

# SIMULATION OF CLASSICAL AND QUANTUM ACTIVATED PROCESSES IN THE CONDENSED PHASE

GIOVANNI CICCOTTI

*INFM - Dipartimento di Fisica, Università di Roma "La Sapienza"  
00185 Roma, Italy*

E-mail: ciccotti@roma1.infn.it

MAURO FERRARIO

*INFM - Dipartimento di Fisica, Università di Modena"  
41100 Modena, Italy*

E-mail: ferrario@unimo.it

DANIEL LARIA

*Departamento Química de Reactores, Comisión Nacional de Energía Atómica  
(1429)Buenos Aires, Argentina*

E-mail: dhlaria@cnea.edu.ar

and

RAYMOND KAPRAL

*Chemical Physics Theory Group, Department of Chemistry, University of Toronto  
Toronto, Ontario M5S 1A1, Canada*

E-mail: rkapral@gatto.chem.utoronto.ca

## ABSTRACT

A computationally efficient molecular dynamics approach based on the method of holonomic constraints is described with reference to the task of estimating the rate constants of rare events that occur by activated processes. The system is constrained at "bottleneck" regions on a general many-body reaction coordinate and molecular dynamics trajectories sample phase space according to a biased configurational distribution. Averages can be taken from such a new, *Blue Moon*, ensemble with suitable reweighting and correct momentum distribution sampling, to study rare events. Applications to a variety of classical and quantum activated processes in the condensed phase are discussed together with the implementation of a highly parallel algorithm based on message passing.

## 1. Introduction

An activated process is a transition between stable states separated by a region of low probability and can be viewed as the crossing of a free energy barrier. When the free energy barrier is much higher than  $k_B T$  the crossing is an infrequent event which cannot be simulated using conventional techniques. If the activated process can be described by a phenomenological rate equation, then the associated rate constants can be calculated within the reactive-flux correlation function formalism<sup>1,2</sup>.

Classically transitions between stable states in a many-body dynamical system can be often described by a suitable function of the configuration space called the

reaction coordinate or progress variable, denoted by  $\xi(\mathbf{r})$ , where  $\mathbf{r}$  represents the set of coordinates of all the degrees of freedom of the system. The progress variable  $\xi$  can be thought of as a collective variable whose values describe the state of advancement of the reaction when going from one stable state to the other. The probability density to find the system in any point of the configuration space such that  $\xi(\mathbf{r})$  equals a prescribed value  $\xi'$  is :

$$P(\xi = \xi') = \langle \delta(\xi(\mathbf{r}) - \xi') \rangle \equiv C e^{-\beta W(\xi')} \quad (1)$$

where  $\langle \dots \rangle$  is the canonical ensemble average,  $\beta = \frac{1}{k_B T}$  and the last equality is a definition of the potential of mean force  $W(\xi)$  associated with the reaction coordinate.  $W(\xi_1) - W(\xi_2)$  expresses the reversible work needed to take the system from some reference value  $\xi_2$  of the variable  $\xi$  to  $\xi_1$ . The constant  $C$  is determined by the normalization condition on  $P(\xi)$ . If  $\xi^\ddagger$  is a local maximum for  $W(\xi)$  we can identify the hyper-surface  $\xi(\mathbf{r}) = \xi^\ddagger$  as the transition region and  $P(\xi^\ddagger)$  is one of the important quantities we wish to calculate. Of course special methods have to be used to determine the probability of a highly improbable value of  $\xi$ . They include special sampling schemes; for example, umbrella sampling<sup>3</sup>, to compute mean force potential differences  $W(\xi') - W(\xi)$  or the mean force on  $\xi$ ,  $F(\xi) = -\frac{dW(\xi)}{d\xi}$ . The rate constant in turn can be computed, knowing this probability, as the product of the  $P(\xi^\ddagger)$  and the average fraction of trajectories which, starting at the transition point  $\xi^\ddagger$ , end in the product region. The classical treatment of chemical reactions we have just described was first introduced by Keck<sup>4,5</sup> and Anderson<sup>6</sup> to treat gas phase chemical reactions and later applied by Bennett<sup>7,8</sup> and Chandler<sup>1</sup> to the treatment of condensed-phase rate processes.

The extension of the classical treatment of a chemical reaction to quantum, activated transition processes in condensed media is an important goal since in many cases, such as proton or electron transfer reactions, quantum effects may become important. When the quantum character of the reaction under study can be reduced to a few degrees of freedom immersed in a classical bath it is possible, in some approximation, to treat the reaction in almost classical terms. The properties that have to do with the energetic of the reaction can be obtained by using the Feynman path-integral formulation of quantum statistical mechanics, while in the limit of adiabatic dynamics (i.e. when the time evolution of the quantum degrees of freedom is confined to a single Born-Oppenheimer surface) the dynamical properties follow by computing the Schrödinger equation for the quantum particle (which depends on the instantaneous solvent configuration) and Newton's equations for the solvent (which depend on the quantum state). The approach is computationally intensive but straightforward.

In this chapter we discuss our approach to the classical problem and its adaptation to simple mixed quantum-classical cases. In Section 2 we recall the reactive flux correlation formulas. In Section 3 we present the approach to free energy and reaction rate calculations for activated processes known as the *Blue Moon Ensemble* approach.

In Section 4 we illustrate the approach with a few simple applications for classical systems. In Section 5 we extend the approach to quantum activated processes, in particular to charge transfer reactions in a polar solvent. Section 6 deals with diffusion controlled reactions. In the Appendix we discuss a parallel implementation of a molecular dynamics code for *Blue Moon* simulations using the Parallel Virtual Machine package (PVM)<sup>9</sup>. An explicit fortran code is given at the end of the chapter.

## 2. Reactive Flux Correlation Formulas

In this section we dust off the cobweb-encrusted theory for autocorrelation function expressions for the rate constant of a reaction. These expressions were first derived by Yamamoto<sup>10</sup> in 1960 and have continued to fascinate researchers in the area as is evidenced by the number of times they have been re-derived and reinterpreted in the literature<sup>2,11,12,13,14</sup>. Part of this stimulus comes from the fact that these formulas form the basis of an efficient algorithm for the computation of the rate constant for activated processes, and this has motivated the presentation given below.

### 2.1. Derivation of chemical rate laws

Consider a simple chemical reaction of the form



taking place in a solvent  $S$ . The mass action rate law for the densities of dilutely-dispersed, reacting species  $A$  and  $B$ ,  $\bar{n}_A$  and  $\bar{n}_B$ , respectively, reads

$$\frac{d\bar{n}_A(t)}{dt} = -k_f\bar{n}_A(t) + k_r\bar{n}_B(t). \quad (3)$$

Using the fact that the chemical potential of species  $\alpha$  is  $\mu_\alpha = \mu_\alpha^{(0)} + \beta^{-1} \ln \bar{n}_\alpha$  where  $\alpha = A, B$  and the expression for the equilibrium constant  $K_{eq} = \bar{n}_B^{eq}/\bar{n}_A^{eq} = e^{-\beta(\mu_B^{(0)} - \mu_A^{(0)})} = k_f/k_r$ , this equation may also be written in the form

$$\frac{d\bar{n}_A(t)}{dt} = -k_f\bar{n}_A(t) \left(1 - e^{-\beta\mathcal{A}(t)}\right). \quad (4)$$

This form makes evident the fact that the driving force of the reaction is the chemical affinity  $\mathcal{A}(t) = \mu_A(t) - \mu_B(t)$ <sup>15,16</sup>. Finally, noting that the deviations of the  $A$  and  $B$  species densities are related by

$$\delta\bar{n}_A = \bar{n}_A - \bar{n}_A^{eq} = -\delta\bar{n}_B = -(\bar{n}_B - \bar{n}_B^{eq}), \quad (5)$$

we may also write the rate law as

$$\frac{d\delta\bar{n}_A(t)}{dt} = -(k_f + k_r)\delta\bar{n}_A(t) = -k_f(1 + K_{eq}^{-1})\delta\bar{n}_A(t). \quad (6)$$

We wish to derive this equation from the microscopic equations of motion and, in the process, obtain autocorrelation function expressions for the reaction rate constants. In order to begin this analysis it is first necessary to introduce operators that characterize the  $A$  and  $B$  chemical species. At this point it is only necessary to consider formal expressions for such operators, which we denote by  $\hat{N}_\alpha$ ,  $\alpha = A, B$ ; explicit expressions will be given below when calculations are carried out.

We suppose the system lies close to equilibrium and let  $\delta\hat{N}_\alpha = \hat{N}_\alpha - \hat{N}_\alpha^{eq}$  denote the difference between the number operator for species  $\alpha$  and its value at equilibrium. The quantum mechanical equation of motion for  $\delta\hat{N}_\alpha(t)$  is the Heisenberg equation,

$$\frac{d\delta\hat{N}_\alpha(t)}{dt} = \frac{i}{\hbar} [\hat{H}, \delta\hat{N}_\alpha(t)] \equiv i\hat{\mathcal{L}}\delta\hat{N}_\alpha(t), \quad (7)$$

where  $\hat{H}$  is the Hamiltonian of the system. Eq. (7) can be cast into a generalized version of Eq. (6)<sup>17</sup> using projection operator methods<sup>18</sup>.

Let  $\mathcal{P}_\alpha$  be a projection operator onto the number operator of species  $\alpha$ :

$$\mathcal{P}_\alpha \hat{\mathcal{O}} = (\hat{\mathcal{O}}, \delta\hat{N}_\alpha^*) (\delta\hat{N}_\alpha, \delta\hat{N}_\alpha^*)^{-1} \delta\hat{N}_\alpha, \quad (8)$$

where the scalar product is defined as

$$\begin{aligned} (\hat{C}, \hat{D}^*) &= \frac{1}{\beta} \int_0^\beta d\lambda \text{Tr} e^{\lambda\hat{H}} \hat{C} e^{-\lambda\hat{H}} \hat{D}^* \hat{\rho}_e \\ &= \frac{1}{\beta} \int_0^\beta d\lambda \langle e^{\lambda\hat{H}} \hat{C} e^{-\lambda\hat{H}} \hat{D}^* \rangle, \end{aligned} \quad (9)$$

and the angle bracket signifies an average over the equilibrium density matrix in the canonical ensemble,

$$\hat{\rho}_e = \frac{e^{-\beta\hat{H}}}{\text{Tr} e^{-\beta\hat{H}}}. \quad (10)$$

We begin by writing Eq. (7) as

$$\frac{d\delta\hat{N}_\alpha(t)}{dt} = e^{i\hat{\mathcal{L}}t} i\hat{\mathcal{L}}\delta\hat{N}_\alpha, \quad (11)$$

and defining the projection operator  $\mathcal{P} = \mathcal{P}_A + \mathcal{P}_B$  onto the  $A$  and  $B$  species number operators. The complement of  $\mathcal{P}$  is  $\mathcal{Q} = 1 - \mathcal{P}$ . Substituting the operator identity (see, for instance McQuarrie<sup>19</sup>)

$$e^{i\hat{\mathcal{L}}t} = e^{i\mathcal{Q}\hat{\mathcal{L}}t} + \int_0^t d\tau e^{i\hat{\mathcal{L}}(t-\tau)} \mathcal{P} i\hat{\mathcal{L}} e^{i\mathcal{Q}\hat{\mathcal{L}}\tau}, \quad (12)$$

into Eq. (11) we obtain

$$\frac{d\delta\hat{N}_A}{dt} = - \int_0^t d\tau K_f(\tau) \delta\hat{N}_A(t-\tau) + \int_0^t d\tau K_r(\tau) \delta\hat{N}_B(t-\tau) + R_A(t), \quad (13)$$

where the rate kernels are defined as

$$\begin{aligned} K_f(t) &= (e^{i\mathcal{Q}\hat{\mathcal{L}}t} i\hat{\mathcal{L}}\delta\hat{N}_A, i\hat{\mathcal{L}}\delta\hat{N}_A^*)(\delta\hat{N}_A, \delta\hat{N}_A^*)^{-1} \\ K_r(t) &= -(e^{i\mathcal{Q}\hat{\mathcal{L}}t} i\hat{\mathcal{L}}\delta\hat{N}_A, i\hat{\mathcal{L}}\delta\hat{N}_B^*)(\delta\hat{N}_B, \delta\hat{N}_B^*)^{-1}, \end{aligned}$$

and

$$R_A(t) = e^{i\mathcal{Q}\hat{\mathcal{L}}t} i\hat{\mathcal{L}}\delta\hat{N}_A, \quad (14)$$

is the random reactive flux. The correlation functions  $(\delta\hat{N}_\alpha, \delta\hat{N}_\alpha^*)$  may be evaluated using equilibrium fluctuation theory to yield<sup>20</sup>

$$(\delta\hat{N}_\alpha, \delta\hat{N}_\alpha^*) = \beta^{-1} \left( \frac{\partial\mu_\alpha}{\partial\bar{n}_\alpha} \right)_{eq}^{-1} = \bar{n}_\alpha^{eq}. \quad (15)$$

Using this result, together with  $i\hat{\mathcal{L}}\delta\hat{N}_A = -i\hat{\mathcal{L}}\delta\hat{N}_B$ , the generalized rate law may also be written as

$$\frac{d\delta\hat{N}_A(t)}{dt} = - \int_0^t d\tau K(\tau)\delta\hat{N}_A(t-\tau) + R_A(t), \quad (16)$$

where

$$K(t) = (1 + K_{eq}^{-1})(e^{i\mathcal{Q}\hat{\mathcal{L}}t} i\hat{\mathcal{L}}\delta\hat{N}_A, i\hat{\mathcal{L}}\delta\hat{N}_A^*)(\delta\hat{N}_A, \delta\hat{N}_A^*)^{-1}. \quad (17)$$

In this derivation equilibrium fluctuation theory was used to compute the species number fluctuations.

