UPTEC X 04 028
APR 2004

ISSN 1401-2138

MARTIN ANDÉR

Implementation of a generalized Born/surface area algorithm for solvation energy calculations in the molecular dynamics program *Q*

Master's degree project

UPPSALA UNIVERSITET

Molecular Biotechnology Programme

Uppsala University School of Engineering

UPTEC X 04 028	Date of issue 2004-04

Author

Martin Andér

Title (English)

Implementation of a generalized Born/surface area algorithm for solvation energy calculations in the molecular dynamics program *Q*

Title (Swedish)

Abstract

The generalized Born/surface area (GB/SA) method for calculating free energies of solvation was implemented in the molecular dynamics program package Q. The implementation is based on an analytical method for calculating Born radii and the Linear Combination of Pairwise Overlaps (LCPO) algorithm for calculating solvent accessible surface areas. A set of nine molecules, ranging in size from 6 to 1029 atoms, was used to test the implementation against the results from similar studies and experimentally determined energies. Using this test set of molecules, the GB/SA implementation in Q produces solvation energies with an average unsigned error of 0.99 kcal mol^{-1}, to be compared with a reported implementation with an average unsigned error of 1.06 kcal mol^{-1}.

Keywords

Computer simulations, molecular dynamics, solvation energy, generalized Born equation, solvent accessible surface area

Supervisors

Johan Åqvist
Department of Cell & Molecular Biology, Uppsala University

Scientific reviewer

Anders Karlén
Department of Medicinal Chemistry, Uppsala University

Project name	Sponsors
Language **English**	Security
ISSN 1401-2138	Classification
Supplementary bibliographical information	Pages **22**

Biology Education Centre Biomedical Center Husargatan 3 Uppsala
Box 592 S-75124 Uppsala Tel +46 (0)18 4710000 Fax +46 (0)18 555217

Implementation of a generalized Born/surface area algorithm for solvation energy calculations in the molecular dynamics program Q

Martin Andér

Sammanfattning

Datorsimuleringar spelar en allt större roll inom molekylärbiologin. Idag är beräkningskapaciteten hos en vanlig kontorsdator fullt tillräcklig för att genomföra enklare simuleringar, även av medelstora system. Ett område inom vilket resultat från datorsimuleringar kan vara till stor hjälp är läkemedelsdesign. Så kallad "rational drug design", där molekyler helt designas i datorn för att binda till strukturer i cellen som är relaterade till något sjukdomstillstånd, kan innebära stora förbättringar när det gäller möjligheten att ta fram effektiva läkemedel.

När man utför beräkningar på hur molekyler binder till varandra är energiförändringar hos de inblandade molekylerna av central betydelse. Solvatiseringsenergin, den förändring i energi som beskriver förhållandet mellan en molekyl i vakuum och samma molekyl omgiven av ett lösningsmedel, är intressant i dessa sammanhang.

I det här examensarbetet har ett existerande program för molekylära simuleringar utökats med en metod för att beräkna solvatiseringsenergier. Metoden kallas för generalized Born/surface area (GB/SA) och använder sig av en implicit beskrivning av lösningsmedlet, vilket innebär att lösningsmedlet endast representeras av en dielektrisk konstant, ϵ. Som namnet antyder består metoden av två delar. En del tar hand om elektrostatiska interaktioner mellan den simulerade molekylen och lösningsmedlet och den andra delen behandlar effekter relaterade till molekylens gränsyta mot lösningsmedlet. Denna metod är förhållandevis snabb, och har visat sig ge resultat som stämmer tillräckligt väl överens med experimentella värden för att vara användbar i många sammanhang.

Examensarbete 20p i Molekylär bioteknikprogrammet
Uppsala Universitet april 2004

Contents

1 Introduction

1.1 Computer simulations

Computer simulations play an important role in molecular biology. Over the last two decades, the computational capacity of computers has increased tremendously; an average desktop workstation[1]of today would have been considered a supercomputer twenty years ago. Together with more and more refined methods, this leads to better results and an expanded range of possible simulations.

Naturally, the speed of a simulation decreases as the number of simulated atoms increases, since more interactions have to be considered. This is one of the main problems with simulations of biomolecules, since they are often both large and require an aqueous environment – adding solvent water to a system may increase the number of atoms by an order of magnitude. However, instead of explicitly including solvent water molecules in the simulation, the effects of solvation may be approximated implicitly by describing the surroundings as a dielectric continuum. The concept is illustrated in figure 1.

