fisica, e la sua strabiliante collezione di libri. Alcune persone in questi anni sono rimaste dei veri amici, anche se non erano qui in Olanda. L'onore mio e del Norwold è salvo grazie al mio gruppo di D&D, Alessandro Vito Scuraro, Marco Houdini, Massimo Scotus, Massimo Xarabas, Nicola Aragon, e il nostro unico vero master, Edo. Gli altri amici di vecchia data ringrazio per ricordarsi di sopportarmi ancora, Annalisa e Antonio da Avezzano (originariamente), Annalisa da Francavilla al mare, Daniele e Sandra da Cambridge, Livia e Lorenzo da Roma, Gabriele ormai da 23 anni, i miei cugini e in particolare Martina, Emiliano, Matteo e Marco da quando sono nati. Luca, ora entrambi siamo riusciti a sconfiggere il nostro Ganondorf. L'inizio di questa avventura lo devo ad Andrea Giansanti, amico non solo per la scienza, e ad Alessandra, il cui contributo di allora continua ad essere importante anche adesso.

The *amsterdammers* have been a fantastic series of people. Rubèn, llegamos juntos y juntos pasamos muchas aventuras y problemas, momentos buenos y malos. Cuenta siempre conmigo. Esther, espero poder experimentar pronto tu técnica Alexander. Miriam, siempre me agradeces demasiado por tantas cosas, aquí quiero ser yo el que te dé las gracias. I owe uncountable moments of Kriterion, Palorma, fun, beer, dinners, parties, emails, and more to Annemarieke, Diego, Fatima, Jordi, Jorge, Juan, María, Manu, Manuel, Misha, Nacho, Salvo. Special credits deserves our official photographer Leonid. It is just a pity that this other people could not join us more: Alejandro, Andrea, Tristan (well, maybe later), Javier, José, Marina, Mónica, Nuria, Silke, Viney, recently Nanda, Elena and Simone, and Chiara, Nuria, Juan, Simona from Leiden. Met een aantal mensen heb ik echt leuke uren van volleybal samenge-speld. Onder anderen de Funblockers, Barbara, Hanneke, Jan, Karin, Pieter, Sylvia. Met Artem hoop ik in de toekomst de beach(volley) ervaringen te herhalen. La *kleine*

Italië di Amsterdam è così grande che è stato impossibile conoscerla tutta. Quando arrivai per la prima volta incontrai quelli che sarebbero stati gli amici di tante altre occasioni, Maddalena, Annamaria e Fabrizio, Barbara e Marco, e Luca, che festeggiava il suo compleanno, a casa di Elio, che però non c'era. Ma ci sarebbe stato spesso in seguito. Grazie per avermi più volte guidato (anche letteralmente) a districarmi nella vita (di Amsterdam). Ai tanti ricordi hanno contribuito in seguito Caterina, Luca (Tomassini), Manola, Nicola, e la simbolica ala italiana di AMOLF, Angelo, Beatrice, Chantal, Fabiana, Ivan, Marco J. In zona VU e non, Giovanna e Valentina, Marinella, Alois, Diego, Elena, Ludovica, Sergio, Stefania. I also wish to remember the roeterseiland people, of the present and the past. From the group, Abdon, Arjen, Bastiaan, Christophe, David, Edith, Elske (bedankt om de samenvatting te checken), Jan Willem, Jantien, Jarek, Jocelyne, Jochem, Live, Marieke, Martijn, Menno, Merijn, Ranieri, Rene, Tim, Vincent. Luca for the many (often weekend) office hours. Ik beloof sint (backup) Gooitzen om voorzichtiger te zijn met mijn data. From the chemistry faculty, David, Enrico, Enrico, Fabrizio, Filippo, Fiona, Gadi, Jurriaan, Mauro, Maya, Nicolas, Olja, Paolo, Sandro, Simona, Susana, Tommaso. From physics, David, Gerardo, Liat, Raoul, Srdjan, Vinh, via Stephan to the economists Astrid, Paolo, Pietro, Sebastiano. From the Diemen world, Håkon, Jasper, Evert, Pedro, and in a sense Rossana and Nicandra. Jia, I still hope we reach China one of these days. Elisa, come in questi ultimi nove mesi, spero anche alla promotie di vedervi entrambi.

Undead creatures give special apologies to my D&D players, Alejandro, Beatrice, Ivan, Miriam, and Chinmay, Mark, Rafa. It took a handful of characters, but now you are really my heroes.

A number of people are somehow connected to Amsterdam, for having spent some time here, or wishing to do so. I specially want to remember Céline, Emanuela, Els, Hilary, Mangalika, Katia and Hans, Serena and Gerrit (and Aldo), Veronica, Paola, Martina, Daniela, Eleonora, Maria Maddalena. To my 'first neighbor' Bill I hope we can meet again one of these days. The same holds for Chiara, Antonio, and Vicki.

My admiration for their (scientific) achievements and as a person to Tanja, now in Mainz, and Sofia, now in Sevilla. I thank Elio and Rubèn for being my *paranimfen*, Nacho for the photo of me on the back cover, and Christoph Dellago for inspiring the cover picture.

A todos mis amigos españoles debo, entre otras cosas, la capacidad de escribir estas mismas palabras.

All the people mentioned here, and many more, have been very important for me during these many years. Thanks, without you I probably would not have reached this point.

