Calculation of classical trajectories with a very large time step: Formalism and numerical examples

Roberto Olender and Ron Elber
Department of Physical Chemistry, Department of Biological Chemistry, The Fritz Haber Research Center for Molecular Dynamics, and The Wolfson Center for Applied Structural Biology, The Hebrew University, Givat Ram, Jerusalem 91904, Israel

(Received 8 July 1996; accepted 13 August 1996)

A new algorithm to compute long time molecular dynamics trajectories is presented. The technique is based on the stochastic path integral of Onsager and Machlup. Trajectories of fixed length of time are computed by path optimization between two end points. Modes of motion with periods shorter than the discrete time step are automatically filtered out, making the trajectories stable for almost an arbitrary time step. Several numerical examples are provided, including motions on the Mueller potential and a conformational transition in alanine dipeptide. Paths similar to the usual molecular dynamics trajectories are obtained, employing time steps 100 times larger than those used in straightforward molecular dynamics. © *1996 American Institute of Physics.* [S0021-9606(96)51043-3]

I. INTRODUCTION

Molecular dynamics (MD) simulations are a standard tool in computational studies of biological macromolecules.[1] Thermodynamical variables, as well as time correlation functions and other average dynamical properties can be extracted from the simulations providing valuable atomic information on the system of interest.

It is therefore a misfortune that the time scale of the simulations is severely limited. Nowadays there are clear limits on the dynamics that we can explore, as well as on the conformational space that can be sampled in a single run. For systems of more than 1000 atoms, the present limit is a few nanoseconds. This is a serious drawback considering the typical time scales of motions in proteins. For example, the subnanosecond time scale accessible to atomically detailed simulations covers only extremely rapid processes (e.g., geminate recombination of a ligand and a protein[2]) which are only a small fraction of biochemical phenomena. Allosteric transitions in proteins occur on a time scale of microseconds,[3] and folding of a protein can last for seconds.[4]

Another related limitation of the molecular dynamics field is the necessity of doing the computations precisely, even if high precision is not required nor warranted. It is not possible to use molecular dynamics approximately, while still maintaining the complete atomic level picture of the system. In contrast, electronic structure calculations suggest a hierarchy of techniques that differ in their accuracy (the simplest example is the variation in the size of the basis set). The hierarchy makes it possible to investigate the molecular system at the desired level of accuracy.

A number of models of proteins that enable the computation of long time dynamics have emerged in the last few years.[5] However, without exception, these models give up on the atomic resolution of the protein and of the solvent. To understand biological processes such as enzymatic reactions,[6] or processes involving specific interactions of a few amino acids, atomic models are required. Therefore, there is a need for approximate methodologies that do not reduce the number of degrees of freedom.

Another approach to study ''dynamics'' is the reaction path method (RP). There, one seeks a low energy path connecting reactants and products. The calculated path provides a qualitative description of the structural changes as a function of the reaction progress. We define ''reaction progress'' as a coordinate displacement along the reaction path from reactants to products (figure 1). The reaction path is also useful in computations of rates that are based on statistical approaches such as the transition state theory (TST).[7]

The reaction path approach does not suffer from the time scale limitation of molecular dynamics, since explicit time is not employed. The computational effort is determined primarily by the roughness of the energy surface and by the coordinate curvature. The lower the roughness and the curvature of the computed path, the larger the change in the structure that we can examine in a single step.

The potential energy is expected to be a slowly varying function of the reaction coordinate (at least compared to other degrees of freedom in the system). It is therefore expected that the energy roughness along the reaction path is much lower than the roughness sampled by a regular MD trajectory. Hence, the ''step'' in RP, in which the fast motions are damped, can be larger than in MD.

The obvious drawback of the reaction path approach is the indirect connection to the real motions of the system and to the rate. One commonly used approach to connect the RP to the rate of the process is the transition state theory[7] that was mentioned earlier. In practical applications of the TST, we have to rely on the assumption that real trajectories pass in the neighborhood of the reaction path and to further rely on statistical estimates of the weights of different configurations. In other words, we have to assume local equilibrium *at least* on a surface dividing reactants and products. This is a strong assumption that is hard to justify in the general case, e.g., in systems that react far from equilibrium.

Moreover, an application of the reaction path approach

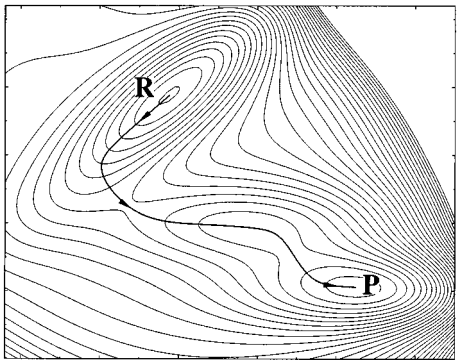

FIG. 1. A schematic presentation of a reaction coordinate (thick line with arrows) between reactant (R) and product (P). The Mueller potential energy surface (Ref. 22) is used in this example.

and the transition state theory (TST) in condensed phase problems must address an additional problem of path multiplicity. In condensed phases many minimum energy paths leading from reactants to products are possible. It is likely that many of these are relevant and contribute significantly to the rate of the process. Searching and identifying the different paths can be a significant computational burden.[8] It is not at all obvious that different paths are in thermal equilibrium with each other on the time scale of the reaction. Rapid exchange between paths is assumed, at least implicitly, when a single reaction coordinate is employed.

To summarize the discussion up to now, atomic detailed simulations of biological macromolecules can be pursued today either by computing molecular dynamics trajectories or by computation of reaction paths. Molecular dynamics trajectories are severely restricted in their length of time (to a few nanoseconds), and computed reaction paths are difficult to interpret and to convert to experimentally accessible information. It is therefore desirable to have a computational technique that could, in some way, benefit from the advantages of both, providing plausible paths of slow processes with a direct estimate of the time of the trajectory (i.e., providing the coordinates as a function of time).

In the present paper an approximate computational technique that computes molecular dynamics trajectories for long times is suggested. The proposed algorithm employs computational techniques similar to the approach which was developed by Czerminski and one of the authors to calculate reaction coordinates.[9] The computational algorithm is also similar in spirit to the work of Gillian *et al.*[10] and Cho *et al.*[11] In a related protocol it uses optimization of functionals to compute classical trajectories. However, our functional is different, and (as we shall argue), more useful for extending time scale of simulations.

This paper is organized as follows. In the next section the Onsager-Machlup (OM) action[12] is reviewed and rederived in the context of the present application. The OM action is the base for the suggested computational method. The numerical implementation of the OM action, formal properties of the algorithm, and its dependence on the time step are described in section III. Numerical examples are found in section IV, and conclusions and final remarks in section V.

Throughout, the following notations are used: A vector is denoted by a bold face and a matrix by an underline. Differentiation with respect to a vector denotes the corresponding gradient. Higher order differentiation corresponds to the appropriate tensor of derivatives.

II. THEORY

We consider a trajectory for which the total time — t_{end} — and the starting and the ending positions are known and are denoted by the vector coordinates $\mathbf{R}(t_i)$ and $\mathbf{R}(t_f)$ ($t_f - t_i = t_{end}$). $\mathbf{R}(t)$, the path that connects the two end points, is the target of the calculations. This is different from what is usually done in molecular dynamics simulations, in which the coordinates and the velocities are specified at some initial time.

If the starting and the ending points of the process are known, then there is no reason not to use this information (if we knew how). The known final point can add significantly to the stability and to the accuracy of the computed trajectory. We are not aware of a known way of using this information in molecular dynamics trajectories to obtain time dependent data. It is difficult, if not impossible, to ''hit'' a desirable product using a condensed phase MD trajectory. Examples of cases in which starting and ending points are known include ordinary chemical reactions with prespecified reactants and products, and conformational changes within a single molecule.

On the other hand, when the end points are not known molecular dynamics is the prime choice. An example of a problem that is difficult to investigate by a boundary value formulation is that of protein folding.

Given that a formulation as a boundary value problem is suitable, the dynamics of the system is conveniently described in terms of a conditional probability. That is, if the starting point is at the coordinate vector $\mathbf{R}(t_i)$, we ask: What is the probability of ending at the coordinate vector $\mathbf{R}(t_f)$ after time t_{end}? The answer — the conditional probability — is denoted by $P(\mathbf{R}(t_i)|\mathbf{R}(t_f);t_{end})$. Questions that are related to the rate can be addressed by computations employing the above conditional probability. Appropriate formulae and numerical examples will be described in forthcoming publications. In this paper we focus on finding a trajectory or trajectories that maximize the above probability. Hence, the goal of the research is to find most probable trajectories that start and end at given positions and also have a fixed length of time — t_{end}.

It will be shown how a time step which is larger by orders of magnitude than the time step employed by ordinary molecular dynamics can be implemented in the scheme. The inevitable result of such a large time step is an approximate trajectory. Nevertheless, we shall demonstrate that the trajectory is stable and quite accurate. This is in contrast to regular MD trajectories that rapidly loose precision and stability as the time step increases (see also figure 4).

As a result of a finite (and large) time step that is used in the numerical integration, the solution of the trajectory is not exact. For special cases it is possible to estimate the errors in the trajectory by comparing approximate solutions to the exact answer. In Appendix A we provide such an estimate for the harmonic oscillator. This estimate is useful in establishing a certain weight or a probability for a given approximate trajectory.

Alternatively, it can be argued that the approximate trajectories generated by the numerical computations do not satisfy exactly the *differential* equations of motion. We therefore write for the approximate solution of Newton's equations

$$\underline{M}d^2\mathbf{R}_{app}/dt^2 + dU/d\mathbf{R}_{app} = \boldsymbol{\varepsilon}(t), \tag{1}$$

where $\mathbf{R}_{app}(t)$ is the approximate solution of the trajectory $\mathbf{R}(t)$, U is the potential energy function, and $\boldsymbol{\varepsilon}$ is a vector of error components for each of the individual coordinates. $\underline{M}$ is the diagonal mass matrix.

The essential hypothesis of the present study is that the errors $\boldsymbol{\varepsilon}(t)$ are sampled in time from identical and uncorrelated Gaussian distributions. The width of the Gaussian decreases as the time step is reduced. The Gaussian hypothesis is an ad-hoc assumption which we cannot and do not attempt to justify for the general case. Nevertheless, we shall show that this assumption can carry us quite far. The first two moments of the error distribution are given by

$$\langle \boldsymbol{\varepsilon}(t)\rangle = \mathbf{0}; \langle \boldsymbol{\varepsilon}(t)\cdot\boldsymbol{\varepsilon}(t')\rangle = \sigma^2\delta(t-t'), \tag{2}$$

where $\langle ... \rangle$ denotes an ensemble average, $\delta(t-t')$ is the Dirac delta function, and σ^2 is the standard deviation of the distribution of the errors.

In the limit of small deviations from the exact trajectory, a quadratic expansion (and the Gaussian) is the leading term. The Gaussian distribution is widely used for the estimation of errors, especially when additional information is not available.

We obviously would like to compute $\mathbf{R}_{app}$ in the best way that we can. When we make the hypothesis about the errors, there is clear danger that the choice of errors will influence the quality of the solution. It is therefore important to provide information comparing the properties of $\mathbf{R}_{app}$ to the exact trajectory. We shall do it later in this paper.

The same functional form of the conditional probability density of trajectories would have been obtained if we had used the Langevin equation. In the Langevin equation, and in the limit of short times, the random force gives rise to a Gaussian distribution around the exact trajectory. In spite of the apparent similarity to the Langevin equation, we emphasize that the formalism described below does not have an explicit randomness or phenomenological stochastic term as in the Langevin equation. Rather, Newton's equations of motion are directly used. Our errors are assumed to be a result of the discretization of the trajectory, and they can be estimated by direct comparison to an exact trajectory (see Appendix A).

