Université d'Ottawa · University of Ottawa

Université d'Ottawa · University of Ottawa

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES

FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

Sébastien OSBORNE

AUTEUR DE LA THÈSE - AUTHOR OF THESIS

M. Sc. (Chemistry)

GRADE - DEGREE

Department of Chemistry

FACULTÉ, ÉCOLE, DÉPARTEMENT - FACULTY, SCHOOL, DEPARTMENT

TITRE DE LA THÈSE - TITLE OF THE THESIS

Implementation and Application of a GCOSMO Solvation Model within DeFT and a Hybrid DeFT/Mopac Monte Carlo Algorithm

A. St-Amand

DIRECTEUR DE LA THÈSE - THESIS SUPERVISOR

CO-DIRECTEUR DE LA THÈSE - THESIS CO-SUPERVISOR

EXAMINATEURS DE LA THÈSE - THESIS EXAMINERS

P. Mayer

J. Wright

J.-M. De Koninck, Ph.D.

LE DOYEN DE LA FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES

SIGNATURE

DEAN OF THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

Implementation and Application of a GCOSMO Solvation Model within DeFT and a Hybrid DeFT/Mopac Monte Carlo Algorithm

SÉBASTIEN OSBORNE

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements
for the degree of Master of Science in Chemistry

The Ottawa-Carleton Chemistry Institute
University of Ottawa

Your file Votre référence
ISBN: 0-612-90341-9
Our file Notre référence
ISBN: 0-612-90341-9

Table of Contents

Abstract 1

1) Condensed Phase Calculations
1.1. Importance of condensed phase calculations 2
1.2. Problems associated with traditional explicit solvent calculations 5

2) Continuum Models
2.1. A quick overview 8
2.2. The cavity 10
2.3. Advantages and disadvantages of continuum models 12
2.4. Generalized COnductor-like Screening MOdel 16

3) Solvent Interaction Potential Radii
3.1. Theory and equations 23
3.2. Results and discussion of SIP calculations 27

4) GCOSMO Applications with DeFT
4.1. Fixed vs. SIP radii 32
4.2. Solvation of glycine and tautomerization 2-hydroxypyridine/2-pyridone 37
4.3. Versatility and limitations of the method 40

5) Free Energy Calculations
5.1. Importance of free energy calculations 43
5.2. Traditional ways of calculating free energies and their limitations 46
5.3. Monte Carlo in depth 52
5.4. Hybrid DeFT/Mopac Monte Carlo algorithm 57

6) Applications of the Algorithm
6.1. Ring opening of cyclobutene 62
6.2. Diels-Alder reaction of ethylene and butadiene 67

7) Conclusions 74

Acknowledgments 76

References 77

Abstract

A Generalized COnductor-Like Solvation MOdel (GCOSMO) was implemented within our own Density Functional Theory software, DeFT, combined with a Solvent Interaction Potential (SIP) algorithm to assign the atomic radii. The SIP radii were tested against a set of fixed radii from the literature. We concluded that the SIP radii outperform the fixed radii. We successfully reproduced the solvation of glycine and the tautomerization of 2-hydroxypyridine/2-pyridone in solution with our GCOSMO/SIP approach.

A hybrid DeFT/Mopac Monte Carlo (MC) algorithm was also implemented. The algorithm differs from purely ab initio MC by inserting a secondary semi-empirical MC chain between each ab initio configuration of the main chain. This leads to a reduction of the correlation between the configurations and a more efficient sampling of the potential energy surface. Two reactions were studied: the conrotatory ring opening of cyclobutene and the Diels-Alder reaction of ethylene and butadiene to form cyclohexene.

1) Condensed Phase Calculations

1.1. Importance of Condensed Phase Calculations

In the early days of computational chemistry, calculations were carried out on small isolated molecules because the computers of those days could not handle larger systems. These calculations are referred to as "in vacuo" or "gas phase" calculations since the calculated properties are those of a single molecule or a small cluster of molecules isolated from everything else. Even though gas phase chemistry is useful in the understanding of some processes, most experimental chemistry is undeniably done in the condensed phase. Gas phase properties are nonetheless insightful but often do not tell the entire story. Computational chemistry must be able to extend its successful description of systems in the gas phase to the condensed phase for it to be a truly great and useful tool for chemistry in general.

The importance of being able to successfully study systems in a condensed medium cannot be overstated. All chemical reactions of biological or biochemical relevance take place in solution. An accurate theoretical description of such reactions cannot neglect the effect of the environment. This is even truer today, now that we live in an era in which major fields like catalysis and drug design make extensive use of theoretical chemistry. For example, a computational chemist cannot test new drug candidates in vacuo and then expect his predictions, as to which candidate will work best, to hold true in vivo [1].

Nearly every field of chemistry is affected by solvents in one way or another. Without a doubt, thermodynamic measurements are greatly affected by the presence of a solvent. Predicted variations of enthalpy or free energy in the gas phase can be completely wrong and unreliable in solution, even more so when ionic species are involved. Reactions that have been extensively studied in the gas phase might end up behaving in a completely different manner in solution. The reaction could take a different pathway to get to the products or even lead to entirely different products. To prove the point that solvent effects are crucial, one only has to look at how the simplest reactions in organic chemistry, the substitution reactions, are affected by the presence of a solvent. S_N1 reactions generally will not happen in the gas phase but are common in solution. The rate determining step of an S_N1 reaction is the formation of a carbocation intermediate, which is greatly stabilized by polar solvents. Stabilizing the intermediate reduces the activation energy and leads to a faster reaction.

Table 1. Reactivity of a typical S_N1 reaction, solvolysis

Substrate	Solvent	relative rate [a]
t-butBr	100% H_2O	400 000
t-butBr	10%H_2O, 90% acetone	1
t-butBr	gas phase	much slower

[a] Data from reference [2]

In the case of S_N2 reactions, the nature of the nucleophile will have a great impact on the kinetics and the resulting products of the reaction. In the gas phase, the nucleophilicity of the halogen anions decreases when proceeding down the periodic table, in accordance with the

correlation between basicity and nucleophilicity. However, in water, that order is reversed because the smaller fluoride anion is more heavily and tightly solvated, impeding its capacity to react, whereas the bigger iodide anion is less solvated. In aprotic solvents, like N,N-dimethylformamide, the order is restored to what it was in the gas phase because of the inability of the solvent to create hydrogen bonds with the nucleophile. The solvation of the nucleophile is the most important factor to take into account when studying S_N2 reactions.

Table 2. S_N2 reaction of iodomethane with a chloride anion

solvent	classification	relative rate [a]
methanol	protic	1
formamide	protic	12.5
N-methylformamide	protic	45.3
N,N-dimethylformamide	aprotic	1 200 000
gas phase	------	much faster

[a] Data from reference [2]

As tables (1) and (2) show, two very similar reactions in nature (S_N1 and S_N2) exhibit completely different behaviors when the media changes from the gas phase to the condensed phase, demonstrating how crucial a proper description of the environment is to the understanding of chemistry.

As the years passed, the rapid evolution of computer power as well as software packages allowed chemists to perform calculations on systems of ever-increasing size. Inevitably, with the

ability to study larger systems with higher levels of theory, computational chemists began to ponder upon various ways of incorporating solvent effects in their calculations. Numerous methods and algorithms have been developed over the years on many levels, from molecular mechanics to semi-empirical to ab initio theory, in an effort to accurately and efficiently describe various processes occurring in the condensed phase. The leap toward calculations in solution is essential if computational chemistry is to become a more useful chemical "tool". A current challenge in the computational chemistry world is to develop methodologies to study processes in solution. It is safe to say that practically any chemical or physical property of a given (reasonably-sized) system can be accurately calculated in the gas phase with the current methods. Performing the same calculations in solution, with results that are within experimental error, is a real challenge for computational chemists.

1.2. Problems associated with traditional explicit solvent calculations

Incorporating the effect of a solvent in a theoretical model can be done in many different ways. Explicitly adding solvent molecules seems to be the most logical and straightforward choice. What could be simpler than studying a system surrounded with solvent molecules rather than this same system alone in the gas phase? Unfortunately, many problems can be encountered when this approach is adopted. To keep the calculations feasible, the number of solvent molecules that can be added is limited. Depending on the resources available and the objectives of the calculation, a computational chemist might be limited to a few hundred solvent molecules at a low ab initio level of theory to a few tens of thousands molecules at the molecular mechanics level. Considering that

a volume as small as one milliliter of water contains approximately $3x10^{22}$ water molecules, it is easy to realize that, no matter how sophisticated and fast computers become in the future, it will always remain impossible to explicitly include enough molecules to reproduce bulk solvent. Having an insufficient number of solvent molecules causes a problem because molecules at the surface, or within a few layers of the surface, behave differently from molecules in the bulk [1]. In a macroscopic amount of liquid, a very small percentage of the molecules are at the surface. In a cluster of a few tens of thousands of molecules, the percentage of molecules at the surface is still unrealistically high.

Further complications arise now since we are at a stage where studying huge molecules like proteins and enzymes is very popular. Bigger systems will obviously require a greater number of solvent molecules to be quasi-submerged in the fluid. To solve this problem, it is common to add a "drop of water" (like a 25Å sphere) around the active site of an enzyme rather than completely surround it. This reduces the total number of atoms, and thus reduces the length of the calculation. But even the cheapest ab initio level is still far too expensive for such a system. QM/MM or other hybrid schemes, where the active site is treated at a quantum mechanical level and the rest of the molecule, along with the drop of solvent, is treated at the molecular mechanical level, are often the best options. Another very important problem with this method is the fact that the list of potential solvents is extremely short. Water is the most important solvent because of its biological applications, but some chemists might be interested in studying other solvents also. Fortunately for us, a water molecule is made of only three atoms. It is much more difficult and time consuming to use bigger molecules like acetone, propanol or hexane as solvents. For a given number of solvent molecules, and a generous quadratic scaling factor, using acetone instead of water would translate

to a calculation that is roughly 11 times longer. Considering that a simulation in water can easily take up to a month of computer time, it is highly improbable that other solvents would even be considered.

In order to fix some of the shortcomings discussed above, periodic boundary conditions are employed. Periodic boundary conditions eliminate the surface problem since the system of interest is trapped in a box filled with solvent (usually of cubic shape but it can be of any shape capable of completely filling space) replicated in all directions to infinity, thus eliminating surfaces altogether. This approach must always be used with caution. Obviously, the box has to be completely filled with solvent in order to prevent having empty spaces between the mirror images. It also has to be large enough to minimize what is called the mirror image problem. If it is not sufficiently large, the solute will see the duplicate copies of itself in the other boxes and feel artificial electrostatic interactions with the copies. It is essential to create a buffer between mirror images of the solute that is thick enough to minimize artificial interactions. Once again, for larger molecules, a large box is required, which comes at a great computational cost. This approach works beautifully if pure liquids are to be studied (does not have to worry about the mirror images problem) but is flawed for polar or ionic solutes. Periodic boundary conditions are routinely used in empirical force fields codes but not in semi-empirical or ab initio programs. Implementing periodic boundary conditions within a molecular orbital formalism is extremely difficult.

2) Continuum models

2.1. A quick overview

The recreation of bulk solvent in an ab initio calculation will never be possible. If one wishes to strictly use the ab initio level of theory, even with the cheaper methods (like Hartree-Fock or Density Functional Theory), smaller basis sets, and a good computer, only a few hundreds of solvent molecules can be added explicitly. That number is not nearly sufficient to recreate bulk solvent. It is true today and will most likely remain so in our lifetime. If the presence of the solvent cannot be recreated explicitly, then the next best thing is to try to approximate or mimic the effects of the solvent. Continuum approaches constitute a simple and affordable alternative to perform condensed phase calculations.

Rather than explicitly adding solvent molecules, these continuum models treat the solvent as a uniform polarizable continuum fluid. This fluid is characterized by one property only, the dielectric constant of the liquid in question [3]. The reaction field generated by the polarizable continuum will modify the energy and properties of the solute. A boundary is defined between the volume occupied by the solute and the surface where the continuum starts. Cavities of different shapes and sizes, a topic that will be discussed later, have been devised and tested for various solutes. Two opposite factors will determine the solvated properties of the solute. The first factor is the electrostatic interactions of the nuclei and the electrons of the solute with the surface of the fluid. These interactions are always favorable. The second factor is the creation of the cavity itself, which is always unfavorable for the fluid. Entropic in nature, this non-electrostatic component is

often referred to as the cavitation and dispersion-repulsion energy. An unfavorable decrease in entropy is observed when the solvent molecules must make room for the solute, rearranging and ordering themselves. The non-electrostatic contribution is proportional to the surface area of the cavity. The magnitude of the two opposite factors will judge whether or not the solvation of the solute is favorable.

A number of continuum models have been developed in recent years. In the early 1990s, Klamt and Schüürmann developed the COnductor-like Screening MOdel, which was initially created for molecular mechanics applications and was later implemented with the highly popular semi-empirical program MOPAC [4]. A few years later, Truong and Stefanovich extended the COSMO model to the ab initio level of theory (namely HF, DFT and MP2). They also included dispersion, repulsion and cavity formation contributions. Their new model was named the Generalized COndoctor-like Screening MOdel (GCOSMO) [5]. The polarizable continuum model (PCM), also developed in the early to mid 1990s, is probably the most popular model of its kind. It was created by Barone, Cossi and Tomasi and implemented in the commercially available and highly popular Gaussian94 software [6]. PCM can calculate solvation free energies at the Hartree-Fock, Density Functional Theory, n^{th} order Moller-Plesset perturbation theory as well as Quadratic Configuration Interaction with Single and Double excitations (QCISD) levels. However, the analytical first derivatives are only available at the HF and DFT levels. A variation of PCM, the Self-Consistent Isodensity PCM (SCIPCM) was also developed and implemented within Gaussian94 by the same group. SCIPCM tries to deal with cavity size problems, but repeated convergence failures have

hampered its popularity. Also worth mentioning are the Finite Difference Poisson-Boltzmann (FDPB) approach [7] and the SMx models of Cramer and Truhlar [8].

2.2. The cavity

Without a doubt, the continuum model approach's performance will greatly depend on the quality of the cavity used to encapsulate the solute in the fluid. Many shapes of various sizes could potentially serve that role. The simplest choice for the shape of the cavity is a sphere. Although it makes sense for spherical molecules, other molecules with long chains or planar molecules do not fit well in a spherical cavity. Unfortunately, most molecules are far from spherical. For long narrow molecules, an ellipsoidal shape makes sense but again only in cases of long chains without any bulky side chains attached to it (like n-alkanes). Spherical and ellipsoidal cavities were initially used because they greatly simplify the calculations, providing analytical solutions to the problem of calculating the charges on the surface. Onsager first solved that problem analytically some 70 years ago [9]. Unfortunately, their usage is limited to the few molecules that have a shape suited to a spherical or ellipsoidal cavity.

For the cavity to be acceptable for any kind of molecule, including transition states, neutral, charged and zwitterionic species, it needs to be able to adapt to any shape or size possible. A more suitable approach is the "overlapping spheres" model, in which a sphere of a given radius is centered on each nucleus. The surface created by the outer edge of the overlapping spheres defines the cavity and is named the "molecular surface". The position and the radius associated with each nucleus are the parameters that will determine the exact shape and size of the cavity. Spheres located on

hydrogen atoms are much smaller than spheres centered on heavy atoms (for obvious reasons) and only slightly affect the surface. Some people use cavities with no sphere around the hydrogen atoms. In that case, the hydrogen atoms are engulfed inside the sphere of the heavy atom to which they are bound. It is believed that neglecting the creation of spheres on hydrogen atoms can improve the numerical stability of the solvation calculation, especially for geometry optimizations [10].