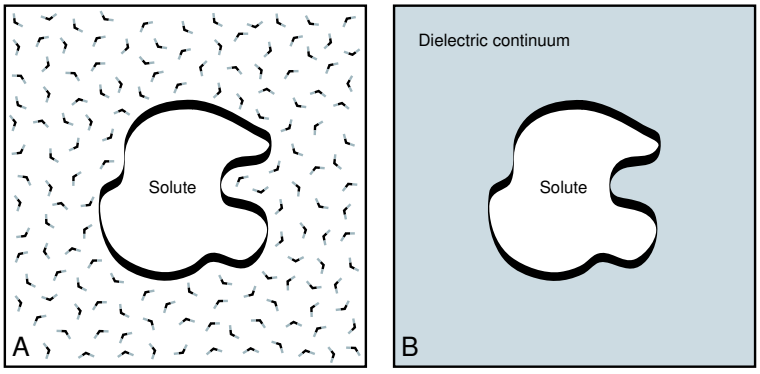

Figure 1: A: The solvent is represented explicitly using an atomic model of the solvent molecules. B: A dielectric continuum is used to represent the solvent molecules implicitly. The continuum has the average properties of water and is defined by the dielectric constant, ϵ.

In this work, an algorithm for calculating the free energy of solvation for a molecule, called the generalized Born/surface area (GB/SA) method [1, 2], was implemented into an existing piece of software – the molecular dynamics package *Q* [3]. In the implementation described here, the method is entirely analytical and it is thus easily differentiable. This means that the method may be used to calculate forces in molecular dynamics [4, 5, 6].

[1]The simulations presented in this work were all performed on a single-CPU Intel P4 workstation.

1.2 Ligand binding

Structure-based rational drug design is an application of computational chemistry that is anticipated to have a great impact on drug discovery. Calculating binding energies in ligand-receptor complexes is of fundamental importance in finding a candidate drug molecule in this approach [7]. An article written by Zhou *et al.* [8], describes how the GB/SA method can be adapted to be used in conjunction with the Linear Interaction Energy (LIE) method [9] to calculate ligand binding energies.

However, this project does not deal with calculating binding energies for ligand-receptor complexes, but rather implementing the GB/SA method in its original form. The implementation is intended to be adapted to ligand binding simulations in the near future.

2 Theory

2.1 Molecular mechanics

There are several ways to describe the properties of a molecule, or a system of molecules. A quantum mechanical approach is perhaps the most obvious choice, since most of the action is related to the electrons. However, the applicability of quantum mechanical methods is severely limited by their computational cost.

In the molecular mechanics approach, instead of modelling electrons explicitly, entire atom-like particles are modelled as spheres with a predefined mass, radius, and partial charge. Furthermore, interactions between atoms are explicitly defined and modelled with classical potentials; a potential function, or force field, is used to parameterize these potentials. An obvious drawback with this approach is that it is only possible to simulate the ground state of the molecule.

Generally, a force field contains two types of terms: intramolecular terms associated with interactions between covalently bound atoms, and intermolecular terms associated with non-bonded interactions. The potential function $V(\mathbf{r}^N)$ is simply the sum of these terms:

$$V(\mathbf{r}^N) = V_{intra}(\mathbf{r}^N) + V_{inter}(\mathbf{r}^N) \tag{1}$$

where $\mathbf{r}^N = \mathbf{r}_1, ..., \mathbf{r}_N$ is the set of spatial coordinates for all N atoms in the molecule.

Simple harmonic potentials may be used for the bond lengths and angles, which gives the intramolecular potential function the following form:

$$V_{intra} = \sum_{\text{bonds}} K_r(r-r_{eq})^2 + \sum_{\text{angles}} K_\theta(\theta-\theta_{eq})^2 + \sum_{\text{dihedrals}} \frac{V_n}{2}[1+\cos(n\phi-\gamma)] \tag{2}$$

where r, r_{eq}, θ, and θ_{eq} represent bond lengths and angles and their equilibrium values, K_r, K_θ, and V_n are force constants, and ϕ, γ, and n represent dihedral angles, phase shift (location of first maximum), and periodicity, respectively. See figure 2.