Returning to the approximate solution of the differential equations of motion, the computation of ''short time'' dynamics is considered. To construct a ''short time'' trajectory, it is separated into two components: a deterministic part which is our zero order guess of where the system would like to go, and a random component sampled from a Gaussian distribution of errors. Since the discrete time step leads to an approximate solution, a volume of uncertainty is added to the neighborhood of the computed new location. The size of the volume should reflect and be proportional to an estimate of the errors in our integrator.

The first component of the solution after a time step Δt is $\mathbf{R}_g(t_i+\Delta t)$. $\mathbf{R}_g(t_i+\Delta t)$ is obtained from $\mathbf{R}(t_i)$ in a deterministic calculation, i.e., $\mathbf{R}_g$ is the guessed coordinate vector at the center of the Gaussian. It is the solution when $\boldsymbol{\varepsilon}$ is zero, where $\boldsymbol{\varepsilon}$ is the error vector that was defined in equation (2). In a discrete time representation, the error correlation [equation (2)] is approximated by

$$\langle \boldsymbol{\varepsilon}(t_i)\cdot\boldsymbol{\varepsilon}(t_j)\rangle = \sigma^2\delta(t_i-t_j) \approx \sigma^2\delta_{ij}/\Delta t. \tag{3}$$

The next expression of interest is the probability of ending after a time interval Δt at $\mathbf{R}(t_i+\Delta t)$, if at time t_i the system was at $\mathbf{R}(t_i)$. The above inquiry defines a conditional probability which for the problem at hand can be written as

$$P(\mathbf{R}(t_i)|\mathbf{R}(t_i+\Delta t);\Delta t) = [1/(2\pi\langle\varepsilon^2\rangle)]^{d/2}\exp[-(\mathbf{R}(t_i+\Delta t)-\mathbf{R}_g(t_i+\Delta t))^2/(2\langle\varepsilon^2\rangle)]. \tag{4}$$

The difference vector $\mathbf{R}(t_i+\Delta t)-\mathbf{R}_g(t_i+\Delta t)$ is exactly $\boldsymbol{\varepsilon}$ (d is the system dimensionality). The term in the exponent is therefore proportional to Δt [see equation (3)].

A long time solution can be obtained by multiplying many short time conditional probabilities and summing up over all the intermediate positions. Since t_f is the final time of the desired trajectory—$t_f = t_i + N\Delta t$, we write

$$P(\mathbf{R}(t_i)|\mathbf{R}(t_f);N\Delta t) = \int\{\prod_j d\mathbf{R}_j\}\{\prod_k P(\mathbf{R}(t_k)|\mathbf{R}(t_k+\Delta t);\Delta t)\}. \tag{5}$$

In the limit in which the number of points approaches infinity and Δt is infinitesimal we obtain a path integral.[13] It is also convenient to accumulate all the Gaussians into a single exponential function, that is,

$$\prod_i P(\mathbf{R}(t_i)|\mathbf{R}(t_i+\Delta t);\Delta t) \propto \exp\left[-(\Delta t/2\sigma^2)\sum_j (\mathbf{R}(t_j+\Delta t)-\mathbf{R}_g(t_j+\Delta t))^2\right] \approx \exp\left[-(1/2\sigma^2)\int dt\boldsymbol{\varepsilon}(t)^2\right]. \tag{6}$$

The above result is quite striking, representing the weight of a trajectory (or the probability that it is correct) as the square of the amplitude of the transient noise. It is, of course, not surprising that the trajectory is most accurate when the noise is zero. The real question is how to proceed from this point in order to get the desired trajectory.

The above formula of the weight of an individual trajectory was derived by Onsager and Machlup[12] in the context of irreversible thermodynamics and stochastic trajectories. The most probable trajectory is found when the transient noise is zero, everywhere along the trajectory. This means that the usual deterministic equations of motions are satisfied everywhere in time and that they are ''error free.'' In the present case these are Newton's equations of motion. Hence, the weight for a *single* trajectory can be also written based on equation (4) as

$$P(\mathbf{R}(t_i)|\mathbf{R}(t_f);\mathbf{R}(t)) \propto \exp\left[-1/(2\sigma^2)\int dt(\underline{M}d^2\mathbf{R}/dt^2+dU/d\mathbf{R})^2\right]. \tag{7}$$

To obtain the above equation, $\varepsilon(t)$ in equation (6) was replaced by the left hand side of equation (1). Note, that if the precision was infinite ($\sigma^2=0$), only a single trajectory would remain. If the complete conditional probability with pre-determined σ^2 is desired, then all possible trajectories must be summed up to obtain a path integral[13]

$$P(\mathbf{R}(t_i)|\mathbf{R}(t_f);t_f-t_i) \propto \int D\mathbf{R}(t)\exp\left[-1/(2\sigma^2)\int dt(\underline{M}d^2\mathbf{R}/dt^2+dU/d\mathbf{R})^2\right] = \int D\mathbf{R}(t)\exp\{-1/(2\sigma^2)\,S[\mathbf{R}(t)]\}, \tag{8a}$$

$$S[\mathbf{R}(t)]\equiv\int dt(\underline{M}d^2\mathbf{R}/dt^2+dU/d\mathbf{R})^2, \tag{8b}$$

where $D\mathbf{R}(t)$ denotes path integral summation,[13] and $S[\mathbf{R}(t)]$ is the Onsager-Machlup action,[12] a functional of the path $\mathbf{R}(t)$. The most probable trajectory is obtained, when the Onsager-Machlup action is minimal. In this paper we suggest that optimization of S can be employed to compute molecular dynamics trajectories.

Before analyzing further the properties of the Onsager-Machlup action, it is useful to discuss alternatives. The Onsager-Machlup (OM) action is one of the possible functionals from which one may compute molecular dynamics trajectories. One may ask what is the advantage of the present formulation. Consider the well known classical action

$$S_{cl}=\int[1/2(d\mathbf{R}/dt)\underline{M}(d\mathbf{R}/dt)-U(\mathbf{R})]dt. \tag{9}$$

Here a classical trajectory is the path $\mathbf{R}(t)$ for which S is stationary.[14] This is already a difficulty, since a stationary solution can also be a maximum or a saddle point, which are unstable with respect to small perturbations. It is considerably more difficult to locate a saddle point than to identify a minimum. In the optimization of the OM action we seek a minimum; the minimum corresponding to a trajectory with the highest conditional probability. Furthermore, the OM action is a non-negative functional which is bound by zero. This bound makes the identification of the global minimum easier, since the lowest possible value of the OM action is zero.

A second point, why the OM numerical computations are advantageous compared to calculations that employ the classical action, is related to the representation of the time integral and to its stability upon optimization. The simplest approach (which we employed to deal with systems with many atoms) is to use a discrete time step. For the classical action we have

$$S_{cl}=\sum_i \Delta t[1/(2\Delta t^2)(\mathbf{R}_{i+1}-\mathbf{R}_{i-1})\underline{M}(\mathbf{R}_{i+1}-\mathbf{R}_{i-1})-U(\mathbf{R}_i)], \tag{10}$$

where the indices refer to the discrete time. Numerical optimization of S_{cl} is with respect to all its independent variables $\mathbf{R}_j$. That is, we search for a minimum of S_{cl} along the gradient of S_{cl}, $\{\partial S_{cl}/\partial\mathbf{R}_j\}$. To demonstrate the computational difficulties associated with a large time step (in the framework of the classical action optimization) consider the one dimensional harmonic oscillator

$$S_{cl}=\sum_i \Delta t[(1/2m)(X_{i+1}-X_i)^2/\Delta t^2-1/2\,kX_{i+1}^2], \tag{11}$$

where X is the displacement from the equilibrium position, m the oscillator mass and k the force constant. At a finite Δt the distance between sequential points in time is determined by opposing factors. The kinetic energy term is holding X_{i+1} and X_i together while the potential energy term is an inverted parabola which upon minimization pushes the points away. It is therefore evident that as Δt increases, S_{cl} changes from a function with a minimum to a function with a maximum, making a stable calculation for a variety of time steps more difficult.

In contrast, the Onsager-Machlup action, even in its discrete form, is non-negative. It therefore has a minimum regardless of the size of the time step

$$S=\sum_i \Delta t[(\underline{M}/\Delta t^2)(\mathbf{R}_{i+1}+\mathbf{R}_{i-1}-2\mathbf{R}_i)-dU/d\mathbf{R}_i]^2. \tag{12}$$

Caution must be exercised when comparing the classical and the OM action. At the first glance it seems that both formulations lead to Newton's equations of motion. This is not exactly so. For analysis purpose (as opposed to the computational goal of the present manuscript) it is useful to consider now the functional variation of either of the actions—S or S_{cl}—with respect to the path—$\mathbf{R}(t)$, that is, $\delta S/\delta\mathbf{R}(t)$.[15]

Following the well known Euler-Lagrange procedure, the classical action provides Newton's equations of motion[14]

$$\underline{M}d^2\mathbf{R}/dt^2+dU/d\mathbf{R}=\mathbf{0} \tag{13}$$

while the equations of motion that are derived from the OM action in a similar fashion are of higher order

$$\underline{M}d^4\mathbf{R}/dt^4+2\underline{M}(d^2U/d\mathbf{R}^2)d^2\mathbf{R}/dt^2+\underline{M}(d^3U/d\mathbf{R}^3) \times(d\mathbf{R}/dt)^2+(d^2U/d\mathbf{R}^2)(dU/d\mathbf{R})=\mathbf{0}. \tag{14}$$

The connection between equations (13) and (14) can be demonstrated with the help of **Q** defined below

$$\mathbf{Q} = \underline{M}d^2\mathbf{R}/dt^2 + dU/d\mathbf{R}. \tag{15}$$

Equation (14) is now decomposed into two more instructive equations

$$\underline{M}d^2\mathbf{R}/dt^2 + dU/d\mathbf{R} = \mathbf{Q}, \tag{16a}$$

$$\underline{M}d^2\mathbf{Q}/dt^2 + (d^2U/d\mathbf{R}^2)\mathbf{Q} = \mathbf{0}. \tag{16b}$$

It is therefore evident that the OM equations are more general than Newton's equations of motion. Only in the special case in which **Q** is identically zero [and therefore also S is zero, see equation (17) below], the two equations of motion coincide. The OM calculation requires more boundary conditions than are required by the classical equations of motion (after all it is a fourth order differential equation). The boundary conditions that we employed are of fixed end points and of zero **Q** at the edges of the path.

In order to develop some understanding of equation (14), it is useful to return to the good old one dimensional harmonic oscillator. The fourth order differential equation yields four solutions. The two solutions obtained when **Q** is set identically to zero are also the solutions of Newton's equations of motion: $\mathbf{R}_1 = \mathbf{A}\cos(\omega t)$ and $\mathbf{R}_2 = \mathbf{B}\sin(\omega t)$. The other solutions are for non-zero **Q** and are obtained by solving for **Q** first [equation (16b)]. Since $(d^2U/d\mathbf{R}^2)$ is a constant for the harmonic oscillator, the two solutions for **Q** are similar to the above solution to **R**, $\mathbf{Q}_1 = \mathbf{A}'\cos(\omega t)$ and $\mathbf{Q}_2 = \mathbf{B}'\sin(\omega t)$.

Substituting for **Q** in equation (16a) we find that the solutions for **R** are therefore those of a forced harmonic oscillator with a driving force exactly in resonance. These solutions are required if the boundary conditions do not conserve energy and it is necessary to feed energy into the system in order to satisfy the boundary conditions. In this case, the errors (and **Q**) fulfill a similar role to the random force of the Langevin equation that heats up the system.

Note that the OM action can also be written as

$$S = \int dt(\mathbf{Q}\cdot\mathbf{Q}) \tag{17}$$

where $\cdot$ denotes a scalar product. Clearly, when **Q** is zero, the value of S is minimal, and we avoid picking the second solution that does not conserve the energy. Therefore, significant effort must be devoted to find the lowest minimum of S that money (and computer time) can buy.