Three types of molecular surfaces are commonly used: the van der Waals (VDW) surface, the Solvent-Accessible Surface (SAS) and the Solvent-Excluded Surface (SES). The VDW surface is the envelope surface of a set of intersecting spheres with given radii centered on the nuclei of selected atoms in a molecule. The SAS is defined by the center of a solvent molecule (considered as a rigid probe sphere) when it rolls along the VDW surface. The SES is the surface envelope of the volume excluded to the solvent probe when it rolls along the VDW surface (a smoothed VDW surface in other words) [11].

A well-known algorithm called GEPOL (GEnerating POLyhedra) is able to build all of the above surfaces. GEPOL divides each sphere into small tesserae, all of which will be given a polarization charge for the solvated calculation. The spheres are usually replaced by an inscribed pentakisdodecahedron with 60 triangular faces. A finer division of the spheres can be used, but the consensus is that the gain in accuracy is not sufficient to justify it. The tesserae that are completely buried are discarded and those that are cut by other spheres are replaced by suitable polygons. GEPOL's standard input includes the coordinates of the nuclei, the radius associated with each nucleus and the level of tessellation of the spheres. A default value of 60 tesserae per sphere was

used. In the case of the SAS and SES, the radius of the solvent probe is required and was estimated at 1.4Å for water. For the SES, the overlapping factor and the radius of the smallest sphere that can be created are also needed. Default values of 0.8 and 0.5Å respectively were used. GEPOL's standard output is sufficient to get all the information that will be needed later on. At the end of the calculation, GEPOL gives the area and the coordinates of the center of each tessera, along with the sphere to which it belongs.

2.3. Advantages and disadvantages of continuum models

If explicit solvent calculations at the ab initio level were possible, then there would be no use for continuum models. Continuum models have been developed to remedy the problems of explicit solvent calculations and provide a much quicker alternative for adding solvent effects. Because the solvent effects are included in the solute's Hamiltonian, the solvated properties can be calculated with a single SCF procedure that is identical, in principle, to the SCF procedure of a gas phase calculation [12]. Depending on the specific model used, the level of theory and the molecule in question, the "solvated" calculation will be 10% to 40% longer than the corresponding gas phase calculation. Unless the gas phase energy is already known, both the solvated and gas phase calculations need to be performed. In that case, supposing the gas phase calculation is done first, the converged electronic density in the gas phase is an ideal initial guess to the density in solution. A good initial guess makes the solvated calculation even faster because fewer cycles are required to achieve self-consistency. Without a doubt, the biggest asset of continuum models is their computational efficiency.

Continuum models are also very versatile. It was written earlier that ab initio calculations with explicit solvent molecules are limited to HF or DFT, with a small basis set, and most likely, with water as the solvent. This is absolutely not the case with continuum models. They have been implemented within virtually all levels of theory including some very sophisticated correlated methods (HF, DFT, MBPT, GVB, MC-SCF, CISD, CCSD, etc.) [13]. Hybrid methods like QM/MM and ONIOM also have the potential to be used within a continuum model without any problem. Once the continuum model is implemented within a program, any calculation that is feasible in the gas phase is equally feasible within the solvation model. The choice of the solvent does not affect the computing time in continuum models, i.e., the size of the solvent molecules is not an issue.

The availability of analytical expressions for first and second derivatives of the energy is a great attribute of continuum models. The gradient in solution is easily calculated with these expression and requires roughly 25% more CPU time than the evaluation of the gradient in the gas phase. These first and second derivatives allow easy geometry optimizations and vibrational frequency analyses in solution. Optimizing the geometry of a system when explicit solvent molecules are added is very difficult because of the huge number of degrees of freedom. The other important application requiring the gradient is molecular dynamics (MD). MD simulations in solution could easily be performed with continuum models and would not take much longer than gas phase simulations. Many other applications, which are not applicable for calculations with explicit solvent molecules, work very well with continuum models. To give a practical example of the versatility of continuum models, consider a chemist who is interested in the infrared spectrum of a compound in solution. A vibrational frequency analysis will be necessary, an impossible task to

perform if a large number of explicit solvent molecules are included. However, with a continuum model, the same task would not be any harder than the same calculation in the gas phase.

Some will argue that, in many cases, the presence of actual solvent molecules is essential to the chemical process. For example, if a process involves a solute hydrogen bonding with water molecules, and this is of vital importance to its proper description, then the simple application of a continuum model would not be adequate. This can easily be avoided by adding a certain number of explicit solvent molecules to the solute and then surrounding this solute-water complex with the polarizable continuum. In any process in solution, the most important solvent molecules are those that are closest to the solute. Adding a shell or two of solvent molecules and then representing the solvent beyond those shells by a polarizable continuum can only render the solvated calculations more accurate.

Unfortunately, any method that is an approximation of a real phenomenon has its share of flaws and disadvantages. Continuum models make the assumption that a solvent can be represented by charges on a surface, which is not reality. In a real solvated system, there is no boundary where the solute stops and the solvent starts. As in any model, the quality of the results will depend on the validity of the approximation made, which in this case is the replacement of solvent molecules by a polarizable continuum. Continuum models do not constitute ab initio methods per say. They require quite a bit of parameterization against experimental data and fitting of the radii used to define the cavity in order to obtain the desired accuracy. Free energies of solvation calculated with continuum models are very sensitive to the radii. The sensitivity is such that it is practically

impossible to select a universal set of radii and use them within any continuum model, level of theory or solvent. The parameterization procedure is usually quite lengthy and will only be useful for a specific model, level of theory and solvent. It is also easy to imagine situations where a single fixed radius for a given atom simply does not make sense. The oxygen atoms of acetic acid obviously have different chemical natures and should have different radii, whereas the oxygen atoms in acetate are identical and should have the same radius. Also, the oxygen atoms of acetate should possess a far different radius than those of the oxygen atoms in acetic acid since they must bear a large negative charge. Another similar situation arises for the protonation of ammonia in solution. The radius associated with nitrogen should be different in ammonia and ammonium. Fortunately, there is a way around this problem. Algorithms exist that assign atomic radii according to the chemical environment of the atoms in question. Such an algorithm (Solvent Interaction Potential) was used in our study and tested to verify if it indeed outperforms a fixed-radii scheme.

Another deficiency of continuum models is that some of the electronic density inevitably lies outside of the cavity as it tails off to infinity. The solvent cannot see or feel the effect of the density outside the cavity, which introduces errors in the calculation. To account for such a loss of electronic density, some have tried to multiply the surface charges by a given factor, without much success. A second option is to distribute additional charges on the surface according to the solute electronic density in each point of the surface to substitute for the density lost outside of the cavity [10]. Other models, namely the isodensity model, try to improve the situation by defining the cavity based on the actual electronic density around the solute. The isodensity surface from a gas phase QM calculation is used as the outer edge of the cavity. A small threshold on the density (usually on

the order of 0.0004 to 0.001 a.u.) is used as a cutoff value. Unfortunately, it was found that a single value for the cutoff cannot simultaneously give satisfying results for anions, neutrals and cations. The cutoff is also sensitive to the basis set used. Problematic and ambiguous situations arise for larger molecules with many different electron-rich and electron-poor sites or, at the extreme case, zwitterionic species.

2.4. Generalized COnductor-like Screening MOdel

The COSMO approach describes the effect of the solvent by means of polarization charges distributed on the molecular surface. These polarization charges are evaluated by imposing the condition that the total electrostatic potential on the surface vanishes. Two different kinds of interactions can be described with this boundary condition, suited for cavities in conducting media: the interaction between molecules and metals and the solvation process in polar liquids [13]. Obviously, the rest of this description of GCOSMO will relate to the solvation application. Even though conductor-like models are physically less founded than dielectric models (like PCM), they remain attractive because the boundary conditions are computationally simpler, especially for the evaluation of energy gradients. In the GCOSMO approach, the cancellation of the ESP on the surface is mathematically represented by [12]:

$$(1) \quad \sum_i \frac{Z_i}{|r - R_i|} - \int_V \frac{\rho(r')}{|r - r'|} d^3r' + \int_S \frac{\sigma(r')}{|r - r'|} d^2r' = 0$$

where r is on the surface S, Z_i and R_i are the atomic number and coordinates of nucleus i, ρ is the solute electronic density and σ is the surface charge on S. In the case of spherical or ellipsoidal cavities, the above equation can be solved analytically. However, in the case of molecular surfaces, the surface needs to be divided into small segments to find a numerical solution. Two matrices (A and B) and a vector (c) need to be defined in order to obtain the numerical expression for the surface charges. For a surface divided into M segments,

$$(2) \quad A_{uv} = \frac{1}{|t_u - t_v|}, A_{uu} = 1.07\sqrt{\frac{4\pi}{S_u}}$$

$$(3) \quad B_{ui} = \frac{1}{|t_u - R_i|}$$

$$(4) \quad c_u = -\int \frac{\rho(r)}{|r - t_u|} d^3r$$

where t_u is the position vector of the center of segment u on the surface and S_u is the area of a segment u of the surface. Matrices A and B are of dimension MxM and MxN, respectively, while vector c is of dimension M, where N is the number of nuclei of the solute and M is the number of segments on the surface. Once the matrices and the vector have been evaluated, the vector of surface charges can be calculated with a simple equation involving A, B and c:

$$(5) \quad q = -f(\varepsilon)A^{-1}(Bz + c)$$

where z is the vector of N nuclear charges and $f(\varepsilon)$ is a necessary scaling factor to ensure that the Gauss theorem for the total surface charge is respected. This scaling factor is a function of the dielectric constant of the solvent:

$$(6) \quad f(\varepsilon) = \frac{\varepsilon - 1}{\varepsilon}$$

Unlike the PCM algorithm, which uses a scaling function to account for numerical errors [14], the scaling function of the GCOSMO model is theoretically motivated. It is also worth noting that using the scaling function introduces a relative error of approximately $1/\varepsilon$, which is less than 1 kcal/mol for the hydration free energy of most solutes. The error is minimized since water is by far the most utilized solvent and has a large dielectric constant. Now that all the equations relating to the GCOSMO model have been laid out, all that remains to be done is to implement them within a molecular orbital formalism to allow solvation calculations at the ab initio level. Matrices A and B depend on fixed parameters only, namely the positions of the segments on the surface and the positions of the nuclei. For that reason, they will remain constant throughout the SCF procedure of a single point energy calculation need only be evaluated once. On the other hand, the vector c depends on the electronic density, which changes from one iteration to the next. It is necessary to evaluate the vector c at every cycle of the SCF. By expanding the vector in a basis set, equation (4) becomes:

$$(7) \quad c_u = -\left\langle \Psi \left| \frac{1}{|r - t_u|} \right| \Psi \right\rangle = \sum_{\mu\nu} P_{\mu\nu} L^u_{\mu\nu}$$

$$(8) \quad L^u_{\mu\nu} = -\left\langle \varphi_\mu \left| \frac{1}{|r - t_u|} \right| \varphi_\nu \right\rangle$$

where $P_{\mu\nu}$ is the density matrix element corresponding to atomic orbitals φ_μ and φ_ν. The above formalism is simple to implement in any HF or DFT code. It has also been implemented at the MP2

level of theory, with the exception that the expression for the vector c is more complex. Since this research focuses on the GCOSMO application within a DFT program, the MP2 related equations will not be discussed. The total energy of the system in solution is given by:

$$(9) \quad E_{tot} = \sum_{\mu\nu} P_{\mu\nu}(H_{\mu\nu} + \frac{1}{2}G_{\mu\nu}) - \frac{1}{2}f(\varepsilon)z^{+}B^{+}A^{-1}Bz + E_{nn} + E_{non-els}$$

where H is a matrix representing the energy of a single electron in a field of bare nuclei, obtained from the core Hamiltonian, G is the usual two-electron matrix [35], E_{nn} is the nuclear repulsion energy of the solute and $E_{non\text{-}els}$ is the non-electrostatic contribution to the solvation free energy, which will be discussed in more detail later. A superscript "+" sign denotes matrix transposition. The last three terms of equation (9) are all independent of the electronic density and remain constant throughout the calculation. They only need to be evaluated once. The H and G matrices of the solute-solvent system are obtained by adding the same matrices for the solute only and contributions arising from solute-solvent interactions:

$$(10) \quad H = H^{0} + H^{solv}, G = G^{0} + G^{solv}$$

$$(11) \quad H_{\mu\nu}^{solv} = -f(\varepsilon)z^{+}B^{+}A^{-1}L_{\mu\nu}$$

$$(12) \quad G_{\mu\nu}^{solv} = -f(\varepsilon)c^{+}A^{-1}L_{\mu\nu}$$

In contrast with the PCM algorithm, the effects of the solvent reaction field are directly included in the operator. This direct inclusion greatly simplifies the analytical first derivatives of the energy and renders the explicit evaluation of the surface charges unnecessary for single point energy calculations.

The non-electrostatic contribution to the solvation free energy is also called the cavitation and dispersion-repulsion energy. As mentioned earlier, it results from the unfavorable loss of entropy as the solvent rearranges and orders itself to make room for the solute. It is always unfavorable ($E_{non\text{-}els}$ is always positive) and proportional to the surface area of the solute. There is no explicit mathematical equation to be derived from theoretical chemistry to account for this contribution. However, it seems that the surface area of the SAS of a solute is correlated to the $E_{non\text{-}els}$ contribution [7]. The simplest way to estimate the non-electrostatic contribution is to do a linear regression of $E_{non\text{-}els}$ plotted versus the surface area of the SAS for a series of molecules. It is impossible to separate experimental solvation free energies into electrostatic and non-electrostatic contributions. However, electrostatic contributions are readily calculated (with a continuum model) and then removed from the experimental solvation free energy to get an estimate of $E_{non\text{-}els}$. Linear and branched alkanes are usually chosen for this fitting procedure as their electrostatic contributions are very small. The fitting procedure should be done with the same set of radii, level of theory and basis set that will be later used for solvation calculations.

Analytical first and second derivatives of the energy have been derived for GCOSMO. However, the non-electrostatic contribution has no analytical derivative and is usually not accounted for in geometry optimizations. This is not a big concern since the optimized gas phase geometry should not change to a point where $E_{non\text{-}els}$ will be significantly altered during the optimization in solution. It was mentioned earlier that the explicit charges on the surface were not necessary for energy calculations. However, they are required for the gradient evaluation. The charges are easily

calculated at the end of the SCF procedure with equation (5). The gradient in solution is calculated exactly like the gradient in the gas phase with the addition of a new contribution given by:

$$(12) \quad \nabla_{Ri}^{solv} = z^{+}(\nabla_{Ri}B^{+})q + (\nabla_{Ri}c^{+})q + \frac{1}{2f(\varepsilon)}q^{+}(\nabla_{Ri}A)q$$

Differentiating equations (2), (3) and (7) with respect to the nuclear coordinates yields the three terms of equation (12). If we assume that the radii are fixed and that each surface element remains associated with the same atom throughout, then the derivative of the non-diagonal elements of matrix A is given by:

$$(13) \quad \nabla_{Ri}(A_{uv}) = -\frac{(t_u - t_v)_i}{|t_u - t_v|^3}(\theta_{ui} - \theta_{vi})$$

where θ_{ui} is equal to one if the surface element u is associated with atom i and equal to zero otherwise. In the case of the diagonal elements of matrix A, the derivative depends on the change in area of the surface element with respect to the change in atomic positions. The only elements whose area would change as atoms move are those at the overlap regions of adjacent spheres. Those elements are well defined in the case of a van der Waals surface only. Other surfaces, like the more appropriate Solvent-Excluded Surface, have no analytical expression to express the dependance of the surface element's area with respect to the position of the nuclei. However, experiments have shown, with numerical differentiation, that these derivatives are very small and can be ignored, thus leading to:

$$(14) \quad \nabla_{Ri}(A_{uu}) = 0$$

A very similar expression is obtained by differentiating equation (3):

$$(15) \quad \nabla_{Ri}(B_{uj}) = -\frac{(t_u - R_j)_i}{|t_u - R_j|^3}(\theta_{ui} - \delta_{ij})$$

where θ_{ui} has the same definition as above and δ_{ij} is a typical Kronecker delta. Finally, only part of equation (7) needs to be differentiated since the partial derivative of the density matrix is already taken into account in the gas phase gradient equations. The partial derivative of the L matrix in equation (7) is given by:

$$(16) \quad \nabla^*_{Ri}(c_u) = -\sum_{\mu\nu} P_{\mu\nu} \left\langle \varphi_\mu \left| \frac{(r - t_u)_i}{|r - t_u|^3} \theta_{ui} \right| \varphi_\nu \right\rangle$$

By adding the contributions from equations (13), (14), (15) and (16) to the gradient, a geometry optimization within a continuum representation of the solvent can be done using typical optimization algorithms. Analytical expressions for the second derivatives have also been derived but have not been implemented within this project. This would be pointless since the software code to which we are adding it does not have the ability to handle analytical second derivatives in the gas phase. Thusly, second derivatives will not be discussed. Ignoring the non-electrostatic contribution's derivative and the element's surface area derivative is believed to be a cause of instability during the optimization procedure. However, all but a few of our test cases have been optimized successfully with the above procedure. The few molecules that have failed to converge were mostly ions. Nothing directly links the cause of this failure to the procedure itself.