The intermolecular term itself consists of two terms: A Lennard-Jones (LJ) 6-12 term representing induced dipole-dipole moment interaction and hard-sphere repulsion, and a Coulomb term representing electrostatic interactions. The resulting expression for the intermolecular potential function is:

$$V_{inter} = \sum_{i<j} 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] + \sum_{i<j}\frac{q_i q_j}{r_{ij}} \tag{3}$$

where r_{ij} is the Euclid distance between atoms i and j, ϵ_{ij} is the depth of the LJ potential, σ_{ij} is the interatomic distance at which the interaction energy between atoms i and j is zero, and q_i and q_j are the partial charges of atoms i and j. The LJ potential is illustrated in figure 3.

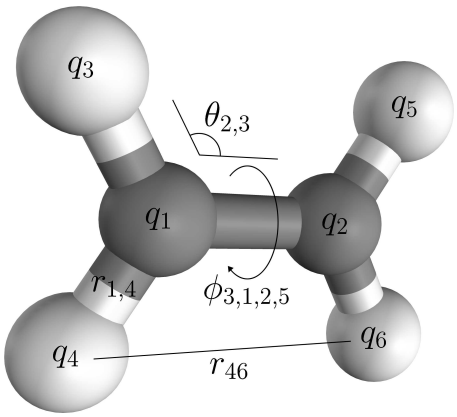

Figure 2: Examples of some of the force field parameters describing an ethylene molecule. Note that $r_{i,j}$ denotes the bond length between atoms i and j whereas r_{ij} denotes the Euclid distance between the atoms.

In practice, a force field is a set of values for the above parameters that has been optimized for a certain task. Different force fields may be aimed at accurately describing different types of molecules, or different types of calculations. Experimental data as well as quantum chemical calculations are used to tune the parameters.

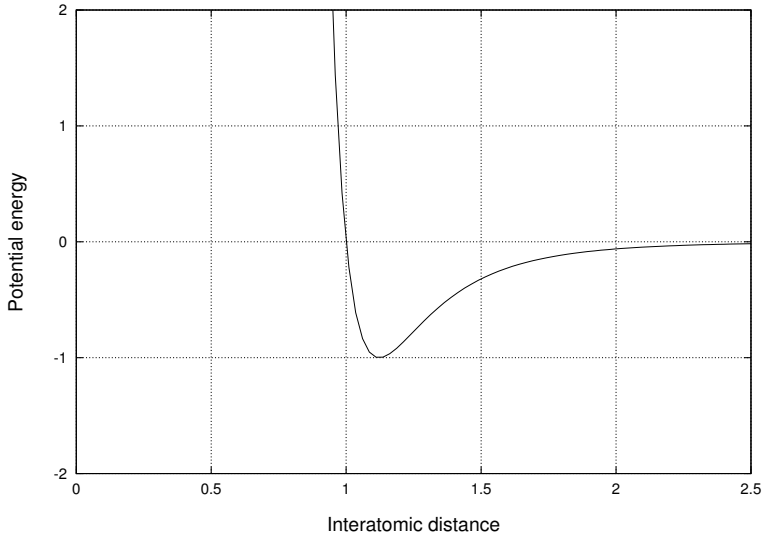

Figure 3: The Lennard-Jones potential curve, shown here with $\epsilon_{ij} = \sigma_{ij} = 1.0$. Note that the potential is zero at $r_{ij} = \sigma_{ij} = 1.0$, when the two atoms make van der Waals contact. The energy minimum is at $r_{ij} = r^*_{ij} = \sqrt[6]{2}\sigma_{ij}$

2.2 Molecular dynamics

In molecular dynamics (MD) [1, 4], the motion of the individual particles in a system consisting of N particles (i.e. N atoms in a molecule) is simulated by solving Newton's equation of motion. The force acting on particle i, $\mathbf{F}_i$, is calculated from the potential energy function of the system, $V(\mathbf{r}^N)$:

$$\mathbf{F}_i = -\frac{\partial V(\mathbf{r}^N)}{\partial r_i} \tag{4}$$

$V(\mathbf{r}^N)$ is calculated using equations (1) – (3) as described in section 2.1. Provided that the mass of particle i, m_i, is known, equation (4) can be inserted into Newton's second law, thereby providing a means to calculate $\mathbf{a}_i$, the acceleration of particle i:

$$\mathbf{a}_i = \frac{\mathbf{F}_i}{m_i} \tag{5}$$

With $\mathbf{a}_i$ being the second derivative of the position $\mathbf{r}_i$ with respect to time, the dynamic behavior of the system is described by the second order differential equation:

$$\frac{\partial^2 \mathbf{r}_i}{\partial t^2} = \frac{\mathbf{F}_i}{m_i} \tag{6}$$