## 2.2. Derivation of the macroscopic law

Since the affinity  $\mathcal{A}$  is the thermodynamic driving force that couples to the species numbers, we consider a local-equilibrium density matrix corresponding to a generalized canonical ensemble where the system Hamiltonian is augmented with the contribution  $-\mathcal{A}\delta\hat{N}_A^*$ :

$$\rho(0) = \frac{e^{-\beta(\hat{H} - \mathcal{A}\delta\hat{N}_A^*)}}{\text{Tr} e^{-\beta(\hat{H} - \mathcal{A}\delta\hat{N}_A^*)}}. \quad (18)$$

The macroscopic species number densities may be associated with averages of the microscopic quantities over this initial non-equilibrium density matrix:

$$\delta\bar{n}_\alpha = \text{Tr} \rho(0)\delta\hat{N}_\alpha. \quad (19)$$

Provided the corresponding average of the random reactive flux vanishes, this procedure will yield a generalized (with memory) form of the macroscopic chemical rate law. Making use of the operator identity Eq. (12) in the form

$$e^{-\beta(\hat{H} - \mathcal{A}\delta\hat{N}_A^*)} = e^{-\beta\hat{H}} + \int_0^\beta d\lambda e^{-(\hat{H} - \mathcal{A}\delta\hat{N}_A^*)(\beta-\lambda)} \mathcal{A}\delta\hat{N}_A^* e^{-\hat{H}\lambda}, \quad (20)$$

for small deviations from complete equilibrium we may expand this expression for the density matrix to first order in the affinity to obtain<sup>18</sup>

$$\rho(0) = \hat{\rho}_e \left( 1 + \int_0^\beta d\lambda e^{\lambda \hat{H}} \delta \hat{N}_A^* e^{-\lambda \hat{H}} \mathcal{A} \right) . \quad (21)$$

The average of the random force over this ensemble is given by

$$\bar{R}_A(t) = Tr \rho(0) R_A(t) = \int_0^\beta d\lambda Tr R_A(t) \delta \hat{N}_A^* (-i\hbar\lambda) \hat{\rho}_e \mathcal{A} . \quad (22)$$

In writing this equation we have used the fact that  $Tr \hat{\rho}_e R_A(t) = 0$  since  $i\hat{\mathcal{L}}e^{-\beta\hat{H}} = 0$ . The condition that  $\bar{R}_A(t)$  vanishes is just the condition that the random reactive flux correlations with the particle number fluctuations vanish. Thus, imposing the previous condition, the mean evolution of the density variables follows the generalized macroscopic rate law,

$$\frac{d\delta\bar{n}_A(t)}{dt} = - \int_0^t d\tau K(\tau) \delta\bar{n}_A(t - \tau) . \quad (23)$$

The standard chemical rate law follows from the assumption of rapid decay of the memory kernels, a fact that must be verified in specific applications, and the rate constant is identified with the zero frequency limit of the rate kernel:

$$k = \int_0^\infty d\tau K(\tau) . \quad (24)$$

It is also convenient in some contexts to define a time-dependent rate constant by

$$k(t) = \int_0^t d\tau K(\tau) , \quad (25)$$

whose properties will be analyzed in detail below.

### 2.3. Properties of projected rate kernel

The rate kernel that enters in the generalized chemical rate law Eq. (23) is given in Eq. (17). The appearance of projected dynamics in its definition is both crucial for its structure and leads to complications in its computation. For this this reason it is necessary to examine the consequences of replacing projected dynamics by ordinary time evolution. Let

$$K_u(t) = (1 + K_{eq}^{-1})(e^{i\hat{\mathcal{L}}t} i\hat{\mathcal{L}}\delta\hat{N}_A, i\hat{\mathcal{L}}\delta\hat{N}_A^*)(\delta\hat{N}_A, \delta\hat{N}_A^*)^{-1} , \quad (26)$$

be the corresponding rate kernel with ordinary time evolution.

It is well known that the projected rate kernel that defines the rate constant of the reaction has very different decay properties from the analogous quantity where the

projected dynamics is replaced by ordinary dynamics. It is a simple matter to derive the relation between the rate kernels  $K_u(t)$  and  $K(t)$ . Taking the scalar product of Eq. (16) with  $\delta\hat{N}_A^*$ , multiplying by  $(\delta\hat{N}_A, \delta\hat{N}_A^*)^{-1}$  and using the properties of the scalar product to shift the position of the time derivative, we obtain

$$-\frac{(\delta\hat{N}_A(t), \delta\dot{\hat{N}}_A^*)}{(\delta\hat{N}_A, \delta\hat{N}_A^*)} = -\int_0^t d\tau K(\tau) \frac{(\delta\hat{N}_A(t-\tau), \delta\hat{N}_A^*)}{(\delta\hat{N}_A, \delta\hat{N}_A^*)}. \quad (27)$$

Differentiation of this equation with respect to time and use of the Leibnitz's rule yields

$$-K_u(t) = -K(t) - \int_0^t d\tau K(\tau) \frac{(\delta\dot{\hat{N}}_A(t-\tau), \delta\hat{N}_A^*)}{(\delta\hat{N}_A, \delta\hat{N}_A^*)}. \quad (28)$$

Laplace transformation gives

$$-\hat{K}_u(s) = -\hat{K}(s) + \frac{1}{s}\hat{K}(s)\hat{K}_u(s), \quad (29)$$

which may be rearranged to obtain<sup>21</sup>

$$\hat{K}_u(s) = \frac{s\hat{K}(s)}{\hat{K}(s) + s}, \quad (30)$$

where  $\hat{K}_u(s)$  and  $\hat{K}(s)$  are the Laplace transforms of the rate kernels with unprojected dynamics and projected dynamics, respectively.

The different properties of these two rate kernels can now be understood from the following considerations. The projected rate kernel is constructed so that the dynamics of the (assumed) slow species variables do not contribute to its decay. If the species variables are the only slow variables one may assume that  $K(t)$  will decay on some microscopic time scale  $t_{mic}$ . As such, its infinite time integral,  $\lim_{s \rightarrow 0} \hat{K}(s)$ , is some well-defined constant. If this is so, then it is clear from the  $s \rightarrow 0$  limit of Eq. (30) that  $\hat{K}_u(0)$  must vanish, implying that the infinite time integral of  $K_u(t)$  vanishes. Thus, this quantity has positive and negative time portions that exactly cancel. In terms of the time-dependent rate kernel defined in Eq. (25) we see that  $\lim_{t \rightarrow \infty} k(t) = k$ , the macroscopic rate constant, while the same limit of the analogous quantity for ordinary dynamics is,  $\lim_{t \rightarrow \infty} k_u(t) = 0$ .

These considerations lead one to a discussion of the plateau value problem. Assuming there is time scale separation between the decay of the species numbers and other microscopic rates in the system we expect the decay of the rate kernel  $K_u(t)$  to be similar to that of  $K(t)$  for times  $t$  such that  $0 < t < t_{mic}$  so that the time-dependent rate kernel will exhibit a plateau for times  $t^*$  such that  $t_{mic} < t^* < t_{chem}$  where  $t_{mic}$  and  $t_{chem}$  are microscopic and chemical relaxation times, respectively. The rate constant of the reaction ( $k = k_f + k_r$ ) may then be approximately identified with

$$k \approx k_u(t^*) \equiv \int_0^{t^*} d\tau K_u(\tau), \quad (31)$$

which is nearly independent of the time  $t^*$ . Of course, the rate constant may be rigorously identified as the infinite time integral of the projected rate kernel as noted above.

#### 2.4. Structure of the rate kernel

Consider the rate kernel in Eq. (25) with  $K(\tau)$  replaced by  $K_u(\tau)$ . This may be written in the form

$$k_u(t) = \left(\frac{1}{\bar{n}_A^{eq}} + \frac{1}{\bar{n}_B^{eq}}\right)\beta^{-1} \int_0^\beta d\lambda \text{Tr} \delta\hat{N}_A(t - i\hbar\lambda)\delta\dot{\hat{N}}_A\hat{\rho}_e. \quad (32)$$

Carrying out the integration over  $\lambda$ , this equation may be rewritten as,

$$k_u(t) = \left(\frac{1}{\bar{n}_A^{eq}} + \frac{1}{\bar{n}_B^{eq}}\right)\frac{1}{i\hbar\beta}\text{Tr} [\delta\hat{N}_A(t), \delta\hat{N}_A]\hat{\rho}_e. \quad (33)$$

From this expression it is clear that  $k_u(t = 0) = 0$  since the commutator vanishes at  $t = 0$ . The corresponding classical expression for the rate kernel follows from the replacement of the commutator by the Poisson bracket and the trace over the quantum density matrix by the classical canonical ensemble average:

$$k_u(t) = \left(\frac{1}{\bar{n}_A^{eq}} + \frac{1}{\bar{n}_B^{eq}}\right)\frac{1}{\beta}\langle\{\delta N_A(t), \delta N_A\}\rangle, \quad (34)$$

which may also be expressed in the form

$$k_u(t) = \left(\frac{1}{\bar{n}_A^{eq}} + \frac{1}{\bar{n}_B^{eq}}\right)\langle\dot{N}_A\delta N_A(t)\rangle, \quad (35)$$

Equations (33), (34) and (35) constitute the the reactive flux rate coefficient expressions that will be used in the subsequent calculations.

To be more explicit in the classical case let us now give a microscopic description of the reaction. This entails a measure of the progress of the reaction. In general this is achieved by defining a reaction coordinate  $\xi$ , i.e. a possibly complicated function of the coordinates of all the particles, such that the reactant and product regions can be defined as the regions where  $\xi$  is, respectively, less than or greater than a preset value,  $\xi^\dagger$ . The value  $\xi^\dagger$  characterizes the transition state of the reaction. In this case we may say that the species variables are defined by the populations in the left and right sides of the transition state value thus, for example,  $N_A(\xi) = \theta(\xi^\dagger - \xi)$ , where  $\theta(\cdot)$  is the Heaviside function. Using this definition the formula (35) can be written as

$$k_u(t) = \left(\frac{1}{\bar{n}_A^{eq}} + \frac{1}{\bar{n}_B^{eq}}\right)\langle-\dot{\xi}\delta(\xi - \xi^\dagger)\theta(\xi^\dagger - \xi(t))\rangle = \left(\frac{1}{\bar{n}_A^{eq}} + \frac{1}{\bar{n}_B^{eq}}\right)\langle\dot{\xi}\delta(\xi - \xi^\dagger)\theta(\xi(t) - \xi^\dagger)\rangle. \quad (36)$$

When written in this form one can see that the classical rate kernel  $K_u(t)$  (or  $K(t)$ ) has a singular contribution for  $t = 0^+$ . For small positive times  $t \rightarrow 0^+$  we may write  $\xi(t) = \xi^\dagger + \dot{\xi}t$ , so that  $\theta(\xi(t) - \xi^\dagger) = \theta(\dot{\xi}t) = \theta(\dot{\xi})$ . Thus, we find

$$k^{TST} = k_u(t = 0^+) = \left( \frac{1}{\bar{n}_A^{eq}} + \frac{1}{\bar{n}_B^{eq}} \right) \langle \dot{\xi} \delta(\xi - \xi^\dagger) \theta(\dot{\xi}) \rangle. \quad (37)$$

Note that for the classical expression given above the  $t \rightarrow 0^+$  limit gives the transition state theory estimate of the rate constant<sup>1</sup>. Finally, using the transition state theory expression, we may write the time dependent rate constant  $k_u(t)$  in terms of the time dependent transmission coefficient  $\kappa(t)$  as

$$k_u(t) = k^{TST} \kappa(t), \quad (38)$$

which measures deviations due to dynamical effects associated with re-crossings of the species dividing surface.

Alternatively we can separate the static and dynamic contributions to the rate constant by multiplying and dividing the right hand side of Eq. (36) by  $\langle \delta(\xi - \xi^\dagger) \rangle$ . We then obtain

$$k_u(t) = \left( \frac{1}{\bar{n}_A^{eq}} + \frac{1}{\bar{n}_B^{eq}} \right) \left[ \frac{\langle \dot{\xi} \delta(\xi - \xi^\dagger) \theta(\xi(t) - \xi^\dagger) \rangle}{\langle \delta(\xi - \xi^\dagger) \rangle} \right] \langle \delta(\xi - \xi^\dagger) \rangle. \quad (39)$$

The last term on the right hand side is the probability density to find the system at the transition state while the ratio in the square brackets is a conditional average, namely the average of the product  $\dot{\xi} \theta(\xi(t) - \xi^\dagger)$  given that  $\xi = \xi^\dagger$ .

### 3. Blue Moon Ensemble

By simulation it is difficult to determine properties that depend on rare events. In the particular case of rare but fast events it is possible to circumvent the difficulty. For example to estimate the transition rate a computationally efficient way is to choose initial states localized at the hyper-surface that separates the stable regions. With such states one can compute the conditional average indicated in square brackets in Eq. (39). In addition one has to compute the equilibrium probability density for the system to be at the transition state. A closely related idea, based on the separation indicated in Eq. (38), was introduced to treat gas phase chemical reactions<sup>4,6</sup> and later applied to condensed phase rate processes<sup>7,1</sup>. In this case one computes the equilibrium one way flux across the transition state, i.e. the Transition State Theory value of the rate constant, and then the transmission coefficient.

Assuming that the activated process is described by the reaction coordinate  $\xi(\mathbf{r})$ , where  $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N)$  represents the  $3N$  Cartesian coordinates of the system composed of  $N$  particles with masses  $m_i$  ( $i = 1, \dots, N$ ) and, for simplicity, without internal structure, we want to compute ensemble averages of static or time dependent

dynamical variables containing a delta function that localizes the system initially at a prescribed (rare) value of  $\xi(\mathbf{r}) = \xi'$ . The generalization to the case of molecules is straightforward unless constraints are used to freeze some of the intramolecular degrees of freedom<sup>22</sup>.

The presence of the delta function is suggestive of the ensemble which describes our system when the reaction coordinate is constrained to the value prescribed by the delta function. To see the analogy let us derive the statistical mechanics in Cartesian coordinates of a system subjected to a holonomic constraint. Usually when there are constraints one introduces a set of generalized coordinates  $\mathbf{q}$  and conjugate momenta  $\mathbf{p}^q$  such that  $\mathbf{r} = \mathbf{r}(\mathbf{q})$ . In general it is not possible to invert this relation since there are more  $\mathbf{r}$  coordinates than  $\mathbf{q}$  coordinates. However by including the expression(s) for the constraint(s) one has extra generalized coordinate(s) and one recovers the one-to-one correspondence  $\mathbf{r} = \mathbf{r}(\mathbf{q}, \text{constraint expression}(s))$ .