3) Solvent Interaction Potential radii

3.1. Theory and equations

As mentioned in section 2.3, it often does not make sense to have a fixed radius assigned to a specific atom and to use that radius in any chemical environment. Ideally, the cavity (the atomic radii) would be able to adapt to the different types of environment in which the atoms might find themselves. A few algorithms have been developed in an effort to build such versatile cavities. The Solvent Interaction Potential (SIP) algorithm has been implemented within our GCOSMO model and will be discussed in further detail [15].

In the SIP algorithm, the atomic radii are determined by the potential existing between the solute and a solvent probe. The final radii are those that best reproduce the Solvent-Accessible Surface (SAS), which is defined as the local minima of the potential. Four different components will contribute to the SIP: the Lennard-Jones (E_{L-J}) potential and the potential generated by the interaction of the solute's partial charges with the dipole (E_{i-d}), quadrupole (E_{i-q}) and induced-dipole (E_{i-id})of the solvent probe. The partial charges of the solute are obtained from gas phase QM calculations while the probe's dipole, quadrupole and polarizability are physical properties of the chosen solvent. The global SIP is thus given by:

$$(17) \quad E_{SIP} = E_{L-J} + E_{i-d} + E_{i-q} + E_{i-id}$$

A cartesian grid is laid out around the solute and the SIP is evaluated at three different points in space for each grid point. The grid points are equally spaced by 0.1Å in all directions and stop

when it is physically impossible for an atom in the solute to have a large enough radius to reach that particular point. The largest possible radii were estimated to be 1.5Å for hydrogen and 2.5Å for heavy atoms. They were purposely overestimated to make sure the size of the grid would never prevent the proper evaluation of a radius. Each component of the SIP is individually evaluated. The LJ potential is obtained with:

$$(18) \quad E_{LJ} = \sum_{i=1}^{N} 4\varepsilon_i \left\{ \left(\frac{\sigma_i}{r} \right)^{12} - \left(\frac{\sigma_i}{r} \right)^{6} \right\}$$

where ε is defined as the well depth of the interaction between the probe and the atom. It is given the value of 0.17kJ/mol for hydrogen and 0.34kJ/mol for first-row atoms [16]. σ is defined by the following equation, where r_e is the sum of the atom's LJ radius, r_{LJ}, and the solvent probe radius, r_p:

$$(19) \quad \sigma = \frac{1}{\sqrt[6]{2}} r_e$$

For the other contributions to the SIP, the experimental gas phase dipole moment and quadrupole moment tensors of the solvent are scaled by a Langevin factor, L. This factor takes into account the fact that thermal energy will still cause fluctuations in the solvent molecules positions as they orient themselves about the solute molecule [15].

$$(20) \quad \mu' = L\mu, Q' = LQ$$

$$(21) \quad L = \frac{e^x + e^{-x}}{e^x - e^{-x}} - \frac{1}{x}$$

$$(22) \quad x = \frac{\mu F}{kT}$$

At any point around the solute, the Langevin dipole of the probe is assumed to lie in the opposite direction of the electrostatic field created by the partial charges of the atoms, forming angles, θ, with the vectors connecting that point and the atoms. If we assume that the quadrupole can freely rotate about its principal axis, the dipole, quadrupole and induced dipole interaction energies are given by:

$$(23) \quad E_{i-d} = \sum_{i=1}^{N} \frac{-\mu' q_i}{\varepsilon r_i^2} \cos\theta_i$$

$$(24) \quad E_{i-q} = \frac{1}{2} \sum_{i=1}^{N} \left[\frac{Q'_{zz} q_i}{\varepsilon r_i^3} (3\cos^2\theta_i - 1) \right]$$

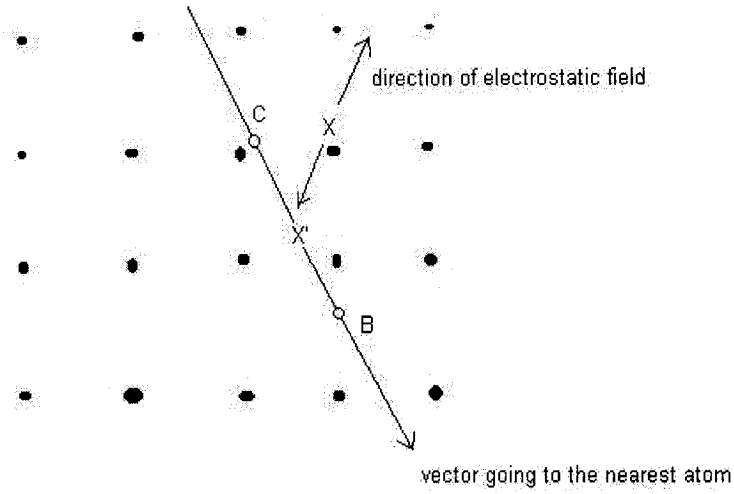
$$(25) \quad E_{i-id} = \frac{1}{2} \sum_{i=1}^{N} \left[\frac{q_i^2}{r_i^4 \varepsilon^2} (\alpha_{zz} \cos^2\theta_i + \frac{(\alpha_{xx} + \alpha_{yy})}{2} \sin^2\theta_i \right]$$

The Langevin function is not involved in the induced dipole interaction potential because that particular potential is not affected by it. Now that the SIP potential can be calculated for any point in space, all that is left to do is to decide whether that grid point is part of the SAS or not. The center of mass and the probe center do not necessarily coincide. An offset is applied in the direction opposite to the electrostatic field. The offset is also scaled by the Langevin factor:

$$(26) \quad \Gamma' = L\Gamma$$

The position X' is obtained by applying the offset at grid point X. X will be part of the SAS if the SIP at X' is a minimum along the vector direct towards the the nearest atom, as illustrated in Figure (1).

Figure 1. Evaluations of SIP for a grid point



Three distinct evaluations of the SIP are necessary to determine if grid point X is part of the SAS or not (at points B, C and X' in figure 1). If the SIP at X' is smaller than the SIP at B and C, then X is deemed to be on the SAS. The final step of the entire process is to assign a radius to each atom by trying to reproduce the SAS as well as it can be done with atom-centered interlocked spheres. First, the Solvent-Excluded Volume is evaluated by counting how many grid points lie within the SAS. An initial guess of the radii is made and the same volume is evaluated, but with the surface created by the radii. The mismatch between both volumes is evaluated by counting how many points belong to one volume but not the other. The best radii are those that will minimize this mismatch and are obtained by adjusting each radius individually until the number of mismatches is minimized.

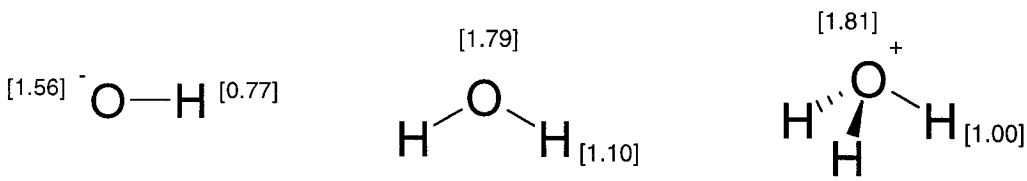
3.2. Results and discussion of SIP calculations

The SIP algorithm has been coupled to our GCOSMO model implemented within the Density Functional Theory software package DeFT. The input required for SIP calculations can be separated into two parts: the solute and the solvent. For the solute, atomic charges, atomic positions and Lennard-Jones radii are required. Charges obtained from the Molecular ElectroStatic Potential (MESP) were chosen over Mulliken charges since the latter is arbitrary and very sensitive to the level of theory and basis set used. Each atom is assigned a Lennard-Jones radius, which is used in the evaluation of the LJ potential. Our database of molecules included the following atoms: hydrogen, carbon, nitrogen, oxygen and fluorine. We manually optimized the LJ radii associated to these atoms and found values of 1.04Å, 2.29Å, 2.12Å, 1.90Å and 1.72Å respectively. Water has been the only solvent tested, although incorporating other solvents would be trivial, requiring only to change the physical constants that apply in the SIP model. For our calculations with water, we used the experimental gas phase dipole moment of 1.86 Debye and quadrupole tensor of -0.13 Buckinghams [17]. The experimental polarizabilities were also required and equal $1.528 \times 10^{-24} cm^3$, $1.415 \times 10^{-24} cm^3$ and $1.468 \times 10^{-24} cm^3$ in the x, y and z directions respectively, assuming the molecular axis lies on the z axis [17]. The temperature is set to 298.15K and the radius of the probe is given the usual value of 1.4Å for water.

Looking at the individual components of the SIP, we realized that the LJ and dipole components totally dominate the other components of the potential, especially for neutral species (more than two orders of magnitude larger). As a matter of fact, for neutral species, the dominance

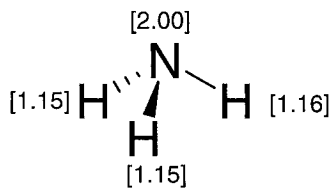
is such that ignoring the quadrupole and induced dipole components give the exact same radii obtained when included. For that reason, we dropped these two terms for neutral molecules in an attempt to make the SIP calculation more efficient. Unfortunately, we were unable to obtain satisfying results with the use of a single offset value for both charged and neutral molecules. Offsets of $0.11a_0$, $0.25a_0$ and $0.42a_0$ were assigned to cations, neutrals and anions, respectively. The offset effect will be felt more strongly by anions than cations, as the next examples will show. In the odd case of a zwitterionic species, the total charge on the molecule still decides which offset is used. In other words, the zwitterion nature of the solute is not taken into consideration.

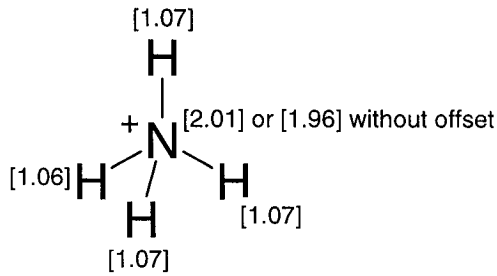
The oxygen radius in water, hydroxide and hydronium truly shows the effect of the applied offset. The oxygen radius is much more important than the hydrogen radius for the subsequent solvation calculation because the spheres around the hydrogen atoms are practically engulfed by the oxygen sphere. The radius of oxygen goes from 1.56Å in hydroxide, to 1.79Å in water, to 1.81Å in hydronium:



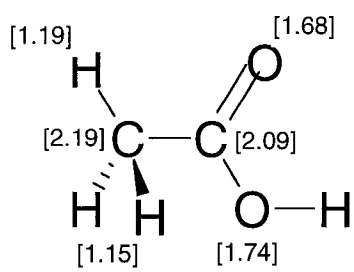
The electrostatic field around the molecule will dictate the value of the offset. Generally, the electrostatic field increases in the direction of a negative charge and decreases in the direction of a positive charge. This leads to a contraction of the SAS in regions where the electrostatic field increases in the direction of the solute and to an expansion for the opposite scenario. The oxygen

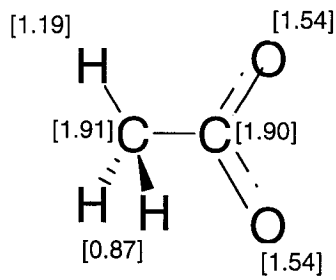
atom in hydroxide bears a large negative charge, causing the electrostatic field to increase in its direction. This considerably reduces its radius. For water, the charge is less negative and the Langevin function greatly reduces the offset effect for neutrals. This leads to a large increase in the radius. Even though it seems the effect of the offset is not very important for hydronium, it is important to realize that without the offset, its oxygen radius would be 1.75Å. That means that the electrostatic field around the hydronium cation points away from the oxygen atom, slightly increasing its radius. Ammonia and ammonium constitute another pair of molecules where the same situation arises:



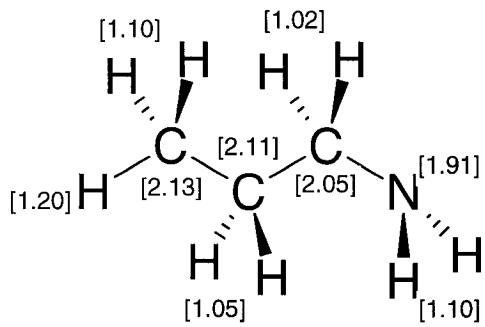


The following examples will demonstrate how atoms of the same element but found in different chemical environments within a molecule need different radii. These will expose the deficiencies of fixed-radii schemes. For instance, the oxygen atoms of acetic acid should have different radii whereas the same atoms in acetate should have the same radius. Simple Lewis dot structures are sufficient to make these predictions.





The predictions hold true with the SIP method. This example also demonstrates an important fact about carbon atoms in different hybridization states. In fixed-radii schemes, sp or sp^2 hybridized carbons are often given a radius that is smaller than that of sp^3 hybridized carbons. It is also the case for the SIP method, as the carbon of the methyl group has a larger radius than that of the carbonyl carbon. In all the test cases, carbon atoms that are part of a carbonyl group, and most carbon atoms in aromatic rings, have a smaller radius than sp^3 carbons. It is important to note that this arises naturally as part of the SIP model. Unlike previous schemes, the user need not assign hybridation to any atom. The radius of a carbon atom in a chain containing electron donating groups like alcohols, amines or halogens increases as you move away from such groups. For example, as you move away from the nitrogen atom in propylamine, the radii of the carbon atoms increase toward the value of the carbon radius in methane, which is equal to 2.15Å:



The same trend holds for the hydrogen atoms attached to these carbon atoms. It may look odd that the hydrogen atom capping the chain (in the same plane as the chain) on the terminal carbon atom is always larger than the other two hydrogen atoms attached to the same carbon atom. However, note that even though the three hydrogen atoms are equivalent over a timeframe because of the rotable C-C bonds, they are not equivalent in frozen pictures of molecules like the ones used

for static calculations. Because it is further away, the terminal hydrogen atom is not engulfed as much by the spheres of the heavy atoms as compared to the other two hydrogen atoms.

Buried atoms have smaller radii. However, the effect of the radius of a buried atom is lessened because it does not contribute much to the overall size of the cavity. For example, the nitrogen radius in ammonia, methylamine, dimethylamine and trimethylamine is 2.00Å, 1.91Å, 1.88Å and 1.80Å respectively. The same trend would be observed for the central carbon atom in the methane, ethane, propane, isobutane and neopentane series. In the case of neopentane, or any case of a heavy atom bonded to four or more heavy atoms for that matter, determining the radius of a buried atom could be ignored in SIP calculations as its radius may not affect the cavity size at all.

