



ELSEVIER

Biophysical Chemistry 51 (1994) 217–233

Biophysical
Chemistry

Molecular dynamics simulations of water using a floating polynomial force field and an interpolating electrostatic field representation

Raúl E. Cachau¹

*Structural Biochemistry Program, Frederick Biomedical Supercomputing Center, PRI / Dyn Corp.,
National Cancer Institute-Frederick Cancer Research, and Development Center, Frederick, MD 21702-1201, USA*

Received 20 December 1993; accepted 23 February 1994

Abstract

The ability to accurately describe the force field of a molecule is of great importance in spectroscopic and drug design studies. However, the fitting of accurate potential energy functions has proved to be a highly complex task. The description through a simple generic formula of all conformations of a molecule has proved to be a seldom reliable procedure, while more complex representations are increasingly difficult to fit, slower to compute, and difficult to program. In this work, alternative procedures are explored: (1) the intramolecular force fields are expanded in a floating polynomial representation; (2) a fast treatment for the non-bonded interactions is applied. The advantage of these treatments is in their ability to describe highly accurate representations of molecules in a very efficient manner. The main difficulty is a heavy trade off in computer memory usage. Some of the more frequently used force fields for water, and a first principle force field are used as a test of these techniques.

Key words: Molecular dynamics; Water; Floating polynomial force field; Interpolating electrostatic field representation

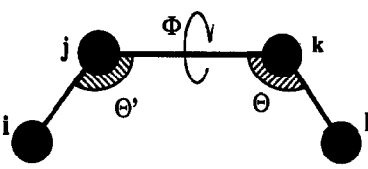
1. Introduction

Molecular structures can be interpreted in terms of a ball-and-stick model. The role of a molecular mechanics (MM) representation is to make this model quantitative. The MM approach makes it possible to compare and rank different conformations by an *energy* value. The concept of a bond length, or of a bond angle, is simple

enough in terms of a ball-and-stick model and a naive MM force field representation [1]. In a real molecule, however, atoms are undergoing vibrational motions. These vibrations are, in general, anharmonic in nature. This means that the simplest ideas about vibrational motion may be insufficiently accurate. Modern experimental techniques, together with sophisticated *ab initio* calculations, have helped in the development of highly refined MM force fields [2]. These force fields usually contain a variety of anharmonic terms as well as cross term corrections. The de-

¹ E-mail cachau@ncicrf.gov

velopment of force fields for molecules of biological interest is especially challenging, given the size and intrinsic complexity of the systems involved [3]. To obtain the accuracy required for the study of macromolecules, it is necessary to use a relatively complex form for the empirical potential functions. It is also necessary to optimize the values of the parameters that determine the magnitudes of different contributing terms. In general, a force field will include terms that depend, not only on the relative positions of all pairs of atoms, but also on certain triplets and quadrupoles of atoms. Based on this approach a number of force fields for use with polyatomic systems have become available in recent years [2]. Features common to most of them include: a harmonic restoring force between bonded nearest neighbors, which is frequently improved by using a Morse-like force term; a penalty for deforming an angle between three neighboring atoms; a dihedral torsional potential to allow for a hindered rotation of groups around a bond; and non-bonded interactions between separated atoms. Currently available force fields might also incorporate higher-order terms as well as cross terms. The bonded interactions for a typical modern force field are schematically described in Fig. 1. The sums (Eq. (1)) are usually truncated using



$$E = \sum_{ij} D_{ij} \left[1 - e^{-\alpha_{ij}(R_{ij} - R_{ij}^0)} \right]^2 + 1/2 \sum_{ijk} K_{\theta_{ijk}} [\theta_{ijk} - \theta_{ijk}^0]^2 + 1/2 \sum_{ijkl} K_{\Phi_{ijkl}} [1 + \cos(\delta_{ijkl} - \Phi_{ijkl})] + \sum_{ij} \sum_{kl} K_{R_{ij}R_{kl}} (R_{kl} - R_{kl}^0)(R_{ij} - R_{ij}^0) + \sum_{ij} \sum_{klm} K_{R_{ij}\theta_{klm}} (\theta_{klm} - \theta_{klm}^0)(R_{ij} - R_{ij}^0) \quad (1)$$

Fig. 1. Schematic representation of a typical molecular force field. R_{ij} is the distance between atoms i and j , θ_{ijk} is the angle between atoms i , j and k , and Φ_{ijkl} is the dihedral angle between atoms i , j , k and l . The 0 superscript indicate equilibrium distances and angles.

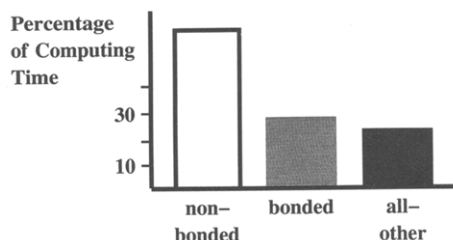
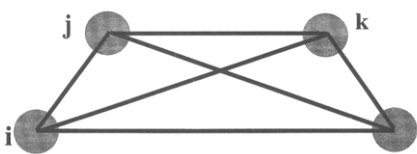


Fig. 2. Ratio of CPU time required for different tasks in an MD calculation. The example is from a calculation on a peptide of sequence GRGDS [29] surrounded by 117 water molecules. The non-bonded interactions accounted for 66% of the computing time, bonded interactions accounted for 21% and all other calculations for 12%.

a chemical heuristic approach. For example, while any three atoms define an angle, only a few of them are considered. The selection of bond distances, angles and dihedrals follows, in most cases, the intuitive description of a molecule in terms of chemical bonds, angles defined between bonds and dihedrals as rotations around the bonds (Fig. 1). The interaction between non-bonded (usually called the *non-bonded* interaction term) atoms are usually accomplished by representing the atoms as points with assigned properties, like electric charge, and hardness described with van der Waals radii [4,5]. Because of a large number of pairs of atoms that are included in the list of non-bonded atoms in a molecular dynamics (MD) simulation, the computation of the non-bonded interactions takes most of the computer time required for MD calculations. Therefore, most of the effort in speeding up MD calculations has been focused on optimizing the calculation of the non-bonded interaction term. The non-bonded term is usually arranged as a long list of entries to which exactly the same operation is applied, making a high level of optimization possible. The expression in Eq. (1) contains variables (e.g. angles), that cannot be directly calculated from the Cartesian coordinates of the atoms [6], but require the use of computationally expensive transcendental functions in their evaluation. Fig. 2 shows a profile of the time consumed in a MD calculation by the different segments of the program. A typical ratio for computing time usage between non-bonded and bonded interactions is



$$E = \sum_i \sum_j K(R_{ij}) R_{ij}^l + \sum_i \sum_m \sum_n \sum_{ijk} K(R_{lmnijk}) R_{ij}^l R_{ik}^m R_{jk}^n + \sum_i \sum_m \sum_n \sum_o \sum_p \sum_q \sum_{ijkl} K(R_{lmnopqijkl}) R_{ij}^l R_{ik}^m R_{il}^n R_{jk}^o R_{jl}^p R_{kl}^q \quad (2)$$

Fig. 3. FFFF representation of the force field truncated to 4 atom clusters (compare with Fig. 1). The indexes ($l \dots q$) are the polynomial power. The sets $\{K_{ij}\}$ are parameters. These parameters index the force constants. The index is related to the variable R_{ij} , where R_{ij} is the distance between two atoms, by: $\{K_{ij}\} = \text{int}(\text{scale} * R_{ij} + \text{shift})$ where *scale* and *shift* are properly chosen to optimize the memory use.