As discussed above, the goal which we keep in mind is the possibility of computing trajectories with a very large time step. Equation (12) is the discretized form employed in the calculations. While it is stable for an arbitrary time step, it is nevertheless rather primitive, and uses a simple finite difference formula to estimate the time derivatives. The obvious questions are how accurate is it for different time steps, and can one get an idea on the type and the order of magnitude of the errors and their dependence on the time step. The discussion below attempts to address this point by examining the effects of the approximations used on the exact representation of the coordinate evolution in time.

As the discussion continues, it will become evident that an exact and analytical estimate of the errors for an arbitrary potential is not available. However, we gained some understanding of the characteristics of motions that are eliminated when the large time step is employed. At the end of Appendix A it will be outlined how the ideas discussed next can be used in numerical estimates of σ^2.

We consider an exact expansion of the coordinate as a function of time in the interval $[0,\delta t]$

$$\mathbf{R}(\tau) = \mathbf{a} + \mathbf{b}\tau + 1/2\,\mathbf{c}\tau^2 + \sum_k \mathbf{d}_k \sin(\omega_k\tau);\ (\omega_k = k\pi/\delta t), \tag{18}$$

where **a**, **b**, and **c** are used in a three point expansion of the path to be described below. τ is the time variable. The Fourier expansion in the sine functions describes the deviations of the exact path from the three point expansion. Only the sine expansion (no cosine) is used since **a**, **b**, and **c** satisfy the boundary condition of the coordinates

$$\mathbf{R}(\delta t) = \mathbf{R}_{i+1};\quad \mathbf{R}(0) = \mathbf{R}_{i-1}. \tag{19}$$

Hence, it is possible to write $\mathbf{R}(0) = \mathbf{a}$ and $\mathbf{R}(\delta t) = \mathbf{a} + \mathbf{b}\delta t + 1/2\mathbf{c}\delta t^2$.

For the derivation below it is convenient to set the boundary conditions as close as possible to our three point interpolation. The change in the velocity during the above interval is set to: $\Delta\mathbf{V} = \mathbf{c}\delta t$. The last equation adds a condition on the $\mathbf{d}_k$-s as follows: The magnitude (and the change) of the velocity are estimated employing also the coordinate at $\delta t/2$—$\mathbf{R}(\delta t/2) = \mathbf{R}_i$: $\mathbf{V}(\delta t/4) \cong 2(\mathbf{R}_i - \mathbf{R}_{i-1})/\delta t$ and $\mathbf{V}(3\delta t/4) \cong 2(\mathbf{R}_{i+1} - \mathbf{R}_i)/\delta t$. An estimate for the acceleration is therefore: $2\Delta\mathbf{V}/\delta t = \mathbf{c} = (\mathbf{R}_{i-1} + \mathbf{R}_{i+1} - 2\mathbf{R}_i)/(\delta t/2)^2$. In the spirit of the three point approximation the condition becomes

$$\Delta\mathbf{V} = 2(\mathbf{R}_{i-1} + \mathbf{R}_{i+1} - 2\mathbf{R}_i)/\delta t. \tag{20}$$

For the above to be exact (i.e., when the sine series is also included) another condition must be met

$$\sum_k \mathbf{d}_k\omega_k[\cos(\omega_k\delta t) - \cos(0)] = \mathbf{0}, \tag{21}$$

$\Sigma_{k=1,3,5,\ldots}\mathbf{d}_k(k\pi/\delta t) = \mathbf{0}$ (i.e., the sum is over odd elements only). The analysis of the accuracy of the solution is now continued. It is divided into two steps: First, the time integral using the exact expansion of the coordinates in time is performed. Second, the action S is minimized as a function of all the coefficients—$\{\mathbf{d}_k\}$—in an attempt to estimate their values. Clearly, the larger their values, the poorer the approximation used.

In the integration, an approximation is employed which is consistent with the above three points expansion of the path. The force is set to a constant throughout the interval. The approximate integration and the following analysis suggest how the approximation used behaves numerically. The analytical integration of the forces in time is not possible and therefore less useful. It is the question of the consequence

of this approximation which is our prime interest here. So, in that sense, the calculations below are not a ''true'' error estimate.

In principle, it is possible to go beyond the constant force approximation (see Appendix B). The expressions are likely to be useful for further analytical studies, and for the derivation of an approximate long time conditional probability. The formulae are rather messy and are therefore placed in Appendix B. The basic idea, however, is rather simple and is outlined below: The ''fixed force'' $-dU/d\mathbf{R}_i$ is replaced by a linear expansion of the force with the path, i.e., $-dU/d\mathbf{R} \approx -dU/d\mathbf{R}_i - d^2U/d\mathbf{R}_i^2(\mathbf{R}(t)-\mathbf{R}_i)$, where $d^2U/d\mathbf{R}_i^2$ is the second derivative matrix of the potential computed at $\mathbf{R}_i$. Since the integrals are (at most) quadratic with $\mathbf{R}(t)$, they are doable analytically, as is shown in Appendix B.

However, for calculations of a *single optimal trajectory* the computations are complex and numerically intensive. Therefore, the correction terms were not employed in the numerical computations described in section IV. Rather than using higher order expansion of the integrand, the properties of the solution as a function of the size of the time step and the number of grid points were examined.

The exact action integral is written below (by notation, vectors on the left of the matrix are row vectors)

$$\begin{aligned} S &= \int_0^{\delta t} dt(\underline{M} d^2\mathbf{R}/dt^2 + dU/d\mathbf{R})^2 \\ &= \int dt[(d^2\mathbf{R}/dt^2)\underline{M}][(\underline{M}d^2\mathbf{R}/dt^2)] \\ &\quad + 2\int dt[(d^2\mathbf{R}/dt^2)\underline{M}][dU/d\mathbf{R}] \\ &\quad + \int dt[dU/d\mathbf{R}][dU/d\mathbf{R}]. \end{aligned} \tag{22}$$

The first integral on the right hand side can be evaluated exactly

$$\begin{aligned} &\int dt[(d^2\mathbf{R}/dt^2)\underline{M}][(\underline{M}d^2\mathbf{R}/dt^2)] \\ &= \mathbf{c}\underline{M}\underline{M}\mathbf{c}\delta t + \sum_k \mathbf{d}_k \underline{M}\,\underline{M}\mathbf{d}_k (k\pi/\delta t)^4. \end{aligned} \tag{23}$$

The other two integrals cannot be solved in a closed form for a general potential. However, employing the same approximation as the one used in our numerical calculations, i.e., $dU/d\mathbf{R} \approx dU/d\mathbf{R}_i$, the second integral becomes $2\mathbf{c}\underline{M}(dU/d\mathbf{R}_i)$ and the third is $(dU/d\mathbf{R}_i)^2\delta t$. Within our approximation, only the first integral depends on $\mathbf{d}_k$. The contribution is also non negative. That is, a minimum of S with respect to all the $\mathbf{d}_k$ is when $\partial S/\partial \mathbf{d}_k = \mathbf{0}$, which corresponds to $\mathbf{d}_k = 0$. This solution also satisfies the boundary condition (19) and it is therefore the lowest action solution for our problem.

The sine series was employed to capture the motions with frequencies higher than $\approx 1/\delta t$. If we approximate the forces by a constant in this interval, then the solution eliminates (in an automatic way) all these high frequency motions.

Elimination of rapid motions (such as bond stretching) to make study of long time processes possible, has been the target of many algorithms, for example, SHAKE,[16] RATTLE,[17] or the Back Euler (BE)[18] algorithms. The first two approaches are conceptually different from the OM protocol. In the RATTLE or the SHAKE algorithm, a decision of which are the fast and which are the slow coordinates must be made prior to the beginning of the computations. The OM automatically finds the fast and the slow coordinates compared to the time step δt. The BE algorithm removes energy from the fast modes and in that sense it is similar to the optimization of the OM action. However, since the BE is formulated as an initial value problem (as opposed to a boundary value problem in the OM), the gain from the increase in the time step is much smaller in BE than in the OM protocol.

It is the removal of rapid motions that makes the OM action an especially stable algorithm. It is nevertheless clear that as the time step is increased, accuracy is lost. Rigorous estimates of the errors are difficult to make. In Appendix A we consider the error for the simple example of the harmonic oscillator. As the time step increases and passes a critical value, the oscillations are frozen and the errors become the motions that were eliminated. This is in contrast to the molecular dynamics approach, in which large time steps can lead to unbound, exponentially growing errors (see also figure 4).

This concludes the more formal discussion of the paper, that is aimed at introducing a new numerical algorithm.

III. THE NUMERICAL ALGORITHM

In order to compute and to optimize the Onsager-Machlup action, the simplest possible discretization scheme already outlined in equation (12) is employed. S is a function of N intermediate coordinate vectors $\{\mathbf{R}_i\}$. It is minimized using a variety of options, such as the conjugate gradient algorithm with the restart option of Powell[19] and simulated annealing.[20] Multigrid techniques are also applied.[21]

The initiation of the optimization process requires a starting path for the optimization, i.e., a starting set of $\{\mathbf{R}_i\}$. The simplest starting path which we used at the beginning of this research and also in past reaction path calculations[9] is the straight line interpolation. This initial guess is the limit of very high kinetic energy in which the system ''travels'' at a constant speed over the potential energy surface. In this limit, the potential energy surface is irrelevant. This is the expected solution in the limit of a short trajectory time and relatively large distances to be traveled.

Clearly, for sufficiently large number of points (or a sufficiently small time step) we should converge to the ''right'' answer. However, some care must be exercised since the optimization of the trajectory is a global optimization problem. The most probable trajectories are the ones with the globally lowest S.

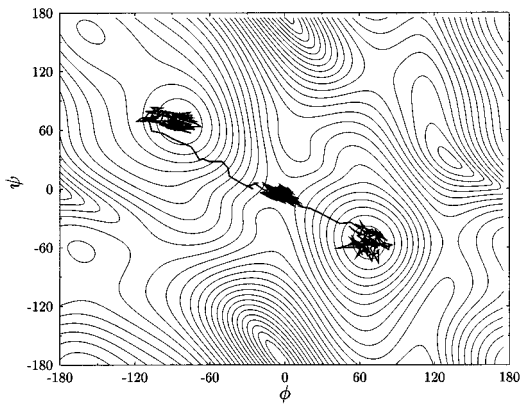

FIG. 2. An example for a local minimum with a high value for the Onsager-Machlup action. A conformational transition in alanine dipeptide is considered. The initial guess for the path was a straight line interpolation from reactants to products. The initial guess was minimized directly using a conjugate gradient protocol. The local minimum that was obtained includes significant delay of the system in the neighborhood of the transition state, (a fixed point), in addition to the more intuitive delays at the different minima.

In practice, the straight line interpolation leads, many times, to local minima of the action S. One phenomenon for which we were not able to provide detailed theoretical analysis is the existence of ''fixed'' points along the path. There are relatively high energy configurations at which the minimized path is ''trapped.'' The path cannot get away from such a coordinate set by a straightforward conjugated gradient minimization (figure 2). There is no reason for a complex function such as S **not** to have fixed coordinates (i.e., a coordinate set $\{\mathbf{R}_i\}$ for which $\delta S/\delta \mathbf{R}_i = 0$, and therefore in direct minimization $\mathbf{R}_i$ will not change). It is therefore important to employ global optimization techniques that are able to get out from a local minimum.

Three different computational setups were tried in order to obtain better paths than those obtained in the calculations that use conjugate gradient and straight line interpolation. The first approach was to use a global minimizer, i.e., simulated annealing. This attempt failed. Simulated annealing was successful in locating different local minima of S. However, it was difficult to ''convince'' the algorithm to look for the minimum with the lowest value of S. Reasonable variations in the length of the simulations and in the cooling protocols were not successful.

