The Dynamo Library for Molecular Simulations Using Hybrid Quantum Mechanical and Molecular Mechanical Potentials

MARTIN J. FIELD, MARC ALBE, CÉLINE BRET, FLAVIEN PROUST-DE MARTIN, ALINE THOMAS

Laboratoire de Dynamique Moléculaire, Institut de Biologie Structurale—Jean-Pierre Ebel, 41 Rue Jules Horowitz, F-38027 Grenoble Cedex 01, France

ABSTRACT: The Dynamo module library has been developed for the simulation of molecular systems using hybrid quantum mechanical (QM) and molecular mechanical (MM) potentials. Dynamo is not a program package but is a library of Fortran 90 modules that can be employed by those interested in writing their own programs for performing molecular simulations. The library supports a range of different types of molecular calculation including geometry optimizations, reaction-path determinations and molecular dynamics and Monte Carlo simulations. This article outlines the general structure and capabilities of the library and describes in detail Dynamo's semiempirical QM/MM hybrid potential. Results are presented to indicate three particular aspects of this implementation—the handling of long-range nonbonding interactions, the nature of the boundary between the quantum mechanical and molecular mechanical atoms and how to perform path-integral hybrid-potential molecular dynamics simulations.   © 2000 John Wiley & Sons, Inc.   J Comput Chem 21: 1088–1100, 2000

Keywords: Dynamo module library; Fortran 90; molecular modeling; molecular simulations; hybrid potentials

Introduction

Hybrid quantum mechanical (QM) and molecular mechanical (MM) potentials are proving to be powerful tools for the investigation of reaction processes in condensed phase systems. The concept of a hybrid potential was first introduced by Warshel and Levitt in 1976[1] in their seminal study of the catalytic mechanism of the enzyme, lysozyme. To tackle this problem, Warshel and Levitt employed a QM potential to calculate the energy of the parts of the substrate and the protein that were directly involved in the reaction, and a more approximate MM force field for the energy of the remaining atoms in the system. The partitioning of the system in this way allowed calculations on systems much larger than would have been possible with QM potentials alone, and it meant that processes, such as reactions, could be studied for which MM potentials would not have been appropriate. Since then a number of hybrid potentials have been developed, and they have been applied to a wide range of problems in condensed phase chemistry, including the study of solvation effects, of reactions in solution and in proteins and of solvent-induced spectral shifts (for reviews, see ref. 2, 3, and 4).

Because of the interest in hybrid potentials, an increasing number of academic and commercial molecular modeling programs implement them. In this article, we present a program library, called the Dynamo library, which we have developed in our laboratory for the simulation of molecular systems using hybrid QM/MM potentials. As its name suggests, Dynamo is not a single program but a set of compatible program modules that can be employed when users write their own programs. The library itself is copyrighted, thus limiting some of the uses to which it may be put, but otherwise it is free and may be downloaded from the web at the address http://www.ibs.fr/ext/labos/LDM/projet6/welcome_en.htm.

The outline of this article is as follows. Section 2 describes the Dynamo module library and some of its features, Section 3 gives examples of the use of the library, and Section 4 summarizes.

The Dynamo Library

A recent monograph[5] describes in detail a molecular-mechanical version of the Dynamo library and gives many examples of how it can be used. To avoid duplicating what was said there, we shall restrict ourselves in this article to a brief overview of the library, and shall present in depth only those portions of it concerned with hybrid potentials. Interested readers seeking more should refer to ref. 5.

GENERAL STRUCTURE OF THE LIBRARY

The Dynamo library is written in Fortran 90.[6] Fortran 90, and its more recent update Fortran 95, are modern programming languages that contain significant enhancements over Fortran 77. Some of their more useful features are a compact notation for specifying array operations, the ability to dynamically allocate arrays, composite data structures, optional procedure arguments, and the concept of modules. Other features that are unfortunately not present in the language but that may be included in future upgrades are object-orientation and standard interfaces for interacting with the operating system environment and with C programs.

The Dynamo library is currently built up of about 80 Fortran 90 modules. There are a number of reasons for using modules in this way. First, modules provide a convenient way of grouping together data structures and the related functions and subroutines that operate on them. Second, it is possible to divide the data and the procedures in a module into private and public parts. This means that most of the working of a module can be hidden and protected from the outside by making the majority of the module's elements private and only allowing certain data structures or procedures to be publically accessible. The final reason for using modules is that their use enables many run-time errors typical of Fortran 77 programs to be eliminated. This is because modules provide well-defined interfaces, and so Fortran 90 compilers can check at compilation time whether, for example, the calling sequence of a particular module procedure is correct or whether the type of a module variable is consistent.

The Dynamo modules may be conveniently grouped into five categories which are:

1. Utility modules that perform basic programming tasks, such as error processing, file handling, and parsing. These modules provide the framework for the rest of the library and, for the most part, do not need to be used directly.
2. Parameter modules that store mathematical, physical, and chemical data.
3. Mathematical modules that perform standard numerical operations such as the optimization of a multivariate function or matrix diagonalization.

4. The modeling and simulation modules that handle calculations on molecules.
5. The Dynamo module, which declares all the other modules in the library.

This last module needs more explanation, which is best illustrated with an example. Suppose that the library is to be used to calculate the energy of a water molecule using the AM1 semiempirical QM method.[7] A suitable program to do this is shown in Figure 1. The basic structure of the program is self-explanatory. It starts with a PROGRAM statement that defines the name of the program, in this case EXAM-PLE, and is terminated by an END PROGRAM line. In between there are two declaration statements followed by six execution statements. The first execution statement CALL DYNAMO_HEADER prints out a header that indicates the version of the Dynamo library that is being used, whereas the two statements TIME_PRINT print out the time at the start and the end of the job.

The real work of the program is done by the three intervening statements. The first is a call to the subroutine XYZ_DEFINE, which reads in a set of coordinates, in XYZ format, from the external file water.xyz, and at the same time uses this information to set up the Dynamo data structures that

```
!--------------
PROGRAM EXAMPLE
!--------------

!  .  Module declarations.
USE DYNAMO

IMPLICIT NONE

!  .  Print out the header.
CALL DYNAMO_HEADER

!  .  Initialization.
CALL TIME_PRINT

!  .  Define the system by reading in an XYZ file.
CALL XYZ_DEFINE ( "water.xyz" )

!  .  Define all the atoms to be AM1.
CALL MOPAC_SETUP ( "AM1" )

!  .  Calculate and print an energy.
CALL ENERGY

!  .  Finish up.
CALL TIME_PRINT

END PROGRAM EXAMPLE
```

FIGURE 1. A program for calculating the energy of a water molecule using the AM1 semiempirical method. All lines starting with an exclamation mark are comments.

define the composition of the molecular system that is to be studied. The second statement is a call to the subroutine MOPAC_SETUP with the single character argument "AM1," which means that the potential energy of all the atoms in the currently defined system is to be calculated using the AM1 semiempirical QM method. The third call is to the subroutine ENERGY, which calculates and prints the potential energy for the system.