3:1. Even though this ratio is sensitive to the particular problem and to the force field used, the non-bonded and bonded segments of the MD calculation generally account for 85% to 95% of the total computation time [7]. Great improvements have been shown recently in the development of algorithms for the fast computation of the non-bonded interactions [8]. However, those

algorithms seldom refer to improvements in the efficiency of bonded interaction calculations [8]. Modern force fields contain several cross terms as well as anharmonic terms. When cross terms are considered, the typical ratio for non-bonded to bonded computing time usage shifts to 2:1. Hence, while the optimization of the computation of non-bonded interactions is still the area where the most significant gains could be made, the improvement of *only* this segment of a MD code could only lead to a maximum speedup of about $\frac{2}{3}$ of the initial speed of the code. A breakdown of the computing time use in the *all other* segments (Fig. 2) shows several subroutines, using 1% or less of the total computing time. These routines usually involve intensive input/output or large list updates. These can be effectively dealt with by the use of modern large memory machines, where most of the list updates can be speeded up by using redundant secondary lists and where the input–output can be optimized by using internal files.

Therefore, simultaneous improvement of both the non-bonded *and* the bonded interaction computations is required to maximize the speed of MM or MD computations.

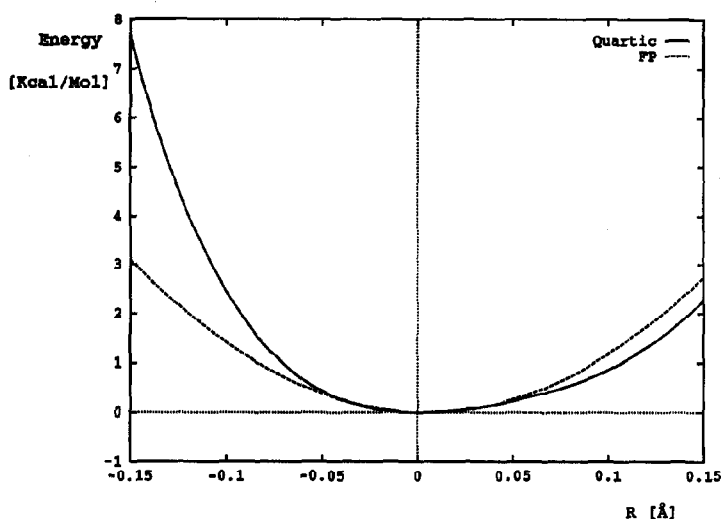


Fig. 4. Water quartic force field for the symmetric stretch of the r and r' bonds (see Table 1) and the corresponding diagonal term of the FFFF representation for the range $[-0.05, +0.05]$ Å. Note the local character of the fit.

2. Intramolecular force field

Modern supercomputers require algorithms where operations are applied to long lists of variables in either a parallel or a vector processing fashion. One of the main problems with Eq.

(1) (see Fig. 1), is that it contains heterogeneous variables, like bond angles, dihedral angles, and distances, which increase the complexity of the algorithm. Common occurrences of exceptions in Eq. (1) complicate this task even further. Moreover, the energy and its derivatives are not easily

Table 1

M force field parameters. Quartic force field from Jensen [13]. All distance units are in Å. Units for the quartic force field are in millidyne/Å^{nr+nr'-2}, where *nr* and *nr'* represent the exponents of *r* and *r'* respectively. The charge units for non-bonded interaction terms are in electrons. Fraga's parameters for the correction of polarizability and repulsive terms, and the multiclassification scheme are reported in the literature [4,14]

Intermolecular parameters

f_{rr}	8.43938 (19)
$f_{rr'}$	-0.10515 (16)
$f_{r\theta}$	0.30641 (23)
$f_{\theta\theta}$	0.70700 (12)
f_{rrr}	-55.40 (33)
$f_{rrr'}$	-0.318 (20)
$f_{rr\theta}$	-0.252 (55)
$f_{rr'\theta}$	-0.447 (25)
$f_{r\theta\theta}$	-0.3383 (62)
$f_{\theta\theta\theta}$	-0.7332 (70)
f_{rrrr}	306.0 (47)
$f_{rrrr'}$	2.57 (40)
$f_{rrr\theta}$	-6.14 (91)
$f_{rrr'r'}$	1.93 (15)
$f_{rrr'\theta}$	-3.22 (32)
$f_{rr\theta\theta}$	-0.950 (51)
$f_{rr'\theta\theta}$	0.1150 (62)
$f_{r\theta\theta\theta}$	0.87 (13)
$f_{\theta\theta\theta\theta}$	-0.238 (19)
r^0	0.9572
θ^0	104.52

Molecular multipoles

μ_z	-0.7296
Π_{zz}	-0.098
Π_{xx}	1.96
Π_{yy}	-1.862
Ω_{zzz}	1.8908
Ω_{xxx}	-3.1745
Ω_{yyz}	1.2837

Atomic multipoles

q_{hydrogen}	-0.32
μ_x	0.041
μ_y	0.000
μ_z	0.023
Π_{xx}	0.30
Π_{yy}	0.27
Π_{zz}	0.00
Π_{xy}	0.00
Π_{yz}	0.00
Π_{xz}	0.32

q_{oxygen}

q_{oxygen}	-0.68
μ_x	0.000
μ_y	0.000
μ_z	0.040
Π_{xx}	0.45
Π_{yy}	0.40
Π_{zz}	-0.02
Π_{xy}	0.00
Π_{yz}	0.00
Π_{zx}	0.00

Molecular polarizabilities

α_{zz}	9.907
α_{xx}	10.311
α_{yy}	9.549
β_{zz}	-0.09
β_{xx}	-0.15
β_{yy}	-0.10

related to Cartesian coordinates. In order to solve this problem a polynomial expansion in interatomic distances can be used (Fig. 3). The form of the force field in Eq. (2) is suitable for parallel or vector machines since the operations required can be reduced to simple matrix operations. It is important to notice that the set $\{R_{ij}\}$ is now a set of *parameters* that indexes the matrix of force constant and R_{ij} are the active variables for the energy calculation (Fig. 3).

This form of representation of a function by a polynomial, whose coefficients are indexed on a discrete form of the expansion variables is common place in a number of areas (i.e. the use of *splines* in graphics) [9]. In our case the fitting procedure does not satisfy the restrictions that the spline definition imposes. Thus we will call this particular expansion a *floating polynomial*, as this functional form used to be called [9]. While Eq. (2) appears to be better suited to parallelization or vectorization than Eq. (1), three main difficulties in its implementation could be encountered: (1) the equation has a very strong local character (see Fig. 4); (2) a larger memory is required to load the $K(\{R_{ij}\})$ matrices; and (3) a well established protocol for the fitting of the polynomial coefficients for any given molecular force fields is missing.

The challenge presented by the local character of the representation can be overcome by using higher-order polynomials. However, a use of matrices of higher order than 3 is impractical. Alternatively, products of lower order matrices can be used:

$$K(\{R_{ij}\})_{(1\dots n)(i\dots k)} \cong K(\{R_{ij}^*\}) \times K(\{R_{kl}^*\})_{(1\dots n)(i\dots k)}, \quad (3)$$

where $\{R_{ij}^*\}$ are subsets of distance parameters. This representation increases the number of matrices used, but each of them has a smaller dimension. Obviously, the optimal form of the force field (as in Eq. (3)) will contain a certain group of matrices of force constants which retain the most important features of the energy potential. This approximation is strictly valid when the matrices in Eq. (2) are very sparse.