In the second trial, different initial conditions for the paths were employed. Rather than a straight line interpolation, we equally divide the intermediate points between the minimum of the reactants and the minimum of the products. In contrast to the straight line interpolation in which the forces component of the action is highly excited in the initial path, in the last setup the time derivative part of the action requires significant relaxation.

In practice, the last protocol produces, many times, low energies and low S paths. It successfully avoids the difficulty of high energy fixed points that was mentioned above. However, using more than one initial guess is a good idea, since the optimization results strongly depend on the underlining energy surface. There is no general ''best'' initial guess for the trajectory.

Another approach that was employed to identify better local minima of the OM action is multigrid.[21] This is an exceptionally important protocol for large systems with different time scales. Multigrid makes it possible to recover the correct behavior of different modes efficiently, in separate calculations. It was found to be most important for the computational example of alanine dipeptide.

The straightforward approach to optimize S is to represent it by a set of N discrete points and to minimize the discrete representation of the functional. In multigrid, a sequence of different grid approximations to the functional are optimized keeping the total trajectory time fixed. Each representation is most effective in recovering part of the modes. For example, when the grid includes only a small number of points, low frequency modes are computed most effectively. The solution of one grid is used to initiate the computation of the next grid.

There is considerable mathematical literature proving that the sequence of optimizations using different grids converges much faster that the optimization of a single grid.[21] The so-called ''V cycle'' was employed in the OM computations. The beginning is a path with high resolution and many grid points which is the upper left corner of the letter ''V.'' This path is optimized until the convergence rate becomes slow.

Next, a path with a lower resolution is constructed from the optimized solution. This is done by selecting (for example) each second grid point. The lower resolution path is optimized and the solution is employed in yet another path with even smaller number of grid points. This process continues until the path with the lowest resolution of the cycle (the lower corner of the ''V'') is reached.

At this point the ''right arm'' of the V is investigated. The resolution is increased in an inverse sequence to the decrease in the resolution. The new coordinate sets are constructed by a linear interpolation between two ''old'' coordinate sets. At each step the time integral is optimized and employed to initiate the next step. The final solution is obtained from the optimization of the upper right corner of the ''V.''

There is still considerable room for improvement in the application. Improvement can be made in the design of better and more effective grids, and development of different transition protocols between the grids. Computational enhancements may include automated algorithms to modify the grid whenever difficulty is encountered. These are topics of future work.

In the next section, a number of numerical examples that differ in their complexity and relevance to biological systems are provided. The numerical calculations are employed to illustrate the analytical properties of the algorithm that were discussed previously, as well as the promise of the algorithm in producing very long time trajectories.

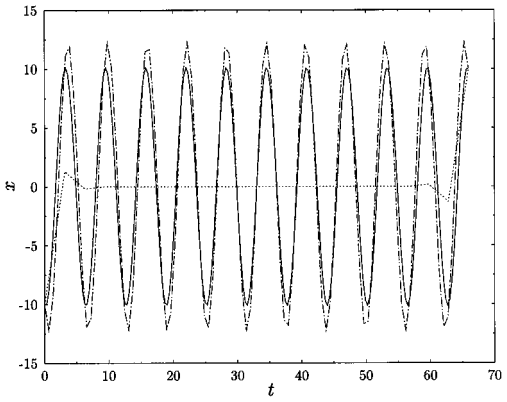

FIG. 3. Optimization of the Onsager-Machlup action for the harmonic oscillator using different time steps. The total time considered includes 10.5 periods of oscillations and different number of grid points were employed to approximate the trajectory. Below, we exclude from the number of configurations the two fixed end points. Thus only the intermediate points are counted: 200 grid points (solid line), 100 grid points (dotted dashed line) and 20 grid points (dotted line). The 200 grid points provide a solution that is essentially exact. Note the complete ''annihilation'' of the oscillation amplitude once the number of points is reduced to 20.

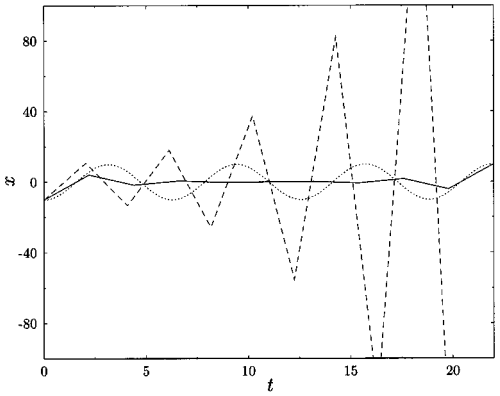

FIG. 4. The stability of the solutions of the OM action optimization and of the Verlet algorithm. The trajectory of the harmonic oscillator is examined as a function of the time step. Three and a half oscillations are considered. The exact solution (dotted line), an OM solution with 11 grid points ($\Delta t=2.2$, solid line) and a solution with the Verlet algorithm ($\Delta t=2.04$, dashed line) are shown. For the Verlet algorithm the initial coordinate is the same as in the OM and the initial velocity is set to zero. Attempts to use a time step for the Verlet algorithm which is larger than 2.04 failed. The solution ''explodes'' immediately.

IV. NUMERICAL EXAMPLES

A. A one dimensional harmonic oscillator

It does not come as a surprise that the harmonic oscillator can be solved analytically also in the Onsager-Machlup framework (see section II). It is therefore convenient to compare the exact solution of the harmonic oscillator to numerically derived approximations. We present the solution as a function of time in figure 3. The units of time are such that $\omega=1$, the mass was set to one, and the boundary coordinates are at ± 10. Throughout the computations presented below the value of **Q** at the boundaries is zero. The initial guess for the path in this case is irrelevant since there is only one solution once four boundary conditions are specified.

As anticipated from our previous analysis of the high frequency modes, when the time step is such that $\Delta t>\pi/\omega$ the oscillations die out. This behavior is not surprising considering the analytical proof we provided in section II. Nevertheless, it is a demonstration of one of the most important features of the OM algorithm, filtering out high frequency motions with periods smaller than the time step employed.

The argument above suggests that the algorithm will be stable under *all conditions*. This is not precisely so and instabilities are possible for a limited range of Δt values. Rarely, we encountered the instabilities in computations. The existence of problematic Δt's can be demonstrated as follows: For the harmonic oscillator the discretized version of the Onsager-Machlup action is quadratic, i.e., S can be written as $S=\Sigma(a_{ij}\mathbf{R}_i\mathbf{R}_j)$. The a_{ij}'s are determined by opposing factors—the time derivatives and the potential derivatives. For a small Δt the time derivatives ''win'' and for a large Δt the potential derivatives are larger. At some intermediate Δt they cancel each other leading to a flat surface of S and possible numerical instabilities. Computations should therefore avoid this narrow window of a time step.

It is of interest to examine at this point the behavior of usual integrators (we employed the velocity Verlet) for time steps that are of the same order as the oscillation period (figure 4). As is evident from the figure, the integrator looses its stability and provides a useless trajectory. This is in contrast to the OM optimization that overdamps modes which it cannot possibly follow.

B. A two dimensional harmonic oscillator

A better way of demonstrating the effectiveness of the optimization in filtering out *only* the high frequency components is to examine a two dimensional harmonic oscillator. We consider an oscillator with two well separated frequencies — $\omega_1=1$ and $\omega_2=5$, i.e., $U(x,y)=1/2(x^2+25y^2)$.

In the computed time interval (figure 5), the slow oscillator finishes 3.5 cycles while the fast oscillator completes 17.5 oscillations. If 500 grid points are employed (excluding the end points), there are about 28 points to describe a fast oscillation, which is more than enough. The fast oscillations are reasonably described. When the number of grid points is reduced to 20 (about 1 point per fast cycle and about 6 points for a slow oscillation) the fast motions are eliminated and only the slow oscillations remain.

As mentioned in section II, the OM action filtering out high frequency motions is a physically appealing approximation. In bond stretching, for example, the spatial deviation of bonds from their equilibrium position is not large. Some quantitative changes in the properties of the trajectories are expected, however, many properties remain unchanged and the computed approximate trajectories are qualitatively similar.

The OM optimization automatically eliminates *all* of the components with frequencies which are larger than $1/\Delta t$.

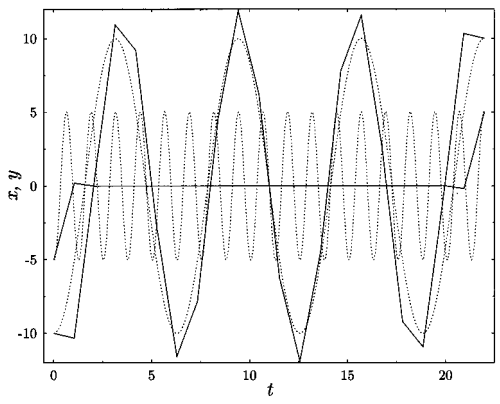

FIG. 5. Optimization of the Onsager-Machlup action for the two dimensional harmonic oscillator. The potential energy employed is $U(x,y)=1/2(x^2+25y^2)$, the mass is 1 and 3.5 slow oscillations are considered. The boundary conditions are $x(0)=-x(t_{end})=-10$ and $y(0)=-y(t_{end})=-5$. Two trajectories are shown. The first consists of 500 intermediate points (dotted lines) which provide an essentially exact solution. The x component corresponds to the slower and larger amplitude motion. The second trajectory employed only 20 intermediate points (solid lines). Note that for the 20 points the fast oscillations are overdamped and frozen while the slow oscillations are reasonably well reproduced.

This is advantageous, since we are not required to identify the fast motions prior to the beginning of the computations. Other features of the dynamics that are difficult to follow using a large time step are rapid jumps between minima and fast passage over barriers. To demonstrate the difficulties and to further advance our understanding of the approximations involved we consider next trajectories on the Mueller potential.

C. The Mueller potential (Ref. 22)

A more ambitious two dimensional model is the Mueller potential (figure 6). It was invented as a nontrivial test for reaction path algorithms since the intrinsic reaction coordinate and the saddle points are somewhat difficult to locate. The Mueller potential is also helpful in demonstrating numerical properties of the Onsager-Machlup optimization protocol for highly anharmonic systems: information that is impossible to obtain from the previous examples. Here we are interested in trajectories between the two extreme minima on the energy surface.

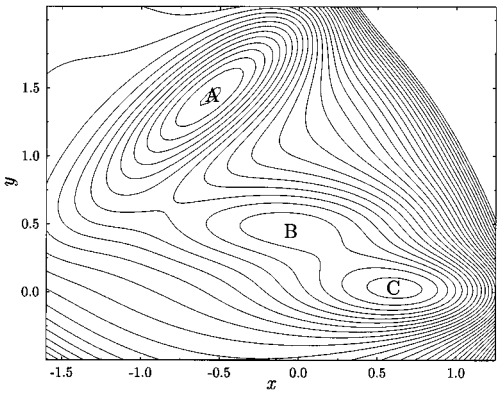

FIG. 6. A contour plot of the Mueller potential (Ref. 22) that we employed as one of the tests of the OM calculations. The potential is of the form: $U(x,y)=\Sigma_{i=1,\ldots,4}A_i \exp[a_i(x-x_i)^2+b_i(x-x_i)(y-y_i)+c_i(y-y_i)^2]$, where the parameters are $\mathbf{A}=(-200,-100,-170,15)$, $\mathbf{a}=(-1,-1,-6.5,0.7)$, $\mathbf{b}=(0,0,11,0.6)$, $\mathbf{c}=(-10,-10,-6.5,0.7)$, $\mathbf{x}=(1,0,-0.5,-1)$, $\mathbf{y}_i=(0,0.5,1.5,1)$ The three local minima are labeled A, B and C. The separation between the contour lines is 15.