The two declaration statements at the start of the program are USE DYNAMO and IMPLICIT NONE. The latter forces all variables to be explicitly typed in the program but it is not really needed in this simple example, as no extra variables, either scalars or arrays, have been declared. In contrast, the line USE DYNAMO is required because what the Fortran 90 USE keyword does is to make visible, or "import," data structures, functions, and subroutines that exist in other Fortran 90 modules. Thus, the statement USE DYNAMO makes accessible to the program in Figure 1 all the public data structures and procedures of the Dynamo library because the module DYNAMO itself contains a list of all the usable data structures and procedures in the library. The advantage of declaring the different elements of the Dynamo library in this way is one of economy as otherwise the different subroutines used in the program would have to be imported from their own modules directly using separate USE statements, i.e., TIME_PRINT from the module TIME with the statement USE TIME, and so on.

CAPABILITIES OF THE LIBRARY

The library has modules for performing a wide range of different types of calculations on molecular systems. Most of these are described in ref. 5, but they include:

1. Reading and writing of the coordinates of a system in a number of formats, including Dynamo's own format, Protein Data Bank format,[8] XYZ format and as a Z-matrix.
2. Manipulations of coordinates, including the interconversion of Cartesian and internal coordinates, the rotation and translation of coordinate sets, and the superposition of structures.
3. Calculation of the potential energy of a system using QM, MM, and hybrid QM/MM potentials. At present, Dynamo uses the all-atom OPLS force field[9] as its MM potential and the semiempirical AM1, MNDO, and PM3[7, 10, 11] methods as its QM potentials. Although we shall not discuss it in this article, the library

also has an interface to a range of *ab initio* QM potentials.[12]

4. Description of condensed-phase systems using the minimum image periodic boundary condition approximation. Long-range interactions are truncated using an atom-based force-switching method[13] for hybrid potentials, but an Ewald method is also supported for MM potentials.

5. Optimization of the geometry of a system with algorithms that use both the first and the second derivatives of the system's potential energy with respect to its atoms' coordinates. Both minima and saddle points can be located with these algorithms.

6. Reaction path determinations using steepest-descent and self-avoiding walk[14] methods.

7. Calculation of the harmonic frequencies of a system and the analysis of its normal modes.

8. Classical and path-integral molecular dynamics simulations. Both constant temperature and constant pressure/constant temperature algorithms are provided.

9. Monte Carlo calculations in the canonical and isobaric–isothermal ensembles. Currently, these are limited to simulations of rigid molecules using MM potentials.

10. The application of geometric constraints to a system during a simulation. Dynamo can handle constraints that freeze the position of groups of atoms or that restrain particular geometric parameters in a system using harmonic potentials. The latter are particularly useful for the determination of free-energy profiles using umbrella sampling techniques.

It is important to emphasize that the Dynamo library makes use of dynamically allocated data structures throughout, and so it is never necessary to recompile the library with bigger array dimensions to handle larger systems. The only limit on the size of the system that can be treated is imposed by the amount of computer memory available and the speed of the CPU. The library has been tested and run on a number of different types of scalar and parallel computer systems, including PC/Linux, DEC Alpha, SGI, and Cray T3E machines.

THE HYBRID POTENTIAL

Both semiempirical and *ab initio* versions of a hybrid potential are implemented in the Dynamo library, but here we limit the discussion to the semi-empirical version and will publish details pertinent to the *ab initio* potential in due course.[12] The structure of the hybrid potential follows that of the potential developed by Field et al.[15] and so only a brief outline will be given here.

Dynamo's hybrid potential partitions the atoms in a system into two regions—a QM region and an MM region. As either region may contain zero atoms, pure QM or pure MM calculations can also be performed. Once the partitioning has been decided, an effective Hamiltonian $\hat{H}_{\text{eff}}$ for the system is constructed which consists of a sum of three terms—one for the atoms in the QM region, one for those in the MM region and a term which describes the interactions between the two:

$$\hat{H}_{\text{eff}} = \hat{H}_{\text{QM}} + \hat{H}_{\text{MM}} + \hat{H}_{\text{QM/MM}} \tag{1}$$

In Dynamo's semi-empirical hybrid potential, the Hamiltonian for the QM region $\hat{H}_{\text{QM}}$ takes the form that is appropriate for the QM method being used, either AM1,[7] MNDO[10] or PM3.[11] Likewise, the Hamiltonian for the MM region $\hat{H}_{\text{MM}}$ employs the form for the all-atom OPLS force field[9] but, as the latter does not contain any operators, this term reduces to the normal OPLS energy for the MM atoms.

The QM/MM interaction Hamiltonian is more complicated. In those cases where there are no covalent bonds between the atoms of the QM and MM regions, $\hat{H}_{\text{QM/MM}}$ consists of the sum of an electrostatic term and a Lennard–Jones term and may be written:

$$\begin{aligned}
\hat{H}_{\text{QM/MM}} = &- \sum_i \sum_m Q_m \hat{o}_{\text{electron}}(\boldsymbol{r}_i, \boldsymbol{r}_m) \\
&+ \sum_q \sum_m Z_q Q_m \hat{o}_{\text{core}}(\boldsymbol{r}_q, \boldsymbol{r}_m) \\
&+ \sum_q \sum_m 4\sqrt{\epsilon_{qq}\epsilon_{mm}} \left\{ \left(\frac{\sqrt{\sigma_{qq}\sigma_{mm}}}{r_{qm}} \right)^{12} \right. \\
&\left. - \left(\frac{\sqrt{\sigma_{qq}\sigma_{mm}}}{r_{qm}} \right)^6 \right\}
\end{aligned} \tag{2}$$

In this equation, the subscripts i and q refer to the electrons and the nuclei of the QM region, and m to the atoms of the MM region, Z_q is the core charge of the qth QM atom, Q_m is the charge of the mth MM atom, $\boldsymbol{r}$ is a coordinate vector, ϵ and σ are parameters for the Lennard–Jones interaction and r_{qm} is the distance between atoms q and m. The operators $\hat{o}_{\text{electron}}$ and $\hat{o}_{\text{core}}$ determine how the electrostatic interactions between the electrons and the nuclei of the QM atoms and the charges of the MM atoms are calculated. For *ab initio* wave functions,

of course, they take the form r_{im}^{-1} and r_{qm}^{-1}, respectively. The AM1, MNDO, and PM3 semiempirical methods use alternative forms, however, which are the same as the exact forms at long range but differ at short range.[15] The interaction Hamiltonian needs additional terms when there are covalent bonds between the atoms of the QM and MM regions, but these will be described in detail later.