In order to obtain the maximum efficiency on a given machine architecture (i.e. vector or paral-

lel machines), different programming algorithms must be considered for each of them. However, the compact form of this formulation of the force field makes it possible to adapt the programs to different platforms with a relatively minor effort.

3. Application of the floating polynomial force field (FPFF) approximation to the water molecule

In what follows we will consider the water molecule as a workbench for some of the ideas described above. The very small size and high symmetry of the water molecule simplify its description. In this case we will truncate the polynomial to the following expression:

$$E_{\text{water}} = K_{(rr',r'')}^0 + K_{(r)}r + K_{(r')}r' + K_{(r'')}r'' + K_{(rr')}rr' + K_{(rr'')}rr'' + K_{(r'r'')}r'r'' + K_{(r'r'r'')}rr'r'', \quad (4)$$

where the meaning of the coefficients is immediate from Table 1.

Elaborated forms of fitting the coefficients on these expressions must be developed before this treatment can be applied to highly heterogeneous systems. However, since liquid water is a complex system built on replicas of a *single* molecule, this problem is greatly reduced. In fact, water is a *planar* system with no dihedral angles, which reduces to a minimum the extra memory required for a MD simulation using a FPFF representation. All force fields used in this work were decomposed internally using for their representation Eq. (4). The memory required for this FPFF representation of the water force field is 250 Kb (Kb = kilo byte) using a 0.1 Å grid for all distances in the range [0.5 Å, 3.1 Å]. Given this very small range of distances, due to the local character of the fitting (Fig. 4), all calculations can be carried out using operations on integers. This is especially important in the implementation of this algorithm in machines that are poor floating point performers.

In order to prove the validity of this approach we will use it with some of the most common water models mentioned in the literature (SPC [10], MCY [11], ST2 [12]), as well as with an

extended model, based on Jensen [13] quartic force field for water (Table 1) and Fraga's pair potential [14].

The accuracy of the procedure is schematically described in Fig. 4. The FPFF representation (Eq. (4)) perfectly matches the quartic force field. The FPFF representation requires the calculation of 8 terms, which can be reduced to fewer calculations in the machine implementation. The quartic force field [13] requires, in comparison, 31 terms. A simple inspection of Eqs. (1) and (4) (using for Eq. (1) the quartic force field presented in Table 1) shows that, for the number of operations involved in a *typical* implementation, the FPFF is expected to be approximately three times faster than the traditional implementation of a quartic force field due to the smaller number of operations involved. However, aspects other than the total number of operations must also be considered. For instance, the quartic representation in Fig. 1 requires the evaluation of Θ , which is

not easily related to the Cartesian coordinates, which make the FPFF implementation even more favorable. On the other hand, the need of accessing parameters *at random* for the FPFF functional form can slow down this calculation. Thus, the advantages of the different implementations of a force field are complex to evaluate and depend on many variables. In order to have a balanced estimate of the ratio of speed of the different procedures, 10^7 random conformations of a water molecule were evaluated using sample codes which only proceed through the force field calculation (no I/O or other operations are involved). The results are summarized in Table 2. The FPFF implementation is, on average, a factor of 7 faster than any of the alternative expressions of similar quality in any platform tested. The numbers presented in Table 2 for the MASPAR machine are extrapolations from reported benchmarks [15]. A particular quality of this representation is especially evident in this platform,

Table 2

Upper part: Rate of speed in different platforms. *Float* indicates that the FPFF representation use floating point operations. *Int* indicates that integer calculations are used. MASPAR numbers are extrapolated from benchmark values [15]. Lower part: CPU time rates for benchmark runs. The rate expresses the ratio of (CHARMm CPU time versus M CPU time) or (M* CPU time versus M CPU time). M* describes our implementation of the traditional MD calculation using a truncation scheme for the non-bonded interactions. The calculations are 15 ps MD trajectories, with the exception of the 216 water molecule simulations which were run over 150 ps. For the complete description of the 216 water molecule simulation see the main text. A 12 Å cutoff with a switching function to 13 Å was used for the CHARMm and M* calculations, and a cutoff of 9 Å for the M calculations. The atoms not included within this radius in the M calculations are accounted for by the field interpolation technique. All results reported here were calculated in an otherwise idle SGI/4D440 machine. Similar results were obtained for simulations carried out on a CrayYMP8/128 machine

Computer	Morse	Quartic	FPFF int	FPFF float
RS6000	9	6	1	1.32
R4400	7	6	1	1.12
Cray XMP	6	5	1	1.0
MASPAR	12	17	1	12.3
Intel80486	21	19	1	11.5
Problem			CHARMm	M*
216	water molecules		1.3	1.1
343	water molecules		2.7	2.0
749	water molecules		5.7	4.5
3375	water molecules		21.0	12.0
1331	argon atoms (125 charged)			10.8
12167	argon atoms (343 charged)			32.1

given that, for a certain range of r , r' and r'' only a set of parameters is selected, both the values of the variables (R_{ij}), and the value of the parameters ($\{R_{ij}\}$) are within a given (known) range, floating point calculations can be avoided. The Weitek processors currently used in the MASPAR machine are poor floating point CPUs.

4. Intermolecular force field

The implementation of the intramolecular force field, an accurate representation of intermolecular interactions, and its application to water will be discussed first. The procedures that speed up calculations of non-bonded interactions will be discussed afterwards. Subsequently, water will be used as an application example of the algorithms. In biomolecular dynamics simulations protein motions are more realistic when the discrete solvent is explicitly included [16]. Hence, the knowledge gathered from water MD simulations will be applicable to the treatment of biomolecules in solution. In simulations there is a never fulfilled balance between the contradictory goals of maximum realism and greatest simplicity. Trends toward simplicity are a natural consequence of the limitations in the computer power available. A simple functional form for a generic

force field that describes interactions between two non-bonded atoms can be written as follows:

$$E_{ij} = \frac{q_i q_j}{R_{ij}} + \frac{(q_i^2 \alpha_j + q_j^2 \alpha_i)}{R_{ij}^4} + \frac{\alpha_i \alpha_j}{[(\alpha_i/n_j)^{1/2} + (\alpha_j/n_i)^{1/2}] R_{ij}^6} + \frac{C_{ij}}{R_{ij}^{12}}, \quad (5)$$

where E_{ij} is the interaction energy between atoms i and j ; q_i , α_i and n_i are, respectively, the atomic effective charge, polarizability and the number of electrons for the atom i . This is the functional form used, for example, in Fraga's pair potential representation [14–17,18]. The conventional assignment of different terms is, from left to right: the Coulomb interaction, polarization, dispersion, exchange, contact, and repulsion terms. The listed parameters represent a convenient decomposition of the molecular interaction terms used for numerical reasons (these parameters are, in a first approximation, transferable between homologous atoms in different molecules). Other force fields frequently used [19,20] have similar functional forms, although the $1/R^4$ term is seldom used.