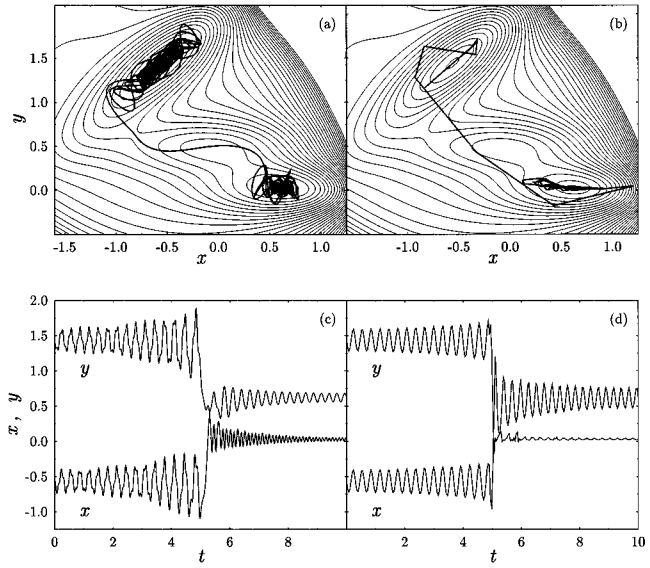

FIG. 7. Onsager-Machlup optimal trajectories on the Mueller potential. The mass was set to 1. The total time of the trajectories was 10 and two different time steps were considered: 0.01 and 0.04. The initial guess for the path was half of the points in minimum A and half of the points in minimum C. The final optimized actions are not zero but equal 136 and 2150, respectively. (a) A display of the OM trajectory with a time step of 0.01 on the contour plot. (b) A display of the OM trajectory with a time step of 0.04 on the contour plot. (c) The x and y coordinates of trajectory (a) as a function of time. (d) The x and y coordinates of trajectory (b) as a function of time.

To start the optimization of the path, half of the grid points (the time intermediates) are placed at the upper left minimum (labeled A in figure 6) and the rest of the points at the lower right minimum (labeled C). At the boundaries, **Q** was set to zero. The optimizations were pursued directly by conjugate gradient minimizations.

In figure 7 we demonstrate the effects of filtering that may occur on a highly anharmonic energy surface. The computations are for a particle with mass 1. Trajectory (a) is a detailed computation with 1000 intermediate points. The particle remains most of the trajectory at the minimum in which it was initiated and the transition between the minima occurs rapidly (see also figure 7(c) in which the different degrees of freedom are plotted as a function of time).

We identify two types of rapid motions in the system. One type corresponds to the vibrations in the wells and the second type to the transition over the barrier. In figures 7(b) and 7(d), we examine the changes in the trajectory after the time step is increased by a factor of four. The size of the new time step is similar to the period of the rapid oscillations in the narrower directions of the two minima and to the time scale required to pass the barrier. The changes in the trajec-

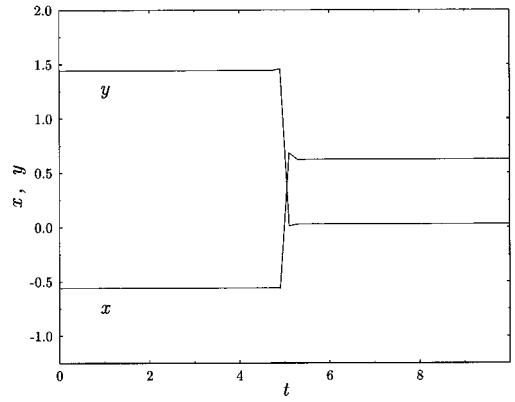

FIG. 8. The time evolution of the x and y coordinates on the Mueller potential [the same as figures 7(c) and (d)], this time with a time step of 0.2.

tory are evident when examining the new motion on the two dimensional energy surface [figures 7(a) and 7(b)]. The oscillations in the direction that corresponds to more rapid motions in the well are factored out.

In figures 7(c) and 7(d) the time dependence of the x and the y coordinates is shown. The vibrations in the minimum at the lower right corner can be decomposed (approximately) to oscillations along the x axis (slow) and the y axis (fast). As expected from the description above and from the display of the trajectory on the energy surface [figure 7(b)], the fast y motions are almost completely overdamped. The oscillations along the x axis remain similar.

This brings us to the behavior of the oscillations at the second minimum at the upper left corner of the energy surface (labeled A). The upper minimum is narrow similarly to the lower minimum with one slow and one fast direction. In contrast to the lower minimum discussed above, the fast and the slow modes do not follow the x or the y axes. Therefore, the apparent narrowing of the trajectory on the energy surface is harder to detect in the time dependent plot. Both the **x** and the y coordinates have a component of the slow motion that is not overdamped when the time step is increased. Some degradation in the amplitude is therefore observed, but not annihilation of the vibrations as for the y direction in the lower minimum (labeled C).

The last piece of the trajectory that changes its shape when the fast motions are filtered out is the passage over the barrier. This motion is, of course, not a bound vibration, however, it is executed over a short time interval which is difficult to follow using a large time step. The time dependent plot shows that the time required to jump over the barrier is somewhat shorter when a large time step is used. The spatial description emphasizes the abrupt characteristic of the jump which is represented by a small number of points (four).

An example that is going to the extreme is the use of a time step which is larger than *any* time scale in the system. In figure 8 we show the time dependent of the barrier crossing in this case. In this trajectory nothing moves until the time to jump to the other side (which is arbitrary) arrives. After the jump occurs the system returns to absolute rest.

In the above example it was shown that the OM performs as expected from the theoretical analysis. Rapid oscillations are filtered out, producing a trajectory with slow components only. Nevertheless, it is worth emphasizing that the characteristics of the *slow* degrees of freedom, e.g., the typical frequencies and amplitudes, exemplified by the time dependent plots, are similar to the ''exact'' trajectories.

If a fast and rare movement, such as barrier crossing, is of interest it is possible to use multilevel time grids and to describe this motion in more detail. Computationally this refinement involves two steps: First, the identification that a rapid jump indeed occurred must be made. This can be done by searching for unusually large changes in structure over a few time steps. The second step is interpolation of the time interval of interest with more time points. Such a refinement is straightforward to do using the OM action, however, it is beyond the scope of the present paper.

The rapid passage over a barrier presents a significant challenge to the OM protocol. Nevertheless, the OM algorithm has a significant advantage in the study of barrier crossing as compared to regular MD. In the OM algorithm a boundary value problem is solved. It is therefore possible to compute a *single* trajectory that will pass the barrier. It is difficult to guarantee a crossing trajectory in MD employing a room temperature condensed phase simulation.

To focus our attention on different types of barrier crossing in the OM we return next to a simple one dimensional model, i.e., a double well system.

D. Double well potential

In this section trajectories are computed using only a small time step. Such a step is appropriate for the investigation of rapid barrier crossing. Instead of focusing on the dependence of the solution on the size of the time step [as was done in sections IV A–IV C], emphasis is made on the possibility of multiple solutions.

It was emphasized that the conjugate gradient optimization of the Onsager-Machlup action finds the *local* minimum which is the closest to our initial guess. The OM optimization may have numerous solutions for systems with nonlinear potentials. This makes the OM path optimization a global optimization problem which, in spite of recent advances in the field, is a complex problem with no general solution.

Demonstration of path multiplicity is better done by a simple model system, and for that purpose a double well potential is considered. It is shown that markedly different solutions may be obtained by using different initial guesses to start the optimization.

The details of the calculations are provided in the legend of figure 9. Two different paths were employed as initial guesses for the trajectory: The first is a straight line connecting the two end points: $R(t)=R(0)-2R(0)t/t_{end}$ ($t_{end}=47.86$) and the second is the solution of the harmonic part of the potential $R(t)=R(0)\cos(2\pi t/T)$, where T is the period of the corresponding classical trajectory. The different

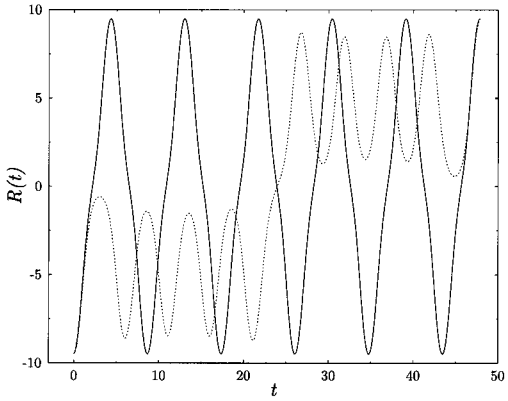

FIG. 9. The minimization of paths using the OM action for the double well potential $U(R)=1/2(R^2+A\exp(-\alpha R^2)-B)$, where $B=1/\alpha[\log(\alpha A)+1]$, $A=80$ and $\alpha=0.04375$. The boundary conditions are such that $R(0)=-R(t_{end})$, with $R(0)$ satisfying $U(R(0))=20$ and $Q(0)=Q(t_{end})=0$. The barrier height of the potential defined above is about 14.3. The dotted line is the solution obtained from a straight line interpolation between the end points as an initial guess. The solid line is the optimized solution when the initial guess is $R(t)=R(0)\cos(2\pi t/T)$, where T is the period of the Newtonian trajectory (≈ 8.7). In both cases the path has 500 intermediate points, the mass is one and the total length of time is 5.5 periods. The corresponding actions were 10^{-8} (solid line) and 2.4 (dotted line).

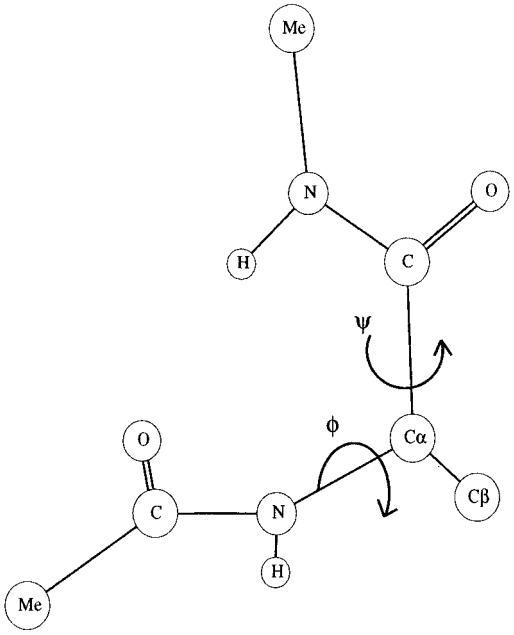

FIG. 10. Sticks and balls picture of alanine dipeptide. Note the two soft degrees of freedom—ϕ and ψ—that determine the conformation of the chain backbone.

guesses were optimized and the solutions are presented in figure 9. Which of the different optimized paths is correct, or more correct? One possible approach is to examine the value of the action. The value of the action for the classical trajectory is 10^{-8} while the value of the action of the second trajectory is 2.4. Hence, the classical path is to be preferred.

The path with the lower value of the action is the classical Newtonian trajectory, however, the other trajectory is meaningful too. It is a trajectory that is trapped in one well and is provided with the required additional energy to pass the barrier by the ''errors'' in the solution. Alternative interpretation follows the Langevin equation in which the errors are associated with the random force and the fluctuations of the environment that ''kick'' the system over the barrier. If the time step is large, the accuracy is reduced and the total energy is not known precisely, it becomes more difficult to distinguish between the two solutions and both trajectories are possible. Of course, if σ^2 is very small only the classical trajectory will have significant probability

E. Alanine dipeptide

The last numerical example considered is of a larger system: a conformational transition in alanine dipeptide. The system includes 12 particles and therefore 36 degrees of freedom (all the CH_n groups are modeled as spheres). The significant increase in complexity and the existence of multitime scales in the system is, of course, of prime interest. The use (and the need) of an additional path optimization tool (the multigrid scheme[21]) will be demonstrated. In spite of the additional complexity, alanine dipeptide is simple enough, so detailed analysis of the characteristics of the solution can be made.