The statistical mechanics is easily formulated in terms of the generalized coordinates  $\mathbf{q}$ . However it may be useful to have the equivalent formulation of the statistical mechanics in terms of the original Cartesian coordinates. We now derive this expression for our  $N$ -particle system subjected to the holonomic constraints  $\sigma(\mathbf{r}) = \xi(\mathbf{r}) - \xi' = 0$ . The dynamics of the system is described in Cartesian coordinates by the Lagrangian

$$\mathcal{L}(\mathbf{r}, \dot{\mathbf{r}}) = K(\dot{\mathbf{r}}) - V(\mathbf{r}) = \sum_{i=1}^N \frac{1}{2} m_i \dot{\mathbf{r}}_i^2 - V(\mathbf{r}) , \quad (40)$$

to which we add the constraints  $\sigma = 0$ . The corresponding equations of motion are the Lagrange equations of first kind<sup>23</sup>. The set of  $3N - 1$  generalized coordinates  $\mathbf{q}$  plus  $\sigma$  can be taken as a new set of equivalent coordinates denoted collectively by  $\mathbf{u}$ . We have

$$\mathbf{r} = \mathbf{r}(\mathbf{q}, \sigma) = \mathbf{r}(\mathbf{u}) , \quad (41)$$

or

$$\mathbf{u} = \mathbf{u}(\mathbf{r}) . \quad (42)$$

In the new variables the Lagrangian is given by

$$\mathcal{L}'(\mathbf{u}, \dot{\mathbf{u}}) = \frac{1}{2} \dot{\mathbf{u}}^T \mathbf{M} \dot{\mathbf{u}} - V'(\mathbf{u}) , \quad (43)$$

where  $\mathbf{u}^T$  is the transpose of vector  $\mathbf{u}$  and  $\mathbf{M}$  is the metric matrix with elements given by

$$M_{\mu\nu} = \sum_{i=1}^N m_i \frac{\partial \mathbf{r}_i}{\partial u_\mu} \cdot \frac{\partial \mathbf{r}_i}{\partial u_\nu} . \quad (44)$$

The Lagrangian of the constrained motion is easily obtained by putting  $\sigma = \dot{\sigma} = 0$ , i.e.

$$\mathcal{L}_c(\mathbf{q}, \dot{\mathbf{q}}) \equiv \mathcal{L}'(\mathbf{q}, \sigma = 0, \dot{\mathbf{q}}, \dot{\sigma} = 0) . \quad (45)$$

To derive the statistical mechanical ensemble we need the Hamiltonian description of the dynamical system. The Hamiltonian in  $\mathbf{u}$  coordinates is given by

$$\mathcal{H}'(\mathbf{u}, \mathbf{p}^u) = \frac{1}{2} \mathbf{p}^{uT} \mathbf{M}^{-1} \mathbf{p}^u + V'(\mathbf{u}) , \quad (46)$$

where

$$\mathbf{p}^u = \frac{\partial \mathcal{L}'}{\partial \dot{\mathbf{u}}} = \mathbf{M} \dot{\mathbf{u}} , \quad (47)$$

and the inverse of the metric matrix  $\mathbf{M}^{-1}$  can be written explicitly as

$$(M^{-1})_{\mu\nu} = \sum_{i=1}^N \frac{1}{m_i} \frac{\partial u_\mu}{\partial \mathbf{r}_i} \cdot \frac{\partial u_\nu}{\partial \mathbf{r}_i} . \quad (48)$$

To obtain the constrained motion we have to compute the Hamiltonian at  $\sigma = 0$  and  $p^\sigma$  satisfying the constraints  $\sigma = \dot{\sigma} = 0$ . Since

$$\dot{\sigma} = \left( \mathbf{M}^{-1} \mathbf{p}^u \right)_{3N} \equiv \mathbf{E} \mathbf{p}^q + Z p^\sigma , \quad (49)$$

where  $\mathbf{E}$  and  $Z$  are suitable submatrices of  $bfM^{-1}$ , the condition  $\dot{\sigma} = 0$  yields  $p^\sigma = -\tilde{Z}^{-1} \tilde{\mathbf{E}} \mathbf{p}^q$  where  $\tilde{\phantom{x}}$  means that the matrices have to be evaluated at  $\sigma = 0$ . Now we have for the constrained Hamiltonian

$$\mathcal{H}_c(\mathbf{q}, \mathbf{p}^q) \equiv \mathcal{H}'(\mathbf{q}, \sigma = 0, \mathbf{p}^q, p^\sigma = -\tilde{Z}^{-1} \tilde{\mathbf{E}} \mathbf{p}^q) . \quad (50)$$

Notice that Eq. (49) implies that

$$p^\sigma + Z^{-1} \mathbf{E} \mathbf{p}^q = Z^{-1} \dot{\sigma} . \quad (51)$$

Calling  $\rho_\xi(\mathbf{q}, \mathbf{p}^q)$  the probability density corresponding to any possible ensemble for the constrained dynamical system, we have

$$\begin{aligned} \rho_c(\mathbf{q}, \mathbf{p}^q) d\mathbf{q} d\mathbf{p}^q &= \rho'(\mathbf{u}, \mathbf{p}^u) \delta(\sigma) \delta(p^\sigma + Z^{-1} \mathbf{E} \mathbf{p}^q) d\mathbf{u} d\mathbf{p}^u \\ &= \rho(\mathbf{r}, \mathbf{p}^r) \delta(\sigma) \delta(Z^{-1} \dot{\sigma}) d\mathbf{r} d\mathbf{p}^r \\ &\equiv \rho_\xi(\mathbf{r}, \mathbf{p}^r) d\mathbf{r} d\mathbf{p}^r , \end{aligned} \quad (52)$$

where in the second line of Eq. (52) we have used the fact that the variable transformation  $(\mathbf{u}, \mathbf{p}^u) \leftrightarrow (\mathbf{r}, \mathbf{p}^r)$  is a canonical, volume conserving, transformation, since it is a point transformation<sup>24</sup>.

For our purposes it is convenient to rewrite this probability as

$$\rho_\xi(\mathbf{r}, \mathbf{p}^r) = \rho_\xi^r(\mathbf{r}) \rho_\xi^p(\mathbf{p}^r | \mathbf{r}) , \quad (53)$$

where the configurational probability density,  $\rho_\xi^r(\mathbf{r})$ , is obtained by performing the integration over momenta of the full probability density and the conditional probability density of the momenta given the configuration,  $\rho_\xi^p(\mathbf{p}^r | \mathbf{r})$ , is obtained by taking

the ratio indicated in Eq. (53). To be more definite let us restrict our considerations to the canonical ensemble. Then the configurational probability density is given by

$$\rho_{\xi}^r(\mathbf{r}) d\mathbf{r} = Q_c^{-1} |Z|^{\frac{1}{2}} \exp[-\beta V(\mathbf{r})] \delta(\sigma) d\mathbf{r}, \quad (54)$$

where  $Q_c$ , the normalization factor, is the partition function of the constrained system and the conditional probability density of the momenta given the configuration is

$$\rho_{\xi}^p(\mathbf{p}^r | \mathbf{r}) d\mathbf{p}^r = |Z|^{-\frac{1}{2}} \exp[-\beta K] \delta(Z^{-1} \dot{\sigma}) d\mathbf{p}^r, \quad (55)$$

where  $K$  is the kinetic energy and  $Z$  is defined by

$$Z = \sum_{i=1}^N \frac{1}{m_i} \frac{\partial \sigma}{\partial \mathbf{r}_i} \cdot \frac{\partial \sigma}{\partial \mathbf{r}_i}. \quad (56)$$

The physical meaning of the quantity  $Z$  has been discussed by several authors<sup>25,26,27,28</sup> and has its origin in the restriction imposed in the momentum space by the validity of the constraint  $\sigma = 0$  at all times which, in turn, requires that the generalized velocity  $\dot{\sigma}$  vanishes at all times.

The probability densities we have been discussing up to now, more particularly the configurational probability density in Eq. (54), has to be compared with the configurational probability of the unconstrained system

$$\rho^r(\mathbf{r}) d\mathbf{r} = Q^{-1} \exp[-\beta V(\mathbf{r})] d\mathbf{r}, \quad (57)$$

or, even better, with the probability density to be at  $\mathbf{r}$  and at  $\xi = \xi'$

$$\rho^r(\mathbf{r}) \delta(\xi(\mathbf{r}) - \xi') d\mathbf{r} = Q^{-1} \exp[-\beta V(\mathbf{r})] \delta(\xi(\mathbf{r}) - \xi') d\mathbf{r}. \quad (58)$$

At this stage we can take the essential step. Recall that, to improve statistics, we wish to express the conditional average of any configurational property of our system in terms of the  $\xi$ -constrained ensemble introduced above. Indeed, while by definition the value  $\xi'$  we want to sample is rare in the original ensemble, only configurations with  $\xi = \xi'$  are sampled in the  $\xi$ -constrained ensemble. By simple inspection of Eq. (54) and Eq. (58) one finds that

$$\frac{\langle O(\mathbf{r}) \delta(\xi(\mathbf{r}) - \xi') \rangle}{\langle \delta(\xi(\mathbf{r}) - \xi') \rangle} = \frac{\langle |Z|^{-1/2} O(\mathbf{r}) \rangle_{\xi'}}{\langle |Z|^{-1/2} \rangle_{\xi'}}, \quad (59)$$

where the observable  $O(\mathbf{r})$  is some function of the configuration space,  $\langle \dots \rangle$  denotes the usual canonical average and  $\langle \dots \rangle_{\xi'}$  denotes an average over the constrained ensemble with  $\xi = \xi'$ . Eq. (59) represents the fundamental result of the *Blue Moon* approach since it permits one to estimate the conditional average on the left hand side of this equation. This result was implicit in the work of Bennett<sup>7</sup>

and Chandler<sup>1</sup> who did not use constraints but equivalent localizing procedures. To be useful Eq. (59) should be complemented with a numerically-workable procedure to compute  $P_\xi(\xi') = \langle \delta(\xi - \xi') \rangle$ . To this end let us proceed as follows. Call  $W(\xi') = -\frac{1}{\beta} \ln \frac{P_\xi(\xi')}{C}$ , where  $C$  is a normalization constant, the reversible work needed to bring the system from a given reference state to  $\xi = \xi'$ . The associated thermodynamic force

$$F(\xi') = -\frac{dW(\xi')}{d\xi'}, \quad (60)$$

can be expressed as the conditional average of a suitable observable. Then, using Eq. (59) and a thermodynamic integration of  $F(\xi')$  over  $\xi'$  we can obtain the desired probability.

To derive the explicit form of the thermodynamic force let us perform the derivative indicated in Eq. (60) with the help of a change of variables which introduces the Jacobian of the transformation,  $|J|$ . The result is

$$\begin{aligned} F(\xi') &= \frac{1}{\beta} \frac{\frac{d}{d\xi'} \int d\mathbf{r} e^{-\beta V} \delta(\xi - \xi')}{\int d\mathbf{r} e^{-\beta V} \delta(\xi - \xi')} \\ &= \frac{1}{\beta} \frac{\langle \left( \frac{\partial}{\partial \xi} \ln |J| - \beta \frac{\partial V}{\partial \xi} \right) \delta(\xi - \xi') \rangle}{\langle \delta(\xi - \xi') \rangle} \\ &\equiv \frac{\langle \hat{F} \delta(\xi - \xi') \rangle}{\langle \delta(\xi - \xi') \rangle}, \end{aligned} \quad (61)$$

where  $\hat{F}$  is composed of two terms, the first term  $\frac{1}{\beta} \frac{\partial}{\partial \xi} \ln |J|$  represents the apparent force acting on the system due to the use of generalized (non-inertial) coordinates, while the second term corresponds to the standard component of the force along the generalized coordinate  $\xi$  arising from the potential  $V$ .

The above result expresses the thermodynamic force as a conditional average; therefore it can be computed numerically by using the *Blue Moon* result, Eq. (59). To obtain this result we have proceeded as follows: Consider the numerator on the rhs of the first line of Eq. (61) and apply the transformation from the Cartesian coordinates  $\mathbf{r}$  to the variables  $\mathbf{u}$ . Performing the derivative and transforming back to the original coordinates one finds

$$\begin{aligned} \frac{d}{d\xi'} \int d\mathbf{r} e^{-\beta V} \delta(\xi - \xi') &= \int d\mathbf{q} d\xi \left[ -\frac{\partial}{\partial \xi} \delta(\xi - \xi') \right] |J| e^{-\beta V} \\ &= \int d\mathbf{q} d\xi \delta(\xi - \xi') \frac{\partial}{\partial \xi} [|J| e^{-\beta V}] = \int d\mathbf{r} |J|^{-1} \delta(\xi - \xi') \frac{\partial}{\partial \xi} [|J| e^{-\beta V}], \end{aligned} \quad (62)$$

which gives the desired result.

There are two nasty quantities to be computed in Eq. (61), the Jacobian and the partial derivative with respect to  $\xi$ . Ruiz and Frenkel<sup>29</sup> have shown how to simplify

this calculation by using as generalized coordinates  $\mathbf{u}$  a set of orthogonal variables, i.e. a set such that

$$\nabla u_\alpha \cdot \nabla u_\gamma = |\nabla u_\alpha|^2 \delta_{\alpha\gamma}, \quad (63)$$

where  $\nabla$  is the  $3N$  dimensional vector given by the partial derivative with respect to the Cartesian coordinates of all particles  $\mathbf{r}$ . Moreover, the coordinates  $\mathbf{q}$  can be chosen so that  $|\nabla q_\alpha| = 1$ ,  $\alpha = 1, 3N - 1$ . Under these conditions it is easy to show that  $|J| = |\nabla \xi|^{-1}$  and that

$$\frac{\partial}{\partial \xi} = \frac{1}{|\nabla \xi|^2} (\nabla \xi) \cdot \nabla. \quad (64)$$

The constrained ensemble we have obtained can be generalized to give the biased configurational sample we already know and the correct distribution of momenta. This new *Blue Moon* ensemble can be easily obtained by multiplying the  $\xi$ -constrained configurational probability density by the correct (Maxwellian) conditional probability of the momenta.

$$\rho_{BM}(\mathbf{r}, \mathbf{p}^r) = \rho_\xi^r(\mathbf{r}) \rho^p(\mathbf{p}^r | \mathbf{r}), \quad (65)$$

This ensemble provides a natural method for the computation of time correlation functions. What is required is to take an average over a *Blue Moon* ensemble of

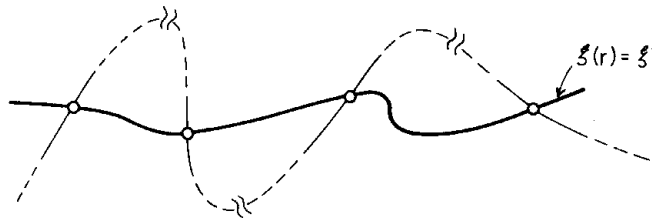


Fig. 1. Schematic representation of Blue Moon sampling. The bold line represents the constrained ( $\xi(\mathbf{r}) = \xi'$ ) dynamical evolution in phase space. The natural evolution of the system is indicated by the broken line. Open circles represent common points in configuration space which are the initial conditions of the activated trajectory sampling. Note that these points are not real crossings in phase space since the two trajectories differ in the momentum space. Interruptions of the broken line stand for long time segments in the natural trajectory and indicate that “crossings” are very rare events. The dynamics represented by the light lines in the vicinity of the crossing points gives the dynamical information needed in Eq. (66).

initial conditions of trajectories obtained by following the complete time evolution of

the system after release of the constraint  $\xi$  (see Fig. 1). Hence given two arbitrary observables  $O'(\mathbf{r}, \mathbf{p}^r)$  and  $O''(\mathbf{r}, \mathbf{p}^r)$  we may write

$$\frac{\langle O'(\mathbf{r}, \mathbf{p}^r) O''(\mathbf{r}(t), \mathbf{p}^r(t)) \delta(\xi(\mathbf{r}) - \xi') \rangle}{\langle \delta(\xi(\mathbf{r}) - \xi') \rangle} = \frac{\langle |Z|^{-1/2} O'(\mathbf{r}, \mathbf{p}^r) O''(\mathbf{r}(t), \mathbf{p}^r(t)) \rangle_{\xi'}}{\langle |Z|^{-1/2} \rangle_{\xi'}}, \quad (66)$$

This general approach may now be directly applied to the calculation of the rate constant of a chemical reaction.