Once the effective Hamiltonian for the system has been defined, the potential energy of the system E and the wave function for the electrons on the QM atoms Ψ are obtained by solving a time-dependent Schrödinger equation:

$$\hat{H}_{\text{eff}}\Psi = E\Psi \tag{3}$$

Other quantities, such as the forces on the QM and MM atoms, are straightforwardly determined when the energy E is known. In practice, Dynamo employs a single-determinant restricted or unrestricted Hartree–Fock approximation to represent the electronic wave function for the system, and resolves eq. (3) using a self-consistent field technique every time the geometry of the system changes or, in other words, at each step of a geometry optimization or of a molecular dynamics or Monte Carlo calculation.

Examples

The Dynamo library has been developed over a number of years in our laboratory, and we have completed a number of research projects using it. Among them are two recent studies in which we investigated the reaction mechanisms of two enzyme systems—neuraminidase from the influenza virus,[16] and spinach acetohydroxy acid isomeroreductase.[17] In both cases, we calculated free-energy profiles for various steps in the reactions using an umbrella sampling method and a semiempirical QM/MM potential.

In this section, we present results that illustrate some of the more technical aspects of Dynamo's current semiempirical QM/MM potential. Specifically, these are the nature of the truncation approximation used to treat the long-range nonbonding interactions, how we deal with the QM/MM interface, and our implementation of a path-integral molecular dynamics simulation algorithm.

NONBONDING INTERACTIONS

The determination of the long-range electrostatic interactions is critical for any molecular simulation

methodology. For MM potentials, the calculation of the electrostatic and Lennard–Jones interactions is usually the most time-consuming part of a simulation. This may not be the case for QM potentials, because other operations, such as diagonalization of the Fock or Hamiltonian matrix, can dominate. Nevertheless, the efficient evaluation of the long-range electrostatic interactions remains an important aspect of any QM algorithm.

There are essentially two strategies that can be adopted for calculation of the long-range interactions.[5] The first of these is to try an exact calculation. The problem with this, though, is that the number of interactions scales as the square of the number of particles, and so the time required becomes prohibitively large as the size of the system increases. To avoid this problem, algorithms with more favorable scaling properties have been developed that do not evaluate the long-range interactions exactly but only to a certain precision. These algorithms include the fast multipole methods and the Ewald and Ewald-like methods that are appropriate for periodic systems.

The second strategy, which has been widely used for simulations of large systems with MM potentials, is to ignore interactions between particles that are greater than a certain cutoff distance apart. As abrupt truncation is not usually a good idea, functions of various forms are applied to the interaction so that it is smoothed or tapered to zero at the cutoff distance. Truncation can be supplemented with other techniques, such as reaction-field methods, that attempt to estimate the neglected interactions beyond the cutoff. The advantage of truncation is that the time required for the evaluation of the nonbonding interactions scales linearly with system size, but the disadvantage is that the omitted long-range interactions can have a significant influence upon a system's properties.

Ultimately in Dynamo we intend to have algorithms that allow the "exact" evaluation of the nonbonding interactions for both QM and MM potentials. These will include a fast algorithm for periodic systems combined with a reaction-field method for the representation of solvent and an Ewald-like method for periodic systems. In the current version of the library, however, we have plumped for a comprise approach consisting of:

1. All the nonbonding interactions may be evaluated using an exact (or $O(N^2)$ where N is the number of particles) algorithm. This method works for QM, MM, and hybrid potential sim-

ulations but is limited to relatively small systems in vacuum.

2. A standard Ewald method may be used for the simulation of periodic systems with MM potentials.[18]

3. The nonbonding interactions are determined with truncation. This is the default method in Dynamo, and it can be used to study periodic and vacuum systems with MM and hybrid potentials. The truncation is applied to interactions between MM atoms and between QM and MM atoms but not to those between QM atoms. This should normally be a reasonable approximation, as the QM region will usually be of relatively small size and embedded in a much larger MM region.

Many different types of truncation approximation have been employed, and numerous tests have been done to see what effect they have on the properties of the system being simulated (see, e.g., ref. 19). The truncation potential that we have used is the atom-based force-switching method developed by Steinbach and Brooks,[13] who showed that it gave superior results for the simulation of both neutral and charged macromolecular systems when compared to other atom- and group-based truncation schemes.

The principle behind the force-switching scheme is that a truncation or switching function $S(r)$ is applied to the force between two particles, not the potential. Thus, if $F(r)$ is the "true" force, the force actually used becomes $S(r)F(r)$. For the electrostatic interactions, we use a switching function that has a value of 1 up to a distance of r_{on}, a value of 0 after a distance of r_{off}, and tapers smoothly and continuously in between. Its exact form is:

$$S(r) = \begin{cases} 1 & r \leq r_{on} \\ \dfrac{(r_{off}^2 - r^2)^2(r_{off}^2 + 2r^2 - 3r_{on}^2)}{(r_{off}^2 - r_{on}^2)^3} & r_{on} < r \leq r_{off} \\ 0 & r > r_{off} \end{cases} \tag{4}$$

To ensure that the potential and the force are consistent, it is necessary to modify the functional form of the interaction. Thus, if $V(r)$ is the "true" potential, the modified potential $V'(r)$ becomes:

$$V' = \begin{cases} V(r) + \delta V & r \leq r_{on} \\ -\int_{r_{off}}^{r} dr' \, S(r')F(r') & r_{on} < r \leq r_{off} \\ 0 & r > r_{off} \end{cases} \tag{5}$$

where δV is a constant to ensure continuity at r_{on}. These formulae are straightforward to implement

for the normal r^{-1} electrostatic potential, but for the Lennard–Jones interaction, we use a slightly simpler switching function that was suggested by Steinbach and Brooks. Readers should refer to ref. 13 for the exact formulae.