The small differences observed for the radii of equivalent atoms (like the hydrogen atoms in ammonia and ammonium in the previous examples) are caused by the rather coarse grid used. A cartesian grid may not be the best choice but it was used nonetheless because of its simplicity. Grid points separated by 0.1Å are not ideal either, but a finer grid would greatly affect the computational efficiency of the SIP method, as three SIP evaluations are required for each grid point. Our SIP method uses the charges derived from a gas phase calculation. Getting the SIP radii takes less than half the time of the corresponding gas phase geometry optimization. In our opinion, a fast algorithm was necessary for it to be used on the fly rather than having a separate calculation just to get the radii.

4) GCOSMO Applications with DeFT

4.1. Fixed vs. SIP radii

A database of 35 molecules (22 neutral non-aromatics, 7 neutral aromatics, 6 ions) of various chemical nature was used to test our GCOSMO algorithm, implemented within our own DFT software package, DeFT. The gas phase calculation is first performed, followed by assigning the radii or the calculation of the SIP radii. The final converged density matrix serves as an initial guess for the subsequent solvated calculation. The final solvation free energy is given by the difference in energy of the gas phase calculation and the solvated calculation, including the non-electrostatic contribution. The parameterization procedure was carried out using the VWN (Vosko/Wilk/Nusair) functional with a 6-31G* basis set. Diffuse functions were not added to anions because our model, at this point, cannot correct for the escaped electronic density. The presence of diffuse functions drives a larger part of the electronic density outside of the cavity. Without a correction procedure for that extensive loss of density, the solvated calculation is heavily penalized with respect to the gas phase calculation and results in a solvation free energy that is not nearly as negative as it should be. The non-electrostatic contribution to the solvation free energy is obtained from our own fitting procedure with a series of alkanes, as explained in section 2.4. Our non-electrostatic contribution is given by the following linear regression:

$$E_{non-els} = 1.09 + 0.005 * A_{SAS} \quad (27)$$

where A_{SAS} is the surface area (in $Å^2$) of the solvent-accessible surface given by GEPOL. Larger molecules will have a larger $E_{non-els}$ contribution. The solvent-excluded surface obtained from

GEPOL is always used for the actual solvated calculation. It is more realistic than the VDW surface and does not take any more time to generate. The only disadvantage of the SES is a possible source of instability during geometry optimizations.

For fixed radii, the set of radii of Stefanovich and Truong was used [5]. According to the authors, that particular set of radii can be used with PCM or GCOSMO and multiple levels of theory, including DFT, HF and MP2. Hydrogen, oxygen and fluorine are given a radius of 1.172Å, 1.576Å and 1.28Å, respectively. Carbon and nitrogen possess more than one radius, depending on the environment around the atom. sp^3 hybridized carbon atoms are given a radius of 2.096Å while all other carbon atoms have a radius of 1.635Å. If a nitrogen atom is bound to at least one hydrogen atom, its radius is 1.738Å, otherwise it is 2.126Å. Table (1) summarizes the results obtained with fixed radii.

Table 1. GCOSMO results with fixed radii

Solute	Experimental free energy of solvation (kcal/mol) [a]	Calculated free energy of solvation (kcal/mol)	Difference (kcal/mol)
Neutral non-aromatic			
hydrogen fluoride	-7.5	-7.05	0.45
ethanol	-5.0	-4.92	0.08
dimethylamine	-4.3	-2.55	1.75
butanone	-3.6	-4.41*	-0.81
acetone	-3.9	-4.84	-0.94
water	-6.3	-8.87	-2.57
acetaldehyde	-3.5	-4.24	-0.74
acetic acid	-6.7	-7.67	-0.97
acetamide	-9.7	-11.17	-1.47
ammonia	-4.3	-6.34	-2.04
N-methylacetamide	-10.0	-8.87	1.13
methane	2.2	0.81	-1.39
trimethylamine	-3.2	-0.35	2.85

propylamine	-4.4	-4.13	0.27
2-methoxypropane	-2.0	-1.96*	0.04
2-methoxyethanol	-6.8	-6.61	0.19
ethyl acetate	-3.1	-4.84	-1.74
ethane	1.8	0.93	-0.87
methyl formate	-2.8	-4.91	-2.11
piperazine	-7.4	-6.15	1.25
fluoromethane	-0.2	-1.24	-1.04
propenol	-5.0	-6.02	-1.02
RMS ERROR			1.39
Neutral aromatic			
benzene	-0.9	-2.18	-1.28
toluene	-0.9	-2.13	-1.23
4-methylpyridine	-4.9	-3.20	-1.70
aniline	-4.9	-7.34	-2.44
acetophenone	-4.6	-6.10	-1.50
benzaldehyde	-4.0	-5.49	-1.49
phenol	-6.6	-7.60	-1.0
RMS ERROR			1.58
Ions			
ammonium	-80.6	-86.6	-6.0
N-butylammonium	-69.0	-71.2*	-2.2
hydronium	-105.4	-98.6	6.8
hydroxide	-109.5	-102.4	7.1
fluoride	-109.5	-107.4	2.1
propionate	-79.0	-72.3*	6.7
RMS ERROR			5.6

[a] From reference [18], [19] and [20]

Note that the few asterisks in Table 1 mean that the optimization of the solvated geometry failed. In such cases, the gas phase optimized geometry was used for a solvated single point energy calculation. The procedure for our GCOSMO calculations with SIP radii has been discussed earlier in section 3.2. The exact same set of molecules was retested with the same basis set and functional, now using the SIP radii. Table 2 summarizes the SIP results.

Table 2. GCOSMO results with SIP radii

Solute	Experimental free energy of solvation (kcal/mol) [a]	Calculated free energy of solvation (kcal/mol)	Difference (kcal/mol)
Neutral non-aromatic			
hydrogen fluoride	-7.5	-6.39	1.11
ethanol	-5.0	-4.88	0.12
dimethylamine	-4.3	-2.47	1.83
butanone	-3.6	-4.07*	-0.47
acetone	-3.9	-3.99*	-0.09
water	-6.3	-7.51	-1.21
acetaldehyde	-3.5	-2.62	0.88
acetic acid	-6.7	-7.33	-0.63
acetamide	-9.7	-9.95	-0.25
ammonia	-4.3	-4.10	0.20
N-methylacetamide	-10.0	-8.35	1.65
methane	2.2	1.18	-1.02
trimethylamine	-3.2	-2.17	1.03
propylamine	-4.4	-4.16	0.24
2-methoxypropane	-2.0	-2.88	-0.88
2-methoxyethanol	-6.8	-7.23	-0.43
ethyl acetate	-3.1	-4.41	-1.31
ethane	1.8	1.15	-0.65
methyl formate	-2.8	-3.01	-0.21
piperazine	-7.4	-7.87	-0.47
fluoromethane	-0.2	-0.37	-0.17
propenol	-5.0	-6.19	-1.19
RMS ERROR			0.88
Neutral aromatic			
benzene	-0.9	-1.99	-1.09
toluene	-0.9	-2.61	-1.71
4-methylpyridine	-4.9	-4.04	0.86
aniline	-4.9	-7.70	-2.80
acetophenone	-4.6	-5.61	-1.01
benzaldehyde	-4.0	-4.48	-0.48
phenol	-6.6	-8.29	-1.69
RMS ERROR			1.55
Ions			
ammonium	-80.6	-83.0	-2.4
N-butylammonium	-69.0	-72.9*	-3.9
hydronium	-105.4	-101.0	4.4
hydroxide	-109.5	-105.2	4.3
fluoride	-109.5	-112.3	-2.8
propionate	-79.0	-75.4*	3.6
RMS ERROR			3.6

[a] From reference [18], [19] and [20]

Comparing Tables 1 and 2, we come to the conclusion that SIP radii do indeed outperform the fixed radii approach. The RMS (root mean square) errors are significantly smaller in the case of SIP radii for neutral non-aromatics and ions. In the case of aromatics, the RMS error is identical for both methods. Unfortunately, this improvement does not come without a cost. GCOSMO calculations with SIP radii take longer because of the evaluation of the radii. However, this extra cost is not exorbitant. As previously mentioned, the determination of the radii takes less time than the optimization of the gas phase geometry.

For the SIP calculations, RMS errors of 0.88 kcal/mol, 1.55 kcal/mol and 3.6 kcal/mol for neutral non-aromatics, neutral aromatics and ions respectively are quite satisfying. Some types of molecules prove to be more of a problem than others. Substituted amines and amides are known to be problematic in such calculations. The exact cause of this problem is still unclear at this point. In our calculations, ammonia and acetamide almost agree perfectly with experiment, while dimethylamine, trimethylamine and N-methylacetamide show large positive deviations with experiment. It is worth noting that several groups have carried out explicit solvent simulations with methylated amines and amides and still encountered the same problems [7]. This leads to the assumption that the problem does not lie within the continuum model approximation. Many research groups are studying this phenomenon to remedy the situation.

Toluene, aniline and phenol all have a large negative deviation from experiment. This is odd since the other aromatics compounds all agree within 1 kcal/mol. Because the other aromatic

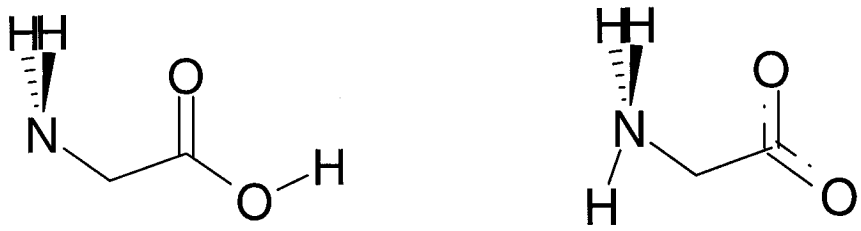
compounds agree, the problem must not be with the ring itself, but rather with the substituents. It is also surprising that the fixed-radii approach gives a better solvation free energy for those three specific molecules, which leads us to believe that the deficiency comes from the SIP calculation and not the GCOSMO model. It is also interesting to note that electron-withdrawing substituents (benzaldehyde and acetophenone) work well while electron-donating substituents (aniline, phenol, toluene) do not. It seems that the SIP model cannot properly account for electron donation, systematically underestimating the radii of the substituent and producing solvation free energies that are too negative. Finally, testing only six ions is not nearly enough to get a good assessment of the algorithm. However, it is encouraging to see the SIP approach working so well, reducing the RMS error by 2 kcal/mol when compared with the fixed-radii scheme. The results of the ions could not be further improved without hurting the results of the neutrals. More complicated charged systems, including zwitterions, have been tested and are the subject of the next section.

The errors associated with the experimental free energies of solvation were not included in tables 1 and 2. In the case of ionic species, the error is between 1 and 2 kcal/mol, and even smaller for non-ionic species. The simple fact that many different sources obtain the same experimental free energy of solvation (to a tenth of a kcal/mol) for these molecules show that they are trust worthy. The errors are small enough to use the center value with confidence.

4.2. Solvation of glycine and tautomerization of 2-hydroxypyridine/2-pyridone

Glycine exists in two forms, neutral and zwitterionic. In the gas phase, the preferred tautomer is neutral, with the carboxylic acid and amine groups intact. However, in water, the

zwitterionic tautomer is favored. A proton is transferred from the carboxylic acid group to the nitrogen atom in the zwitterionic form.



Experimentally, the solvation enthalpy of glycine, going from the neutral form in the gas phase to the zwitterionic form in water is -19.2±1 kcal/mol [21]. We tried to reproduce this solvation process with our GCOSMO model. We used the VWN functional with a 6-31G* basis set and the SIP radii option. Table 3 summarizes our calculations and compares our results with experimental values as well as B3LYP calculations from the literature.

Table 3. Glycine tautomerization

	Present Work (kcal/mol)	B3LYP / 6-31G** (kcal/mol) [a]	Experiment (kcal/mol)
ΔG_{solv} (NT)	-11.06	-11.59	---
ΔG_{solv} (ZT)	-38.18	-42.47	---
ΔG ($NT_{solv} \rightarrow ZT_{solv}$)	-5.95	-3.09	-7.67 [b]
ΔH ($NT_{gas} \rightarrow ZT_{solv}$)	-20.01	-17.67	-19.2 ± 1 [c]

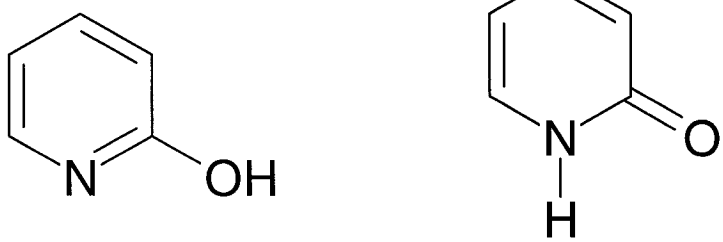
[a] From reference [12]

[b] From reference [22]

[c] From reference [23]

ZT stands for zwitterion while NT stands for neutral. Our calculations, which only took around 90 minutes, give a solvation enthalpy for glycine that falls within the experimental error, which is quite remarkable. From Table 3, we can conclude that it is more advantageous to use the SIP approach with a cheaper functional than fixed radii with a more accurate and expensive functional. Our VWN/SIP scheme does indeed outperform the B3LYP/fixed-radii scheme used in the literature. Note that the entropy contribution is taken as 3 kcal/mol at 298K [23] to calculate the enthalpy of solvation.

The second example is the tautomerization equilibrium of 2-hydroxypyridine and 2-pyridone. This is very similar to the previous example. The alcohol proton in 2-hydroxypyridine migrates to the nitrogen atom in the aromatic ring to form 2-pyridone.



Experimentally, the free energy of solvation of 2-hydroxypyridine is 4.3 kcal/mol above that of 2-pyridone in water [24]. We carried out a GCOSMO calculation on both molecules to see if we could reproduce that difference in energy. Once again, we used the VWN functional with a 6-31G* basis set and the SIP radii approach.

Our calculations yielded a solvation free energy of -6.60 kcal/mol for 2-hydroxypyridine and -10.20 kcal/mol for 2-pyridone, a difference of 3.60 kcal/mol. Based on the results presented in section 4.1, the solvation free energy of 2-hydroxypyridine is most likely too negative because of the electron donating alcohol group on the aromatic ring. If this problem were corrected, our SIP would be even in greater agreement with experiment. Nonetheless, two very simple calculations that took approximately an hour each (with 4 900 MHz UltraSPARC III Cu processors in parallel) got within 0.7 kcal/mol of the experimental results. This proves the accuracy and efficiency of continuum models.

4.3. Versatility and limitations of the method

Our GCOSMO implementation within our own developed density functional theory software package, DeFT, is very versatile but also has limitations. As with any continuum model, GCOSMO can be used with any basis set or functional that is available within the software. The analytical first derivatives are available in solution but the second derivatives are not yet implemented. Our first derivatives do not include the non-electrostatic contribution and the derivative of each surface element's area with respect to the nuclear coordinates. The availability of the first derivatives makes it possible to perform geometry optimizations or molecular dynamics simulations in solution. Unfortunately, frequency analyses in solution can only be done via numerical differentiation of analytical first derivatives.