Last, but not least, I wish to thank all those non-Italians who could spell correctly my name, and apologize for disappointing anyone who thought I am a woman.

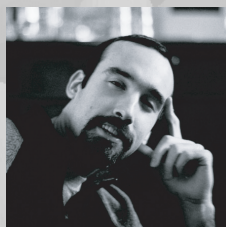
Index

- Algorithm
 - loop-boundary, 105
 - NPH dynamics, 183–186
 - PPTIS, 200–202
 - TIS, 194–200
- Bennett-Chandler procedure, 13
- Biased simulation, 174
- Bond order parameters, 187
- Characteristic function, 8
 - backward/forward, 55
- Classical Nucleation Theory, 177
- Committors, 49
 - and nucleation, 140
 - generalized, 192
- Crossing probability, 59
 - PPTIS long-distance, 83
 - PPTIS one-interface, 83
 - TIS, 61
- Degree of undercooling, 127
- Detailed balance, 153
 - in path ensemble, 29
- Diffusion move, 35
- Dimer model, 68
 - diffusive, 88
- Dividing surface (TST), 10
- Effective positive flux, 58
- Ensemble average, 7
 - constrained, 12
 - weighted, 59
- Equilibrium constant, 84
- Flux calculation, 61
- Flux relation, 167
- Fokker-Planck operator, 153
- Free energy, 11
 - geometrically invariant, 23
- Geissler test, 51
- Implementation
 - free energy in PPTIS, 105
 - PPTIS, 85, 194–202
 - TIS, 61, 194–202
 - TPS, 29
 - TST-BC procedure, 14
- Inertia tensor, 43
- Interfaces, (PP)TIS, 54
- Langevin dynamics, 8
- Lennard-Jones
 - phase diagram, 129
 - potential, 43
- Loop-boundary points, 103
- Mean first passage time, 154
- Mean residence time, 9
- Memory loss function, 87
- Nucleation barrier, 177
- Nucleation cluster
 - analysis, 137
 - identification, 130
- Nucleation mechanism (LJ), 136
- Nucleation problem, 125
- Number of connections per particle, 130
- Onedimensional models
 - bistable, 159
 - diffusive, 17
 - hopping, 81
- Order parameter, 28
- Overall state, 55

- Parallel path swapping, 98
- Partial Path TIS, 80
- Path, 27
- Path reversal, 35
- Path sampling
 - PPTIS, 85
 - TIS, 62
 - TPS, 29
- Pathlength, 27
- Potential energy surface, 42
- Radius of gyration tensor, 137
- Random telegraph, 4
- Rare events, 3
- Rate constant, 152–160
 - for nucleation (CNT), 181
 - for nucleation, by (PP)TIS, 134
 - for random telegraph, 7
 - in BC procedure, 13
 - in PPTIS, 84
 - in RRKM theory, 166
 - in TIS, 56
 - in TPS, 38
 - in TST, 9
- Reaction coordinate, 15
- Reactive flux, 13
- Recrossings, 17
- RRKM theory, 163
- Scaling of Interface Sampling, 111
- Separation of timescales, 3
- Shifting move, 34
- Shooting move, 29
- Simple PPTIS, 92
- Spectral gap, 154
- Sphere, multidimensional integral, 163
- Stable state, 8
- Tail corrections, 184
- Time slice, 27
- Transition Interface Sampling, 54
- Transition path ensemble, 27
- Transition Path Sampling, 26
- Transition State Ensemble, 50
- Transition State Theory, 8
 - variational, 22
- Transmission coefficient, 16
 - effective positive flux, 22, 119
 - improved BC expressions, 19
 - method comparison, 121
 - Ruiz-Montero method, 20
 - unnormalized, 14
- Umbrella sampling, 39
- WCA potential, 68
- Windows
 - in PPTIS, 86, 103
 - in TIS, 65
 - in Umbrella sampling, 39
- Zeldovich factor, 182
- Zone system, 65

So if the event is rare, why do we care? In computer simulations of atomic systems, even present day machines need about 3 years to observe an event that occurs once a μs . Rare events then are not the passage of the Halley comet, but, for example, protein folding, chemical reactions, and nucleation during phase transitions. This thesis is concerned with rare events and the special techniques used to simulate them. We present a new method, named Transition Interface Sampling (TIS), and its diffusive version, Partial Path TIS. Both methods are based on the efficient computation of fluxes through multidimensional hypersurfaces in the phase space, the interfaces. We then successfully applied the methods to a case of crystal nucleation in a simple fluid.

Daniele Moroni was born in Rome, Italy, on 19 March 1976. He graduated in 2000 in Physics at the University "La Sapienza", with the thesis "Dynamics and thermodynamics of a variable-range interacting XY model". Eventually he started a PhD at the group for Computational Chemistry and Physics, Van 't Hoff Institute for Molecular Sciences, Universiteit van Amsterdam.



He defends his PhD on 28 April 2005. Counting from the very beginning it took him about 255163 hours to achieve this, which is comparable to the CPU time the group has available on TERAS, the Dutch national supercomputer. However, he would like to think that in this life there is more than just computer-time.

228 pages, 85 figures

Cover: *Potential Minimum*. The picture was inspired by the 'Dellago logo' for The Berkeley Mini Statistical Mechanics Meeting.