The Coulomb term is by far the most computationally expensive of all terms in the molecular

Table 3

Estimated cost of a typical MD calculation using different number of atoms for the biomolecule, different number of water molecules, and different quality in the representation of the biomolecule atoms (either united atom representation or an all atom representation). The computational cost is expressed in logarithmic scale. The addition of bulk water is clearly the most expensive alternative

Number of protein atoms	Number of water atoms	Estimated cost of the calculation		
		only protein	only water	all atoms
1000 (united atoms)		5		
	500 (3 center)	5	6	6
	500 (4 center)	5	6	7
	1000 (3 center)	5	7	7
	1000 (4 center)	5	7	8
1000		6		
	500 (3 center)	6	6	7
	500 (4 center)	6	6	7
	1000 (3 center)	6	7	7
	1000 (4 center)	6	7	8

force field for two reasons: (1) the inherent expense of evaluating a square root for an Euclidean distance; (2) the number of distances to evaluate grows as the square of the number of *centers* used in the molecular representation. Many of the widely used water models use a three- or even four-charge center representations: SPC [10], ST2 [12], TIPS [21]. In comparison, protein force fields frequently use *less* than a charge per atom by using a united atom representation. In this representation *several* atoms can be described as a single, unified entity (Table 3).

In section 5 simplifications for the treatment of this term will be introduced. In order to test our ideas we will use an extended representation including high-order multipoles, which will ensure the applicability of our results to any generic representation.

The first two terms in Eq. (5) are a simplified description of the electrostatic interactions between molecules. The electrostatic potential of a molecule, however, can be expressed in more general terms as

$$4\pi\epsilon_0 V(R) = Q/|R| - \mu \cdot \nabla(1/R) + \frac{1}{3}\Pi : \nabla\nabla(1/R) + \dots, \quad (6)$$

where $V(R)$ is the electric potential measured from the centroid of charge of the molecule (R), Q is the molecular charge, μ the molecular dipole, and Π the molecular quadrupole.

In the presence of an external field (E) the energy (W) of the charge distribution can be expressed as:

$$W = W_0 - \mu \cdot E - \Pi : E' \dots, \quad (7)$$

where E' is the field gradient tensor, and W_0 the zero field contribution. If the charges are mobile, they will redistribute themselves. Hence the moments μ , Π , ... will change, according to:

$$\mu(E) = \mu_0 + \alpha \cdot E + 1/2!\beta : EE + \dots, \quad (8)$$

where α is the dipole polarizability, and β the first dipole hyperpolarizability. A similar expression can be written for the change in the quadrupole or any higher order multipole.

Formally, the multipole expansion is just a Taylor series expansion of $V(R)$ on an arbitrary

number of centers conveniently chosen. Thus, $V(R)$ is a favorable mathematical expression of the electrostatic potential, but it is by no means a unique decomposition. For instance, an equivalent multicentric expansion can be written based on an atom expansion, or on any arbitrary collection of centers (for an excellent review of the subject refer to the work by Larson [22]). However, only *molecular* multipoles are physical observables. Therefore, in what follows we will constrain the internal parameters of the force field to the values of these molecular observables. The further decomposition of the molecular multipoles in atomic multipoles was chosen for two reasons: (1) we plan to apply the results of our studies in water to the description of larger systems, and atomic properties are far more convenient entities due to the ease with which they can be transferred between similar atoms of different molecules; (2) an atom-centered representation of a force field allows to implement operations which speed up the computations.

5. Evaluation of the multipole terms

An initial set of atomic multipoles were evaluated using Lipscomb prescription [23] with RHF level calculations, using modified Dunning atomic basis sets [24]. These initial multipoles were scaled, using the cumulative fitting of multipoles introduced by Kong and Yan [25], with the restriction that our initial expansion is defined by the above mentioned *ab initio* atomic multipoles. Therefore, the procedure is reduced to a least-squares fitting of a set of scale parameters [25]. The components of the experimental molecular multipoles used for this purpose are reported in Table 1. The initial atomic charges and expansion parameters (as in Eq. (5)) are borrowed from Fraga's force field [14,17]. Three main modifications are introduced: (1) a multiclassification scheme is used [4]; (2) an elliptical correction to the C_{ij} parameter (Eq. (1)) for the oxygen–oxygen van der Waals interaction [26]; and (3) a correction for the anisotropic polarizability (Table 1). The latter correction is a very first approximation to higher-order hyperpolarizabilities (Eq. (8)). The

parameters reported in Table 1 are calculated at the RHF level as the dipole deformation under strong electric fields in the three orthogonal directions. A proper treatment of this matter would require a polarizability tensor (Eq. (8)). Due to the internal organization of the computations this description would not result in extra computational cost but in a larger requirement of memory.

It must be noted that the aim of this work is to prove the feasibility of using the level of representation discussed above to the liquid state simulation. Thus, the chosen parameters, although not arbitrary, have not been optimized. The aim of the following sections will be a discussion of the *implementations* of this kind of force field in computers.

Point charge representation of multipoles. Suppose for simplicity that we represent a molecule as a set of point charges $\{q_1, q_2, \dots, q_n\}$ located at positions $\{r_1, r_2, \dots, r_n\}$. The quantities Q , μ and Π (Eq. (7)) for this charge distribution can be calculated as

$$Q = \sum q_i, \quad (9)$$

$$\mu = \sum r_i q_i, \quad (10)$$

$$\Pi = \frac{1}{2} \begin{pmatrix} \sum q_i(3x_i^2 - r_i^2) & \sum x_i y_i q_i & \sum x_i z_i q_i \\ \sum y_i x_i q_i & \sum q_i(3y_i^2 - r_i^2) & \sum y_i z_i q_i \\ \sum z_i x_i q_i & \sum z_i y_i q_i & \sum q_i(3z_i^2 - r_i^2) \end{pmatrix}. \quad (11)$$

By replacing these expressions in Eq. (7), the electrostatic potential $V(R)$ at a point R due to these point charges is given by

$$4\pi\epsilon_0 V(R) = \sum q_i / |R - r_i|. \quad (12)$$

Eq. (12) strictly converges to the exact potential $V(R)$ (Eq. (7)) when a large number of points and a large enough distance is considered ($i \rightarrow \infty$; $R \rightarrow \infty$). However, for all practical purposes, a few charged centers per atom will result in an almost quantitative approximation of $V(R)$ for values of R near and beyond the van der Waals radii of the atom.

In our program, the internal representation of the potential $V(R)$ results from an expansion of the atomic multipoles in a grid of $3 \times 3 \times 3$ point charges. The representation of multipoles by discrete charge decomposition have been discussed in the literature [21,27]. We must point out though that in our procedure, the charges are projected from *atomic* multipoles into *atomic* grid charges. Hence, the procedure is greatly simplified.

Our strategy can be overviewed as follows: the higher-order molecular multipoles (up to octupoles) are first fitted into lower order (up to quadrupoles) atomic multipoles. Then, these multipoles are *locally* fitted to a grid of point charges (the atomic subcharges). By this factorization procedure we avoid the accumulation of errors and ensure that the overall electrostatic field will be properly represented. One of the greatest advantages of using point charges is that the equations that describe the interactions between molecules are simplified: only the first term of Eq. (5) will be retained to describe these interactions, resulting in a simplified representation with a homogeneous treatment of the interaction energy terms.

6. Non-bonded interaction approximation procedures

Three families of approximations are frequently used for the simplification of the $1/R_{ij}$ calculation: simplified functional forms, truncation schemes, and clustering techniques. None of these approximations is fast or precise enough to be considered in accurate calculations. The main drawbacks of these techniques will be briefly summarized in the following part.