## 4. Applications to Classical Systems

### 4.1. A Simple example: vacancy migration

Perhaps the first important aspect to be considered in any microscopic description of a chemical reaction in solution is an adequate definition of the reaction coordinate  $\xi = \xi(\mathbf{r})$ . In principle, any scalar function involving single or many-body coordinates and which takes well-differentiated values in the reactant and product phase space regions can be considered as a good prospective candidate for  $\xi$ . In activated processes, the evolution of the system may also be viewed as the passage over one or several free energy barriers, which are associated with the intermediate transition states. Therefore, it is also important for any reaction coordinate to provide a proper characterization of these transition states. Normally, the choice of  $\xi$  is guided by physical intuition; however, there are even very simple cases where the choice of the reaction coordinate is not always so straightforward and requires careful thought.

As a first illustration of these features, we begin this section reviewing a simple example which involves the calculation of rates of vacancy migration in solids near the melting transition<sup>7,30,31</sup>. The system is very simple: a perfect Lennard-Jones FCC crystal with an embedded vacant site. The normal dynamics of the atoms involves mostly fast oscillations around their equilibrium positions. In addition, spontaneous thermal fluctuations may eventually induce diffusion of a tagged atom, for example the one originally located at  $\mathbf{r}_1$ , into the vacant site located at  $\mathbf{r}_v$  (see Fig. 2). If the activation energy required for the process to occur is greater than normal thermal energies, diffusion is a rare event. The first description of the diffusion of point defects in solids using computer simulations was performed by Bennett in 1975<sup>7</sup>. For this problem, he proposed the following reaction coordinate:

$$\xi(\mathbf{r}) = \left[ \mathbf{r}_1 - \frac{1}{4}(\mathbf{r}_2 + \mathbf{r}_3 + \mathbf{r}_4 + \mathbf{r}_5) \right] \cdot \hat{\mathbf{j}}, \quad (67)$$

where the versor  $\hat{\mathbf{j}}$  defines a fixed direction in space in terms of the *ideal* lattice positions

$$\hat{\mathbf{j}} = \frac{\mathbf{r}_1^{id} - \mathbf{r}_v^{id}}{|\mathbf{r}_1^{id} - \mathbf{r}_v^{id}|}. \quad (68)$$

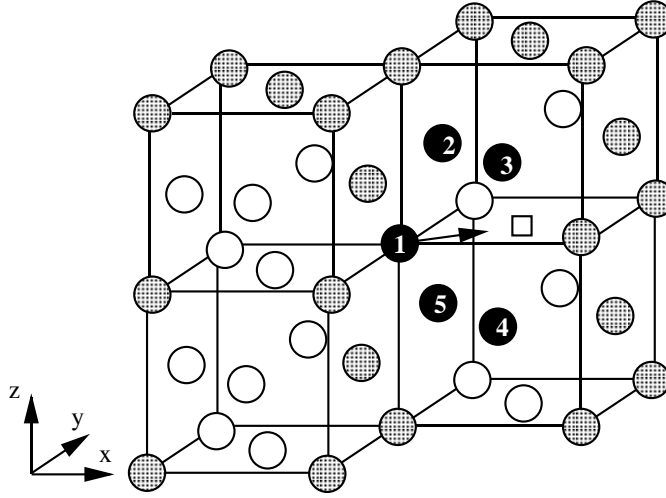


Fig. 2. Geometry of the crystal around the vacancy (square). The reaction coordinate is defined by the distance of the atom 1, the one that is jumping, to the center of mass of the four atoms 2, 3, 4 and 5. It can be seen as a way of measuring the position of the atom which is jumping with respect to its two possible accessible sites.

Actually in a perfect crystal  $\xi$  measures the distance between the prospective reactive particle and a plane equidistant from  $\mathbf{r}_1^{id}$  and  $\mathbf{r}_v^{id}$ ; the position of this plane also identifies the transition state for this reaction. At a first glance, Bennett's definition seems physically sound; however we will see below that it has a serious drawback: in a perfect FCC crystal, a single vacancy possesses 12 equivalent sites occupied by nearest neighbor atoms from where an eventual reactive process may evolve; therefore, it is impossible to predict and to control from which direction a possible jump attempt may take place. Note that the definition in Eq. (67) would be valid only if one atom was allowed to diffuse into the vacancy; unfortunately this is not the case and the description provided by this reaction coordinate is not complete.

To circumvent this difficulty, Paci *et al.*<sup>31</sup> have considered an alternative scheme where all the nearest neighbors of the vacancy except one, are prevented from diffusing. This is achieved by switching on artificial external harmonic fields of the type

$$V(\mathbf{r}_2, \dots, \mathbf{r}_N; \lambda) = \lambda \frac{k}{2} \sum_{j=2}^N |\mathbf{r}_j - \mathbf{r}_j^o|^2, \quad (69)$$

where  $k$  is a restoring constant whose magnitude is similar to that of a typical harmonic constant in the crystal,  $\lambda$  is an adjustable parameter between 0 and 1 and  $\mathbf{r}_j^o$  represents the set of relaxed equilibrium lattice positions. For  $\lambda > 0$ , the diffusion is hindered except for particle 1 so there is full control of the dynamical process. Paci's simulation scheme involves a series of constrained molecular dynamic experiments at different values of  $\lambda$ ; the final result for the rate is obtained after proper extrapolation to the limit  $\lambda \rightarrow 0$ .

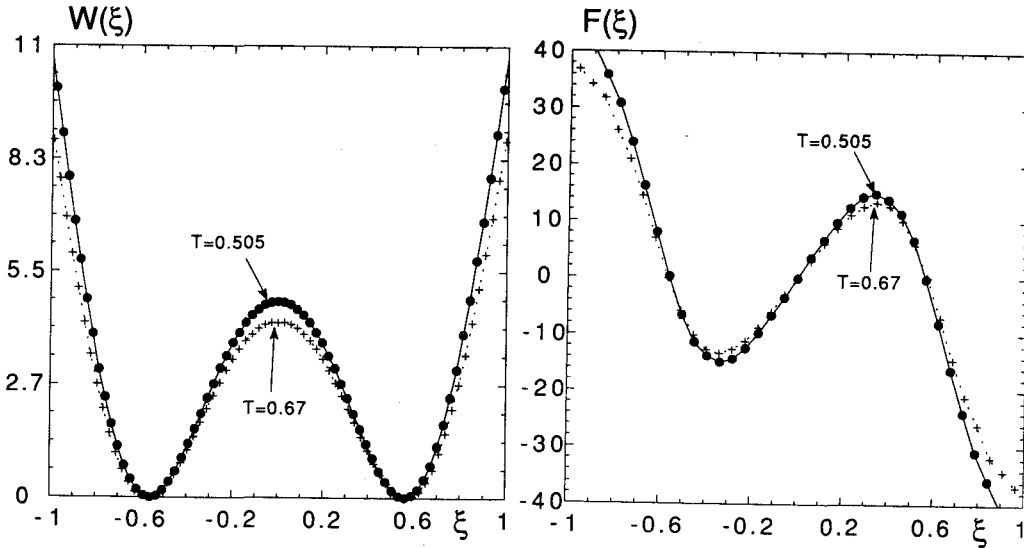


Fig. 3. Mean force and potential of mean force as a function of the reaction coordinate for the two different temperatures indicated in the graphs.

Note however, that for *Blue Moon* experiments, Bennett's definition for the reaction coordinate possesses an attractive feature: the reaction coordinate can be expressed as a linear combination of a subset of Cartesian coordinates of the system. Calculation of free energy barriers by the integration of the mean force in Eq. (61) requires a change to generalized coordinates  $\{\mathbf{r}\} \rightarrow \{\mathbf{u}\} = \{\xi, \mathbf{q}\}$ . If the reaction coordinate is expressed as a linear combination of Cartesian coordinates, this change of variables is straightforward; it suffices to take Eq. (67) and supplement it with  $(q_2, \dots, q_{3N}) = (r_{1y}, \dots, r_{Nz})$ . The matrix transformation  $\mathbf{J}$  can be obtained by inverting Eq. (67); except for the first row, the resulting matrix is diagonal and the computation of its determinant gives  $|\mathbf{J}| = \sqrt{2}$ . Since  $|\mathbf{J}|$  is a constant, the centrifugal correction to the mean constrained force vanishes. In Fig. 3 we show the computed free energies and constrained forces for an FCC Lennard-Jones crystal at two thermodynamic states.

#### 4.2. Potential of mean force

The next example that we will consider is the computation of the potential of mean force  $W(r)$  between a pair of particles dissolved in a fluid. The free energy profile provides important information about structural details of the solvation and is a valuable tool to help understand equilibria and chemical rates for inter-conversion between different solute configurations.

Basically, the evaluation of the potential of mean force is equivalent to the computation of  $P(r)$ , the probability density of finding two tagged particles at a distance

$r$  in a fluid,

$$-\beta W(r) = \ln [P(r)/P_u] , \quad (70)$$

where  $P_u$  is the uniform probability density. In a one-component fluid, this calculation reduces to the standard evaluation of the radial distribution function  $g(r)$

$$P(r) = \frac{4\pi}{V} r^2 g(r) . \quad (71)$$

For multi-component systems it is necessary to resort to different strategies. The *Blue Moon* ensemble is particularly suitable for this kind of calculation and a number of studies have been devoted to this subject. The majority of the studies consider the case of neutral and charged ion pairs dissolved in different media, including a model polar dumbbell solvent<sup>32,33</sup>, SPC model of water<sup>34</sup>, aprotic solvents<sup>35</sup> and cluster environments<sup>36,37</sup>.

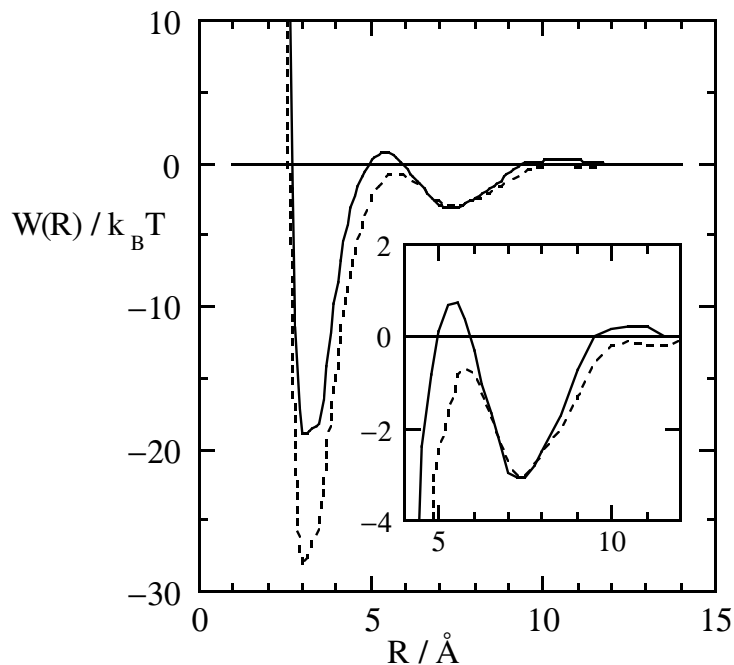


Fig. 4. Potential of mean force between the ion pair (in units of  $k_B T$ ) as a function of the internuclear separation ( $\text{\AA}$ ) for the model polar dumbbell solvent. An expanded view of the barrier region is given in the inset. The solid line corresponds to the solvent with a molecular dipole  $\mu = 3.0$  D while the dashed line refers to the case  $\mu = 2.4$ .

For such systems, a reasonable choice for the reaction coordinate is the distance between the two tagged particles  $\mathbf{r}_1$  and  $\mathbf{r}_2$ ,

$$\xi(\mathbf{r}_1, \mathbf{r}_2) = |\mathbf{r}_1 - \mathbf{r}_2| = r . \quad (72)$$

The simulations involve series of molecular dynamics runs where the ion pair is held fixed at selected distances by holonomic constraints. During these constrained experiments, the average force along the inter-ionic distance exerted by the solvent upon the ion pair is computed. In this case, the centrifugal contribution to the mean force does not vanish. However, the computation of this contribution is not difficult if the change of variables

$$\{r_{1x}, r_{1y}, r_{1z}, r_{2x}, r_{2y}, r_{2z}, \dots, r_{Nz}\} \rightarrow \{\xi, \theta, \phi, \bar{r}_x, \bar{r}_y, \bar{r}_z, \dots, r_{Nz}\}, \quad (73)$$

is considered. Here  $(\theta, \phi)$  are the two angles of the vector  $\mathbf{r}_1 - \mathbf{r}_2$  in spherical polar coordinates and  $\bar{\mathbf{r}}$  represents the coordinate of the center of mass of the ion pair; the remainder of the coordinates are unchanged. The calculation of the Jacobian of the coordinate transformation is straightforward:  $|\mathbf{J}| = r^2 \cos \theta$ . From Eq. (61) and Eq. (60) the expression for the difference of the potential of mean force between two points  $\xi_1$  and  $\xi_2$  is

$$-\beta [W(\xi_2) - W(\xi_1)] = \int_{\xi_1}^{\xi_2} \langle \hat{F} \rangle_{\xi'} d\xi' + 2 \ln \frac{\xi_2}{\xi_1}. \quad (74)$$

In Fig. 4 the potential of mean force for two ions in a model polar solvent is shown for two different values of the permanent dipole moment of the solvent molecule<sup>32</sup>.