The user has the option of choosing which set of radii he wants, fixed or SIP. The set of fixed radii is the same set that was used for the parameterization presented in section 4.1. If requested, the SIP radii are generated instead, as explained in section 3.1. At this moment, a very limited number of atoms have been included for GCOSMO calculations: hydrogen, carbon, nitrogen, oxygen and fluorine. The vast majority of organic chemistry deals exclusively with the previous atoms. However, it would be useful to add silicon, phosphorus, sulfur and the remaining halogens, so as to be able to treat any amino acid, nucleic acid or halogenated compound. A potential user who needs other atoms in a solvated calculation could add them himself, just like parameters in semi-empirical or molecular mechanics codes. Adding the extra atoms would involve a parameterization process with various molecules containing such atoms, for both the fixed and SIP approaches.

The user also has a choice between the VDW surface and the SES. Although the SES is normally used for continuum models, nothing prevents the use of the VDW surface. The SES successfully eliminates crevasses on the VDW surface that cannot actually be reached by the solvent, by adding a multitude of small spheres to the ones centered on the nuclei. However, the extra spheres are believed to be a cause of instability when the time comes to optimize geometries. Using the VDW surface instead would be somewhat less accurate because of the lower quality surface but more stable for geometry optimizations.

Even if applications of continuum models with solvents other than water are not common, five different solvents can be used with our GCOSMO model: water, acetone, hexane, methanol and DMSO (dimethylsulfoxide). Caution is a must when using the other solvents for two reasons. First,

as it was mentioned in the description of the GCOSMO algorithm, an error proportional to ε^{-1} arises from the requisite scaling function. While this error is not a concern with water because of its high dielectric constant (ε=78.4), it certainly becomes a factor to be conscious of if hexane (ε=2.02) is used. Second, all the radii, for both the fixed and SIP approaches, have been parameterized against experimental data in water. Using the same set of radii in a different solvent could potentially lead to substantial errors. Unfortunately, the only way to remedy that situation would be to reparameterize all the radii for every solvent. The lack of reliable experimental solvation free energies in other solvents will surely make such parameterization very difficult.

5) Free energy calculations

5.1. Importance of free energy calculations

A typical calculation in quantum chemistry will give the electronic energy at absolute zero for one particular conformation of a system. In many cases, studying particular systems or reactions by calculating variations of electronic energies only will be sufficient to provide useful insights. At absolute zero, translational energy, rotational energy and entropy can be ignored. Only the zero-point vibrational energy remains. Adding the zero point correction for the vibrational energy to the electronic energy,

$$(28) \quad ZPE = \sum_{i=1}^{nvibrations} \frac{h\nu_i}{2}$$

where ν is the vibrational frequency, gives the internal energy at 0 K. If ones wishes to use a 0 K calculation to study a phenomenon happening at room temperature, he can hope that the effects of translations, rotations and entropy cancel out, or are small enough to be neglected. However, in some cases, the electronic energy and free energy profiles of a reaction will be different, making it necessary to take the calculation one step further, to obtain the free energy profile. A reaction could also very easily proceed via different pathways at room temperature, different from its 0 K energy profile. As the temperature rises, it also becomes possible for the molecules to change their conformation and jump out of the absolute minimum on their potential energy surface. The problem with standard calculations is that they completely neglect thermal effects. In many chemical processes of utmost importance, entropy plays a key role and cannot be tossed aside and ignored.

For example, consider the folding of a protein. When a protein folds, it will do so in a way to create an interior core of non-polar hydrophobic amino acids, whereas the outer surface will mainly be comprised of polar/charged amino acids [1]. At first glance, the reason why a protein folds like that may seem related to enthalpy, but it turns out that entropic effects explain the process. From an enthalpic point of view, the packing of the non-polar amino acids is unfavorable for the protein because weak dipole/induced dipole interactions with water molecules are replaced by even weaker dispersion interactions amongst non-polar amino acids. From an entropic point of view, packing amino acids together is also unfavorable because the structure becomes more ordered. However, one has to remember that water molecules interacting with the protein are not entirely satisfied because their movements are restricted. Upon packing of the non-polar amino acids, a large number of water molecules are released, resulting in a very favorable gain in entropy. The release of the water molecules dominates all other thermodynamic events and explains why proteins fold and form hydrophobic cores. Even though a protein folding computer simulation still has not been done from beginning to end, this example brings forward an important conclusion. In many cases, neglecting the solvent or the entropic effects will lead to completely different and incorrect results.

Another simpler example would be a reaction that is deemed to be not spontaneous from standard 0 K calculations. Since entropy is neglected, this assumption is made because the variation of enthalpy at 0 K is positive. With that calculation alone, it is impossible to predict if the reaction would become spontaneous at some finite temperature, and what indeed that temperature would be. If the partition functions are evaluated, then the entropy change can be calculated in addition to the enthalpy change. A positive entropy change would mean that, at some finite temperature, the

reaction does indeed become spontaneous. That critical temperature can easily be estimated using Gibbs' Law:

$$\Delta G = \Delta H - T\Delta S \tag{29}$$

The main reason why a computational chemist has to be concerned with free energy calculations is simple: we do not live in a world at 0 K without entropy. To reproduce real-life experiments, the ones that are performed in laboratories every day, or make real-life predictions, entropy must be taken into account. This world is governed by the rules of thermodynamics. Every field of experimental chemistry deals with thermodynamics in some way. From statistical mechanics, it is known that any imaginable thermodynamic property of a given system can be obtained if its partition function, Q, is calculated, including entropy, enthalpy, free energy, chemical potential or heat capacity, to name a few. To give a few examples, the following equations show how to calculate the internal energy, entropy, Helmholtz free energy and Gibbs Free energy from the partition function:

$$U - U_0 = kT^2\left(\frac{\partial \ln Q}{\partial T}\right)_V \tag{30}$$

$$S = \frac{U - U_0}{T} + k \ln Q \tag{31}$$

$$A - U_0 = -kT \ln Q \tag{32}$$

$$G - U_0 = -kT \ln Q + NkT \tag{33}$$

Accounting for entropy and other thermal effects in theoretical calculations will allow computational chemists to better predict the behavior of reactions in real situations.

5.2. Traditional ways of calculating free energies and their limitations

The absolute free energy of a system can be estimated with a standard static ab initio calculation, if a frequency calculation is also carried out. The vibrational partition function can be estimated from the frequency calculation, assuming the harmonic normal mode approximation is valid. In theory, if the partition function is known, then any thermodynamic property of the system can be calculated. However, frequency calculations are quite expensive for large systems and the harmonic approximation breaks down when weak interactions (like hydrogen bonds) are involved [25]. This approach might be sufficient for simple reactions involving rather small molecules, but its usage is very limited. The harmonic normal mode approximation is valid for stationary points on the potential energy surface (minimums and saddle points) only. Free energy profiles of reactions cannot be created by this method.

Another important problem arises when larger systems are studied. For such systems, it is often wrong to think of them as sitting in a single potential energy well. Most of the time, larger molecules are very floppy (mainly because of the many degrees of freedom and rotable groups) and their PES has numerous minima that are close to each other energetically. If that is the case, then other low-lying energy minima must be taken into account in addition to the absolute lowest minimum. Obviously, it is not possible to study many conformations of a given molecule with a single energy calculation. Rather than taking a single snapshot of the molecule, the average of many

different configurations of that same molecule is required. Two very different and widely used methods exist to generate the necessary number of geometries: molecular dynamics (MD) and Monte Carlo (MC). The average of any property can be calculated with these two algorithms. MD produces time averages while MC produces ensemble averages.

In MD, the nuclei of the system are given initial positions and velocities. Newton's equations of motion are then integrated, using small timesteps, for a predetermined number of steps to obtain a simulation over a given time span (length of timesteps x # of steps). The method is said to be deterministic because once the initial conditions are set, the trajectory each nuclei will take is set in stone [1]. A typical MD simulation would be on the order of one million steps of one femtosecond for a total time of one nanosecond. To compare any calculated property with its experimental value, the particular property is evaluated at each step and the time average is generated:

$$(34) \qquad A_{average} = \frac{1}{M}\sum_{i=1}^{M} A[p^N(t), r^N(t)]$$

where M is the number of steps taken and A is a function of the nuclei's momenta (p) and positions (r). In MC, the nuclei of the system are also given initial coordinates but the velocities are not considered. The method is not deterministic and relies totally on chance, hence the name Monte Carlo [1]. The configurations are generated by slightly perturbing the positions of the nuclei in a random direction. This updating scheme is referred to as Metropolis Monte Carlo and it is understood that it is the updating procedure used. Some MC algorithms will perturb all the atoms at each step, while others will only perturb one random atom or one atom at a time sequentially. The generated configurations are accepted or discarded based on a random criterion. If a configuration

has a potential energy that is inferior to the previously accepted geometry, then it is automatically accepted and becomes the new "standard". In the case where its potential energy is superior to the previously accepted geometry, the acceptance criterion is evaluated:

$$(35) \quad x = \exp\left\{\frac{-\left[V_{new}(r^N) - V_{old}(r^N)\right]}{k_b T}\right\}$$

which gives a number between zero and one since the numerator is always negative. The acceptance criterion of equation (35) is then compared with a random number, also between zero and one. If the acceptance criterion is larger than the random number then the configuration is accepted and becomes the new "standard", otherwise it is discarded. Note that smaller variations upward in potential energy will produce acceptance criteria that are closer to one, increasing the probability of an accepted configuration. Just like in MD, any particular property can be evaluated for each accepted configuration and the ensemble average is then calculated:

$$(36) \quad \langle A \rangle = \frac{1}{M}\sum_{i=1}^{M} A(r^N)$$

MD and MC each have their own advantages and disadvantages. Obviously, any time-dependent phenomena must be studied with MD. For example, following the progress of a reaction in order to create a movie to visualize the movement of each atom cannot be done with MC because the configurations jump all over the place, whereas MD configurations follow a path dictated by Newton's equations of motion. In typical MD simulations, the sum of the kinetic and potential energy remains constant throughout while the temperature fluctuates, which is not ideal for temperature dependent processes. In the case of MC, the temperature is a fixed parameter and

remains constant while the potential energy varies (kinetic energy is not considered in MC). Both methods are used. Deciding which one to use is often dictated by need rather than preference.

Just as in experimental chemistry, absolute free energies are much more complicated to calculate than variations of free energy. It is important to distinguish between mechanical and thermal properties. Mechanical properties include internal energy, pressure and heat capacity while examples of thermal properties are entropy and free energy or chemical potential. By looking at equations (30) to (33), a notable difference between the expression for the mechanical property, internal energy, and the other thermal properties can be observed. Calculating the absolute mechanical properties of a system is easier because only the derivative of the partition function with respect to temperature, rather than the partition function itself, is required. Evaluating this derivative primarily requires proper sampling of low energy configurations, for which MD and MC have been developed. On the other hand, calculating absolute thermal properties is very difficult because the partition function itself must be evaluated. To get an accurate estimate of the partition function, the entire PES (low and high energy regions) must be properly sampled. Unfortunately, perfect sampling can only be achieved with an infinitely long MD or MC simulation. For that reason, the direct evaluation of absolute thermal properties via theoretical simulations is not really feasible.

On the bright side, one is rarely interested in the absolute entropy or free energy of a system. It is much more important to calculate variations of such properties between various molecules or isomers. Some algorithms have been developed to circumvent the problems mentioned above and make it easier to calculate variations of free energy. Free energy perturbation is routinely used with

empirical force fields to evaluate the free energy change when slight modifications are made to a molecule [1]. Reactions that cannot be done in real life can be done on a computer via non-physical transformations. For example, one can calculate the variation in the binding free energy of a drug caused by replacing a chlorine atom by fluorine or replacing a hydrogen atom by a methyl group. The whole process is usually separated in small windows where the difference in free energy between the initial and final states is on the order of 1-2kcal/mol. A simple addition of all the variations between the windows gives the total free energy change for the entire process:

$$(37) \quad \Delta A(X \rightarrow Y) = \Delta A(X \rightarrow I_1) + \Delta A(I_1 \rightarrow I_2) + ... + \Delta A(I_N \rightarrow Y)$$

where I represents the fictitious intermediate states between real states X and Y.

For the case of simple chemical reactions, the potential of mean force (PMF), derived from the thermodynamic integration technique, is another method that is often used [26]. Creating a free energy profile for a reaction is often impossible with standard MC or MD simulations because the system will take much too long a time to overcome the activation energy barrier. Our computing resources are limited. Waiting for the reactants to increase their energy and reach the transition state (note that MC and MD can go uphill in energy, but preferentially move in downhill directions) is not exactly efficient computing. Even if the system did manage to reach the transition state on its own, it would not stay near the top of the free energy curve long enough to properly sample that part of the PES [27]. The PMF method solves this problem by choosing a reaction coordinate and forcefully dragging the system along that coordinate. In simple cases, a bond length (e.g.; a dissociation reactions), a bond angle (e.g.; isomerization of HCN to HNC), a torsion (e.g.; staggered ethane to

eclipsed ethane) or a simple combination of a few variables can be used as the reaction coordinate. For more complicated reactions, it is useful to prepare an intrinsic reaction coordinate (IRC) path (easily done with programs such as Gaussian) and then follow that IRC in the simulation. Care must be taken when choosing the reaction coordinate since a bad coordinate can result in an unfavorable path that does not go through the transition state, resulting in an overestimation of the activation energy. Traditional MC or MD simulations are used to sample the PES perpendicular to the chosen reaction coordinate. Integrating the force required to drag the system along that reaction coordinate will make it possible to calculate free energy variations:

$$(38) \quad \Delta A_{(0-1)} = \int_0^1 \frac{\partial A(\lambda)}{\partial \lambda} d\lambda = \int_0^1 \left\langle \frac{\partial E(X,\lambda)}{\partial \lambda} \right\rangle_\lambda d\lambda$$

It is understood that the parameter λ smoothly changes from zero to one upon going from the initial to the final state. In practice, the entire process is separated into windows where the value of λ is fixed. A normal constrained MC or MD simulation is done in each window to accumulate data and the results of all windows are summed to get the total change of free energy for the process:

$$(39) \quad \Delta A = \sum_{i=1}^{nwindows} \langle F_i \rangle_\lambda \Delta\lambda_i$$

Alternatively, a method called the slow-growth approach exists where the constraint value is changed in a continuous manner from the initial to the final state [28]. Slow-growth simulations have the advantage that the thermal equilibration only needs to be done at the initial state as opposed to the beginning of each window in the traditional PMF method. Thermal equilibration is necessary to let the system adjust to the constraint and make sure the results are not biased by the initial

configuration. If the constraint does not change rapidly in a slow-growth simulation, then the system can remain, for all intents and purposes,

equilibrated throughout.