A fourth, newer approximation presented in this work, termed field interpolation, does not fit comfortably in any of the above mentioned categories. Given the novelty of this procedure, a separate section is dedicated to its description.

6.1. Simplified functional forms

Two forms: Robson's polynomial representation [28] and a $1/R^2$ form are frequently used instead of $1/R$. The second one, commonly men-

tioned as an R -dependent dielectric constant representation, is a heuristic description that considers the average dielectric effect of water in the electrostatic calculations between the atoms of a protein or small peptide. These approximations might never be used to represent the interactions between water molecules, since these simplifications are designed to effectively take into account the role of water in biomolecular simulations. In addition to the described deficiencies of these functional forms in the description of the electrostatic screening effects [29], these approximations must *not* be used for the calculation of the energy derivatives with respect to the atomic coordinates. In other words, these simplifications should not be used in MD calculations. The reasons are that, even though the distance-dependent dielectric constant model suppose to include the bulk properties of the surrounding solvent, it does so only partially, failing to incorporate friction effects characteristic of the water environment. Thus, the time scale of the simulation and effects like the dumping of low frequencies, natural in the water environment, will not be properly described.

6.2. Truncation schemes

Given that the cost of the electrostatic calculation increases with the square of the number of atoms considered, a most natural approach to reduce this expense is to truncate the number of atoms considered. This can be accomplished by using a radius cutoff (Fig. 5B). The main drawback of this approach is the introduction of very high frequency noise due to the sudden *jump* in energy at the cutoff border. Two partial solutions have been described to deal with this problem [30,31]. The first is a *switching* function, which is applied in the gap region between the cutoff radii and an auxiliary, larger cutoff radius. The switching function smooths the gap by applying a sigmoid numerical correction which varies between the value of the $1/R$ function at the cutoff radius and zero, at the outside of the gap region (Fig. 5C).

Another commonly used procedure is a *shift* function. The shift is also a sigmoid function but

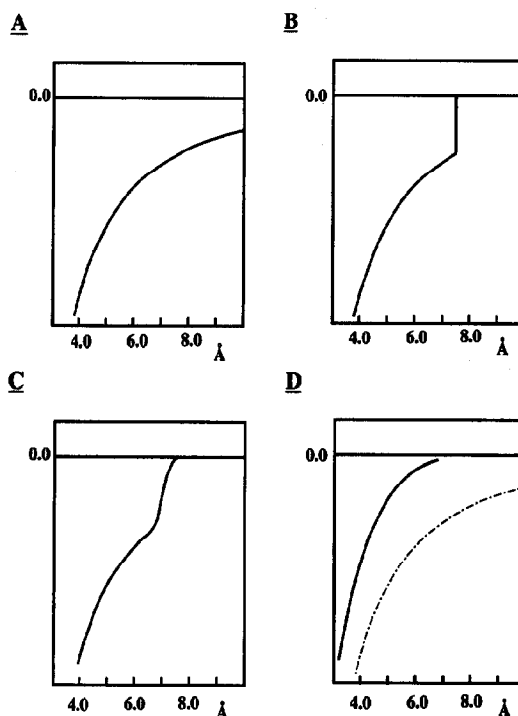


Fig. 5. Scheme of the most commonly used simplifications in the treatment of long-range interactions [30,31]. (A) $1/R$; (B) Truncation; (C) Switching function; (D) Shift function.

it is applied to the whole range in which the electrostatic calculation will be evaluated scaling the $1/R$ function. The restriction that this function imposes is that the electrostatic energy will vary smoothly from the $1/R$ value at very short distances, to zero at the cutoff radii. The effective form of the electrostatic component in this case (Fig. 5D) clearly resembles the $1/R^2$ dependency discussed before.

A hidden problem in these techniques is described in Fig. 6. If the total energy between two complete water molecules is analyzed instead of considering only two atoms as we did in Fig. 5, it soon becomes apparent that the interference between the smoothed functions for each pair of atoms result in a gross artifact at the cutoff region. The solution to this problem entails using *group switching* functions, which consist of applying the same switch function to a *group* of atoms, usually arranged in a molecule or a chemical

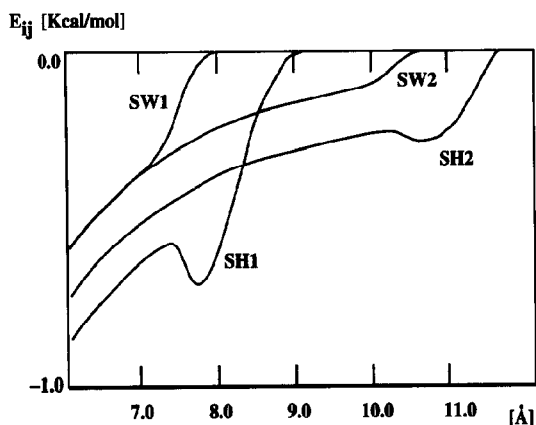


Fig. 6. Interaction energy between two water molecules (in a C2 symmetric arrangement, where the dipole moments vectors are aligned), using the TIPS force field. SH: Shifting functions applied based in atoms. The artifacts created by the interference of the atom representation are evident. The effect is greatly reduced by combining a group description with a switching function (SW1 and SW2, for different cut-offs).

group (Fig. 6). However, this is only a partial solution. As in any hierarchical arrangement, the problem simply propagates to a higher level. If instead of considering a single water molecule as in Fig. 6, we consider *several* water molecules at once, the artifact reappears (although much smoother).

6.3. Clustering techniques

Clustering techniques belong to a different class of algorithms used to speed up non-bonded computations. Cell or grid clustering consists in indexing the atoms by their arrangement through *space* (irrespective of their chemical site). To understand this idea we can imagine the region of the space we want to explore layered in the three orthogonal Cartesian directions. We will then say that an atom belongs to a given cell when its coordinates place it within the boundaries of that cell, defined by the intersection of the layers. We can number these cells, and later assign the index of the cell to a particular atom. When the neighboring atoms must be located (for a certain cutoff

distance) the complete list of pairs must not be scanned, since the neighbor cells for a given cell are known beforehand, and, since all atoms are cell-indexed, the complete list of neighbors can be gathered with minor effort. The main drawback with this technique is that it does not adapt well to modern computer architecture. In fact, most current programs use some form of the switching algorithm and a brute force neighbor list calculation with a delayed refresh rate. The refresh rate is frequently coupled with the size of the window of the switching function. Recent discussions of hierarchical clustering techniques bring new hope for the implementation of these techniques in modern machines at no extra cost [32]. However, after the cell indexing is completed and the list of neighbors built, a treatment for the truncation of the $1/R$ function is again required.

7. Field interpolation

As we have shown in section 6, the distortions introduced by the common simplifications in the treatment of non-bonded interactions are too large to be considered as alternative procedures in accurate calculations.

We have been studying a reverse form of the cell cluster technique. In this case, after the atoms are indexed (clustered) it is the electrostatic *field* that will be calculated in fixed points in space. Then we will index the atoms as they are placed in this grid. In a following step, we will calculate the field generated for the average multipoles in the cells, at the corners of other distant cells. After the values of the electrostatic field have been evaluated at the corners of the cells, the interaction between an atom (or any charged center) with other atoms far away from it, can be easily represented. This is accomplished by calculating the interaction between the *interpolated* value of the field at corners of the cell at the point defined by the coordinates of that particular center, and the charge of that center: $F = E_{\text{int}}Q$, where E_{int} is the value of the interpolated field, Q that of the local charge, and F the force exerted on the center charge by the electric field

cretated by the distant neighbors at the center coordinates.