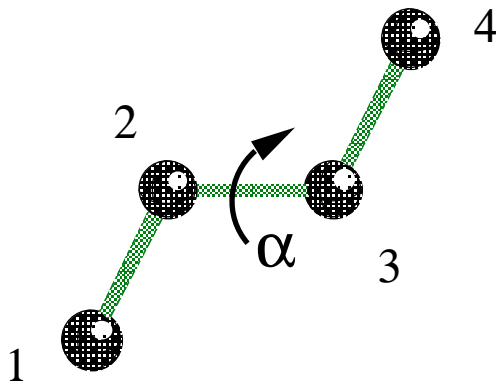


Fig. 5. Dihedral angle for a butane molecule, see Eq. (75).

Before closing this section, we will briefly refer to a recent comparative study for the trans-gauche inter-conversion of *n*-butane performed by Depaepe *et al*<sup>38</sup>. Several authors<sup>39,40</sup> have already considered the study of these equilibria in solution so the system represents a useful benchmark for a critical evaluation of different sampling techniques. The butane molecule is modeled as a fully-flexible, four-site chain and the reactive process is described by the dihedral angle  $\alpha$  (see Fig. 5) defined as

$$\xi(\mathbf{r}) = \cos \alpha = \frac{(\mathbf{r}_{12} \wedge \mathbf{r}_{23}) \cdot (\mathbf{r}_{23} \wedge \mathbf{r}_{34})}{|\mathbf{r}_{12} \wedge \mathbf{r}_{23}| |\mathbf{r}_{23} \wedge \mathbf{r}_{34}|}, \quad (75)$$

where  $\mathbf{r}_j$  represents the coordinate of the  $j$ -th site in the butane molecule and  $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ .

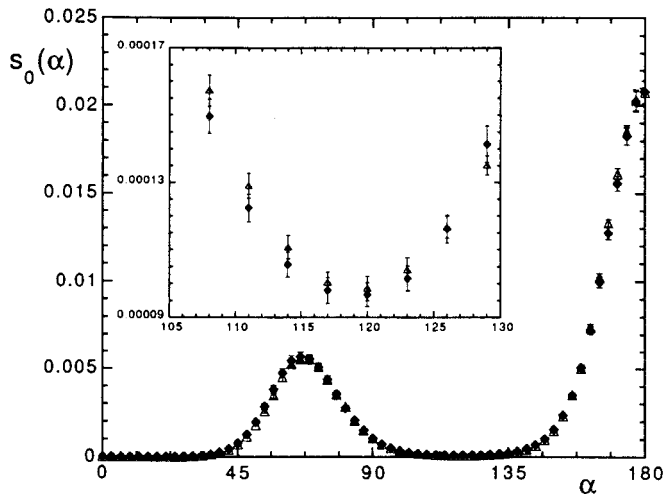


Fig. 6. Probability density  $s_0(\alpha)$  of n-butane in carbon tetrachloride. Results of umbrella sampling (triangles) and *Blue Moon* sampling with the dihedral angle constraint (diamonds) are shown. The probability density around the barrier between *trans* and *gauche* regions is shown in the inset.

Averages computed from sampling using the *Blue Moon* ensemble require the calculation of  $Z$  or, equivalently, partial derivatives  $\partial\xi/\partial\mathbf{r}_i$ . In addition, it is necessary to perform a full transformation of coordinates  $\{\mathbf{r}\} \rightarrow \{\mathbf{u}\} = \{\xi, \mathbf{q}\}$  and compute the associated Jacobian  $|\mathbf{J}|$ . For this particular case, the calculations are tedious but straightforward if one considers a generalised change of variable proposed by Go and Scheraga<sup>41</sup>. The coordinate transformation involves a change from the  $3n$  Cartesian coordinates of the sites into the three coordinates of the centre of mass of the molecule, three angles that characterise the overall molecular orientation supplemented by  $n - 1$  bond lengths,  $n - 2$  bond angles and  $n - 3$  dihedral angles. Full details of these calculations are given elsewhere<sup>38</sup>. In Fig. 6, we present the intramolecular distribution function  $s(\alpha)$  for n-butane in liquid carbon tetrachloride where the results of the *Blue Moon* ensemble are in perfect agreement with those obtained using standard biased umbrella sampling<sup>42</sup>. The quality of the agreement confirms that both methods seem to have similar efficiency for this kind of calculation.

## 5. Activated Quantum Process

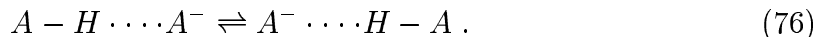
There are many fundamental processes in the area of chemical reactivity in solution where the quantum nature of at least some of the particles cannot be neglected. Charge transfer reactions are perhaps the most clear example; here we refer to the passage of an electron or proton between two charged sites in solution. Normally, the

quantum treatment of the transition process is considered in two regimes: (i) adiabatic dynamics where the Born-Oppenheimer approximation is used to separate the dynamics of the heavy particles and the latter move on a given Born-Oppenheimer surface; (ii) non-adiabatic dynamics which allows for the possibility of transitions between different Born-Oppenheimer surfaces. The design of a tractable algorithm for non-adiabatic dynamics is an active area of research<sup>43,44,45,46,47,48,49</sup> and even adiabatic dynamics presents computational challenges.

### 5.1. Adiabatic dynamics

Adiabatic dynamics represents the limiting regime where transitions between different Born-Oppenheimer surfaces can be neglected. For this hypothesis to be valid, the gap between different energy surfaces should be greater than normal thermal energies for all nuclear configurations. In such cases, the nuclear dynamics is sufficiently slow so that the quantum density of the reactive particle can adjust instantaneously to changes in the nuclear configurations.

We will illustrate some features of adiabatic dynamics by summarising some of the salient features of the study of proton transfer reactions in solution performed by Laria *et al.*<sup>50</sup>. We consider the transfer of a proton between a pair of  $A^-$  anions with fixed internuclear separation dissolved in a polar solvent:



There have been a number of studies of proton transfer for models of this type which have employed both adiabatic and path integral methods (see below)<sup>51,52,53,54</sup>. The only quantum degrees of freedom are the proton coordinates; solvent particles, modeled as a pair of sites with embedded partial charges, are taken to be classical particles. The proton-ion bare potential models a typical strongly hydrogen-bonded system: it possesses a symmetric double-well structure along the inter-ionic axis with a negligible barrier. In absence of solvent, the proton quantum probability corresponding to the symmetric ground state is fully delocalized along the inter-ionic axis. In solution, the presence of local solvent polarization fluctuations introduces considerable asymmetries in the proton potential energy leading to its localization in the neighborhood of one ion. In the adiabatic regime, the dynamics of those solvent polarization fluctuations will actually determine the rate of the transfer process.

Let  $\mathbf{r}_p$  be the coordinate of the quantum proton and  $\{\mathbf{R}\}$  the remainder of the ionic and solvent classical coordinates. The total potential energy of the system  $V$  can be written as sum of two contributions:  $V_{ps}(\mathbf{r}_p, \{\mathbf{R}\})$  and  $V_s(\{\mathbf{R}\})$ ; the former involves proton-solvent and proton-ion interactions while the latter accounts for the interaction between the classical particles. Within the adiabatic approximation, the wavefunction describing the quantum density of the proton satisfies the following

Schrödinger equation

$$\left( -\frac{\hbar^2}{2m_p} \nabla_{\mathbf{r}_p}^2 + V_{ps}(\mathbf{r}_p, \{\mathbf{R}\}) \right) \Psi_n(\mathbf{r}_p; \{\mathbf{R}\}) = \epsilon_n(\{\mathbf{R}\}) \Psi_n(\mathbf{r}_p; \{\mathbf{R}\}) , \quad (77)$$

where  $m_p$  is the mass of the proton,  $2\pi\hbar$  is Planck's constant and the sub-index  $n$  refers to the  $n$ -th eigenvalue and corresponding eigenfunction. The time evolution of a classical particles with mass  $M_i$  follows Newton's equations of motion

$$M_i \frac{d^2 \mathbf{R}_i}{dt^2} = -\nabla_{\mathbf{R}_i} [V_s(\{\mathbf{R}\}) + \epsilon_0(\{\mathbf{R}\})] . \quad (78)$$

The implementation of adiabatic dynamics is straightforward: for a given solvent configuration, the Schrödinger equation is solved according to Eq. (77) by expanding the wave function representing the proton as linear combination of a set of basis functions distributed along the inter-ionic region. The solution of the Schrödinger equation yields a set of eigenvalues and the corresponding eigenfunctions. At each step of the simulation, we verify that the spacing between the ground and the first excited state for the instantaneous Hamiltonian is larger than that of the bare proton-ion potential. Once the ground state wave-function  $\Psi_0(\{\mathbf{r}_p; \mathbf{R}\})$  is computed, the force  $\mathbf{F}_i$  exerted by the quantum proton upon the  $i$ -th solvent particle is calculated according to the Hellman-Feynman theorem<sup>55</sup>

$$\mathbf{F}_i = - \int |\Psi_0(\mathbf{r}_p; \{\mathbf{R}\})|^2 \nabla_{\mathbf{R}_i} V_{ps}(\mathbf{r}_p, \{\mathbf{R}\}) d\mathbf{r}_p . \quad (79)$$

The last step involves the standard integration of Newton's equations for the classical particles in Eq. (78) with forces derived from Eq. (79) and from classical solvent-solvent and solvent-ion interactions.

We wish to estimate the rate for the proton transfer process. Perhaps the most natural choice for the reaction coordinate of this problem is the  $z$  component of the average proton position

$$\xi(\{\mathbf{R}\}) = \bar{z}_p(\{\mathbf{R}\}) = \langle \Psi_0(\{\mathbf{R}\}) | \hat{z}_p | \Psi_0(\{\mathbf{R}\}) \rangle , \quad (80)$$

where we have chosen the  $z$ -axis along the inter-ionic direction. By symmetry, the transition state would be localized midway between the position of the ion pair. Unfortunately, the implementation of trajectories in the *Blue Moon* ensemble with the definition given in Eq. (80) is difficult since the algorithm requires trajectories where the reaction coordinate is constrained at the transition state. This is most easily carried out if the reaction coordinate can be expressed as an analytical function of the coordinates of the particles; in the present case, the value of  $z_p$  is only known numerically.

Due to the characteristics of the adiabatic dynamics it is possible to define a reaction coordinate more suitable for the simulation scheme. Note that in solving the

Schrödinger equation, the instantaneous position of the proton probability density will be fully determined by the fluctuating proton-solvent potential field; then, the characteristics of the proton dynamics will be governed by the dynamical evolution of the solvent polarization fluctuations. Using a very simplified model, we verified that the dynamical behavior of  $\Delta E$  defined as the difference of the solvent electrical potential between two points along the inter-ionic axis, is qualitatively similar to that of  $\bar{z}_p$ . This new reaction coordinate is an analytical function of the nuclear positions  $\{\mathbf{R}\}$  so it is easy to generate constrained trajectories in which the reaction coordinate remains fixed at the transition state.

The rate constant for the proton transfer can be calculated from the limiting behavior of the following reactive flux correlation function<sup>10,1</sup>:

$$k(t) = k^{TST} \kappa(t) = \frac{\langle \dot{\Delta E}(0) \delta(\Delta E(0) - E^\ddagger) \theta[\Delta E(t) - \Delta E^\ddagger] \rangle}{\langle \theta[\Delta E - \Delta E^\ddagger] \rangle}, \quad (81)$$

where  $\langle \dots \rangle$  represents a canonical ensemble average,  $\theta(x)$  is the Heaviside function and  $\Delta E^\ddagger$  is the value of the reaction coordinate at the transition state *i.e.*  $\Delta E^\ddagger = 0$ . Provided a time scale separation exists between the characteristic time scales describing the dynamics of the proton transfer and the other microscopic relaxation processes,  $k(t)$  will exhibit a fast relaxation towards a limiting plateau value from which we may compute the rate constant. The transition state estimate for the rate constant  $k^{TST}$  is given by

$$k^{TST} = \lim_{t \rightarrow 0^+} k(t) = \frac{\langle \dot{\Delta E} \theta(\Delta E) \delta(\Delta E - \Delta E^\ddagger) \rangle}{\langle \theta(\Delta E - \Delta E^\ddagger) \rangle}. \quad (82)$$

Apart from prefactors the transition state rate constant is governed by an Arrhenius term proportional to the probability that the system is in the transition state.

The expression for  $\kappa(t)$  using the *Blue Moon* ensemble can be written as

$$\kappa(t) = \frac{\langle Z^{-1/2} \Delta E \theta[\Delta E(t)] \rangle_c}{\langle Z^{-1/2} \Delta E \theta(\Delta) \rangle_c} \quad (83)$$

where  $\langle \dots \rangle_c$  represents an average taken over unconstrained trajectories initiated at the transition state  $\Delta E = \Delta E^\ddagger$  and generated using the *Blue Moon* ensemble approach.

For our particular model of proton transfer, the computation of  $Z$  in Eq. (56) requires different products of matrices with elements involving intramolecular bond constraints in the solvent particles and gradients of  $\Delta E$ . The final expression is quite simple and reduces to<sup>50</sup>

$$Z = \frac{1}{2m} \sum_i \sum_\alpha (\nabla_{i,\alpha} \Delta E)^2, \quad (84)$$

where the sums over  $i$  and  $\alpha$  extend over all solvent molecules and sites, respectively, and we have assumed that all sites have identical mass  $m$ .