The above scheme can be applied to the MM electrostatic and Lennard–Jones interactions and also to the Lennard–Jones interactions between the QM and MM atoms, but it is not valid for the electrostatic interactions between the QM and MM atoms. In the semiempirical hybrid potential that Dynamo uses, the interaction between the charge on an MM atom m and basis functions μ and ν on a QM atom q is:

$$I_{\mu\nu} = -Q_m(\mu^q \nu^q | s^m s^m) \tag{6}$$

where s^m is a notional s-orbital on atom m and $(\mu^q \nu^q | s^m s^m)$ is a two-electron integral. Likewise, the interaction between the nucleus of a QM atom and a MM atom also involves a two-electron integral but this time of the form $(s^q s^q | s^m s^m)$.[15]

Force-switching truncation can be used for these interactions, but it is necessary to apply the switching function to the derivatives of the integrals rather than to the force. Fortunately, the two-electron integrals used in MNDO-type methods all have a rather simple form:

$$(\mu^q \nu^q | s^m s^m) \propto \sum_i \frac{1}{\sqrt{(r + \alpha_i)^2 + \beta_i^2}} \tag{7}$$

where α_i and β_i are constants,[20] and so it straightforward to derive analytic formulae for the modified integrals equivalent to those for the modified potential of eq. (5). These formulae are too long to be repeated here, but are available from the authors on request.

To show the effect of the truncation approximation, we plot in Figure 2 the energies of interaction between two water molecules as a function of their oxygen–oxygen distance. There are two curves— one calculated without using truncation, and the other calculated using force-switching truncation with inner and outer cutoffs of 8 and 12 Å, respectively. One of the molecules is treated as QM with the AM1 potential, and the other as MM with the all-atom OPLS force field. The symmetry of the complex is C_{2v}, and the internal geometries of the water molecules are fixed along the curves. It is clear that the curves are essentially identical except for a slight distortion that occurs within the switching region.

When studying larger systems, we test, where possible, the effect that the value of the cutoff distance and the breadth of the switching region have
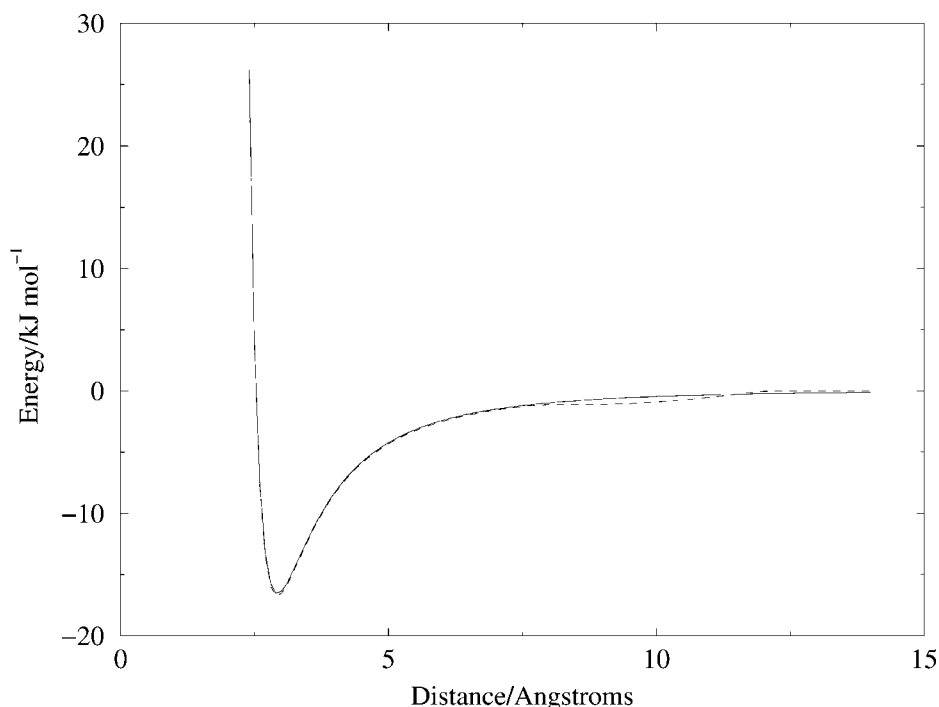
FIGURE 2. Plots of the interaction energies between two water molecules, one QM and one MM, as a function of their interoxygen distance. The solid line was obtained without truncating the nonbonding interactions, whereas the dashed line was obtained applying the atom-based force-switching truncation approximation. Full details of the calculation are given in the text.

on our results. For studies of enzyme and solution reactions, we have found that an inner cutoff of at least 8 Å and a switching region 4 Å broad are normally sufficient to give results that differ little from those that are obtained without truncation.

THE QM/MM INTERFACE

How nonbonding interactions between the atoms of the QM and MM regions are handled in Dynamo's hybrid potential was described earlier. In cases, however, where there are covalent bonds between atoms of the two regions it is necessary to introduce additional approximations. The treatment of such covalent bonds is probably the subject about which there is the most debate in the hybrid-potential community and a range of different methods have been proposed to address the problem. It is not the purpose of the article to comprehensively review such methods here, but we shall mention three broad classes that encompass most of the methods in current use. These are:

1. *Link-atom methods* introduce an extra unphysical "dummy" or "link" atom into the system for each covalent bond between atoms of the

QM and MM regions. The link atom, which is usually a hydrogen, is part of the QM region, and is placed along the bond between the QM and MM atoms at a suitable distance (~1 Å) away from the QM atom. The link atom serves to replace the MM atom in the QM calculation and ensures that the QM atom of the broken bond has no unsatisfied valency. Of course, the use of a link atom in the QM calculation does nothing to maintain the bond between the QM and MM atoms, and so this must be treated with appropriately chosen MM energy terms. Although addition of extra atoms into a system is rather inelegant, link-atom methods are relatively robust and easy to implement and, as such, have been the most widely-used in hybrid potential simulation studies. They were first introduced by Singh and Kollman,[21] and were also used by Field et al.[15] Variations on the link-atom approach have been developed by Morokuma et al.[22] and by Bersuker et al.[23]

2. *Capping-atom methods* use atoms of fictitious elemental type to "cap" the bonds between atoms in the QM and MM regions. Thus, for example, a bond between a QM atom and an

MM sp³-hybridized carbon would be capped by replacing the MM atom by a univalent QM atom that has been parametrized to behave like an sp³ carbon, and would give an appropriate bond length in a QM calculation. Such capping atoms can be hydrogen-like, with one valence electron, or halogen-like, with seven valence electrons. The bonds between the capping atom and the MM atoms to which it is bound are reproduced using MM terms. Semiempirical capping-atom methods exist in the MOPAC[24] and Hyperchem[25] programs, and have been proposed for use with *ab initio* QM/MM hybrid potentials by Zhang et al.[26]

3. *Hybrid- or localized-orbital methods* employ hybrid or localized orbitals on the QM atom at the boundary of the MM region and then freeze out of the QM calculation those orbitals, and their corresponding electrons, which point towards MM atoms. Hybrid-orbital methods were used by Warshel and Levitt in their original hybrid potential work,[2] and have since been extensively exploited by Rivail and coworkers.[27] Another hybrid-orbital scheme has been developed by Gao et al.[28] and, more recently, Philipp and Friesner have used localized orbitals to treat the QM/MM boundary region.[29]

Despite their diversity, the majority of the methods outlined above require some sort of parametrization, especially of the interactions between the atoms in the volume around the bond between the QM and MM atoms.