5.3. Monte Carlo in depth

MC relies totally on chance. MC algorithms have been applied to a wide variety of problems encountered in various fields in the past. To demonstrate the usefulness and simplicity of MC, consider the estimation of π with MC [1]. To do so, a unit circle is put inside a square of dimension 2 by 2. All the randomly generated points fall within the square but only a fraction falls in the circle. The ratio of points that have fallen in the circle over the total number of points will equal the area of the circle over the area of the square:

$$(40) \quad \text{fraction} = \frac{\text{circle}}{\text{square}} = \frac{\pi r^2}{4r^2} = \frac{\pi}{4}$$

The value of π is estimated to be four times the fraction of the points that fall within the circle. The accuracy of the estimation will depend on the total number of points used and the quality of the random number generator used. The same idea can be applied to chemistry, by defining the probability of finding a given configuration as (low energy configurations have larger probabilities):

$$(41) \quad \rho(r^N) = \frac{\exp\left\{\frac{-V(r^N)}{k_b T}\right\}}{\int dr^N \exp\left\{\frac{-V(r^N)}{k_b T}\right\}}$$

Equation (41) divides the Boltzmann factor of a given configuration by all the possible Boltzmann factors for the system. The average potential energy could theoretically be obtained by:

$$(42) \quad \left\langle V(r^N) \right\rangle = \frac{\int dr^N V(r^N) \exp\left\{ \frac{-V(r^N)}{k_b T} \right\}}{\int dr^N \exp\left\{ \frac{-V(r^N)}{k_b T} \right\}}$$

Equation (42) takes all the possible potential energies for the system and multiplies each of them by their probability defined with equation (41) to get the average potential energy, in an effort to account for all the populated configurations. Unfortunately, the integrals in equation (42) cannot be solved analytically or numerically in an efficient manner. MC can be used to generate configurations and create a database to estimate the average potential energy. In the simplest case, the configurations could be generated at random. However, the vast majority of the configurations generated would not make chemical sense. They would have a rather high energy and thus make a small contribution to the integrals because of low probability factors. The way to get around that is to use Metropolis Monte Carlo, which was discussed in the previous section. Metropolis MC preferentially samples the low energy parts of the PES, that is the configurations that have a high probability factor. The trial configurations are generated by slightly perturbing the last accepted configuration. This is easily done in cartesian coordinates by applying a random displacement in the x, y and z directions on each nucleus:

$$(43) \quad \text{coord}_{new} = \text{coord}_{old} + (2\chi - 1)\text{disp}_{max}$$

where χ is a random number between zero and one and $disp_{max}$ is the maximum displacement allowed. Equation (43) allows the nuclei to move anywhere in the [-$disp_{max}$,$disp_{max}$] range in all three directions. Usually, the maximum displacement allowed is given a value that will ultimately lead to an acceptance rate of approximately 50%. A large number of configurations generated with Metropolis MC provide an estimate of the average potential energy:

$$\left\langle V(r^N)\right\rangle = \frac{\sum V(r^N)\exp\left\{\frac{-V(r^N)}{k_bT}\right\}}{\sum \exp\left\{\frac{-V(r^N)}{k_bT}\right\}} \quad (44)$$

Many different applications of theoretical chemistry make use of the Monte Carlo algorithm in some way or another. For example, MC can be very useful to optimize the geometry of a large floppy molecule in cases where conventional optimization algorithms (like steepest descent, conjugate gradient or quasi-Newton methods) fail. In a method similar to simulated annealing, an initial configuration is submitted to a MC simulation. The temperature is then gradually decreased until an energy minimum is found. The process can be repeated a number of times with different starting configurations to verify the validity of the newly found minimum. Some like to use a gradient-aided MC algorithm. In addition to the energy, the gradient is also evaluated for each proposed configuration. The subsequent moves are biased to go against the gradient (towards low energy configurations). In the case of large systems, perturbing the position of each atom may not be particularly efficient. Instead, perturbing the position of groups of atoms can be more intuitive and efficient. For example, in a protein, it may be a good idea to keep some of the side chains rigid and randomly alter the positions of the rigid chains rather than the atoms individually. The same

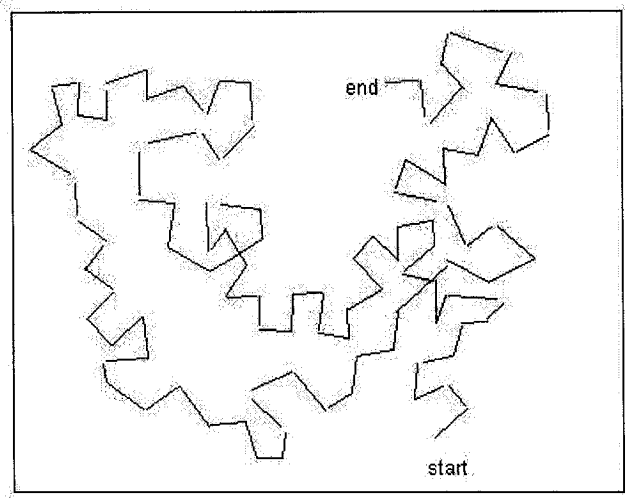
idea can be pushed further, by perturbing specific torsion angles only, in an effort to obtain a better tertiary structure.

MC is easily implemented within any software package and level of theory. It requires very little and basic programming when compared with a MD algorithm. To put it in perspective, all that is needed for MC is a good random number generator and a loop to repeat the energy calculation over and over. MC is very useful for simulations that require a precise temperature. It is also superior to MD for the sampling of a PES containing many high energy barriers. MC can take non-physical pathways to overcome these barriers whereas MD will have to gather enough kinetic energy to do the same, which can take a very long time [1]. The main reason why MC is often better to sample space is because its configurations are not correlated as much. Correlation is used to define the similarity and dependence each configuration has with the previous one. In the case of deterministic algorithms like MD, correlation is at its absolute maximum. Each new configuration is the direct product of the previous configuration. For Metropolis MC, there is still some correlation because each new configuration is obtained by "slightly" perturbing the previous one (in pure MC there is absolutely no correlation), but much less so than in MD.

Our application of the MC algorithm is within a PMF scheme (explained in the previous section). A given reaction is divided into windows and a typical MC simulation is performed for each window. The force required to keep the reaction coordinate fixed, for a given window, is evaluated for each accepted configuration. After the simulation is completed, all the recorded forces are averaged and the variation of free energy up to the next window can be calculated with equation

(39). The key is to accumulate enough data to obtain the correct average force in each window. This is where correlation becomes an important factor. High correlation means the generated configurations are similar to the previous ones, meaning a greater number of configurations are required to properly explore all of the PES. Schematically, one can think of a connect-the-dots type diagram where point 2 is right next to point 1, point 3 is right next to point 2 and so on.

Figure 2. MC sampling when correlation is high



Because of correlation, a larger number of configurations (or points in figure 2) must be generated in order to get a good estimate of the average force (or cover the area of the square in figure 2). The actual number of configurations necessary will depend on many factors, including the desired accuracy and the specific reaction studied. However, that number can be on the order of a few thousands of configurations. In a scenario where a reaction is divided into 10 windows, each requiring 1500 accepted configurations at an acceptance rate of 50%, and estimating the time required for one calculation to be one minute, the total time required to carry out the simulation would be 20.8 days. If that simulation was carried out at a respectable level of theory (DFT or HF),

then the total number of atoms would be on the order of 10. In other words, traditional pure Metropolis MC is not an adequate method to carry out ab initio PMF simulations of medium sized or solvated systems. Pure ab initio simulations are obviously attractive because of the accuracy they could provide but the CPU time could be prohibitive for all but the smallest of systems.

5.4. Hybrid DeFT/MOPAC Monte Carlo Algorithm

Speeding up the MC simulation of a reaction using the PMF approach is not trivial. Three basic variables are responsible for the time it takes: the number of windows, the number of configurations required in each window to attain convergence of the average force and the time required for a single calculation. Improving the time required for a single calculation can be done by either using a cheaper ab initio method or a smaller basis set. However, ab initio simulations are already limited to a very basic level of theory (HF or simple DFT) with a modest basis set. Performing the simulation at a cheaper ab initio level would only go against what originally pushed us towards the use of ab initio methods, which is a need for accuracy. Another solution is to run the calculations in parallel if the software and hardware available allow it. The vast majority of computational chemists have access to supercomputer centers, making it easy to use parallel environments for longer calculations.

Reducing the number of windows for a simulation is probably not a good option. A parallel can be made to Simpson's rule in mathematics to understand why a large number of windows are preferable. The estimation of an integral with Simpson's rule will always be more accurate if a

larger number of points are considered inside the lower and upper bounds. The same principle applies to equation (39). A MC simulation estimates the average force in a particular window. The value of the average force is then multiplied by the requisite amount to get to the next window. It is assumed that the average force is constant between the windows. However, the force is not constant, inevitably producing an error. The error is minimized if the distance between the windows is small (which is achieved by having as many windows as possible) since windows that are close to each other will have similar average forces.

The only remaining parameter that can be altered is the number of configurations required within each window. As it was mentioned in the previous section, correlation between the configurations will determine how many configurations are required. Much correlation will necessitate a larger number of configurations to properly sample the PES. We propose a scheme, similar to the one of Schofield et al. [29], in which each ab initio configuration of the principal MC chain is separated by a secondary semi-empirical MC chain. The idea behind this scheme is to reduce the correlation between the ab initio configurations of the main MC chain. This hybrid scheme is very similar in nature to conventional MC except for the fact that there are two levels of theory (ab initio and semi-empirical) and two chains (main and secondary).

In our implementation of the hybrid scheme, we use our DFT program (DeFT) for the main chain and the freely available MOPAC software for the secondary chains. The algorithm is very simple. A starting configuration is submitted to DeFT and the force is evaluated. The same configuration is then submitted to MOPAC and is the starting point for a semi-empirical MC

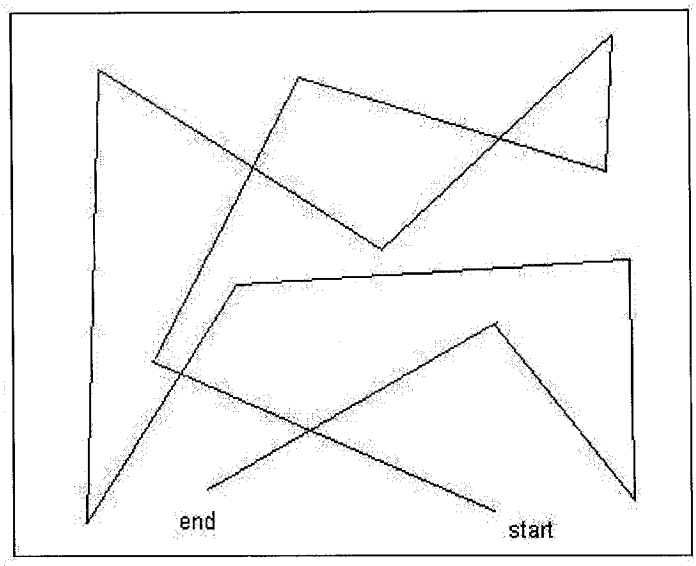
simulation of 500 configurations. The final semi-empirical configuration is accepted or refused according to a new acceptance criterion:

$$(45) \quad x = \exp\left\{-\frac{[E_{DFT}(X_{new}) - E_{MOPAC}(X_{new})] - [E_{DFT}(X_{old}) - E_{MOPAC}(X_{old})]}{k_B T}\right\}$$

If the acceptance criterion is larger than one, then the final semi-empirical configuration is accepted and submitted to DeFT for the force evaluation, becoming the second configuration of the main chain. If it is smaller than one, then it is compared to a random number between zero and one. An acceptance criterion larger than the random number means that the configuration is accepted. It is refused otherwise. Equation (45) needs to consider the MOPAC energies, even though MOPAC is not considered for the actual force calculation, to ensure that the main chain has the correct limiting Boltzmann distribution. In cases where the final semi-empirical configuration is accepted, the same configuration is submitted to DeFT and the cycle is repeated. If it was not accepted, the cycle is repeated with the same initial configuration that was used in the previous attempt (before the secondary semi-empirical chain), exactly like a conventional MC simulation. The cycle is repeated until the average force obtained from the configurations calculated with the DeFT software package has converged.

Correlation between configurations is greatly reduced with this hybrid scheme. We found that the number of required configurations is approximately an order of magnitude smaller than what it is for a conventional ab initio MC simulation. The connect-the-dots picture that was drawn in figure (2) now has dots that are far from each other.

Figure 3. MC sampling when correlation is low



Because the points are far from each other, the square can be properly sampled with fewer points. The only extra cost of this hybrid scheme with respect to the conventional simulation is a smaller acceptance rate and the generation of the fast MOPAC secondary chains. However, the simulation with the hybrid scheme is much faster even with a smaller acceptance rate. For example, reconsider the previous fictitious simulation (10 windows, 1500 accepted configurations per window, 50% acceptance rate, 1 minute per configuration takes 20.8 days). If we improve the sampling by an order of magnitude with the hybrid scheme, only 150 accepted configurations are now required within each window. In a bad case where the acceptance rate would now be only 10%, that same simulation would take half the time required by the conventional method.

The acceptance rate actually observed is directly related to the similarity of the MOPAC and DeFT energies for the system and the length of the secondary chains. If MOPAC and DeFT see similar changes of energy, then the acceptance rate will rise toward the value it would have in a pure

DeFT simulation, which is approximately 50%. However, if they disagree, the observed acceptance rate will be smaller. Unfortunately, hybrid simulations sometimes get stuck if MOPAC and DeFT disagree too strongly, leading to several hundred rejected configurations in a row. For example, if the overall MOPAC energy variation is negative after 500 configurations while the DeFT energy variation is positive, then it is most unlikely that the new configuration will be accepted. To remedy the situation, we introduce a new "helping" factor. Equation (45) can be rewritten to better see how the helping factor affects the acceptance criterion:

$$(46) \qquad x = \exp\left\{-\frac{\Delta E_{DFT} - \Delta E_{MOPAC}}{k_b T}\right\}$$

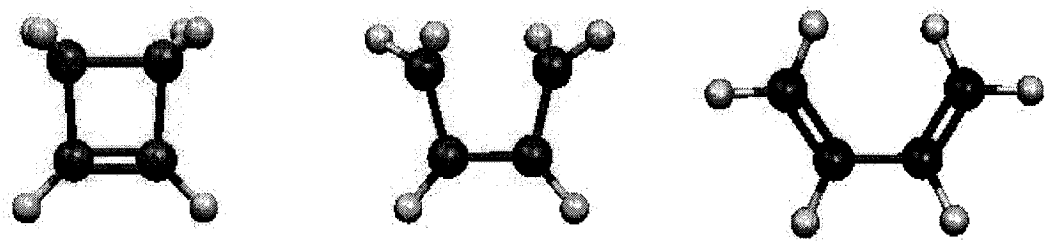
The ideal situation (a large value of x) occurs when a negative ΔE_{DFT} is accompanied by a positive ΔE_{MOPAC}, while the opposite is the worst situation. The helping factor does not come into play for these two cases. However, if ΔE_{DFT} and ΔE_{MOPAC} are of the same sign, then one is helping the situation while the other is making it worse. The helping factor will favor the one that is helping the situation. In a case where they are both positive, ΔE_{DFT} would be multiplied by a factor smaller than one (1-a where a is on the order of 0.1) and ΔE_{MOPAC} would be multiplied by a factor larger than one (1+a). If they are both negative, then the factors are reversed. The helping factor proved to be very useful in avoiding long periods of continuous rejection and raising the overall acceptance rate in our simulations.

6) Applications of the hybrid DeFT/Mopac algorithm

6.1. Ring opening of cyclobutene

The first reaction studied is the conrotatory ring opening of cyclobutene to form the kinetic product, gauche-butadiene. Four main processes are involved in this reaction: breaking of the C-C σ bond, partial elimination of the C-C π bond and formation of two new C-C π bonds, skewing of the carbon skeleton by rotation around the middle C-C bond, and finally, conrotatory movement of the CH_2 groups. Figure (4) shows snapshots of the reactant (cyclobutene), transition state and product (gauche-butadiene).