In this way, the process of calculating the interaction between distant centers is factored into two fully optimized operations: a projection from cell to cell, and an interpolation within each cell for the calculation of the interaction with each cell internal center. This operation is simple to implement and fast to execute since the position of each cell with *respect to each other* is known as a simple function of the running indexes of the cells. In a similar fashion, the values of the transcendent functions (i.e. to calculate the contribution of a dipole center in one cell to the corner of some other cell requires a computationally expensive evaluation of cosine functions) can also be precomputed and properly indexed.

Even more important, the procedure, due to its simplicity, can be implemented in parallel or vector machines. The implementation of this algorithm is clearly machine dependent.

For instance, in a parallel implementation in a mesh architecture, the simplest way of broadcasting the information from one cell to another is by repeated neighbor passes. If this communication is bi-directional the maximum number of broadcasts will be proportional to the square root of the diagonal of the mesh. In other words, such an implementation would have a computational cost that grows only with the total number of charge centers (N). This is faster than the traditional truncation and clustering techniques that in favorable cases have a computational cost which grows with $N \log N$.

Another advantage of this representation is its high accuracy attainable in the representation of the electrostatic interaction.

Exploratory studies of techniques similar to the field interpolation can be found scattered in the literature. A description of some of the best results can be found in the work of Hockney and Eastwood [33]. These calculations were carried out at the end of the 1970s. The traditional clustering technique, combined with truncation schemes, was finally preferred over these other approaches. The reason why these techniques were relegated can be found in the early stage of development of the computer hardware at the

time. The development in the last decade of larger memory machines allows us to reconsider simple and efficient computational techniques, whose implementation a few years ago were very difficult a few years ago.

8. M

M (Molekylarefterapning med Datamaskin, R.E. Cachau, unpublished) is our prototype MD program. M is structured as a collection of highly specialized routines that are arranged in small custom programs as required. This very modular arrangement of the algorithms eases the optimization of the FPFF and field interpolation calculations in any given machine. The projection of molecular multipoles into atomic multipoles, as well as the projection of these atomic multipoles into atomic subcharges, precedes the MD calculation. In a similar fashion the FPFF coefficients are calculated in an early stage. The atom multipole projection into atomic subcharges and most of the FPFF projection scheme are automatic. The projection of the molecular multipoles into atomic multipoles is a more complex task and requires a more detailed manipulation.

After the projections are completed a library of atom types is built, which includes most of the FPFF information as well as an extended topology (i.e. including chirality) and local charges and polarizabilities. In this sense, the MD central program ignores the origin of a particular FPFF representation.

9. Test calculations

A suite of MD trajectories was used to test the algorithms described in this paper. The algorithms were implemented in M. Several models, in addition to the force field described in Table 1, were used for benchmark purposes. These are the SPC [10], MCY [11], ST2 [12], and TIPS [21] models. These models have been thoroughly tested. All model force fields were subject to the same treatment: the force fields were projected in

an FPF representation, and the electrostatic components were used to fill the subcharges representation. In order to estimate the speedup ratios, comparable calculations were carried out with CHARMM and a program written ad hoc to perform traditional MD calculations in simple liquids.

The MD calculations were performed as follows (unless stated otherwise): The non-bonded interactions were split in near interactions, for pairs of atoms closer than 9 Å (treated as discrete centers); and far interactions, simplified with the field interpolation treatment. An additional simplification used is that the subcharges are added up to atom center charges for interactions beyond 6 Å. The grid size used for field interpolation was 1.25 Å. In most MD calculations discussed here the total number of water molecules was 216 for the central box. The size of the system was chosen to facilitate the comparison with results reported in the literature, although the system is

too small to show the full advantage of the field interpolation technique. Simulations were carried out at 300 K. The equilibration time was 30 ps, after 10 ps of heating. The production trajectories were 150 ps long, with an integration interval of 0.5 fs. The environment region was described using neighbor replicas. This is a simple form of describing periodic boundary conditions, where the central box is copied in 26 symmetrically arranged copies surrounding the central box.

Since at the level of calculation of the MD program most force fields are equally complex, due to the use of the FPF representation, the comparison of speed *between* different force fields using our program is not very informative. Several test calculations were carried out instead using CHARMM in a similar size problem. The TIPS [21] water model and an 1 Å cutoff was used, all other conditions being similar. The test runs were done on an SGI/4D440 and a CrayYMP8/128. The speed ratio in each case

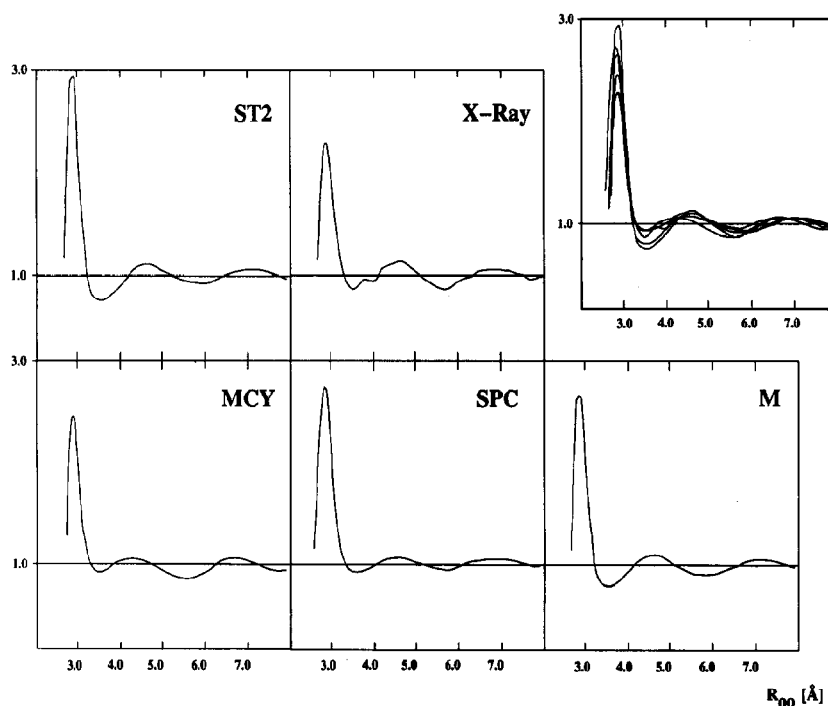


Fig. 7. g_{OO} versus distance R_{OO} for the different water force field tested and the X-ray experimental value [34]. The insert in the upper right corner is a comparison of the different profiles.

was 1:1.3 and 1:1.1 in favor of our implementation in both cases. The size of the system (216 water molecules in the central box) was chosen to favor the comparison with the reported results in the literature. Thus, the size of this particular system is too small to fully show the advantages of the techniques here described. In addition, our representation of the non-bonded interaction requires the consideration of a much larger number of charge centers. To make this test more useful, a 15 Å cutoff in the CHARMM calculation was chosen to use an approximately equally large list of non-bonded interaction in both programs. Therefore, this particular test shows the speedup due to the better organization of the neighbor list in our technique. Larger calculations were carried out with up to 3375 water molecules, and with a mixture of argon and charged argon with up to 12167 particles in the central box for the purpose of speed comparison (Table 2). The ratio of speedup for our implementation when compared with CHARMM or a similar MD program range from 1:5 to 1:32. The cutoff in these calculations (CHARMM) is of 12 Å. The results shown in Table 2 are even more remarkable if we considered that, due to the different quality of representation in both programs, the number of pairs of centers considered in M is much higher than in CHARMM. The 12 Å cutoff is the smallest cutoff that can be used in the CHARMM calculations in order to obtain similar trajectories and energy fluctuations comparable as those obtained with M. The energy fluctuations were studied during the longer water simulations. The typical energy fluctuation with CHARMM was in the range ± 0.4 kcal/mol (98.5% confidence). During the M trajectories this range was of only ± 0.12 kcal/mol. We think that the source of this difference is in the treatment of the boundary condition at the cutoff distance for the $1/R$ function. The switching function used with the CHARMM program seems to introduce higher noise into the system. All other treatment tested (e.g. shift functions) result in even higher levels of noise.