In figure 10 we show a plot of alanine dipeptide. It includes two blocked amide planes and a single side chain—CH_3 (C_β) branched from the C_α position. There are two soft degrees of freedom in this molecule that determine its conformational state. These are the ϕ and the ψ dihedral angles (figure 10). The rest of the internal degrees of freedom are relatively rigid, making only small deviations from their equilibrium values.

It is therefore useful to consider a two dimensional projection of the energy surface of this molecule. A two dimensional adiabatic energy map (figure 11) is constructed. The

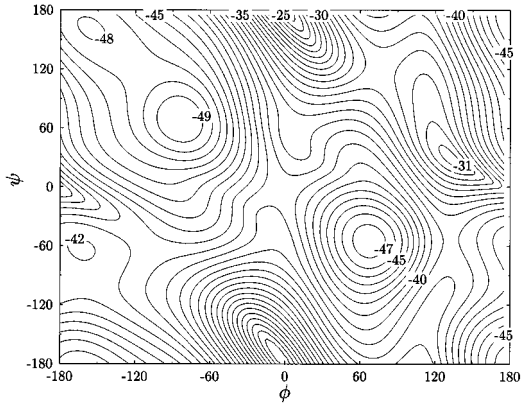

FIG. 11. ϕ and ψ adiabatic energy map of alanine dipeptide. For each fixed value of the pair of the torsion angles the energies of the rest of the degrees of freedom are fully minimized. The energies are in kcal/mol. Here we focused on the computations of trajectories from the equatorial conformation (the minimum at $\phi=67$ and $\psi=-55$) to the axial conformation (the minimum at $\phi=-84$ and $\psi=68$).

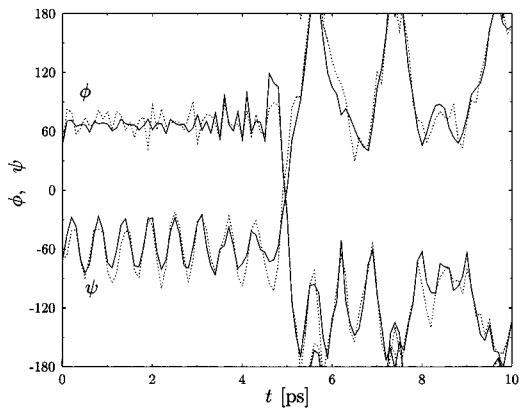

FIG. 12. The dihedral angles—ϕ and ψ—as a function of time. The dotted line is a 10 ps segment of a molecular dynamics trajectory. The trajectory, computed with a 1 fs time step, jumps from the equatorial conformation to the axial conformation. The RATTLE algorithm (Ref. 17) was employed to fix the bond length during the MD simulation. The solid line is an optimized Onsager-Machlup (OM) trajectory. For the initial guess of the OM optimization we used structures from the molecular dynamics trajectory separated by 0.1 ps. The final optimized result is similar to the original, in spite of the significantly larger time step that was employed in the OM calculation.

adiabatic map is generated by minimizing the energy for all degrees of freedom excluding the fixed ϕ and ψ. The corresponding energies are displayed on a contour plot. We compute the trajectory using (of course) all the degrees of freedom, however, it is convenient to represent the trajectory and to analyze it using the two degrees of freedom that dominate the motions.

All the computations (including the optimization of the OM action) were performed using a modified MOIL program.[23] MOIL is a general purpose public domain molecular dynamics package for biomolecules developed in our group. It is also a library of routines, that allows easy patching or addition to the package of a new code. The potential energy in MOIL is the combination of AMBER[24] and OPLS[25] implemented in a reasonably flexible database, making it possible to add or to modify the energy function. The calculations were done without van der Waals or electrostatic cutoffs and with 1–4 factors (8 for van der Waals and 2 for electrostatic) appropriate for the AMBER/OPLS force field.

Suppose that a molecular dynamics trajectory is computed using a standard integrator such as Verlet.[26] If this MD trajectory is now used to initiate OM computations, are we at a local minimum of the action? This is a non-trivial question since in the OM investigations a much larger time step (compared to MD) is employed.

In figure 12 the trajectory computed with the Verlet algorithm (dotted line) is shown. A typical time step of 1 fs was used. The bond lengths were fixed using the RATTLE algorithm.[17] It is difficult to increase the time step to more than 4 fs using standard tools. Significantly larger computational effort must be devoted to the calculation of a *single* step in time if a larger Δt (in the framework of molecular dynamics) is desired.[18,28]

The segment of the trajectory that describes a transition from the conformation C_7 equatorial to C_7 axial is of a total length of 10 ps. The molecular dynamics simulation was much longer than the above mentioned segment. However, most of the time the trajectory spent in the C_7 equatorial minimum without executing ''exciting'' motions.

From the segment of 8.34 ps of the simulation with a time step of 1 fs, a ''lower resolution'' trajectory was constructed. A single point was picked each 100 fs, creating a new trajectory with a time step of 0.1 ps. The new set of structures was given as input to the OM algorithm, and an optimized value of the action (and an optimal trajectory) were computed using the conjugate gradient procedure. The conjugate gradient is an effective algorithm to find the nearest local minimum.

In figure 12 we compare the optimized trajectory (solid line) with the original. We show the time dependence of the soft degrees of freedom—ϕ and ψ. In spite of the increase in the time step by a factor of 100, the resulting trajectory is remarkably similar to the original. Of course, smaller time steps provide even better results. One factor that works for us in this case is the clear separation of time scales between the rest of the degrees of freedom and the soft degrees of freedom. The short time step is essential to follow the vibrations of the bonds. However, it cannot be the whole story since SHAKE or RATTLE, that forces bonds to their equilibrium positions, do not support a step increase to more than a factor of 2 or 3. The OM path remains in the neighborhood of the exact solution for significantly larger time steps than ordinary MD of biomolecules can accommodate.

The dipeptide system includes many modes that differ considerably in their time scales. Bond stretching (femtoseconds) is the most rapid and the conformational transitions of the dihedral angles the slowest. The last are not only slow but also rare. As was argued in section III that dealt with the numerical algorithm, to find a good optimum for a system with multiple time scales, multigrid approach is a very suggestive technique. The principles of the multigrid technique were briefly described in section III and can also be found elsewhere.[21] Therefore, we restrict the discussion below to a report of the optimization parameters and the results.

In figure 13 we show a direct minimization of a 5 ps OM trajectory between the C_7 equatorial and the C_7 axial conformations. The calculation setup is described in the legend. The shape of the trajectory on the two dimensional energy surface seems quite ordinary for the trained eye of the molecular dynamicist. This remains so when examining the time dependence of the soft degrees of freedom [figure 13(b)].

Hence even in the first trial (without multigrid optimization), a reasonable description of the motion of the soft degrees of freedom is obtained. However, the value of the potential energy as a function of time reveals that the configurations sampled are not fully relaxed since the energy reaches high values (see figure 16, dotted line). We therefore continue with the application of multigrid relaxation.

To improve the present trajectory the so-called ''V cycle'' procedure of the multigrid algorithm[21] was employed. The first run was done with 401 grid points with a time step of 0.0125 ps (figure 13). In the next step of the

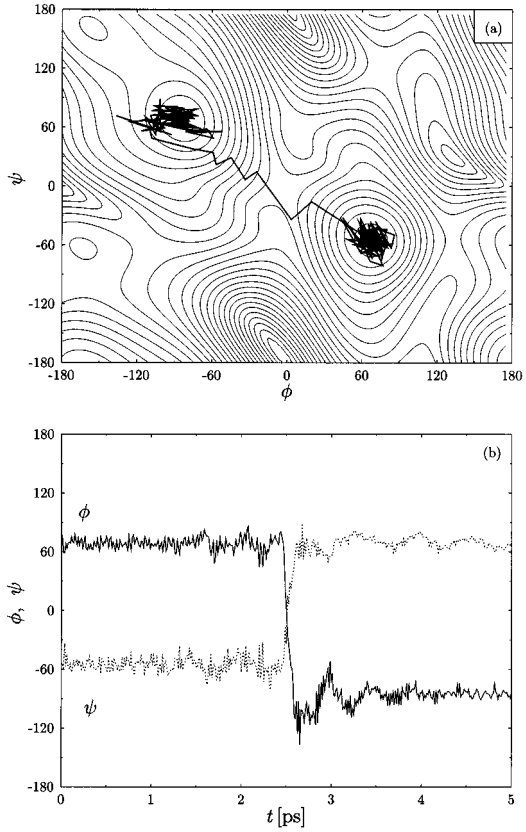

FIG. 13. Minimization of the OM trajectory from the equatorial to the axial conformation of alanine dipeptide. 401 grid points were employed. The time step was 0.0125 ps and the total time was 5.0 ps. The final action was 2.5. (a) The trajectory on the adiabatic map. (b) The time evolution of the two relevant backbone torsion angles. This run was the first in the multigrid cycle.

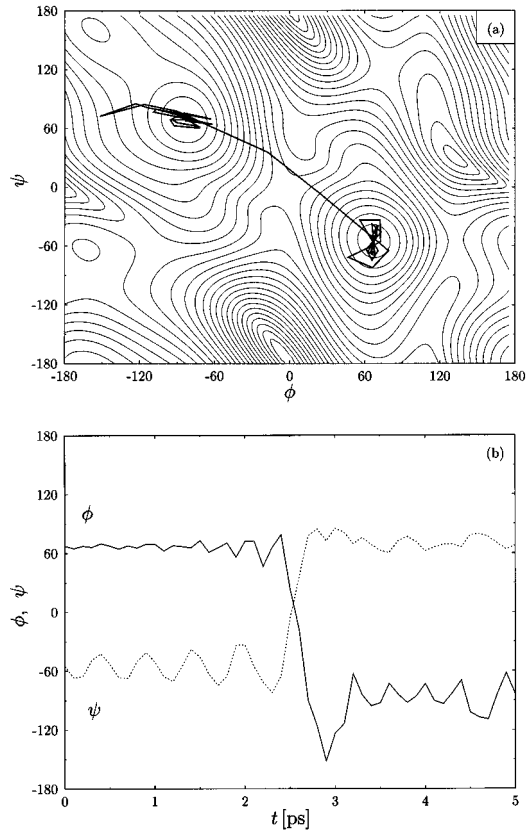

FIG. 14. The same as figure 13, but this time at the lowest corner of the multigrid ''V'' (i.e., the lowest resolution representation that was employed in the multigrid cycle). The time step is 0.1 ps and only 51 grid points are employed. Note the significant similarity of the time evolution of the ϕ and the ψ dihedral angles in figures 14(b), 13(b), and 15(b), demonstrating that the time evolution of the slow degrees of freedom is affected only slightly by the different resolutions employed in the calculations.

optimization we doubled the time step to 0.0250 ps maintaining the same length of trajectory. The total number of points was therefore reduced to 201. Using each second point of the previous optimized solution, we initiated a new optimization with lower time resolution that provided us with a qualitatively similar trajectory.

We continued to the next cycles of 101 and 51 grid points, re-optimizing in each time resolution the final path of the previous run. The optimization for each size of the grid is continued (in general) until the convergence is significantly slowed down. The last trajectory of the series of computations was the lower part of the ''V'' cycle. We then returned to the upper resolution of 401 points by increasing the time step in a similar series of optimizations. The same sequence of calculations as the one employed in the forward direction was used.

An important lesson from the cycle (beside the improvement in the final results described below) is that the qualitative shape of the trajectory and the motion of the soft degrees of freedom is reasonably well reproduced even with only 51 grid points (figure 14).

In figure 16, we show all the potential energies of the paths at the extrema of the ''V'' cycle of the multigrid optimization. The significant improvement and the success of the multigrid procedure are apparent. The new trajectory has potential energy variations comparable to the barrier height separating the two minima in the map. The application of multiple grids also reduced the value of the S from 2.5 to 1.6 (compare figure 13 and figure 15). S is in the usual MOIL units of distance in angstrom, masses in gram/mol and energy in kcal/mol. We therefore note that the relative variation in S can be small. It is therefore useful to examine the potential energy along the trajectory as well, selecting paths that are energetically less demanding.