By integrating equation (6), an expression for $\mathbf{r}_i$ as a function of time is obtained. Assuming that that $\mathbf{F}_i$ is constant during some time step δt, finite difference methods may be used to calculate new velocities and positions given the δt time step. One common algorithm is the Leapfrog Verlet [10] method:

$$\begin{cases} \mathbf{r}_i(t+\delta t) = \mathbf{r}_i(t) + \delta t\, \mathbf{v}_i(t) \\ \mathbf{v}_i(t+\delta t) = \mathbf{v}_i\left(t - \frac{1}{2}\delta t\right) + \delta t\, \mathbf{a}_i(t) \end{cases} \tag{7}$$

where $\mathbf{v}_i(t)$ is the velocity of particle i at time t.

A sampling of the conformational space is obtained by running simulations for many time steps. The required length of the simulation depends on the size and robustness of the molecule being simulated.

2.3 Solvent models

To accurately simulate the interactions between biomolecules, it is important to perform the simulations in an aqueous environment. Modelling the surrounding water molecules (or solvent) explicitly is straightforward, and it is the most accurate way to include the effects of solvation in a simulation. However, the more atoms are included in a simulation, the more computational time the simulation will require. Thus, this method may be very time-consuming. Instead of treating the water molecules explicitly, an implicit solvent model may be used, where the solvent is represented by a dielectric continuum. This way, the number of atoms included in the simulation may be significantly decreased, with a corresponding increase in the speed of the simulation.

Implicit solvation methods have been shown to be fast and relatively accurate as well as having a number of other advantages [5, 6, 11, 12] over their explicit solvent counterparts. No equilibration is needed for a continuous medium; thus convergence may be reached in a fewer number of steps, and the solvent instantly reaches cavities formed in the solute in case of conformational changes.

2.4 The generalized Born equation

The free energy of solvation (G_{sol}) is traditionally considered to be made up of three terms:

$$G_{\mathrm{sol}} = G_{\mathrm{pol}} + G_{\mathrm{vdW}} + G_{\mathrm{cav}} \tag{8}$$

G_{pol} describes the electrostatic interactions between the solute and the solvent, G_{vdW} describes the van der Waals interactions between the solute and the solvent, and G_{cav} describes the energy associated with the formation of a cavity for the solute in the solvent.

The generalized Born equation [13] is an approximate expression for the polarization energy, G_{pol}, for an arbitrarily complex molecule displaced in a dielectric medium. To derive the expression, first consider the electrostatic component of the change in free energy for the displacement of a charged spherical particle in a dielectric continuum. This energy is described by the Born equation [14]:

$$G_{\mathrm{pol}} = -166.0 \left(1 - \frac{1}{\epsilon}\right) \frac{q^2}{\alpha} \tag{9}$$

where ϵ is the dielectric constant of the continuum, and q and α are the charge and the radius of the particle, respectively. Combining equation (9) with Coulomb's law in a dielectric medium, an expression for the total electrostatic free energy (G_{es}) for a system of N widely separated particles is obtained:

$$G_{\mathrm{es}} = 332.0 \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{q_i q_j}{\epsilon\, r_{ij}} - 166.0 \left(1 - \frac{1}{\epsilon}\right) \sum_{i=1}^{N} \frac{q_i^2}{\alpha_i} \tag{10}$$