Figure 4. Cyclobutene in an isomerization reaction to form butadiene

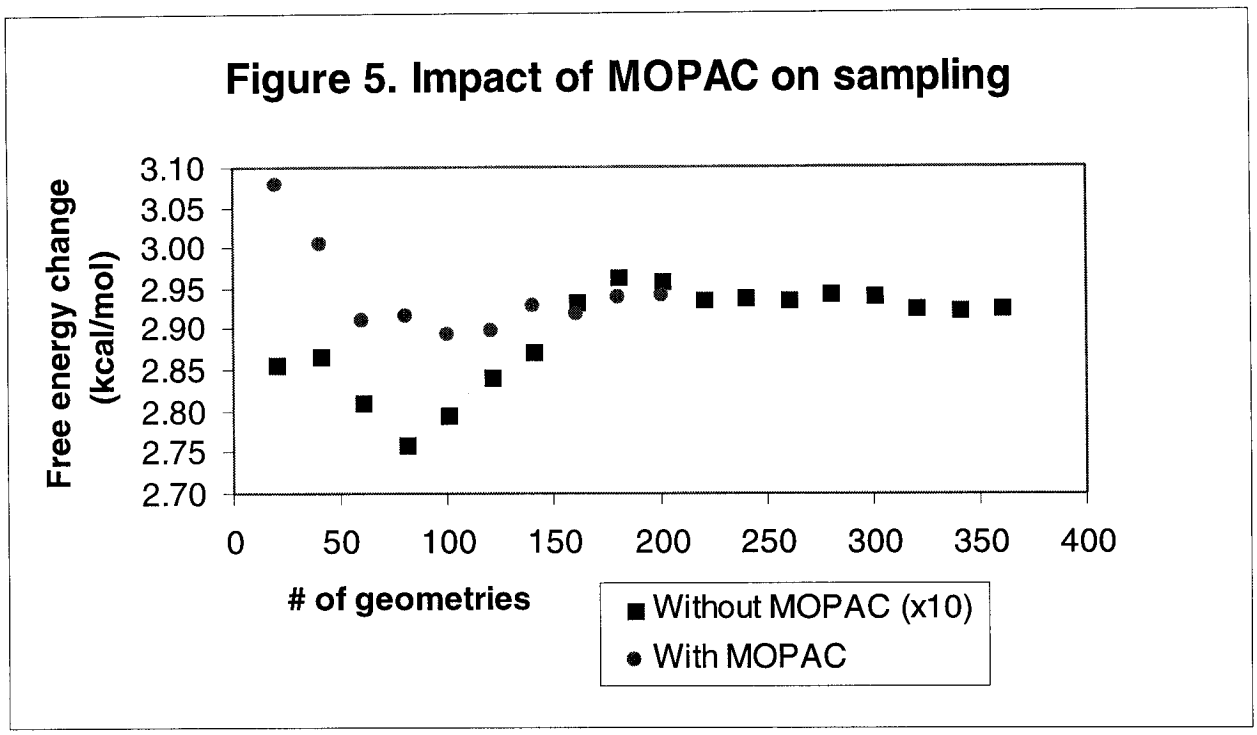


We chose the C1-C4 (upper carbon atoms in the diagram above) distance as the reaction coordinate. All the calculations were carried out using the BP86 functional and a 6-31G* basis set. The entire reaction was divided in 16 windows, in which the C1-C4 distance goes from 1.567Å in cyclobutene to 3.013Å in gauche-butadiene. The positions of the C1 and C4 carbon atoms remain frozen while the other atoms are free to move during the MC simulation. The force required to keep the C1 and C4 atoms fixed is recorded for each accepted configuration, and then averaged at the end of the simulation to calculate the free energy change observed when going to the next window. In

the secondary chains, 500 MOPAC configurations (with the PM3 Hamiltonian) are generated between each DeFT configuration. A helping factor of 10% was applied to increase the acceptance rate of our hybrid DeFT/MOPAC MC algorithm, as explained in section 5.4. Before actually performing the entire simulation, one window was picked at random and tested against a pure DeFT MC simulation to verify that the hybrid DeFT/MOPAC algorithm really diminishes the correlation between each configuration. Correlation is easily monitored by recording the free energy change at regular intervals until it converges to a stable value. For the hybrid algorithm to be advantageous, it must converge much more quickly than the pure DeFT simulation and give the same end result.

Table 4. Correlation of pure DeFT and hybrid DeFT/Mopac MC (distance = 1.799Å)

pure DeFT		hybrid DeFT/Mopac	
# of configurations	Free energy change (kcal/mol)	# of configurations	Free energy change (kcal/mol)
200	2.858	20	3.081
400	2.869	40	3.007
600	2.813	60	2.912
800	2.762	80	2.916
1000	2.798	100	2.894
1200	2.842	120	2.898
1400	2.874	140	2.928
1600	2.935	160	2.919
1800	2.966	180	2.940
2000	2.959	200	2.943
2200	2.937		
2400	2.940		
2600	2.937		
2800	2.944		
3000	2.942		
3200	2.925		
3400	2.927		
3600	2.925		

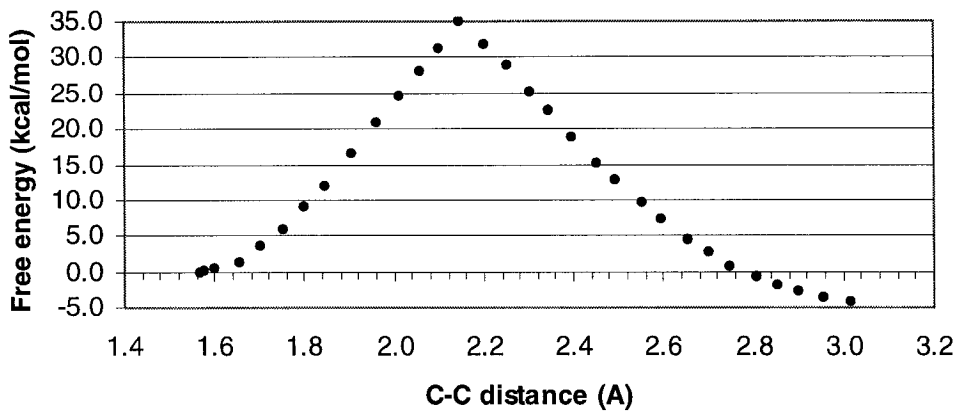


From Table 4 and Figure 5, it is clear that the hybrid scheme greatly reduces the correlation between the configurations. It took approximately 2200 configurations to achieve convergence with the pure DeFT simulation and only 180 configurations for the hybrid DeFT/MOPAC simulation. Considering the acceptance rates of 25% for the hybrid DeFT/MOPAC program and 39% for the pure DeFT program, the total number of DeFT configurations considered is eight times smaller (720 versus 5641) for the hybrid scheme. The only extra cost that could hurt the hybrid scheme comes from the MOPAC secondary MC chains between each DeFT configuration. However, generating the MOPAC chains takes approximately 1.9 seconds in this case, which is insignificantly small when compared with a single DeFT energy calculation. As for the DeFT energy calculations, they take 23 seconds for discarded configurations (energy only) and 41 seconds for accepted configurations (energy and gradient), using four 900 MHz UltraSPARC III Cu processors. In the end, the pure

DeFT MC-PMF window takes approximately 47 hours while the DeFT/MOPAC MC-PMF window is completed in 6 hours.

With that information available, we proceeded to the actual simulation of the reaction. We concluded that 200 accepted configurations were more than enough to ensure a converged average force in each window. Our results, the free energy of reaction and the free energy of activation, will be compared with the experimental values and thermodynamic calculations performed with Gaussian using the normal mode approximation. Figure 6 shows the free energy profile that we obtained.

Figure 6. Ring opening of cyclobutene



The end result is a very smooth free energy profile with an activation free energy of approximately 34.9 kcal/mol and a free energy variation of -4.1 kcal/mol. The average acceptance rate for the 16 windows is 22.2%, with a minimum of 10.1% at 2.4Å and a maximum of 51.9% at 1.57Å. Windows around the transition state seem to have lower acceptance rates, meaning DeFT and MOPAC disagree more strongly in that region. Nonetheless, with acceptances rates in the range

of 10% to 50%, the hybrid DeFT/Mopac algorithm easily outperforms a pure DeFT algorithm in terms of required CPU time.

Table 5. Comparison of the exactitude of the simulation for the ring opening of cyclobutene

	Present Work (kcal/mol)	Gaussian normal mode analysis (kcal/mol)	Experimental values (kcal/mol) [a]
Activation free energy	34.9	31.6	32.8 ± 0.5
Variation of free energy	-4.1	-7.3	-9.7 ± 0.4

[a] From reference [30]

It is not surprising to see that our simulation is overestimating the activation free energy. A proper choice of the reaction coordinate is crucial to accurately estimate the barrier height. With a poor choice of reaction coordinate, the simulation may very well proceed via a pathway that does not actually go through the transition state, resulting in an overestimation of the barrier. Our choice for the reaction coordinate was very crude. Choosing the C1-C4 distance as the reaction coordinate makes the assumption that the separation of the carbon atoms is the only variable affecting the energetics of the reaction (as in a bond dissociation). Other variables are also systematically changing throughout the reaction, like the dihedral angle created by the four carbon atoms as they skew away from planarity. Instead of simply fixing the C1 and C4 atoms in space, it certainly would have been a better idea to somehow include a proper mix of the C1-C4 distance and the dihedral angle in the reaction coordinate.

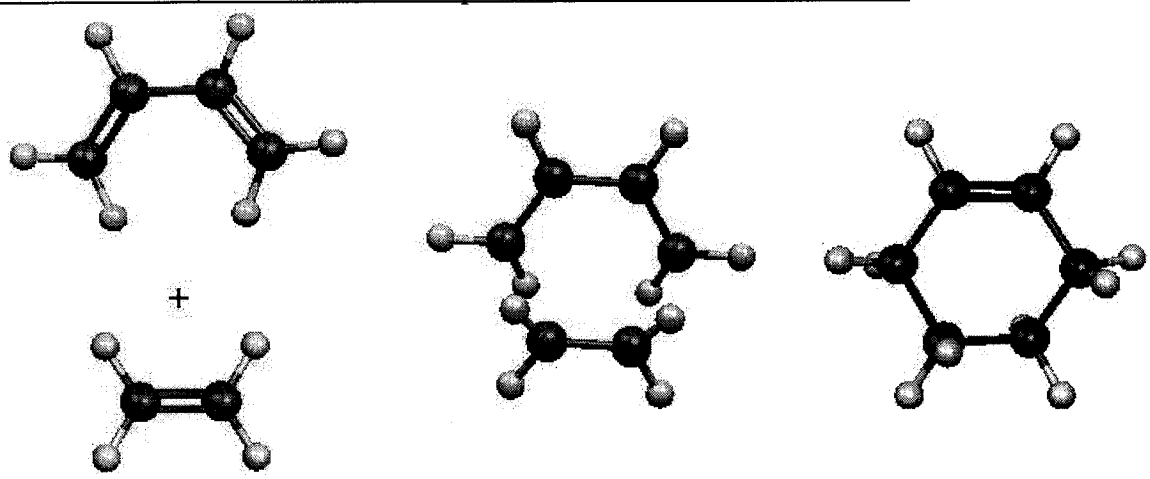
The overestimation of the barrier also affects the variation in the free energy. If the downhill part of the curve is correct, but started from a point higher in energy, then the final point will inevitably by too high as well. It is not a coincidence that our activation free energy and variation of free energy are higher than the Gaussian results by the same amount (3.3 kcal/mol for the free energy of activation and 3.2 kcal/mol for the free energy of reaction). It is also important to realize that sampling errors, if present, are amplified in the later windows. For example, if each window of the downhill part of the curve overestimates the free energy change by a few tenths of a kcal/mol, then the final variation in the free energy will be greatly overestimated because the error is compounded.

Improper sampling in windows around the transition state could also be a potential source of error. The acceptance rates for those windows were quite low (around 10 to 15%). It is possible that the simulation got stuck in a well which it could not escape. If mobility within the simulation is restricted to getting in and out of the well, then poor sampling is observed, resulting in erroneous average forces. A longer simulation, or a different starting point, could verify the validity of the suspect windows. Nonetheless, this simple example nicely demonstrates the advantage of using our hybrid DeFT/MOPAC MC algorithm over a pure DeFT algorithm and reproduces experimental values reasonably well.

6.2. Diels-Alder reaction of ethylene with butadiene

Our first example is successful in proving the sampling efficiency of our hybrid DeFT/MOPAC algorithm. However, the electronic energy and free energy profiles, although not explicitly shown in the previous section, are practically identical for the ring opening of cyclobutene. The capability of the algorithm to correctly incorporate thermal effects was not truly tested. The second reaction studied is the simplest of all Diels-Alder reactions: ethylene and butadiene combining to form cyclohexene. The reactants, transition state and product are shown in Figure 7.

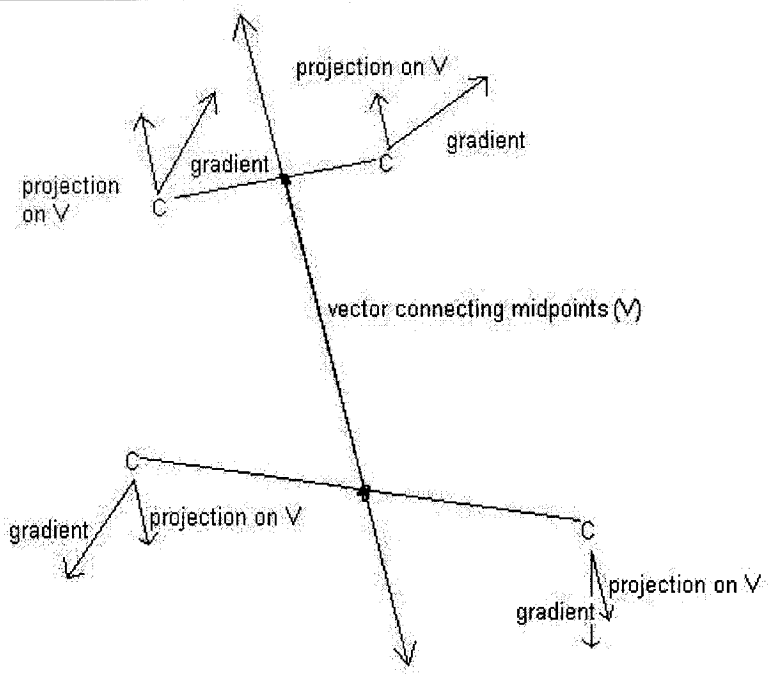
Figure 7. Reactants, transition state and product of the Diels-Alder reaction



In this reaction, two molecules in vacuo combine to form one molecule in vacuo. A considerable loss of entropy will occur and produce a free energy profile that differs significantly from the electronic energy profile. According to Gibb's law, the free energy of reaction and the free energy of activation should be more positive than their electronic counterparts, because of the loss of entropy. The BP86 functional, 6-31G* basis set and 500 MOPAC configurations (with the PM3 Hamiltonian) in the secondary MC chains were used, once again, for this simulation. The only difference in methodology with the previous reaction is the helping factor, which is given a value

of 20% instead of 10%. The reaction coordinate is the distance between the midpoint of the C-C bond in ethylene and the midpoint of the terminal carbon atoms in butadiene. In this simulation, all the atoms are allowed to move. The positions of the four atoms involved in the reaction coordinate are slightly adjusted at each MC step to satisfy the constraint, using a SHAKE type algorithm. To evaluate the force required to satisfy the constraint, the gradients of the four atoms involved are transposed upon the vector connecting the midpoints described above. The average transposed gradient of the two carbon atoms in ethylene is added to the average transposed gradient of the two carbon atoms in butadiene to give the final force, as demonstrated in Figure 8.