In figures 14 and 15 we have shown the trajectory of the slow coordinates—ϕ and ψ—which remains essentially unchanged after the last modifications of the time step.

This summarizes the numerical examples of this paper. The examples aim to demonstrate that the computations are doable, but also include non-trivial difficulties that must be considered. The most important problem is of global optimization of the path. Nevertheless, we showed the significant stability of the algorithm and its relative accuracy for motions with time scales significantly longer than the numerical time step.

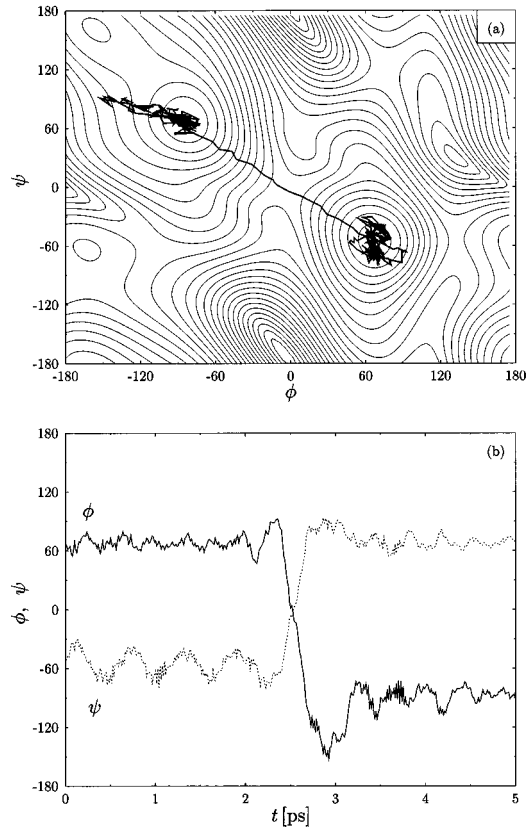

FIG. 15. The same as figures 13 and 14 but this time for the upper right corner of the ''V'' cycle that was used in the multigrid calculations. The time step is the same as in figure (13), i.e., 0.0125 ps.

V. CONCLUSIONS AND FINAL REMARKS

We proposed a new technique for calculating molecular dynamics trajectories. The computational method proposed is different from ordinary molecular dynamics integrators in several respects. First, we consider boundary value problems instead of initial value problems. That is, the two end points and the length of the trajectory must be determined prior to the calculations. Second, we do not solve differential equations, rather we optimize a target function which is a discrete representation of the Onsager-Machlup action. It is a functional of the whole trajectory.

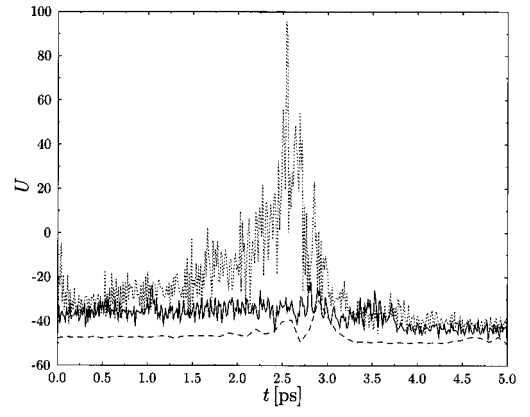

FIG. 16. The potential energy function (in kcal/mol) as a function of time for the paths in figures 13 (401 grid points—dotted line), 14 (51 grid points—dashed line) and 15 (401 grid points—solid line). The paths are the solution at the vertices of the ''V'' multigrid scheme. Note that in addition to the above three runs we also employed computations with 201 and 101 points.

In assessing the numerical properties of the new algorithm we note that for the same accuracy the present approach is significantly more expensive than ordinary molecular dynamics. This is since the complete trajectory is considered at once, in contrast to ordinary MD in which a single structure is followed at a time. In this paper we focused on the comparison of the OM with MD and detailed analysis of the solution accuracy. We did not exploit yet the possibilities of pursuing very long time dynamics and focused instead on times that are covered by both, the MD and the OM protocols.

The advantage of the OM protocol is the systematic and the stable way in which the time step can be increased. Making the time step larger results in filtering out the motions with periods faster than the time step. The filtering maintains the stability of the solution for almost *an arbitrary time step.* An estimate for the magnitude of the errors (see Appendix A) suggests that they are approximately bound by the amplitude of the rapid motions that were factored out. An alternative way to estimate the errors and the action ''width''—σ—(Appendix B) can be therefore based on the prior knowledge of the number of fast oscillations in the system (e.g., from normal mode analysis). Accordingly, a plausible approach of estimating the missing part of the trajectory is to assume that the amplitudes of the fast modes can be computed in the harmonic approximation, assuming a state of equilibrium.

Where do we expect the OM action to be more useful and where do we expect it to fail? The answer to the above question is related to the characteristics and the relative number of the fast modes. The OM filters out oscillations with periods smaller than the time step. It is necessary to keep the time step sufficiently short, so that the relevant motions in the system will remain. This is perhaps one of the reasons that makes the OM action optimization an attractive method for proteins. The very broad range of different time scales observed in protein dynamics[1] makes the systematic screening protocol a potentially useful approach. For proteins, it is anticipated that time steps different by orders of magnitude could be employed and significant motions will remain.

We have preliminary and encouraging results for the R→T transition in hemoglobin and the folding of C peptide. The optimization of trajectories for larger systems corresponds (of course) to a larger minimization problem. These are nevertheless doable calculations since the extensive range of time scales in proteins makes it possible to use a larger time step, reducing the system dimensionality and making it possible to focus on the slow motions that remain. Conformational transitions in C peptide were investigated using 1 ns time step. The time step employed in the study of the R→T transition in hemoglobin was 100 ns.

As a rule of thumb, many degrees of freedom and many

modes are an advantage in an attempt to cover the missing motions. This is evident already in the dipeptide in which the ϕ and ψ movements are well described even with only 51 grid points ($\Delta t = 0.1$ ps). In that sense, the examples presented in here are more demanding. Of course, to test different features of the new technique it is useful to start with simple model systems, as we did. Nevertheless, the present methodology is both more suitable and more useful for much larger systems. The overall success of the examples in reproducing essential features of the dynamics, *in small systems*, is therefore encouraging.

The double well system is one of the more stringent tests of the proposed protocol. There are essentially two rapid motions, one for the motion in the well and the second is for the transition above the barrier. Once the time step is larger than these two rapid motions the system is frozen in one of the wells and ''tunnels'' (in a single step) to the other minimum. It is not possible to obtain a detailed picture of the dynamics unless a small time step is employed. This effect was demonstrated for motions on the Mueller potential.

Nevertheless, as was mentioned in the numerical examples, the OM protocol has another feature that is missing from ordinary molecular dynamics. The OM trajectories are computed between fixed end points. Hence, it is possible to force the system to execute certain movements which are significantly more difficult to sample using ordinary molecular dynamics. For example, (going back to the barrier crossing problem mentioned above) it is possible to use two end points, one before and one after the barrier, and to compute a crossing trajectory. Of course due to the rapid nature of the motions it will be necessary to employ a small time step. However, the probability of getting such a crossing trajectory will be one in the OM protocol and may be practically zero in MD.

Finally we add to the conclusions two more comments that were not discussed in detail in the present work but have significant implications on future work.

This paper is focused on the computations of individual trajectories. Nevertheless, we do appreciate the need to average over an ensemble of trajectories in order to get an estimate of the conditional probability [equation (8)] and eventually the rate. To compute thermal properties of an ensemble of trajectories, the weight of the trajectory should include the Boltzmann factor of the initial coordinates, i.e.,

$$P(\mathbf{R}(t_i),\mathbf{R}(t_f);t_f-t_i)=P(\mathbf{R}(t_i))P(\mathbf{R}(t_i)|\mathbf{R}(t_f);t_f-t_i)$$
$$\propto \exp(-E_i/k_BT)\exp(-S/2\sigma^2), \quad (24)$$

where $P(\mathbf{R}(t_i),\mathbf{R}(t_f);t_f-t_i)$ is the joint probability of observing $\mathbf{R}(t_i)$ at t_i and $\mathbf{R}(t_f)$ at t_f (note the difference between the joint and the conditional probability which we used before). $P(\mathbf{R}(t_i))$ is the probability of observing the initial conformation and E_i is its total energy (including the kinetic energy). Hence, to maximize the weight of the trajectories it is important to seek the paths with low initial energies in addition to the lowest possible S.

On the computational side we also note that the algorithm is simple to parallelize.[27] The path is divided between the processors and the communication needed is for the estimates of time derivatives. The communication required is very small and is restricted to nearest neighbors only. We have a code working on a cluster of workstations using PVM and a code running on the IBM SP2 using MPI.[29] In both cases, the communication overhead was found to be negligible. Parallel and distributed computers are therefore extremely efficient for OM computations. These advanced computers are considerably less effective for ordinary MD computations in which the forces are parallelized. There, the communication overhead is more significant. Moreover, the performance of MD degrades very rapidly with the number of processors, making it impossible to take advantage of massively parallel or distributed machines.

ACKNOWLEDGMENTS

This research was supported by grants from the Israel Science Foundation and from the Israel Science Ministry. The Fritz Haber center is supported by the Minerva fund. Most of the computations were performed at the Wolfson Center for applied structural biology and some were executed at the High Performance Computing Unit of Israel. We thank P. Wolynes, B. Eisenberg and D. Thirumalai for numerous interesting discussions.

APPENDIX A: AN ESTIMATE OF THE ACTION ''WIDTH''—σ

It was argued that each trajectory has a weight of $\exp(-S/2\sigma^2)$. Finding the minimum of S is independent of the value of σ^2, which is a constant. However, in order to assess the accuracy of the computed trajectory and to compute the conditional probability it is useful to have an estimate of what σ^2 might be.

In many respects this is still an open problem: One empirical approach to this problem is presented here. The numerical solution for the harmonic oscillator is considered. As the time step is increased, the computed optimal trajectory becomes different from the exact trajectory (see figure 3).

Let us denote the numerical solution for the path by $\mathbf{R}_{num}(t)$ and the exact solution by $\mathbf{R}_{exa}(t)$. We further denote by S_{num} and S_{exa} the numerical approximation and the exact value of S, respectively. S_{num} is at a minimum at $\mathbf{R}_{num}(t)$ and S_{exa} is at a minimum at $\mathbf{R}_{exa}(t)$. Typically, only S_{num} and $\mathbf{R}_{num}(t)$ are available, but for the special case of the harmonic oscillator we have both. It is therefore possible to ask: What should be the value of σ^2, such that the exact trajectory $\mathbf{R}_{exa}(t)$ will have a weight of 1/e when substituted into S_{num}, i.e., the action width is defined as $2\sigma^2 = (S_{num}[\mathbf{R}_{exa}(t)]-S_{exa}[\mathbf{R}_{exa}(t)])$. Since $S_{exa}[\mathbf{R}_{exa}(t)]$ is zero for the harmonic oscillator, we have $2\sigma^2 = S_{num}[\mathbf{R}_{exa}(t)]$. In figure 17 we estimate the value of the action width as a function of the time step. Since the solution becomes flat for time steps above a certain range, the error becomes constant above that value.