Although we have investigated other approaches (see, e.g., refs. 28 and 30), we make use of a link-atom approach in the current semiempirical QM version of the Dynamo library. Although we are aware that link-atom methods are not necessarily the ideal solution to the problem of how to treat bonds between the QM and MM regions, we have found them to give satisfactory results when reasonable care is taken in their application. It has also been our experience that the errors introduced by employing a link-atom method are relatively minor, it being more important, for example, to improve the quality of the QM method or increase the size of the QM region than to tinker overmuch with the treatment of the covalent bonds at the QM/MM interface.

The implementation of the link-atom method in Dynamo is sufficiently different from others that we have seen that we include a brief description here. In contrast to other implementations, we do not treat a link atom as an independent atom, but as part of the MM atom that has the covalent bond with the QM atom at the QM/MM interface. Each time an energy or gradient calculation is required, the coordinates of the link-atom (L) are generated automatically from the coordinates of the QM and MM atoms that define the covalent bond (Q and M, respectively) using the formula:

$$r_{\mathrm{L}} = r_{\mathrm{Q}} + b_{\mathrm{LQ}}\frac{r_{\mathrm{M}} - r_{\mathrm{Q}}}{|r_{\mathrm{M}} - r_{\mathrm{Q}}|} \quad (8)$$

where r is a position vector and b_{LQ} is a predefined constant, which is the bond distance between the link-atom and the atom Q. Once the position of the link atom is known, it can be employed in the energy calculation and the gradients of the energy with respect to the link-atom coordinates can be transferred to the atoms Q and M by a straightforward application of the chain rule. The advantages of handling link atoms in this way are twofold. First, it is much simpler for users as the determination and placement of link atoms is taken care of automatically and, second, the definition of the link-atom position in eq. (8) ensures that the link atom will always be in the correct place and at an appropriate distance from the atom Q.

The remaining details of the implementation of the link-atom method are similar to those of ref. 15. The link atoms are treated as hydrogen atoms, with a unit charge and a single electron, and enter only into the QM calculation—they do not have Lennard–Jones interactions, and participate in no MM energy terms. As for the other interactions between the atoms around the link atom, the MM bond term between the atoms Q and M is calculated as are any MM bond, angle, or dihedral energy terms that include at least one other MM atom in addition to the atom M that has the bond with the quantum atom Q. This scheme is different from that in ref. 15, and follows more closely that proposed by Eurenius et al.[31] Lennard–Jones interactions between QM and MM atoms are calculated in the normal way as for the OPLS force field with exclusion of 1–2 and 1–3 interactions and scaling of 1–4 interactions.[9]

The only remaining interactions that need to be discussed are the electrostatic interactions between the QM and MM atoms around the link atom. We have tried a number of different schemes to handle these interactions, including the use of scaling factors and of distance-dependent smoothing functions, but the approach that we consistently return to is the one advocated in ref. 15, in which all electrostatic interactions between MM and QM atoms

are calculated, without exclusion or scaling, except for those involving the link atom itself. This method is simple, and although it is not always optimal, it behaves well in a wide variety of circumstances and thus is the default method in Dynamo. We do not present the results of any calculations using the link-atom method here due to the similarities with results that have already been published. Instead, interested readers can refer to refs. 15 and 30 to obtain an idea of the precision to be expected with a link-atom method.

To conclude this section, we make a few general remarks about the use of link atoms in Dynamo. It is, in principle, possible to place link atoms anywhere, but we normally try to choose the atoms in the QM region so that the link atoms fall between sp^3-hybridized carbons. The first reason for this is that it helps to minimize the perturbation to the electronic structure of the QM region. The second reason is more pragmatic, and is due to the fact that the OPLS force field defines neutral CH_2 and CH_3 groups. Partitioning at such a group means that the standard OPLS charges can be used for all the MM atoms, and that no alterations are required to make the MM region have the correct overall charge.

As a final point, we would like to emphasize that the link-atom method is not perfect, and so it is necessary to test its validity in any particular application—both in what effect the siting of the link atoms have on the electronic properties of the QM region and on whether the method of treating the electrostatic interactions at the QM/MM boundary is satisfactory in any particular case.

HYBRID-POTENTIAL PATH-INTEGRAL SIMULATIONS

The great majority of molecular dynamics simulations are done within a classical approximation using Newton's equations of motion, or modifications or them, to describe the dynamics of the particles. Ideally one would prefer to do simulations using a quantum dynamical approach, but existing techniques are too cumbersome to be applied to any but a small number of particles. It is well known, however, that quantum dynamical effects are significant for certain phenomena such as, for example, reactions involving transfer of hydrogen, and so it is important to have some means of estimating their effect.

Although a direct simulation of the quantum dynamics of a system may be difficult, there is a method that can be used to include quantum effects in the determination of the equilibrium properties of a system. This is the path-integral method, which has the great advantage that it can be relatively easily implemented in an existing classical molecular dynamics program. In this approach, a classical simulation is performed with a potential derived by considering the isomorphism that exists between a discretized Feynman path integral representation of the equilibrium QM density operator and a classical system in which the quantum particles are represented by polymers of classical particles.[32, 33] The form of the effective potential is as follows:

$$\mathcal{V}_{PI} = \sum_{i=1}^{N_Q} \frac{P m_i (k_B T)^2}{2\hbar^2} \sum_{p=1}^{P} \left(r_i^{(p)} - r_i^{(p+1)} \right)^2$$
$$+ \frac{1}{P} \sum_{p=1}^{P} \mathcal{V}_{classical}\left(r_1^{(p)}, \ldots, r_{N_Q}^{(p)} \right) \quad (9)$$

where P is the number of beads in each polymer, N_Q is the number of "quantum" atoms, and $r_i^{(p)}$ is the position of the pth bead of atom i (noting that $r_i^{(P+1)} = r_i^{(1)}$). $\mathcal{V}_{classical}$ is the normal, classical potential of the system that will be a function of the pth copy of all the quantum particles as well as the coordinates of the "classical" particles. It should be emphasized that a dynamics simulation with the potential of eq. (9) does not give the dynamics of a real physical system—it is only valuable as a way of calculating the ensemble averages for the corresponding quantum system.