The most sensitive test of the accuracy of a MD procedure was found to be the radial distribution functions (g_{OO}) and their change under

different simulated conditions. Radial distribution functions have a remarkable tendency to *flatten* as the result of noise [10].

The g_{OO} functions at room temperature and pressure are reported, for different models, in Fig. 7. The MCY force field seems to be the most accurate. However, the results shown for our force field seems to be very promising given its early stage of development.

Irrespective of the goodness of each force field, the precision with which the combination of procedures reported here reproduces the published results for the radial distribution functions [10], is a remarkable result. As we have mentioned, radial distribution functions are characteristic tools of analysis since noise sources tend to *flatten* their profile. In this sense, the sharp profiles obtained closely agree with those reported in the literature [10,12,21]. A detailed analysis of these profiles, shown here only as an example of the accuracy of the procedure, is beyond the scope of this presentation. However, it can be noted that there is a resemblance between the first mini-

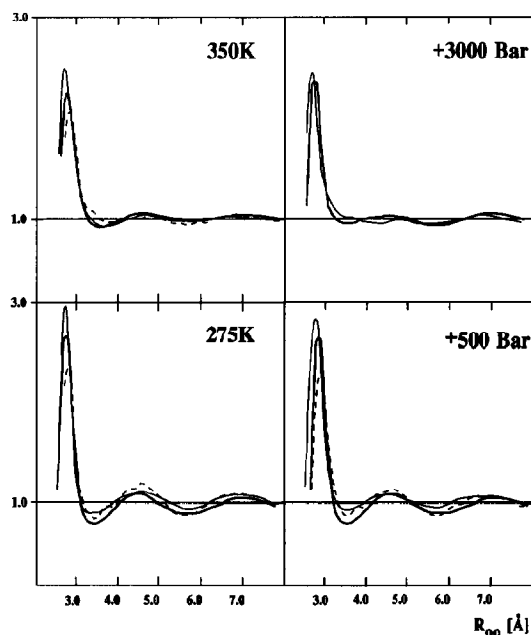


Fig. 8. g_{OO} versus R_{OO} at different pressures and temperatures. M is in thick lines; SPC is in thin lines; experimental values [34] are in dotted lines.

Table 4

A comparison of force field parameters. The force field parameters follow the usual nomenclature. ρ is the average density at constant pressure ($T = 300$ K). R_{eq} is the O–O distance for the dimer energy minima. Note that even though the fitted properties are almost equally well described (see also Fig. 9), the internal components compensate in very different ways for their lack of flexibility in the description of the system. In fact, in empirically fitted potentials the individual contributions cannot be analyzed since the potentials are fitted in such a way that they compensate for the deficiencies of the functional form to reproduce an average property. Thus, for instance, the electrostatic components will contain part of polarization and other effects. To make matters worse these compensations are not equivalent between different potentials. As a consequence apparently similar contributions in different potentials cannot be compared and the physical meaning of the different components within a given potential is not clear and its improvement is difficult

	SPC	BF	TIPS2	M
ρ	0.971	1.181	0.982	0.953
R_{eq}	2.75	2.72	2.25	2.20
r_{OH}	1.0	0.96	0.9572	0.9572
$\angle(HOH)$	109.47	105.7	104.52	104.52
A	629.4	560.4	695.0	595.5
C	625.5	837.0	600.0	603.2
q_O	−0.82	—	—	−0.62
q_H	0.41	0.49	0.535	0.31
q_{LP}	—	−0.98	−1.07	—
r_{OLP}	—	0.15	0.15	—

Dimer Energy [Kcal/Mol]

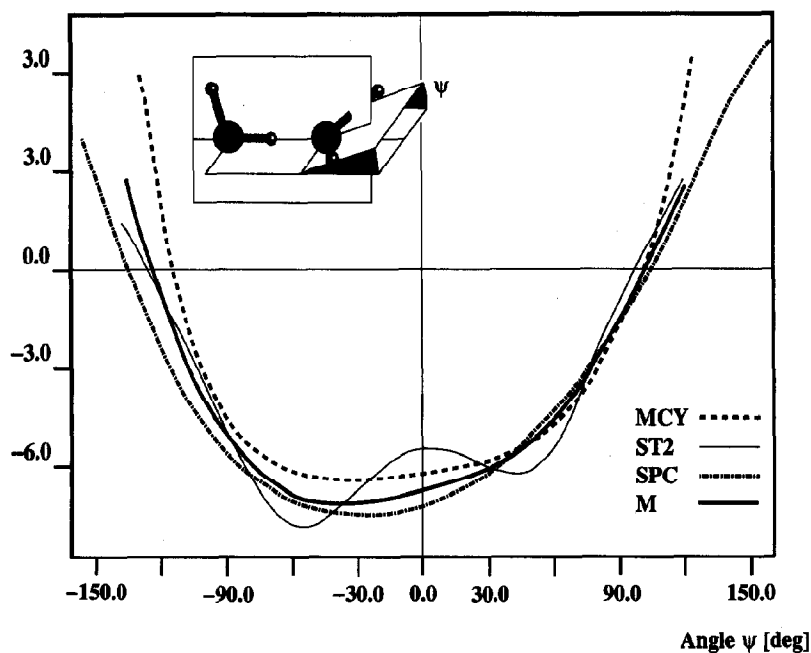


Fig. 9. Librational energy around the angle Ψ for the water dimer. Note the peculiar form of the energy surface for the ST2 potential.

mum of the g_{OO} function for the M and ST2 force fields whose origin is not clear to us, and it requires further investigation.

Another very sensitive test that reveals the overall behavior of a force field and its implementation is the calculation of the *change* of g_{OO} as a function of temperature and pressure. Fig. 8 describes this behavior. The peaks for the radial distribution function for the g_{OO} M force field at higher temperature and pressure can be clearly noticed. The SPC force field shows the reported [10] loss of structure in these simulations. This loss agree with the results described in the literature. No artifacts due to the use of an FPDF or the field interpolation were observed. The g_{OH} and g_{HH} profiles show similar behaviors.

The comparison of the force field parameters (Table 4) is not very informative since most force fields contain *effective* parameters which compensate between themselves but which cannot be considered independently. In this sense, the use of first principle force fields presents a practical advantage. Since the parameters have not been chosen to compensate each other, they truly represent components of the interaction energy. Therefore, different contributions can be studied separately and a physical meaning can be attributed to them. This sort of analysis cannot be carried out with empirically fitted force fields.