Equation (10) may be rewritten as:

$$\begin{aligned} G_{\mathrm{es}} = {} & 332.0 \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{q_i q_j}{r_{ij}} \\ & - 332.0 \left(1 - \frac{1}{\epsilon}\right) \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{q_i q_j}{r_{ij}} \\ & - 166.0 \left(1 - \frac{1}{\epsilon}\right) \sum_{i=1}^{N} \frac{q_i^2}{\alpha_i} \end{aligned} \tag{11}$$

where the sum of the second and third terms is equal to G_{pol}. The generalized Born equation [13] is obtained by combining the two terms:

$$G_{\mathrm{pol}} \approx G_{\mathrm{pol}}^{\mathrm{GB}} = -166.0 \left(1 - \frac{1}{\epsilon}\right) \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{q_i q_j}{f_{\mathrm{GB}}} \tag{12}$$

where f_{GB} is a function of α_i and r_{ij}. One widely used expression [5, 6, 11] for f_{GB} is:

$$f_{\mathrm{GB}} = \sqrt{r_{ij}^2 + \alpha_i \alpha_j \mathrm{e}^{-r_{ij}^2/4\alpha_i\alpha_j}} \tag{13}$$

with α_i being the so called Born radius of atom i. In the simple case of a spherical solute with a centered charge, as in equation (9), the Born radius is simply the same as the van der Waals radius of the solute. In the general case,

the Born radius of atom i depends on the distances between all other atoms in the molecule, and their respective volumes. One way to think about the Born radius is that it is not so much a radius but a kind of average distance to the solute-solvent boundary. Born radii are somewhat complicated to compute, and a number of different numerical and analytical approaches to facilitate these computations have been presented [6, 11].

2.5 Solvent-accessible surface area

The concept of solvent-accessible surface area (SASA) was first described by Lee and Richards in 1971 [15]. By representing a solvent molecule with a sphere, and rolling it over the van der Waals surface of a molecule, the SASA of the molecule is described by the locus of points swept out by the center of the sphere. This is equivalent to defining the SASA as the area of the surface made up of all points where the closest distance to the van der Waals surface of the molecule is exactly the radius of the solvent sphere. See figure 4.

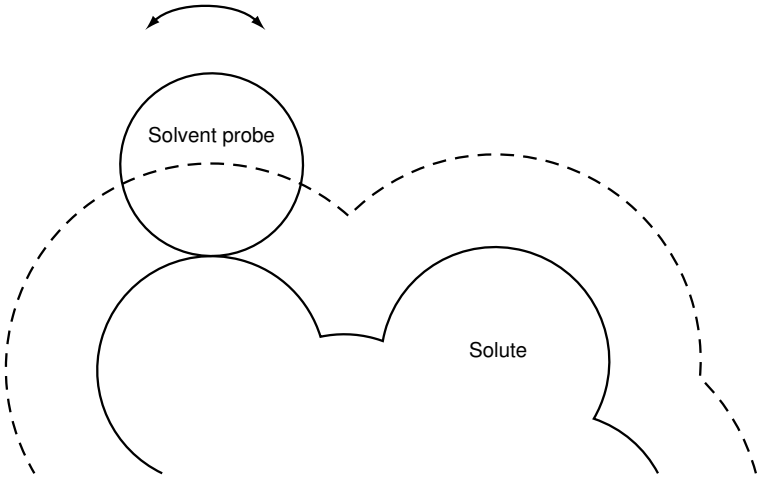

Figure 4: The solvent-accessible surface area can be described by rolling a spherical "solvent probe" over the van der Waals surface of the solute. Here, the concept is presented in two dimensions.

The current interest in SASAs is mainly related to the fact that the free energy of solvation, G_{sol}, for saturated hydrocarbons ($G_{\mathrm{pol}} \approx 0$) has been shown to be linearly related to the SASA of the molecule [16, 17]. This relationship has led to the widely used approximation [18] of $G_{\mathrm{vdW}} + G_{\mathrm{cav}}$ presented in equation (14):

$$G_{\mathrm{vdW}} + G_{\mathrm{cav}} = \sum_{i=1}^{N} \sigma_i \, \mathrm{SASA}_i \tag{14}$$

where SASA_i is the solvent-accessible surface area of atom i and σ_i is an atomic solvation parameter. Together with the GB expression for G_{pol} presented in section 2.4, this provides an approximate expression for the free energy of solvation for any molecule displaced in a dielectric medium.

3 Methods

3.1 *Q*

Q is a molecular dynamics program [3] developed by the Åqvist group at Uppsala University. The package is primarily intended for free energy perturbation (FEP) simulations, empirical valence bond (EVB) calculations of reaction free energies, and linear interaction energy (LIE) calculations of receptor-ligand binding affinities.