It is straightforward to implement the above effective potential for performing molecular dynamics simulations with QM, MM, or hybrid potentials. The first term on the right-hand side of eq. (9) consists of a sum of harmonic terms between adjacent beads of the polymers representing the quantum atoms and is simple to evaluate. The second term consists of a superposition of P classical energies. In the extreme case where all the atoms are treated with path-integrals, this term is simply the sum of P energies that can be calculated independently. More usually, there will be a few path-integral atoms but many more classical atoms, and so it would be wasteful to evaluate the P energies completely independently. Instead, the MM terms and the QM one- and two-electron integrals that involve the classical atoms and that are common to all of the energies in the sum are calculated once and only the terms involving path-integral atoms and their interaction with classical atoms need to be calculated P times. It should be noted, however, that it is still necessary to solve eq. (3) P times for each replica of the quantum atoms whenever QM or hybrid potentials are being used because the wave function and thus the energy

will be different for each of the P copies of the system. This, of course, adds considerably to the time required for a simulation.

In the current implementation of the path-integral methodology in Dynamo, we use the standard Dynamo velocity Verlet integrator with coupling to a Langevin bath when performing a path-integral molecular dynamics simulation.[18] We have found this to be sufficiently efficient for the applications that we have had up to now as long as care is taken to ensure that the time step for the integration is small enough so that the high-frequency motions caused by the harmonic term in eq. (9) can be integrated accurately. Other algorithms that are, in principle, more efficient exist, such as those based upon multiple-time-step and Car–Parrinello-type methods, and we shall consider implementing them in the future.[34, 35]

Many applications of the path-integral molecular dynamics methodology with MM potentials exist in the literature,[18] and, more recently, several examples of such calculations performed with *ab initio* QM methods have been published as well (see, e.g., references 36–39). In this section we illustrate the methodology in Dynamo, using the semiempirical AM1 potential,[7] applied to two simple molecular systems. We have recently performed a more complicated application involving the calculation of the free-energy profile for a hydrogen transfer reaction in a protein using a hybrid potential, and readers should refer to ref. 16 for specific details.

As a first example we examine the molecule CH_5^+ which has previously been studied at an *ab initio* level by Marx and Parrinello.[38] We performed two simulations—one in which all the atoms were treated classically and one in which all the atoms were treated as path-integral atoms with 32 copies of each. Both simulations were done at a temperature of 5 K, using a time step of 0.1 fs, and with coupling to a Langevin bath via a collision frequency of $100 \, ps^{-1}$. The classical simulation was run for 300 ps and the path-integral simulation for 50 ps.

If a geometry optimization is done of the CH_5^+ molecule, it is found that the "classical" ground state structure is of C_s symmetry with a CH_3 tripod and a H_2 moiety loosely bound to the carbon at longer distances. This structure is maintained during the simulations, as can be seen from the graphs that show the distributions of C—H and H—H distances in the CH_5^+ molecule (Figs. 3 and 4, respectively). The major difference between the two sets of results is that the path-integral distributions are much broader than the classical ones. This result is similar to that of Marx and Parrinello,[38] although they also found that there is an exchange or scrambling between the protons of the tripod and of the
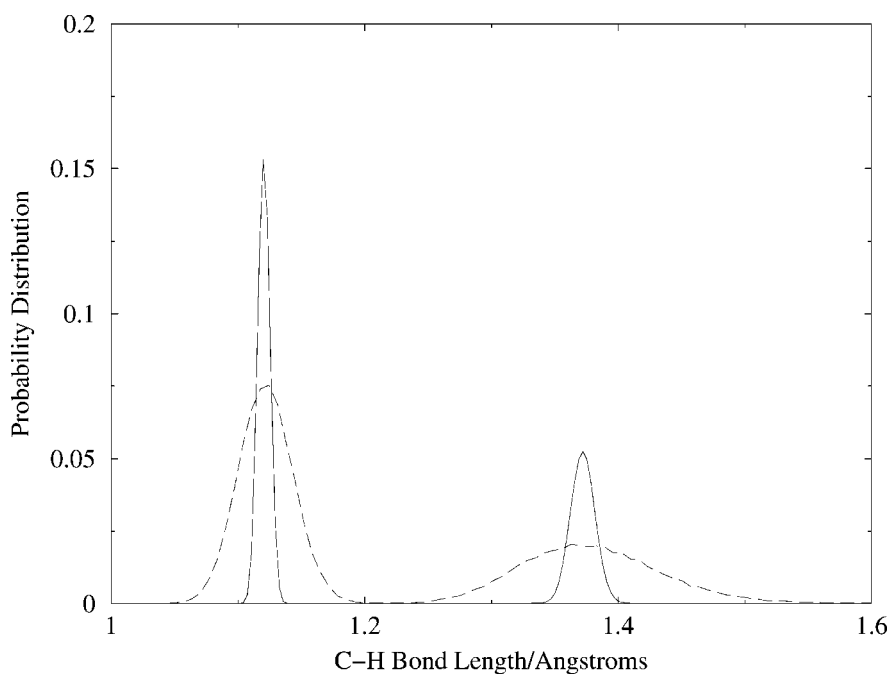

FIGURE 3. A plot of the distributions of C—H bond lengths found in the classical (solid line) and path-integral (dashed line) simulations of the CH_5^+ molecule.
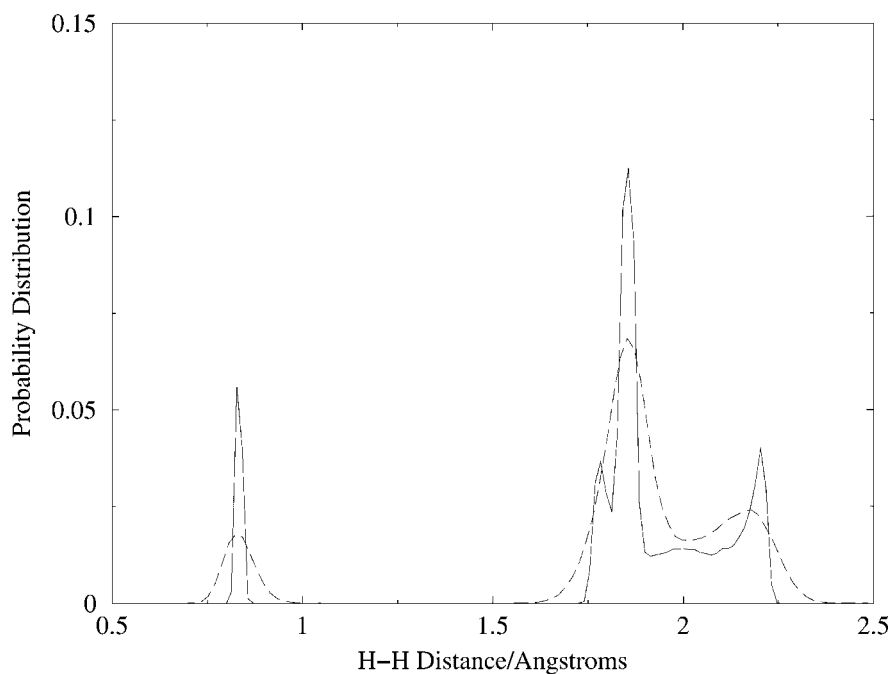
FIGURE 4. A plot of the distributions of H—H bond lengths found in the classical (solid line) and path-integral (dashed line) simulations of the CH_5^+ molecule.