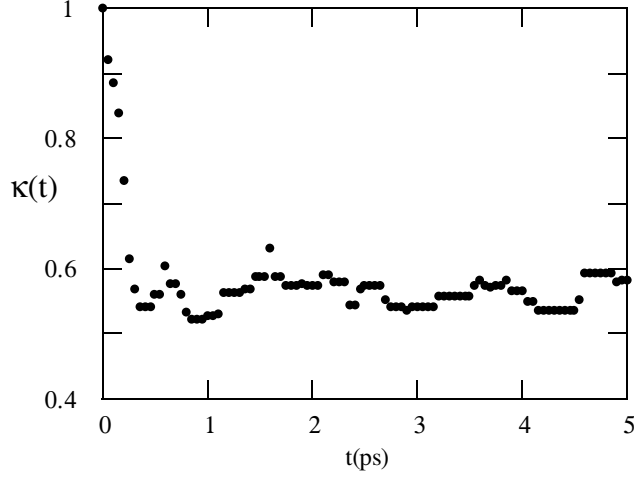


Fig. 7. The time-dependent transmission coefficient for the adiabatic dynamics calculation.

In Fig. 7 we present the transmission coefficient calculated using adiabatic dynamics.

### 5.2. Path integral molecular dynamics

The path integral formulation of quantum mechanics<sup>56</sup> provides an alternative route to compute equilibrium properties for quantum activated processes. Within this formalism, quantum particles are mapped into closed paths  $\mathbf{r}(t)$  in imaginary time  $t$ ,  $0 \leq t \leq \beta\hbar$ <sup>57</sup>. Gillan<sup>58</sup> pioneered this approach to study rates for quantum activated processes by establishing the notion of *centroid* density which has been used in a number of formulations of the rate problem<sup>13,59</sup>. In this scheme the reaction coordinate is identified with the position of the centroid of the quantum path<sup>60</sup>,  $\bar{\mathbf{r}}_p$ , defined according to

$$\xi[\mathbf{r}(t)] \equiv \bar{\mathbf{r}}_p = \frac{1}{\beta\hbar} \int_0^{\beta\hbar} \mathbf{r}(t) dt . \quad (85)$$

In the usual way, a free energy  $W(\xi)$  is associated to the probability density for the centroid  $\xi[\mathbf{r}(t)]$ ,

$$e^{-\beta W(\xi^\dagger)} \propto \langle \delta(\xi[\mathbf{r}(t)] - \xi^\dagger) \rangle , \quad (86)$$

where  $W(\xi)$  represents the reversible work required to move the centroid to the point  $\xi^\dagger$ . For the case where there is only one quantum particle, the average in Eq. (86) can be written as:

$$\langle \delta(\xi[\mathbf{r}(t)] - \xi^\dagger) \rangle \propto \int d\{\mathbf{R}\} \int \mathcal{D}[\mathbf{r}(t)] \delta(\xi[\mathbf{r}(t)] - \xi^\dagger) e^{S[\mathbf{r}(t), \{\mathbf{R}\}] - \beta V_{cl}(\{\mathbf{R}\})} , \quad (87)$$

where  $V_{cl}(\{\mathbf{R}\})$  is the potential energy for the classical degrees of freedom and the

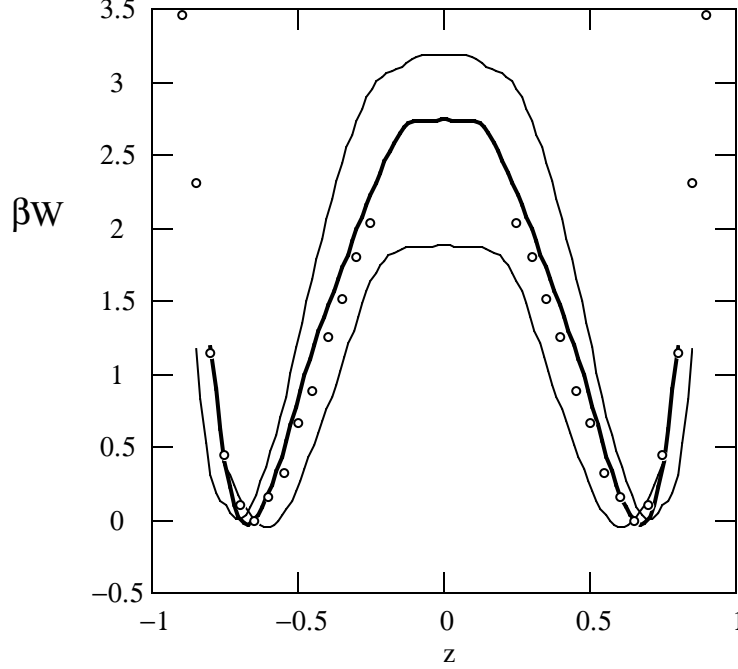


Fig. 8. Free energy along the reaction coordinate. The highest curve is the classical result; the middle curve is for the proton, while the lowest curve is for the muon. Open circles were obtained from an unconstrained MD simulation for the case of the proton.

functional  $S[\mathbf{r}(t); \{\mathbf{R}\}]$  is the action that governs the statistics of the quantum path

$$S[\mathbf{r}(t); \{\mathbf{R}\}] = \frac{1}{\hbar} \int_0^{\beta\hbar} dt \left[ \frac{1}{2} m_p |\dot{\mathbf{r}}(t)|^2 + V_p(\mathbf{r}(t); \{\mathbf{R}\}) \right]. \quad (88)$$

In Eq. (88)  $m_p$  is the mass of the quantum particle and  $V_p(\mathbf{r}(t); \{\mathbf{R}\})$  is the potential energy between the quantum and classical degrees of freedom.

The implementation of a tractable computer algorithm requires the discretization of the closed path and leads to an isomorphism between the Feynman path and a “ring polymer” with  $P$  interaction sites<sup>56,57</sup>. The statistics of this polymer is governed by the following effective potential:

$$V_{eff} = \frac{P m_p}{2 (\beta\hbar)^2} \sum_{i=1}^P (\mathbf{r}_i - \mathbf{r}_{i+1})^2 + \frac{1}{P} V_p(\mathbf{r}(t); \{\mathbf{R}\}) . \quad (89)$$

In fact, the isomorphism is exact in the limit  $P \rightarrow \infty$ . Note that the problem reduces to sampling from a canonical distribution according to the effective potential in Eq. (89). In molecular dynamics, this can be achieved using the following Hamiltonian;

$$H = \sum_{i=1}^P \frac{1}{2} m^* \dot{\mathbf{r}}_i^2 + \sum_j \frac{1}{2} M_j \dot{\mathbf{R}}_j^2 + V_{cl} + V_{eff} \quad (90)$$

where  $M_j$  is the mass of the  $j$ -th classical particle,  $V_{cl}$  is the potential energy of the classical degrees of freedom and  $m^*$  is an arbitrary mass assigned to the polymer beads. The mass  $m^*$  is adjusted to guarantee efficient sampling of the configuration space. At this point, it is worth emphasising that there is no real dynamical meaning to the trajectories generated by the Hamiltonian in Eq. (90). Newton's equations only provide a tool to construct a proper sampling over the distribution given in Eq. (87).

Returning to our previous example of proton transfer, the free energy barrier for the reactive process can be computed by integrating the mean force  $\langle F(\xi) \rangle$  acting on the centroid of the proton-polymer along the reaction coordinate. As in our previous example of diffusion in solids, the resulting reaction coordinate can be expressed as a linear combination of a set of Cartesian coordinates of the system so no  $Z$  factors or Jacobian  $|\mathbf{J}|$  are required to correct *Blue Moon* sampling.

In Fig. 8 we present the free energy profiles obtained for different values of the mass of the quantum particle (classical limit, proton and muon mass) by integration of the mean force and from a histogram for the probability density of the centroid of the proton obtained in a long unconstrained Molecular Dynamics experiment<sup>60,61</sup>. The histogram method is inaccurate in the barrier region since the system trajectory rarely visits these phase regions; consequently, the method is not recommended in the estimation of the shape and height of the barrier. There is a progression in barrier heights, with the classical calculation yielding the highest free energy barrier and the muon the lowest. The locations of the free energy minima corresponding to the stable reactant and product states are shifted to smaller values as the quantum character of the particle increases.

The changes in the free energy barrier can be attributed to the fact that the charge density becomes increasingly dispersed as the mass of the particle decreases. The dispersion of the charge density of the quantum particle can be seen in Fig. 9 which shows snapshots of the path integral polymer for both the proton and muon with the centroid constrained to lie at the transition state and product states. It is clear from this figure that the muon charge density is considerably broader than that of the proton; it extends over a length that is comparable to the distance through which the particle transfers. In this circumstance it is certainly not appropriate to treat the system classically.

The root-mean-square displacement,  $R(t-t')$ , between two points on the quantum path separated by a time increment  $0 \leq t - t' \leq \beta\hbar$  with

$$R^2(t - t') = \langle |\mathbf{r}(t) - \mathbf{r}(t')|^2 \rangle, \quad (91)$$

provides a quantitative estimate of the spatial extent of the quantum particle since the characteristic size of the polymer is  $R = R(\beta\hbar/2)$ . A free quantum particle with mass  $m$  has,  $R^f = \sqrt{3/4}\lambda$ , where  $\lambda$  is  $\lambda = (\hbar^2\beta/m_p)^{1/2}$ . A proton at  $T = 250$  K has,  $R_p^f = 0.38 \text{ \AA}$ , while a free muon has  $R_\mu^f = 1.14 \text{ \AA}$ . Fig. () shows  $R(t)$  versus  $t$  for both the proton and muon with the centroid constrained to lie at the transition

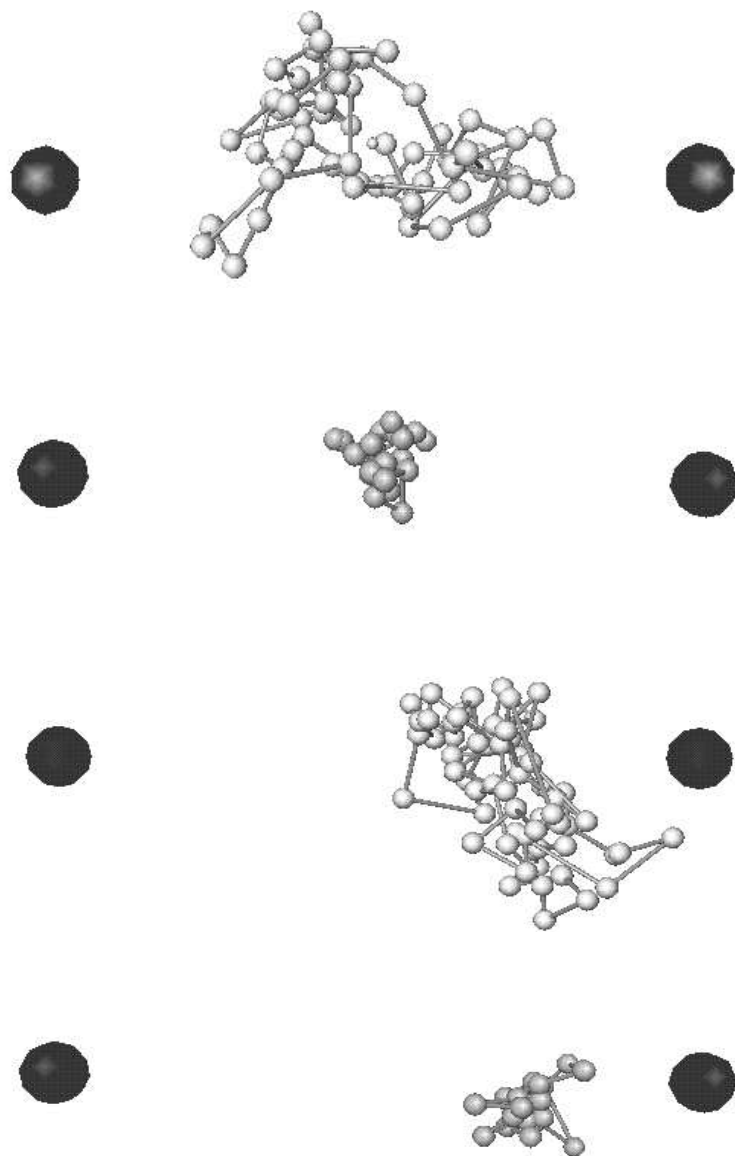


Fig. 9. Muon and proton polymer configurations with the centroid constrained in the transition (top two panels) and product (bottom two panels) states. No significance should be attached to the sizes of the ions or polymer beads.

and product states. The top three curves are for the muon while the bottom three curves are for the proton. The proton results show a small but significant deviation from the free particle curve, with the proton somewhat more confined in the product region than at the transition state. In the product state  $R_p(\hbar\beta/2) \approx 0.30\text{\AA}$  while  $R_p(\hbar\beta/2) \approx 0.32\text{\AA}$  in the transition state.

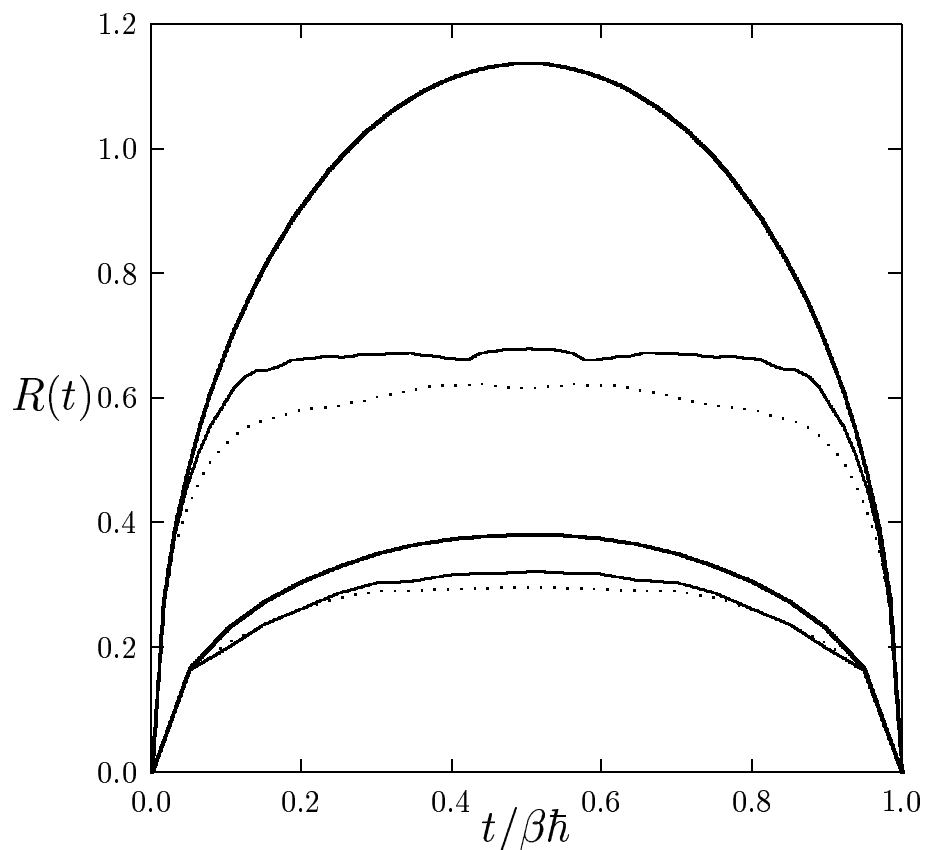


Fig. 10. Root-mean-square displacement,  $R(t)$ , as a function of imaginary time,  $t/\beta\hbar$ , for the proton and muon. The upper three curves are for the muon while the lower three curves are for the proton. The heavy solid lines are for the free particles, the light solid lines are for the system with centroid constrained at the transition state and the dotted lines correspond to the product states.