The idea is to use the asymptotic value of the errors in estimating how the trajectory is affected and the validity of the approximations. Clearly, motions with typical periods

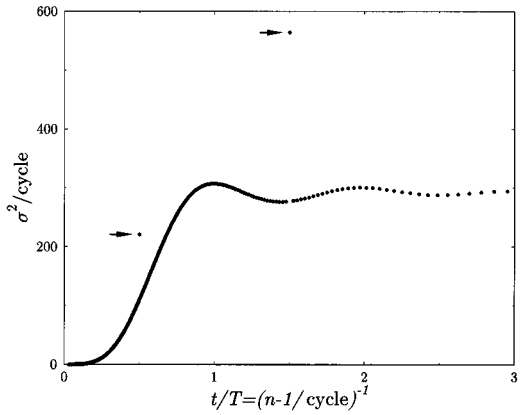

FIG. 17. An example for the estimate of the errors in the evaluation of the Onsager-Machlup action for the harmonic oscillator. The error is computed as a function of the time step—Δt—which is used in the numerical optimization of the trajectory. The time is dimensionless and denoted by the number of points per cycle. The error is normalized per cycle. Note that the error is bound and is stabilized near an asymptotic value. Note also the singular points of significantly larger errors. They correspond to the accidental matching of the time interval and the extrema of the cycle of the oscillator.

faster than the time step are missing. These faster motions will be added as ''noise'' once the width of the action is taken into account. The amplitude of the fast motions, to be substituted in $S_{num}[\mathbf{R}_{exa}(t)]$ can be estimated in several ways. We first note that $S_{num}[\mathbf{R}_{exa}(t)]$ can be expanded in the neighborhood of $\mathbf{R}_{num}(t)$, yielding

$$S_{num}[\mathbf{R}_{exa}(t)] \approx 1/2\,(\mathbf{R}_{exa}(t)-\mathbf{R}_{num}(t))[\delta^2 S/\delta \mathbf{R}^2_{num}(t)] \times (\mathbf{R}_{exa}(t)-\mathbf{R}_{num}(t)). \tag{A1}$$

Hence, by comparing the difference between exact trajectories (i.e., usual MD that includes fast modes) with trajectories for which the fast modes were eliminated, it is possible to obtain an idea of how far we are from the exact trajectory. This approach is feasible since it is not difficult to get reasonable statistics for fast modes.

Another way of estimating the contribution of the fast degrees of freedom is via normal mode analysis and formula (A1). Normal mode analysis provides the period of the motion in addition to the coordinate. Therefore, for a given time step we can immediately identify which are the modes that will survive, and which of the modes will be eliminated. We assume that the fast (and frozen) modes add to the errors in the OM procedure amplitudes—A—that can be estimated from thermal consideration. In one dimension we have $1/2M\omega^2A^2=k_BT$.

APPENDIX B: BEYOND THE CONSTANT FORCE APPROXIMATION FOR THE SHORT TIME PROPAGATOR

Here a formula for the short time OM action is provided that includes the first order correction to the constant force approximation. The starting point is the exact expression of S [equation (8b)]:

$$\begin{aligned} S&=\int_0^{\delta t}(M d^2\mathbf{R}/d\tau^2 + dU/d\mathbf{R})^2 d\tau \\ &=\int_0^{\delta t}(M d^2\mathbf{R}/d\tau^2)^2 d\tau + 2\int_0^{\delta t}(M d^2\mathbf{R}/d\tau^2)(dU/d\mathbf{R})d\tau \\ &\quad + \int_0^{\delta t}(dU/d\mathbf{R})^2 d\tau = I_0+I_1+I_2, \end{aligned} \tag{B1}$$

where I_0 was already computed analytically in the body of the paper. Therefore, the two remaining integrals are the focus of this appendix. The relevant variables are now defined. For completeness some of the definitions that were given in the body of the paper are repeated. The path is separated into a parabolic component $\mathbf{R}^0$ (which is used to estimate the second derivatives in time) and a sine series that accounts for the difference between the exact solution $\mathbf{R}$ and $\mathbf{R}^0$. This difference is denoted by $\mathbf{R}^1$

$$\mathbf{R}=\mathbf{R}^0+\mathbf{R}^1; \quad \mathbf{R}^0=\mathbf{a}+\mathbf{b}\tau+1/2\,\mathbf{c}\tau^2;$$

$$\mathbf{R}^1=\sum_k \mathbf{d}_k \sin(\omega_k \tau); \tag{B2}$$

$$\omega_k=\pi k/\delta t,$$

where the relation between the vectors **a**, **b** and **c** and the coordinate vectors $\mathbf{R}_{i-1}$, $\mathbf{R}_i$ and $\mathbf{R}_{i+1}$ is given in formulae (19) and (20). The forces are approximated in the time interval of interest by

$$dU/d\mathbf{R} \approx dU/d\mathbf{R}_i + (d^2U/d\mathbf{R}_i^2)(\mathbf{R}-\mathbf{R}_i) \equiv \alpha+\beta(\mathbf{R}^0+\mathbf{R}^1), \tag{B3}$$

$$\alpha = dU/d\mathbf{R}_i - (d^2U/d\mathbf{R}_i^2)\mathbf{R}_i; \quad \beta = d^2U/d\mathbf{R}_i^2.$$

Note that α and β are a time independent vector and a time independent matrix, respectively. They are evaluated at the intermediate coordinate $\mathbf{R}_i$. All that is left at present is to substitute the expressions in I_1+I_2, and to integrate the expressions. Here is the final lengthy result

$$I_1+I_2=2\mathbf{c}\underline{M}\alpha\delta t+2\mathbf{c}\underline{M}\beta(\mathbf{a}\delta t+\mathbf{b}\delta t^2/2+\mathbf{c}\delta t^3/6)+8\mathbf{c}M\beta\sum_{k\ odd}\mathbf{d}_k/\omega_k+\sum_k(-1)^k\mathbf{d}_kM\beta\omega_k\delta t[2\mathbf{b}+\mathbf{c}\delta t]-\sum_k\mathbf{d}_kM\beta\mathbf{d}_k\omega_k^2\delta t+\alpha^2\delta t$$

$$+2\alpha\beta[\mathbf{a}\delta t+1/2\,\mathbf{b}\delta t^2+1/6\,\mathbf{c}\delta t^3-2\sum_{k\ odd}\mathbf{d}_k/\omega_k]+\mathbf{a}\beta\beta\mathbf{a}\delta t+\mathbf{a}\beta\beta\mathbf{b}\delta t^2+\mathbf{a}\beta\beta\mathbf{c}t\delta t^3/3+\mathbf{b}\beta\beta\mathbf{b}\delta t^3/3+\mathbf{b}\beta\beta\mathbf{c}\delta t^4/4$$

$$+\mathbf{c}\beta\beta\mathbf{c}\delta t^5/20+4\sum_{k\ odd}\mathbf{a}\beta\beta\mathbf{d}_k/\omega_k-2\sum_k\mathbf{b}\beta\beta\mathbf{d}_k/\omega_k(-1)^k\delta t-4\sum_{k\ odd}\mathbf{c}\beta\beta\mathbf{d}_k/\omega_k^3-\sum_k\mathbf{c}\beta\beta\mathbf{d}_k/\omega_k(-1)^k\delta t^2$$

$$+1/2\sum_k\mathbf{d}_k\beta\beta\mathbf{d}_k\delta t. \tag{B4}$$

The notation that a vector on the left of a matrix is a row vector is used and we further employed the symmetry of the matrices.

The expression is quadratic in $\mathbf{d}_k$. It is therefore possible to obtain a set of linear equations for $\mathbf{d}_k$ after differentiating the approximate S with respect to these variables. The optimal $\mathbf{d}_k$ will no longer be identically zero. This expression is likely to be useful in analytical estimates of the conditional probability. There, an approximate summation of all paths (integration over all the $\mathbf{d}_k$) is of significant interest.

However, we did not employ the above expression in the present work since it is expensive to compute and does not suggest a computational advantage for the study of a single optimal trajectory.

[1] J.A. McCammon and S.C. Harvey, *Dynamics of Proteins and Nucleic Acids* (Cambridge University Press, Cambridge, 1987).
[2] H. Li, R. Elber, and J. Straub, J. Biol. Chem. **268**, 17908 (1993).
[3] J. Hofrichter, J.H. Sommer, E.R. Henry, and W.A. Eaton, Proc. Natl. Acad. Sci. **80**, 2235 (1983).
[4] One protein that was studied in great detail is BPTI. The start was the pioneer investigation of Creighton—T.E. Creighton, J. Mol. Biol. **113**, 275 (1977).
[5] See for instance, A. Kolinski and J. Skolnick, Proteins **18**, 338 (1994).
[6] Arieh Warshel, *Computer Modeling of Chemical Reactions in Enzyme and Solutions* (Wiley, New York, 1991).
[7] S. Glasstone, K.J. Laidler, and H. Eyring, *The Theory of Rate Processes* (McGraw-Hill, New York, 1961); D.G. Truhlar and B.C. Garrett, Annu. Rev. Phys. Chem. **35**, 159 (1984).
[8] R. Czerminski and R. Elber, J. Chem. Phys. **92**, 5580 (1990); R. Elber, in *Recent Development in Theoretical Studies of Proteins*, edited by R. Elber (World Scientific, Singapore, in press).
[9] R. Czerminski and R. Elber, Int. J. Quantum. Chem. **24**, 167 (1990); R. Elber, D. P. Chen, D. Rojewska, and R. Eisenberg, Biophys. J. **68**, 906 (1995).
[10] R.E. Gillian and K.R. Wilson, J. Chem. Phys. **97**, 1757 (1992).
[11] A.E. Cho, J.D. Doll, and D.L. Freeman, Chem. Phys. Lett. **229**, 218 (1994).
[12] L. Onsager and S. Machlup, Phys. Rev. **91**, 1505 (1953); S. Machlup and L. Onsager, *ibid.* **91**, 1512 (1953); S. Machlup, Ph.D. thesis, Yale University, 1952.
[13] R. P. Feynmann and A.R. Hibbs, *Quantum Mechanics and Path Integrals* (McGraw-Hill, New York, 1965).
[14] H. Goldstein, *Classical Mechanics* (Addison-Wesley, Reading, 1980).
[15] P.M. Morse and H. Feshbach, *Methods of Theoretical Physics* (McGraw-Hill, New York, 1953), Chap. 3.
[16] J.P. Ryckaert, G. Ciccotti, and H.J.C. Berendsen, J. Comput. Phys. **23**, 327 (1977).
[17] H.C. Andersen, J. Comput. Phys. **52**, 24 (1983).
[18] P. Derreumaux and T. Schlick, Proteins **21**, 282 (1995).
[19] M.J.D. Powell, Mathematical Programmming **12**, 241 (1977).
[20] R.H.J.M. Otten and L.P.P.P. van Ginneken, *The Annealing Algorithm* (Kluwer Academic, Boston, 1989).
[21] W.L. Briggs, *A Multi Grid Tutorial* (Lancaster, Pennsylvania, 1987).
[22] K. Mueller, Angew. Chem. **19**, 1 (1980); K. Mueller and L.D. Brown, Theoret. Chim. Acta **53**, 78 (1979).
[23] R. Elber, A. Roitberg, C. Simmerling, R. Goldstein, H. Li, G. Verkhivker, C. Keasar, J. Zhang, and A. Ulitsky, Comput. Phys. Commun. **91**, 159 (1995).
[24] S. J. Weiner, P.A. Kollman, D.A. Case, U.C. Singh, C. Ghio, G. Alagona, S. Profeta, Jr., and P. Weiner, J. Am. Chem. Soc. **106**, 765 (1984).
[25] W.L. Jorgensen and J. Tirado-Rives, J. Am. Chem. Soc. **110**, 1657 (1988).
[26] L. Verlet, Phys. Rev. **159**, 98 (1967).
[27] C. Simmerling, R. Elber, and J. Zhang, in *Modeling of Biomolecular Structures and Mechanisms*, edited by A. Pullman *et al.* (Kluwer Academic, Netherlands, 1995), pp. 241–265.
[28] M.E. Tuckerman, B.J. Berne, and G.J. Martyna, J. Chem. Phys. **99**, 5680 (1995).
[29] R. Olender and R. Elber (in preparation).