H_2 moiety. In contrast, the two groups of protons remain distinct in both our simulations. This is due to the greater disparity in the two sets of C—H bond lengths found with the AM1 method ($\sim$1.12 and 1.37 Å) compared to the *ab initio* values ($\sim$1.09 and 1.17 Å).

For our second example we calculate two potentials of mean force (PMFs) for the transfer of hydrogen between the oxygens in acetylacetone (CH_3—COH—CH—CO—CH_3), one PMF in which all the atoms are treated classically and the other in which they are all treated as path-integral atoms with 32 copies of each. For both PMFs, we employed the AM1 semiempirical method as the potential for the whole molecule due to its small size. Acetylacetone has already been the subject of a similar study by Hinsen and Roux,[40] but instead of using a QM method, as we do here, they parametrized an empirical-valence bond potential to describe the potential energy surface for the hydrogen transfer process. Apart from the differences in the potential, we follow essentially the same procedure as that used by Hinsen and Roux to determine their PMFs. Full details of how to calculate PMFs can be found in their article[40] or in refs. 5 and 16.

The reaction coordinate used to model the transfer in the case where all the atoms are classical is defined as the distance resulting from the projection of the vector between the hydrogen H and the point midway between the two oxygens O_1 and O_2 onto the unit vector between the two oxygens, i.e.:

$$\xi(r_H, r_{O_1}, r_{O_2}) = \left(r_H - \frac{r_{O_1} + r_{O_2}}{2}\right)^T \frac{r_{O_1} - r_{O_2}}{|r_{O_1} - r_{O_2}|} \quad (10)$$

The reaction coordinate for the case where all the atoms are path-integral atoms is of the same form except that it is defined in terms of the centroids of the copies of the path-integral atoms rather than the coordinates of the atoms themselves. Thus, for example, instead of using r_H in eq. (10), one uses the centroid c_H:

$$c_H = \frac{1}{P} \sum_{i=1}^{P} r_H^i \quad (11)$$

where P is the number of copies of the path-integral atoms and r^i is the coordinates of the ith copy. With these reaction coordinates, the barrier for the transfer occurs at a value of $\xi = 0$ (because the reaction is symmetric) and the minima are at values of $\xi \sim \pm 0.6$.

To determine the PMFs, we use the method of umbrella sampling,[41] which requires that simulations are performed in the presence of umbrella potentials that constrain the reaction coordinate to be localized about a particular value. Although potentials of any form may be employed, it is common to choose harmonic forms for simplicity. Denoting
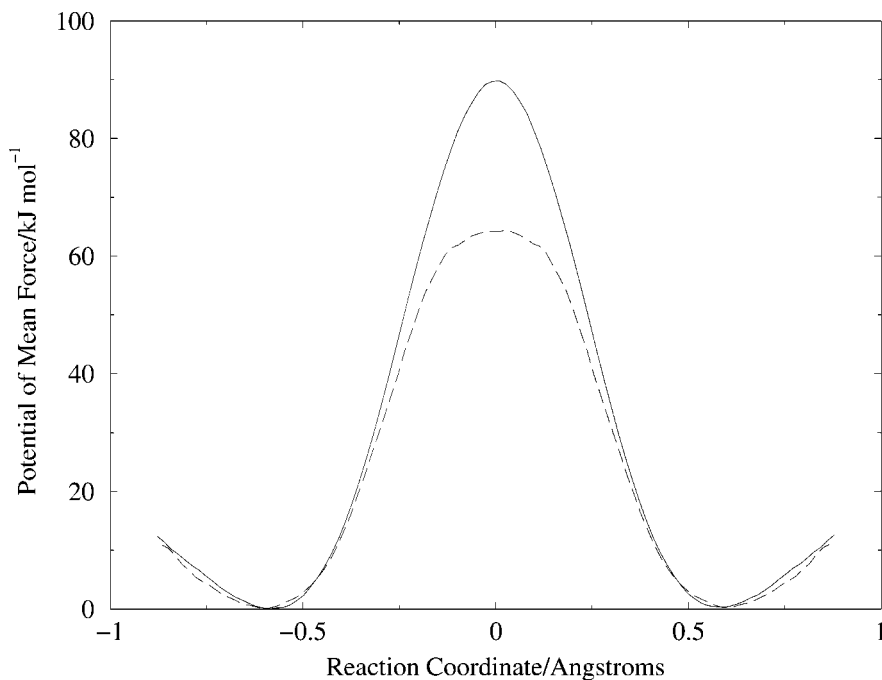
FIGURE 5. A plot of the potentials of mean force for the hydrogen transfer found in the classical (solid line) and path-integral (dashed line) acetylacetone simulations.

the umbrella potential for the wth simulation or "window" as $\mathcal{V}^{\text{umb}}$, we have:

$$\mathcal{V}_w^{\text{umb}}(\xi) = \frac{1}{2}k_w\big(\xi - \xi_w^0\big)^2 \qquad (12)$$

where k_w and ξ_w^0 are the force constant and the equilibrium value for the potential, respectively. We performed 17 independent Langevin molecular dynamics simulations for each PMF with equilibrium values of the umbrella potentials ranging from $\xi_0 = 0.0$ to 0.8 in increments of 0.05 and force constants of 2000 kJ mol^{-1} Å^{-2}. The path-integral simulations were carried out with 32 copies of each of the atoms in the system. Each simulation was performed at 300 K with a time step of 1 fs and a collision frequency for each atom of 10 ps^{-1} and consisted of 4 ps of equilibration followed by 10 ps in which the values of the reaction coordinate were written out at each time step. The PMFs were determined from the reaction coordinate data using the weighted histogram analysis method.[42]

The results of our simulations are shown in Figure 5. It can be seen that the barrier to transfer is about 90 kJ mol^{-1} for the classical simulation, but drops to 65 kJ mol^{-1} for the case in which all the atoms are treated as path-integral atoms. This reduction ($\sim$30%) is of the same order as that found by Hinsen and Roux in their calculations, but both

their barriers are lower, which is probably due to inaccuracies in the description of this transfer by the AM1 potential.