From the shapes of these curves one sees that ground state dominance is somewhat stronger in the product state than in the transition state. There is a much larger effect for the muon. The characteristic size of the muon changes from its free value of  $R_\mu^f = 1.14\text{\AA}$  to  $R_\mu(\hbar\beta/2) \approx 0.62$  and  $0.68\text{\AA}$  for the product and transition states

in solution. Note that the ground state dominates the dynamics.

## 6. Diffusion-Controlled Reactions

We now consider briefly an example when there is not a significant (much larger than  $k_B T$ ) free energy barrier to overcome for passage between the reactant and the product states, or, better, there are other steps in the reaction which are slower than the barrier crossing. This is the case, for example, for the reaction involving the dismutation of the  $O_2^-$  super-oxide ion at the active site of the Cu,Zn super-oxide dismutase enzyme (SOD) in solution. The slowest step that determines the association rate constant is the diffusive motion in solution of the super-oxide ion to a distance close enough to the active site of the SOD enzyme so that the reaction can take place. Once the ion arrives close to the active site the dismutation reaction occurs very rapidly. The overall process is said to be diffusion limited<sup>62,63</sup> and the rate at which the super-oxide ion approaches the active site of the SOD enzyme can be successfully calculated modeling the diffusion process with the Brownian motion of the super-oxide ion in the *external* electric field due to the charges of the enzyme and the solution. This electric field can be considered, within a good approximation, to be fixed independent of time<sup>64</sup>. The association rate constant can be calculated by running an ensemble of Brownian trajectories to compute the probability for the super-oxide ion to arrive at a given distance from the active site of the enzyme<sup>65</sup>. The trajectory propagation is described by the following equation of motion in the Smoluchowski limit:

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \Delta t D \mathbf{F}(t) / k_B T + \mathbf{s} , \quad (92)$$

where  $\mathbf{r}$  is the position vector,  $D$  is the diffusion constant of super-oxide relative to SOD,  $k_B$  is the Boltzmann constant,  $T$  is the temperature,  $\Delta t$  is the time step,  $\mathbf{F}(t)$  is the electrostatic force on super-oxide and  $\mathbf{s}$  is a vector whose components are independent Gaussian numbers determined by the properties:

$$\begin{aligned} \langle s_i \rangle &= 0 \\ \langle s_i^2 \rangle &= 2D\Delta t \quad (i = x, y, z) . \end{aligned} \quad (93)$$

Although the diffusion contribution to the rate constant is merely a probability term its computation follows a route similar to that for the transmission coefficient calculation in the general case. Trajectories are initiated at a given distance  $R_1$  from the protein and are followed until they either *react* by coming close enough to the copper atom or *escape* when they reach a distance from the protein greater than a certain value  $R_2$ . The association rate can be derived starting from the probability  $P$  that a trajectory starting at  $R_1$  hits the target at some later time. A typical calculation of the diffusion contribution to the association rate requires of the order of 10000 independent stochastic trajectories in order to have statistical uncertainties on the

diffusion contribution to the reaction rate constant lower than 2-3%. With this level of modeling it has been possible to reproduce fairly accurately the dependence of the catalytic rate for several different enzymes<sup>65,66</sup>. In the Appendix we report details of the parallelization of the Brownian trajectories on a cluster of workstations using message passing with PVM<sup>9</sup>.

## Appendix: Parallelization of Rate Constant Calculations using PVM

On a coarse grained scale algorithms for the calculation of rate constants can be considered intrinsically parallel since they utilize an ensemble of trajectories, that is usually a large number -from 100 to 10000- of independent trajectories with different initial conditions. So apart from the initial setup the workload can be easily imagined as distributed over several processors. This is best accomplished using a paradigm in which a master process on a selected cpu dispatches processes, each consisting in the integration of a single activated trajectory, over a pool of many slave cpus. However, some caution is needed as the cpu time needed to perform the *elementary* task, i.e. to integrate one trajectory, cannot be completely determined a priori and in all relevant cases one finds significant differences between one task and another. Therefore a strategy that symmetrically distributes the workload, assigning the same number of trajectories to each of the available processors, is usually not very practical and it can lead to a significant loss of performance as cpus remain idle while waiting to synchronize. When the number of trajectories to be run is much larger than the number of available processors, which is usually the case, it is highly rewarding to implement a dynamical *load balancing* strategy for the distribution of workload. The role of the master process is typically to perform the initial setup phase and control the state of slave cpus. As soon as a slave cpu is available the master process dispatches to it the task of integrating one trajectory. Later, when the slave cpu has performed the assigned task, the master process collects the result of the calculation and continues assigning other tasks until the required number of trajectories is computed. This strategy can be implemented easily using a message-passing library such as PVM and can be particularly effective on a non-dedicated, non-homogeneous, parallel system such as a workstation cluster. In particular during the phase of initialization a number of PVM slave processes are started up, one for each available slave processor. Then the master process waits for a READY message from any one of the slave processes. When such a message arrives the master *replies* sending back the data needed to perform an activated trajectory, that is usually the initial values of all particle coordinates and velocities. When the slave process terminates the task of integrating the assigned trajectory, it sends a new READY message to the master process which in turn sends data to start another trajectory. In this way the master program distributes

tasks to the slaves according to their availability and the workload is distributed in a asymmetrical way which maximizes performance. In general it is not necessary to collect immediately the single trajectory results and it can be more efficient to have averages performed separately by each slave process. When the total number of required trajectories is reached the master program exits the distributing stage and replies to READY messages from slaves with a STOP signal, requesting each slave to return to the master all the results accumulated over the locally performed trajectories. The common initialization stage - which needs to be repeated by each slave - and the communication time needed to pass the data needed to start each trajectory are globally much smaller tasks than the computation of trajectories and therefore the performance obtained can be, in principle, very close to the theoretical value. The only tricky part is to devise the master program in such a way so that it always has enough initial conditions at hand to feed the hungry slave processes. In the general case a single initial condition is a large set of particle coordinates and momenta. In a standard *Blue Moon* ensemble calculation the independent initial conditions needed to start the activated trajectories and compute the rate constant are produced by sampling system configurations during the calculation of the mean force, with the reaction coordinate constrained at the value corresponding to the barrier top. These sampled configurations can be stored and later read by the master process and made available to slave processes. Of course Boltzmann sampling of the velocity along the reaction coordinate must be performed to correct for the bias along the constrained trajectory prior to the activated trajectory calculation. Since this requires use of a random number generator it is usually safer to have this task performed by only one process, the master one, before the initial data is actually dispatched to the slave.

The potential-of-mean-force calculation can be also thought of as an intrinsically parallel calculation, but only on a small scale. The number of independent trajectories needed is, in fact, much smaller, usually of the order of 10, while the workload for each trajectory is at least one order of magnitude higher than for the case of the transmission coefficient calculation. Nevertheless, a parallel implementation still gives significant speedup either on small parallel systems or whenever it is possible to resort to a small pool of cpus in such a way that the number of required trajectories is an integer multiple of the number of available processors.

As an illustration of the above method consider the simplified problem of the computation of the rate constant in the dismutation reaction. In particular since stochastic trajectories rely heavily on random numbers, some care was taken concerning the generation of random number sequences by the slave processes. A procedure was set up in the master program to dispatch to each slave a different seed to start the random sequence generator in such a way to have - as far as possible - independent random number sequences for each slave.

Table 1. Timings in seconds for serial to parallel code comparison on the single node and on a five-node homogeneous cluster: *Performance = 4.95 (serial code cpu time/parallel code elapsed time)*

<b>Single NODE: serial code</b>		
	cpu	elapsed
total	29060	30975
<b>Single NODE: parallel code</b>		
	cpu	elapsed
master	11	29517
slave	29392	
<b>5 NODE cluster:</b>		
	cpu	elapsed
master	11	5962
	cpu time	# trajectories
slave 1	5834	1989
slave 2	5918	2026
slave 3	5913	1977
slave 4	5918	1997
slave 5	5911	2011

The program was developed using standard Fortran 77 including explicit message passing through the PVM library (version 3.3.x)<sup>9</sup>. It consisted of two code units, one of which is the self-contained PVM *master* program, while the *slave* was divided into separate subprograms on the base of functionality. However, all message passing instructions have been limited to the two main routines. A number of test runs were performed on a non-dedicated cluster of RISC workstations connected by a high speed FDDI network. On a non-dedicated cluster of workstation the performance that can be actually obtained is only a function of the overall load of the cluster. In this particular case the implemented strategy guarantees that maximum performance will be obtained even on a system composed of computational nodes with different cpu performances.

The results of a few test runs made using up to 5 identical nodes are reported in Table 1. It can be observed that in the most favorable case of a 'quasi' dedicated cluster the ratio between total cpu time of the un-parallelized version and real elapsed time of the parallel case is 4.95, in practice equal to the number of nodes involved. Also the test run with a single slave process shows clearly that time spent for communication is insignificant. Some tests were made with a simplified approach where tasks of 2000 trajectories were symmetrically distributed over five processors. Although the statistical average over the 2000 trajectory resulted in a very similar

cpu time required by each task, the remaining differences would always lead to some relevant idle time while waiting for the end of the slowest task. More important, it was found that on the test platform competition with other user processes - also an unpredictable condition - results in the possibility of one (or more) tasks being two to three times slower than the others with a complete degradation of performance.

Below we report a sample PVM startup session in which a five host configuration is set up. For clarity, PVM command shell is entered on the local system - the master host, `host_1` - and all other slave hosts - `host_2`, `host_3`, `host_4`, `host_5` - are configured with the `add` command. The `conf` command shows current configuration, with details of node identification (DTID), architecture of the node (ARCH) and relative node speed (SPEED), from which it is evident that we are using a cluster of five identical nodes. The `quit` command exits from the PVM command shell, leaving all `pvmd3` daemons running and ready on all hosts.

```

host_1 -> pvm
pvm> conf
1 host, 1 data format

                                HOST  DTID  ARCH  SPEED
                                host_1 40000 RS6K   1000

pvm> add host_2 host_3 host_4 host_5
4 successful

                                HOST  DTID
                                host_2 80000
                                host_3 c0000
                                host_4 100000
                                host_5 140000

pvm> conf
5 hosts, 1 data format

                                HOST  DTID  ARCH  SPEED
                                host_1 40000 RS6K   1000
                                host_2 80000 RS6K   1000
                                host_3 c0000 RS6K   1000
                                host_4 100000 RS6K   1000
                                host_5 140000 RS6K   1000

pvm> quit
pvmd still running.

```

At this point the calculation can be started by launching the master program and afterwards the situation can be monitored entering again the `pvm` command shell environment. Below we report a sample session in which the calculation is started with a background execution command and , entering the `pvm` command shell, one can verify that there is one slave process, `bd_slave`, running on each of the five configured nodes, plus the master process which is running on the local node,

host\_1.

```
host_1 -> bd_master < ox.d > ox.t &
[1] 19726
host_1 -> pvm
pvmd already running.
pvm> ps a

                HOST      TID  FLAG  COMMAND
                host_1    40002  0x4c  -
                host_1    40003  0x4c  bd_slave
                host_2    80001  0x4c  bd_slave
                host_3    c0001  0x4c  bd_slave
                host_4    100001  0x4c  bd_slave
                host_5    140001  0x4c  bd_slave

pvm> quit

pvmd still running.
```

For further inspection the fortran codes for the master *bd\_master.f* and slave *bd\_slave.f* routines are given below, after the bibliography. Uninteresting pieces of code have been omitted and replaced by simple explanations marked by “...” both at the beginning and at the end of the omission.

1. D. Chandler, *J. Chem. Phys.* **68**, 2959 (1978).
2. R. Kapral, *Adv. Chem. Phys.* **48**, 71 (1981).
3. G.M.Torrie and J.P.Valleau, *J. Comp. Phys.* **23**, 187 (1977).
4. J. Keck, *Disc. Far. Soc.* **33**, 173 (1962).
5. J. Keck, *Adv. Chem. Phys.* **13**, 173 (1967).
6. J. B. Anderson, *J. Chem. Phys.* **58**, 4684 (1973).
7. C. H. Bennett, in *Diffusion in Solids:Recent Developments*, Academic Press, 1975.
8. C. H. Bennett, in *Algorithms for Chemical Computation, ACS Symposium Series No. 46*, edited by R. E. Christofferson, p. 63, American Chemical Society, Washington, DC, 1977.
9. J. Dongarra, G. A. Geist, R. Manckek, and V. S. Sunderam, *Computers in Physics* **7**, 166 (1993).
10. T. Yamamoto, *J. Chem. Phys.* **33**, 281 (1960).
11. J. T. Hynes, in *The Theory of Chemical Reaction Dynamics, Vo. IV*, edited by M. Baer, p. 171, CRC, Boca Raton, Fl, 1985.
12. P. Hänggi, P. Talkner, and M. Borkovec, *Rev. Mod. Phys.* **62**, 251 (1990).
13. G. A. Voth, D. Chandler, and W. H. Miller, *J. Phys. Chem.* **93**, 7009 (1989).
14. S. S. W.H. Miller and J. Tromp, *J. Phys. Chem.* **79**, 4889 (1983).
15. S. R. D. Groot and P. Mazur, *Non-Equilibrium Thermodynamics*, North-Holland, Amsterdam, 1962.