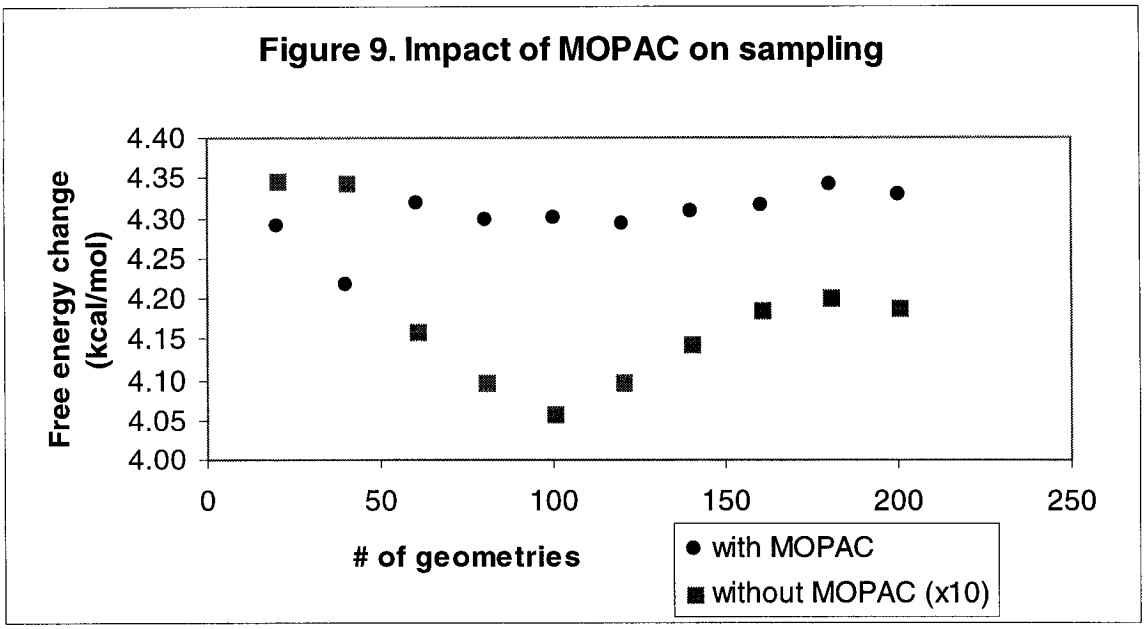
Figure 8. Evaluation of the force in the Diels-Alder reaction



Before actually proceeding to the simulation, the same correlation test performed in the previous simulation, for one specific window, was done as well for this reaction.

Table 6. Correlation of pure DeFT and hybrid DeFT/MOPAC MC (distance = 1.704Å)

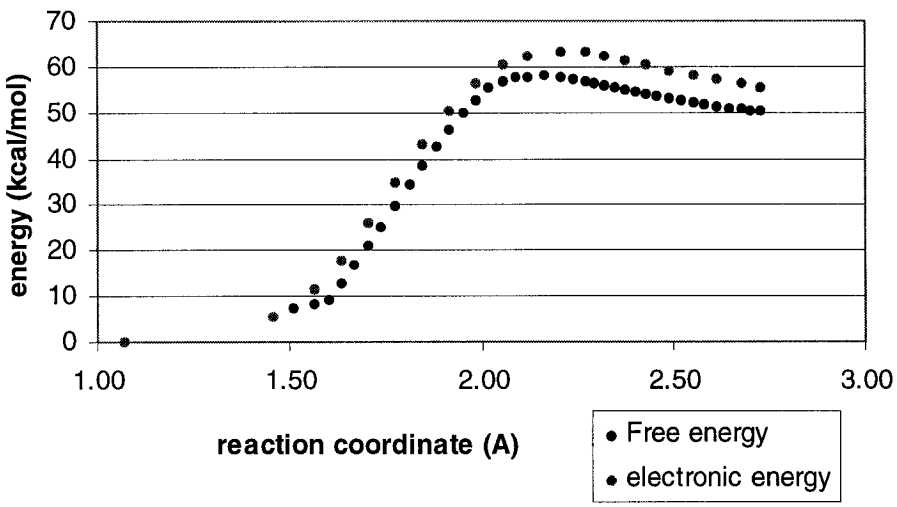
pure DeFT		hybrid DeFT/MOPAC	
# of configurations	Free energy change (kcal/mol)	# of configurations	Free energy change (kcal/mol)
200	4.347	20	4.291
400	4.346	40	4.219
600	4.161	60	4.320
800	4.099	80	4.299
1000	4.060	100	4.301
1200	4.098	120	4.292
1400	4.146	140	4.310
1600	4.187	160	4.317
1800	4.202	180	4.344
2000	4.191	200	4.330



We come to the same conclusions we did in Section 6.2. The hybrid DeFT/MOPAC algorithm greatly reduces the correlation and requires no more than 200 configurations in each window. A difference of 0.1 kcal/mol between the converged pure DeFT and hybrid DeFT/MOPAC simulations is acceptable. The reaction was divided into 20 windows, in which the reaction coordinate varies from 1.455Å to 2.730Å. Note that our reaction coordinate has a value of infinity

for the reactants (ethylene + butadiene) and 1.073Å for the product (cyclohexene) in its most stable conformer. Our simulation started with a stable conformation of cyclohexene in which the reaction coordinate had a value of 1.455Å, rather than the absolute lowest energy conformation that has a value of 1.073Å, for practical reasons. The purpose of our simulation is not to accurately describe cyclohexene going to its absolute minimum, but rather ethylene and butadiene coming together to form one acceptable conformation of cyclohexene. It would be straightforward to perform the simulation of cyclohexene going to its absolute minimum. Obviously, we cannot simulate the separation of ethylene and butadiene all the way to infinity, so we stopped at a distance of 2.730Å. We approximated the free energy change missing at both ends of the curve. Figure 10 displays the electronic energy and free energy curves for our actual simulation.

Figure 10. Diels-Alder reaction



We actually performed this simulation starting with cyclohexene, meaning the curves in Figure 10 represent the reverse reaction. The free energy curve obtained is very smooth again. The average acceptance criterion throughout the 20 windows is 29%. A very important observation can be made with Figure 10. The free energy curve lies below the electronic energy curve. This is to be expected since entropy is gained (entropy gain = lower free energy) upon the distortion of cyclohexene and finally its separation into two distinct molecules. We had to make approximations to estimate the free energy values for the reaction coordinate down to 1.073Å and out to infinity, a necessity for the evaluation of the free energy of activation and the free energy of reaction. The first approximation is that the free energy change is identical to the change in electronic energy upon going from 1.455Å (our first window) to 1.073Å (the global minimum of cyclohexene). Since these two configurations are very similar, their relative entropies should be more or less equivalent. This leads us to believe the approximation is valid. For the latter part of the curve, the situation is more complicated. Separating ethylene and butadiene to infinity produces two distant molecules rather than one complex. The separation will impact the entropy and thermal energy (translations, rotations and vibrations at 298.15K). Because a distance of 2.73Å is still quite close to the transition state, we supposed that the entropy and thermal energy at 2.73Å are identical to that of the transition state. Using Gaussian, we proceeded to calculate the entropy and thermal energy contributions to the free energy change when separating the molecules to infinity. We found contributions of -11.6 kcal/mol for the entropic component and -2.1 kcal/mol for the thermal energy. To that, we added the difference of electronic energy between the configuration at 2.73Å and the infinitely separated molecules, a value of -4.0 kcal/mol.

Table 7. Comparisons of Diels-Alder simulation results

	Present Work (kcal/mol)	Gaussian normal mode analysis (kcal/mol)	Experiment (kcal/mol)
Free energy of activation [a]	28.1	27.5	27.5 [b]
Free energy of activation (reverse reaction) [a]	58.0	56.9	54.4 [c]
Free energy of reaction [a]	-29.9	-29.4	-26.9 [c]

[a] Note that the values are those of the actual Diels-Alder reaction, not the reverse reaction presented in Figure 10

[b] From reference [31]

[c] From references [32], [33] and [34]

Our free energy of reaction and free energy of activation compare well with the Gaussian calculations. The deviations with the experimental values for the free energy of reaction can in part be explained by the fact that we used a single static calculation for butadiene and cyclohexene. In real life, butadiene and cyclohexene will be found in many conformations of varying populations.

Although not perfect, this example shows that our hybrid DeFT/MOPAC algorithm is able to incorporate entropic effects in the calculations. The activation free energy of the reverse reaction is 5.3 kcal/mol lower than its electronic energy counterpart. Once again, it is not surprising that we overestimate the barrier height with respect to the Gaussian calculations. We feel our choice of reaction coordinate is satisfactory, but not ideal. A helping factor of 20% may have contributed to the errors by artificially favoring certain regions of the PES. However, the acceptance rates were not acceptable with smaller helping factors. A better choice of reaction coordinate could be the key to better acceptance rates with smaller helping factors.

7) Conclusions

Our adaptation of the GCOSMO solvation algorithm within our own DeFT software package is satisfactory. The use of SIP radii reduces the RMS error for most molecules included in the parameterization process. Two reactions were studied in further detail, the solvation of glycine and the tautomerization of 2-hydroxypyridine to 2-pyridone in water. In both cases, our results agree well with the experimental values. Future work on the algorithm should focus on the implementation of the second derivatives and further parameterization of the SIP algorithm. The necessity to use different offset values for different types of molecules is a disadvantage which could be addressed.

As the two examples of Section 6 demonstrated, our hybrid DeFT/MOPAC MC algorithm can be used to effectively simulate chemical processes. PMF style simulations work better for particular types of reactions. It is much easier to simulate reactions that go from one reactant to one product, or more generally, from one minimum on the PES to another. If two or more reactants, or products, need to be separated to infinity (as in the case of a Diels-Alder reaction), then errors are introduced since it is impossible to have an infinite number of windows. In such cases, windows must be added to the simulation until the average force is small enough to be neglected. Another option is to use approximations with static free energy calculations to estimate the contributions of entropy and thermal energy, as we did in section 6.2. This particular limitation has nothing to do with our hybrid DeFT/MOPAC MC algorithm, and everything to do with the PMF approach. That being said, the PMF approach works perfectly for isomerization reactions of any kind (like the ring opening of cyclobutene).

The two cases that we presented were reactions in the gas phase. However, the hybrid DeFT/MOPAC algorithm can be coupled with a continuum model to perform the same simulations in solution. The GCOSMO model has been implemented within DeFT, while the COSMO model is an integral part of MOPAC, so all the necessary tools already exist. The derivatives of the energy can be calculated in solution with DeFT, allowing the evaluation of the average force in solution for each window. Simulations in solution, using the continuum models, would be remarkably fast compared with traditional simulations using explicit solvent molecules.

Finally, if pure ab initio levels of theory are used for the principal MC chain, then the total number of atoms in the reaction cannot realistically exceed 20. Unfortunately, most systems of interest easily surpass that limit in number of atoms (enzymes, transition metal complexes, big organic molecules, DNA bases or explicit solvent cases). Our main objective is to couple this hybrid DeFT/MOPAC MC algorithm with our own developed QM/MM scheme (a combination of DeFT and the AMBER ROAR module). The same algorithm can be used, with the exception that the main MC chain is generated at the QM/MM (DFT as the QM level) instead of a pure DFT level of theory. The secondary MOPAC chain would be replaced by a QM/MM calculation where the QM method is either AM1 or PM3 (ROAR alone has this functionality). With a DeFT/MM//Semi-Empirical/MM MC algorithm, systems of impressive sizes could be studied, if the total number of QM atoms stays at roughly 20 or so.

Acknowledgments

First and foremost, I would like to thank my B.Sc. And M.Sc. supervisor for the past three years, Dr. Alain St-Amant, for giving me the opportunity to learn computational chemistry. You always accepted me, as I am, and I truly appreciate it. I could not have picked a better supervisor.

To my lab partners, Etienne Paradis, Michelle Shaw and Delphine Courmier, for showing me the ropes of graduate studies and helping me whenever I needed it. Thanks to all my friends, especially Mario Rios, Sophie Quevillon and Marie-Christine Nolet, for your support and encouragement.

Last but not least, to whoever had to hear me complain and whine about anything, thanks for listening. It helps more than you can ever think.

References

[1] A. Leach; Molecular Modelling principles and applications, Pearson Education Limited, 2nd edition, 2001

[2] Peter, Volhardt, Schore; Organic Chemistry, 2nd edition

[3] C. Cramer, D. Truhlar; Chemical Reviews, 1999, Vol. 99, 2161

[4] A. Klamt, G. Schüürmann; Journal of the Chemical Society Perkin Transactions 2, 1993, 799

[5] T. Truong, E. Stefanovich; Chemical Physics Letters, 1995, Vol. 240, 253

[6] M. Cossi, V. Barone, R. Cammi, J. Tomasi; Chemical Physics Letters, 1996, Vol. 255, 327

[7] D. Tannor, B. Marten, R. Murphy, R. Friesner, D. Sitkoff, A. Nicholls, M. Ringnalda, W. Goddard, B. Honig; Journal of the American Chemical Society, 1994, Vol. 116, 11875

[8] C. Cramer, D. Truhlar; Journal of the American Chemical Society, 1991, Vol. 113, 835

[9] L. Onsager; Journal of the American Chemical Society, 1936, Vol. 58, 1436

[10] V. Barone, M. Cossi, J. Tomasi; Journal of Chemical Physics, 1997, Vol. 107, No. 8, 3210

[11] J.L. Pascual-Ahuir, E. Silla, I. Tunon; Manual of GEPOL93, 1993

[12] T. Truong, E. Stefanovich; Journal of Chemical Physics, 1995, Vol. 103, No. 9, 3709

[13] V. Barone, M. Cossi, J. Tomasi; Journal of Computational Chemistry, 1998, Vol. 19, No. 4, 404

[14] S. Miertus, E. Scrocco, J. Tomasi; Chemical Physics, 1981, Vol. 55, 117

[15] B. Smith, N. Hall; Journal of Computational Chemistry, 1998, Vol. 19, No. 13, 1482

[16] G. Maitland, M. Rigby, E. Smith, W. Wakeham; Intermolecular Forces, Clarendon Press, Oxford, 1981

[17] C. Gray, K. Gubbins; Theory of Molecular Fluids, Clarendon Press, Oxford, 1984

[18] J. Hine, P. Mookerje; Journal of Organic Chemistry, 1975, Vol. 40, 292

[19] S. Cabani, P.Gianni, V. Mollica, L. Lepori; Journal of Solution Chemistry, 1981, Vol. 10, 563

[20] D. Wagman, V. Parker, R. Schumm, I. Halow, S. Bailey, K. Churney, R. Nuttall; Journal of Physical and Chemical Reference Data, 1982, Vol. 11 (Suppl. 2)

[21] J. Gaffney, R. Pierce, L. Friedman; Journal of the American Chemical Society, 1977, Vol. 99, 4293

[22] P. Haberfield, Journal of Chemical Education, 1980, Vol. 57, 346

[23] R. Bonaccorsi, F. Floris, P. Palla, J. Tomasi; Thermochimica Acta, 1990, Vol. 162, 213

[24] P. Beck; Accounts of Chemical Research, 1977, Vol. 10, 186

[25] D. Baveridge, F. DiCapua; Annual Review of Biophysics and Biophysical Chemistry, 1989, Vol. 18, 431

[26] E. Carter, G. Ciccotti, J. Hynes, R. Kapral; Chemical Physics Letters, 1989, Vol. 156, 472

[27] A. Michalak, T. Ziegler; Journal of Physical Chemistry A, 2001, Vol. 105, 4333

[28] T. Straatsma, H. Berendsen, J. Postma; Chemical Physics, 1986, Vol. 85, 6720

[29] J. Schofield, D. Wei, D. Salahub, R. Iftimie; Journal of Chemical Physics, 2000, Vol. 13, No. 12, 4852

[30] D. Spellmeyer, K. Houk; Journal of the American Chemical Society, 1988, Vol. 110, 3412

[31] J. Sauer, R. Sustmann; Angewandte Chemie International Edition in English, 1980, Vol. 19, 779

[32] Handbook of Chemistry and Physics, 1st Student Edition, 1988

[33] W. Steele, R. Chirico, S. Knipmeyer, A. Nguyen, N. Smith, I. Tasker; Journal of Chemical and Engineering Data, 1996, Vol. 41, 1269

[34] C. Beckett; Journal of the American Chemical Society, 1948, Vol. 70, 4227

[35] W. Hehre, L. Radom, P. Schleyer, J. Pople; Ab Initio Molecular Orbital Theory, Wiley, 1986