Summary

In this article we have presented the Dynamo module library, which we have developed for studying molecular systems with hybrid QM/MM potentials. The library supports a relatively wide range of different algorithms for performing molecular calculations, including geometry optimizations, normal mode analyses, and molecular dynamics and Monte Carlo simulations, and is now being used in our laboratory for the majority of our simulation work. Because it is a research tool, the library is continually being added to and changed to take advantage of algorithmic advances and the increasing power of computers. An *ab initio* version of the hybrid potential has recently been implemented, and will be reported soon.[12] Other enhancements that we are working on include an interactive graphical interface that will facilitate the use of the library for many operations, polarizable force fields for describing the atoms of the MM region,[43, 44] and more efficient hybrid-potential simulation algorithms.

References

1. Warshel, A.; Levitt, M. J Mol Biol 1976, 103, 227.

2. Warshel, A. Computer Modeling of Chemical Reactions in Enzymes and Solutions; John Wiley and Sons: New York, 1991.

3. Gao, J. In Reviews in Computational Chemistry; Lipkowitz, K. B.; Boyd, D. B., Eds.; VCH: New York, 1995, p. 119, vol. 7.

4. Amara, P.; Field, M. J. In The Encyclopedia of Computational Chemistry; Schleyer, P. v. R.; Allinger, N. L.; Clark, T.; Gasteiger, J.; Kollman, P. A.; Schaefer, H. F., III; Schreiner, P. R., Eds.; John Wiley & Sons: Chichester, 1998, p. 431, vol. 1.

5. Field, M. J. A Practical Introduction to the Simulation of Molecular Systems; Cambridge University Press: Cambridge, MA, 1999.

6. Metcalf, M.; Reid, J. Fortran 90/95 Explained; Oxford University Press: Oxford, UK, 1996.

7. Dewar, M. J. S.; Zoebisch, E. G.; Healy, E. F.; Stewart, J. J. P. J Am Chem Soc 1985, 107, 3902.

8. Bernstein, F. C.; Koetzle, T. F.; Williams, G. J. B.; Mayer, E. F.; Brice, J. M. D.; Rodgers, J. R.; Kennard, O.; Shimanouchi, T.; Tasumi, M. J Mol Biol 1977, 112, 535.

9. Jorgensen, W. L.; Maxwell, D. S.; Tirado–Rives, J. J Am Chem Soc 1996 118, 11225.

10. Dewar, M. J. S.; Thiel, W. J Am Chem Soc 1977, 99, 4899.

11. Stewart, J. J. P. J Comput Chem 1989, 10, 209; ibid 1989, 10, 221.

12. Amara, P.; Proust–De Martin, F.; Field, M. J. in preparation.

13. Steinbach, P. J.; Brooks, B. R. J Comput Chem 1994, 15, 667.

14. Czermiński, R.; Elber, R. Int J Quantum Chem Quantum Chem Symp 1990, 24, 167.

15. Field, M. J.; Bash, P. A.; Karplus, M. J Comput Chem 1990, 11, 700.

16. Thomas, A.; Jourand, D.; Bret, C.; Amara, P.; Field, M. J. J Am Chem Soc 1999, 121, 9693.

17. Proust–De Martin, F.; Dumas, R.; Douce, R.; Field, M. J. in preparation.

18. Allen, M. P.; Tildesley, D. J. Computer Simulation of Liquids; Oxford University Press: Oxford, UK, 1987.

19. Loncharich, R. J.; Brooks, B. R. Proteins Struct Func Gene 1989, 6, 32.

20. Dewar, M. J. S.; Thiel, W. Theor Chim Acta 1977, 46, 89.

21. Singh, U. C.; Kollman, P. A. J Comput Chem 1986, 7, 718.

22. Maseras, F.; Morokuma, K. J Comput Chem 1995, 16, 1170.

23. Bersuker, I. B.; Leonig, M. K.; Boggs, J. E.; Pearlman, R. S. Int J Quantum Chem 1997, 63, 1051.

24. Stewart, J. J. P. Quantum Chemistry Program Exchange 455; 1996, vol. 6, no. 391.

25. Hyperchem, Hypercube Inc., Waterloo, Ontario, Canada.

26. Zhang, Y.; Lee, T.-S.; Yang, W. J Chem Phys 1999, 110, 46.

27. See, for example: Monard, G.; Loos, M.; Thery, V.; Baka, K.; Rivail, J.-L. Int J Quantum Chem 1996, 58, 153; Assfeld, X.; Rivail, J.-L. Chem Phys Lett 1996, 263, 100.

28. Gao, J.; Amara, P.; Alhambra, C.; Field, M. J. J Phys Chem 1998, 102, 4714.

29. Philipp, D. M.; Friesner, R. A. J Comput Chem 1999, 14, 1468.

30. Amara, P.; Field, M. J.; Alhambra, C.; Gao, J. Theor Chem Accs, in press.

31. Eurenius, K. P.; Chatfield, D. C.; Brooks, B. R.; Hodoscek, M. Int J Quantum Chem 1996, 60, 1189.

32. Feynman, R. P.; Hibbs, A. R. Quantum Mechanics and Path Integrals; McGraw–Hill: New York, 1965.

33. Chandler, D.; Wolynes, P. G. J Chem Phys 1981, 74, 4078.

34. Tuckerman, M. E.; Berne, B. J.; Martyna, G. J.; Klein, M. L. J Chem Phys 1993, 99, 2796.

35. Tuckerman, M. E.; Marx, D.; Klein, M. L.; Parrinello, M. J Chem Phys 1996, 104, 5579.

36. Marx, D.; Parrinello, M. Z Phys 1994, B 95, 143.

37. Cheng, H.-P.; Barnett, R. N.; Landman, U. Chem Phys Lett 1995, 237, 161.

38. Marx, D.; Parrinello, M. Nature 1995, 375, 216.

39. Marx, D.; Tuckerman, M. E.; Hutter, J.; Parrinello, M. Nature 1999, 397, 601.

40. Hinsen, K.; Roux, B. J Chem Phys 1997, 106, 3567.

41. Torrie, G. M.; Valleau, J. P. Chem Phys Lett 1974, 28, 578.

42. Kumar, S.; Bouzida, D.; Swendsen, R. H.; Kollman, P. A.; Rosenberg, J. M. J Comput Chem 1992, 13, 1011.

43. Field, M. J. Mol Phys 1997, 91, 835.

44. Bret, C.; Field, M. J.; Hemmingsen, L. Mol Phys, in press.