16. P. Glansdorff and I. Prigogine, *Thermodynamic Theory of Structure, Stability and Fluctuations*, Wiley, New York, 1971.
17. R. Kapral, *J. Chem. Phys.* **56**, 1842 (1972).
18. H. Mori, *Prog. Theor. Phys.* **33**, 423 (1965).
19. D. A. McQuarrie, *Statistical Mechanics*, chapter 22, p. 574, Harper & Row, New York, 1976.
20. T. L. Hill, *Statistical Mechanics*, McGraw-Hill, New York, 1956.
21. R. Kubo, in *Many-Body Problems*, edited by W. E. P. et al., p. 235, W. A. Benjamin, New York, 1969.
22. E.A.Carter, G.Ciccotti, J.T.Hynes, and R.Kapral, *Chem. Phys. Letters* **156**, 472 (1989).
23. T. C. Bradbury, *Theoretical Mechanics*, Wiley-Interscience, New York, 1968, Chapter 11.
24. H. Goldstein, *Classical Mechanics, 2nd edition*, Addison-Wesley, Reading, Mass., 1980.
25. M. Fixman, *Proc. Natl. Acad. Sci. U.S.A.* **71**, 3050 (1974).
26. N. G. van Kampen, *Appl. Sci. Res.* **37**, 67 (1981).
27. N. G. van Kampen and J. J. Lodder, *Am. J. Phys.* **52**, 419 (1984).
28. D. Chandler and B. J. Berne, *J. chem. Phys.* **71**, 5386 (1979).
29. M. Ruiz and D. Frenkel, , private communication (1994).
30. M. J. Gillan, H. J. Harding, and R. J. Tarento, *J. Phys. C: Solid State Phys.* **20**, 2331 (1987).
31. E. Paci and G. Ciccotti, *J. Phys.: Condens. Matter* **4**, 2173 (1992).
32. G. Ciccotti, M. Ferrario, J. T. Hynes, and R. Kapral, *Chem. Phys.* **129**, 241 (1989).
33. G. Ciccotti, M. Ferrario, J. T. Hynes, and R. Kapral, *J. Chem. Phys.* **93**, 7137 (1990).
34. E. Guardia, R. Rey, and J. Padro, *J. Chem. Phys.* **95**, 2823 (1991).
35. M. Madhusoodana and B. L. Temble, *J. Phys. Chem.* **98**, 7090 (1994).
36. D. Laria and R. Fernández-Prini, *Chem. Phys. Lett.* **205**, 260 (1993).
37. D. Laria and R. Fernández-Prini, *J. Chem. Phys.* **102**, 7664 (1995).
38. J. M. Depaepe, J. P. Ryckaert, E. Paci, and G. Ciccotti, *Mol. Phys.* **79**, 515 (1993).
39. D. W. Robertus, B. J. Berne, and D. Chandler, *J. Chem. Phys.* **70**, 3395 (1979).
40. R. O. Rosemberg, B. J. Berne, and D. Chandler, *Chem. Phys. Lett.* **75**, 162 (1980).
41. N. Go and H. A. Scheraga, *Macromolecules* **9**, 535 (1976).
42. J. P. Valleau, in *Computer Simulations in Material Science*, edited by M. Meyer and V. Pontikis, p. 67, NATO ASI Series No. 205, 1991.
43. P. Pechukas, *Phys. Rev.* **181**, 166 (1969).

44. P. Pechukas, *Phys. Rev.* **181**, 174 (1969).
45. J. C. Tully and R. K. Preston, *J. Chem. Phys.* **55**, 562 (1971).
46. J. C. Tully, *J. Chem. Phys.* **93**, 1061 (1990).
47. S. Hammes-Schiffer and J. C. Tully, *J. Chem. Phys.* **101**, 4657 (1994).
48. D. F. Coker, in *Computer Simulation in Chemical Physics*, edited by M. P. Allen and D. J. Tildesley, p. 315, NATO ASI Series No. 397, Kluwer, 1993.
49. F. Webster, P. J. Rossky, and P. A. Friesner, *Comp. Phys. Comm.* **63**, 494 (1991).
50. D. Laria, G. Ciccotti, M. Ferrario, and R. Kapral, *J. Chem. Phys.* **97**, 378 (1992).
51. D. Borgis, G. Tarjus, and H. Azzouz, *J. Chem. Phys.* **96**, 3188 (1992).
52. D. Li and G. Voth, *J. Phys. Chem.* **95**, 10425 (1991).
53. J. Lobaugh and G. Voth, *Chem. Phys. Lett.* **198**, 311 (1992).
54. S. Consta and R. Kapral, *J. Chem. Phys.* **101**, 10908 (1994).
55. R. P. Feynman, *Phys. Rev.* **56**, 340 (1939).
56. R. P. Feynman, *Statistical Mechanics*, Addison Wesley, Reading, 1972.
57. D. Chander and P. G. Wolynes, *J. Chem. Phys.* **74**, 4078 (1981).
58. M. J. Gillan, *J. Phys. C: Solid State Phys.* **20**, 3261 (1987).
59. J. Cao and G. Voth, *J. Chem. Phys.* **100**, 5093 (1994).
60. D. Laria, G. Ciccotti, M. Ferrario, and R. Kapral, *Chem. Phys.* **180**, 181 (1994).
61. M. Ferrario, D. Laria, G. Ciccotti, and R. Kapral, *J. of Mol. Liquids* **61**, 37 (1994).
62. S. A. Allison, R. J. Bacquet, and J. A. McCammon, *Biopolymers* **27**, 251 (1988).
63. K. Sharp, R. Fine, K. Schulten, and B. Honig, *J. Phys. Chem.* **91**, 3624 (1987).
64. J. J. Sines, S. A. Allison, and J. A. McCammon, *Biochemistry* **29**, 9403 (1990).
65. A. Sergi, M. Ferrario, F. Polticelli, P. O'Neill, and A. Desideri, *J. Phys. Chem.* **98**, 10554 (1994).
66. G. Fiumara, A. Sergi, G. Caristi, M. Ferrario, F. Polticelli, and A. Desideri, *Brownian Dynamics of Diffusion Controlled Reactions. A Parallel Approach Using PVM on a RISC-Cluster*, Technical Report RT 1/204, CNR-PF "Sistemi Informatici e Calcolo Parallelo", 1994.

```

*****
*  bd_master.f
*  Master Program using dynamical load balancing approach
*****
    program bd_master
    implicit real(a-h,o-z)
c ** pvm and local include files **
    include 'fpvm3.h'
    include 'bd_master.fh'
c ** define executable for the slave process **
    exe= 'bd_slave'
c ** read all input data **
    ... read from unit 5 values of all variables:
c ** read number of available processors and number of requested trajectories **
    nproc ntrj,
C ** read other input data
    ofile,idum,dt(1),nprint,rad1,rad2,df,ox,oy,oz,
    radrea,qion,cuix,cuiy,cuiz,cu2x,cu2y,cu2z ...
c ** do the calculations needed to start up: the initial condition for all **
c ** requested trajectories are generated by the master program sampling **
c ** uniformly random points on the surface of a sphere of radius rad1 **
    do j = 1,ntrj
        cte=-1.+2.*ranf()
        te=acos(cte)
        fi=pi2*ranf()
        x(j)=rad1*cos(fi)*sin(te) + ox
        y(j)=rad1*sin(fi)*sin(te) + oy
        z(j)=rad1*cte          + oz
    enddo
c ** seeds for random number generator selected for each slave process **
    midum(1)=idum
    do i=1,nproc-1
        midum(i+1)=midum(i)+int(0.5*(real(idum)*ranf()))
    enddo
c ** BEGINNING OF PVM INSTRUCTIONS **
c ** get identification for master process **
    call pvmfmytid(mytid)
    write(6,*) 'TID_MASTER = ',mytid
c ** start up slave processes on available processors **
    call pvmfspawn(exe,0,'*',nproc,itid,info)
    write(6,*) 'PVM_SPAWN = ',info
    write(6,*) 'itid = ',(itid(i),i=1,nproc)
c ** get identification from each slave process **
    do i = 1,nproc
        call pvmfrecv(itid(i),200,info)
        call pvmfunpack(integer4,ntid,1,1,info)
        write(6,*) i,itid(i),ntid
    enddo

```

```

        enddo
c ** send startup data to the slave processes **
c ** send random number generator seeds individually **
    do j = 1,nproc
        call pvmfinit send(PVMDEFAULT,info)
        call pvmfpack(integer4,j,1,1,info)
        call pvmfpack(real4,midum(j),1,1,info)
        call pvmf send(itid(j),100,info)
    enddo
c ** send other values globally using cast **
    call pvmfinit send(PVMDEFAULT,info)
    call pvmfpack(string,ofile,5,1,info)
    call pvmfpack(integer4,nprint,1,1,info)
    call pvmfpack(real4,cu1x,1,1,info)
    call pvmfpack(real4,cu1y,1,1,info)
    call pvmfpack(real4,cu1z,1,1,info)
    call pvmfpack(real4,cu2x,1,1,info)
    call pvmfpack(real4,cu2y,1,1,info)
    call pvmfpack(real4,cu2z,1,1,info)
    call pvmfpack(real4,df,1,1,info)
    call pvmfpack(real4,dh,ifive,1,info)
    call pvmfpack(real4,dkt,ifive,1,info)
    call pvmfpack(real4,dt,ifive,1,info)
    call pvmfpack(real4,ox,1,1,info)
    call pvmfpack(real4,oy,1,1,info)
    call pvmfpack(real4,oz,1,1,info)
    call pvmfpack(real4,qion,1,1,info)
    call pvmfpack(real4,rad1,1,1,info)
    call pvmfpack(real4,rad2,1,1,info)
    call pvmfpack(real4,radrea,1,1,info)
    call pvmfmcast(nproc,itid,110,info)
c ** control variable: istop=1 signal end of trajectory integration **
    istop=0
c ** main loop: dispatching trajectories to slaves one by one **
c ** until the number of requested trajectories is completed **
    do while(nsend.lt.ntrj)
c ** waiting READY signal from any one of the slave processes **
        call pvmfrecv(-1,205,info)
        call pvmfunpack(integer4,ltid,1,1,info)
c ** sending control variable to selected slave process **
        call pvmfinit send(PVMDEFAULT,info)
        call pvmfpack(integer4,istop,1,1,info)
        call pvmf send(ltid,115,info)
c ** sending data to initiate a Brownian trajectory **
        nsend = nsend + 1
        call pvmfinit send(PVMDEFAULT,info)
        call pvmfpack(real4,x(nsend),1,1,info)
    enddo

```



```

*****
* bdo_slave.f
* Slave program started by the PVM daemon on each available host.
* Calls subroutines needed for performing brownian dynamics simulation
*****
    program bd_slave
c ** pvm and local include file **
    include 'fpvm3.h'
    include 'bd_slave.fh'
c ** declare external subroutines **
    external frcmap
    external move
    external start
c ** get identification for local slave and master process **
    call pvmfmytid(mytid)
    call pvmfparent(mtid)
c ** send local identification to master process **
    call pvmfinit send(0,info)
    call pvmfpack(integer4,mytid,1,1,info)
    call pvmf send(mtid,200,info)
c ** receive startup data from the master process **
c ** receive individual random number generator seed **
    call pvmfrecv(mtid,100,info)
    call pvmfunpack(integer4,mynum,1,1,info)
    call pvmfunpack(real4,idum,1,1,info)
c ** receive other values sent globally using cast **
    call pvmfrecv(mtid,110,info)
    call pvmfunpack(string,ofile,5,1,info)
    call pvmfunpack(integer4,nprint,1,1,info)
    call pvmfunpack(real4,cu1x,1,1,info)
    call pvmfunpack(real4,cu1y,1,1,info)
    call pvmfunpack(real4,cu1z,1,1,info)
    call pvmfunpack(real4,cu2x,1,1,info)
    call pvmfunpack(real4,cu2y,1,1,info)
    call pvmfunpack(real4,cu2z,1,1,info)
    call pvmfunpack(real4,df,1,1,info)
    call pvmfunpack(real4,dh,ifive,1,info)
    call pvmfunpack(real4,dkt,ifive,1,info)
    call pvmfunpack(real4,dt,ifive,1,info)
    call pvmfunpack(real4,ox,1,1,info)
    call pvmfunpack(real4,oy,1,1,info)
    call pvmfunpack(real4,oz,1,1,info)
    call pvmfunpack(real4,qion,1,1,info)
    call pvmfunpack(real4,rad1,1,1,info)
    call pvmfunpack(real4,rad2,1,1,info)
    call pvmfunpack(real4,radrea,1,1,info)
c ** initialise local variables **

```

```

...
c ** call routine responsible for the external field calculation **
    call frcmap
c ** call routine responsible for the trajectory initialisation steps **
    call start
c ** main loop: performing trajectory calculation one by one **
c ** until a STOP signal is received from the master process **
    do while(istop.eq.0)
c ** sending READY signal to the master process **
        call pvmfinit send(PVMDEFAULT,info)
        call pvmfpack(integer4,mytid,1,1,info)
        call pvmf send(mtid,205,info)
c ** receiving control variable: istop=1 signal end of calculation **
        call pvmfrecv(mtid,115,info)
        call pvmfunpack(integer4,istop,1,1,info)
        if(istop.eq.0) then
c ** receiving data to perform one Brownian trajectory **
            itraj=itraj+1
            call pvmfrecv(mtid,120,info)
            call pvmfunpack(real4,x,1,1,info)
            call pvmfunpack(real4,y,1,1,info)
            call pvmfunpack(real4,z,1,1,info)
c ** call routine responsible for trajectory integration **
            call move
        else
c ** sending locally accumulated results to the master process **
            call pvmfinit send(PVMDEFAULT,info)
            call pvmfpack(integer4,mytid,1,1,info)
            call pvmfpack(integer4,itraj,1,1,info)
            call pvmfpack(integer4,mimp,1,1,info)
            call pvmfpack(integer4,mreact,1,1,info)
            call pvmfpack(integer4,mreact1,1,1,info)
            call pvmfpack(integer4,mreact2,1,1,info)
            call pvmfpack(integer4,mtend,1,1,info)
            call pvmfpack(real4,tmap,1,1,info)
            call pvmfpack(real4,tmove,1,1,info)
            call pvmf send(mtid,210,info)
        end if
    end do
c ** exiting from PVM **
    call pvmfexit(info)
c **
end

```