A more thorough analysis of the energy hypersurface reveals, however, the true difference in nature between the different potentials: one of the most remarkable differences can be seen in the librational motion described in Fig. 9. Our description has too deep a minimum, which needs to be corrected. However, the aberrant behaviors exemplified in Fig. 9 for the ST2 model were not noticed with our force field in any of the considered geometries of the water dimmer.

Most average properties estimated (Table 4) were found to be less sensitive to the use of different force fields. This result is not surprising given that the water force field parameters are adjusted to reproduce water bulk properties. However, when these results (Table 4) are compared with the results reported in the literature [9,14,17], it can be noticed that no penalty was paid by using our representation.

10. Conclusions

The direct translation of equations into programs has been a usual way of developing computational methods in many areas of chemistry and physics. A traditional MD program is developed around a particular functional form of the force field. This results, at later stages of development, in the impossibility of improving the functional form of the force field. The use of analytical expressions is useful for a *mathematical* treatment of a problem. However, not all molecular models are amenable to analytical expressions (i.e. Hartree–Fock SCF technique). The most general form that a function can take is that of a look up table. Modern computers control large amounts of memory, making the implementation of numerical approaches feasible. This, together with a very fast development of computers, makes the traditional form of program development essentially obsolete. A modern design of computational chemistry tools should be based as much on the physics that the model describes as on the understanding of the computational details of the model's implementation.

The algorithms presented in this work are based on numerical functions. These functions are at the core of a large number of numerical solutions commonly used in engineering. The use of numerical approximations assures a high speed of execution while preserving the accuracy of the computer representation of a physical model. The FPDF and the field interpolation algorithms are designed to take advantage of newer architectures. This is accomplished by avoiding recursive operations and, instead, organizing the information so that global operations can be applied to long lists of data. The algorithms are designed *independently* of a particular force field to be used. The modular structure of the code ensures maximum flexibility and transportability. M (Molekylarefterapning med Datamaskin) is built as a collection of highly specialized routines which are linked in custom programs as a need for particular tasks required. External to this core set of routines are tools for the manipulation of common force field functional forms and their projection into the FPDF and the atomic sub-

charges. These tasks are completely independent of the central program.

A unique feature of the M representation is its ability to describe diverse data dependencies. The force field functional forms might contain, for instance, conformationally dependent atomic charges. This dependency will be suited for a description similar to that introduced in Eq. (2) for the treatment of the FPF expansion coefficients.

The feasibility of the programming approach describes in this work has been demonstrated in the implementation of known water force fields and the accurate reproduction of the results reported in the literature. On the other hand, our implementation of a first principle force field for water demonstrates the ability of the FPF to handle complex force fields. The quality of the results obtained with this force field encouraging. Extensions of this force field, including higher-order corrections (e.g. three-body terms), are being developed.

Acknowledgement

Structural Biochemistry Laboratory, Frederick Biomedical Supercomputing Center, PRI/Dyn-Corp, National Cancer Institute – Frederick Cancer Research and Development Center, Frederick. Research sponsored by the National Cancer Institute, DHHS. The content of this publication does not necessarily reflect the views or policies of the Department of Health and Human Services (DHHS), nor does mention of trade names, commercial products, or organizations imply endorsement by the US Government.

References

- [1] J.A. McCammon and S.C. Harvey, *Dynamics of proteins and nucleic acids* (Cambridge Univ. Press, Cambridge, 1988).
- [2] N.L. Allinger and J.-H. Lii, *J. Comput. Chem.* 12 (1991) 186; A.T. Hagler, S. Lifson and P. Dauber, *J. Am. Chem. Soc.* 101 (1979) 5122.
- [3] W.F. van Gunsteren and H.J.C. Berendsen, *Angew. Chem. Intern. Ed. Engl.* 29 (1990) 992.
- [4] E.A. Bidacovich, S.G. Kalko and R.E. Cachau, *J. Mol. Struct. THEOCHEM* 210 (1990) 455.
- [5] S. Fraga, *J. Comput. Chem.* 3 (1982) 329.
- [6] T. Schlick, *J. Comput. Chem.* 10 (1989) 951.
- [7] C.L. Brooks III, W.S. Young and D.J. Tobias, *Intern. J. Supercomputer Appl.* 5 (1991) 98.
- [8] J.F. Janak and P.C. Pattnaik, *J. Comput. Chem.* 13 (1992) 533.
- [9] H.W. Press, B.P. Flannery, S.A. Teulosky and W.T. Vetterling, *Numerical recipes. The art of scientific computing* (Cambridge Univ. Press, Cambridge, 1986).
- [10] J.P.M. Postma, Thesis Dissertation, Groningen University (1985).
- [11] O. Matsuoka, E. Clementi and M. Yoshimine, *J. Chem. Phys.* 64 (1973) 1325.
- [12] F.H. Stillinger and A. Rahman, *J. Chem. Phys.* 60 (1974) 1545.
- [13] P. Jensen, *J. Mol. Spectryc.* 133 (1989) 438.
- [14] S. Fraga, *Comput. Phys. Commun.* 29 (1983) 351.
- [15] *Proceedings of the Supercomputing Meeting, Minneapolis* (1993).
- [16] M. Levitt, *Proc. Natl. Acad. Sci. USA* 85 (1988) 7557.
- [17] J.A. Sordo and S. Fraga, *J. Comput. Chem.* 7 (1986) 55.
- [18] E. Clementi, G. Corongiu and G. Ranghino, *J. Chem. Phys.* 74 (1981) 578.
- [19] B.R. Brooks, R.E. Bruccoleri, B.D. Olafson, D.J. States, S. Swaminathan and M. Karplus, *J. Comput. Chem.* 4 (1983) 187.
- [20] W.F. van Gunsteren, in: *Modeling of molecular structures and properties*, (ed. C. Troyanowsky (Elsevier, Amsterdam, 1990).
- [21] W.L. Jorgensen, J. Chandrasekar, J.D. Madura, R.W. Impey and M.L. Klein, *J. Chem. Phys.* 79 (1983) 926.
- [22] E.G. Larson, M. Li and G.C. Larson, *Intern. J. Quantum Chem.* 26 (1992) 181.
- [23] J. Liang and W.N. Lipscomb, *J. Phys. Chem.* 90 (1986) 4246.
- [24] T.H. Dunning, *J. Chem. Phys.* 55 (1971) 716.
- [25] J. Kong and J. Yan, *Intern. J. Quantum Chem.* 46 (1993) 239.
- [26] C. Millot and A.J. Stone, *Mol. Phys.* 77 (1992) 439.
- [27] W.A. Sokalski, D.A. Keller, R.L. Ornstein and R. Rein, *J. Comput. Chem.* 14 (1993) 970.
- [28] B. Robson and E. Platt, *J. Mol. Biol.* 188 (1986) 259.
- [29] A.A. Rashin, *Progr. Biophys. Mol. Biol.* 59 (1993) 395.
- [30] J. Guenot and P.A. Kollman, *J. Comput. Chem.* 14 (1993) 295.
- [31] K. Tasaki, S. McDonald and J.W. Brady, *J. Comput. Chem.* 14 (1993) 278.
- [32] D.B. Beglov and A.A. Lipanov, *J. Biom. Struct. Dyn.* 9 (1991) 205.
- [33] R.W. Hockney and J.W. Eastwood, *Computer simulation using particles* (McGraw-Hill, New York, 1981).
- [34] A.H. Narten and H.A. Levy, *J. Chem. Phys.* 55 (1971) 2263.