Q is written in Fortran 90, and runs on a number of different platforms. Free of charge for use in academic work, *Q* is available for download at the Åqvist group web site [19], both in the form of source code and executables for various operating systems.

3.2 Analytical Born radii

To be able to use the generalized Born equation (9) for calculations of solvation energies, Born radii must be computed. In this work, an analytical method developed by Qui *et al.* [6] was used to calculate Born radii. Values from the OPLS [20, 21] force field was used for van der Waals radii.

Figure 5 is useful in understanding the underlying principle. In a solute consisting of N atoms, we wish to calculate the Born radius of atom i, α_i, as illustrated in figure 5A. The idea is to compute an approximate solvation energy contribution for atom i, $G'_{\mathrm{pol},\,i}$, and then go "backwards" using equation 15 to calculate α_i:

$$\alpha_i \approx \frac{-166.0}{G'_{\mathrm{pol},\,i}} \tag{15}$$

Note that this assumes a unit charge on atom i and that the effect of ϵ has been neglected ($1 - \frac{1}{\epsilon} \approx 1$).

First, all atoms except atom i are temporarily removed from the system, with the result shown in figure 5B. The polarization energy for this simplified system would be given directly by the Born equation (9). Next, the effect of including atom j, see figure 5C, is calculated. The charge on atoms j is set to zero, and the only effect of including this atom is that it displaces the dielectric medium. The change in $G'_{\mathrm{pol},\,i}$ is proportional to the volume of the atom, V_j, and inversely proportional to the interatomic distance between atoms i and j raised to the fourth power, r_{ij}^4. By pairwise inclusion of all atoms $j \neq i$, $G'_{\mathrm{pol},\,i}$ could be calculated. Extending this method with scaling factors for neighboring atoms and a close contact function (f_{CC}) for non-bonded interactions, the equation for $G'_{\mathrm{pol},\,i}$ turns into:

$$G'_{\mathrm{pol},\,i} = \frac{-166.0}{R_{\mathrm{vdW},\,i} + \phi + P_1} + \sum_{1,2} \frac{P_2 V_j}{r_{ij}^4} + \sum_{1,3} \frac{P_3 V_j}{r_{ij}^4} + \sum_{1,\geq 4} \frac{P_4 V_j f_{\mathrm{CC}}}{r_{ij}^4} \tag{16}$$

where $R_{\mathrm{vdW},\,i}$ is the van der Waals radius of atom i, ϕ is the dielectric offset, P_1 is the single atom scaling factor, P_2 is the scaling factor for bonded atom pairs ($1,2$ interactions), P_3 is the scaling factor for atom pairs involved in angle bending ($1,3$) interactions, and P_4 is the scaling factor for non-bonded atoms

$(1, \geq 4$ interactions). The close contact function has the form:

$$\begin{aligned} f_{\mathrm{CC}} &= 1 \ , \ \text{if} \ \left(\frac{r_{ij}}{R_{\mathrm{vdW},\,i} + R_{\mathrm{vdW},\,j}} \right)^2 > \frac{1}{P_5} \\ &\text{otherwise} \\ f_{\mathrm{CC}} &= \frac{1}{4} \left\{ 1 - \cos \left[\left(\frac{r_{ij}}{R_{\mathrm{vdW},\,i} + R_{\mathrm{vdW},\,j}} \right)^2 P_5 \pi \right] \right\}^2 \end{aligned} \tag{17}$$

where P_5 is a soft cutoff parameter.

When calculating the atomic volume of atom j, the subvolumes that lie inside directly bonded atoms k are subtracted:

$$V_j = \frac{4\pi}{3} R_{\mathrm{vdW},\,j}^3 - \sum_k \frac{\pi}{3} h_{jk}^2 \left(3R_{\mathrm{vdW},\,j} - h_{jk} \right) \tag{18}$$

where

$$h_{jk} = R_{\mathrm{vdW},\,j} \left(1 + \frac{R_{\mathrm{vdW},\,k}^2 - R_{\mathrm{vdW},\,j}^2 - r_{jk}^2}{2R_{\mathrm{vdW},\,j}\, r_{jk}} \right) \tag{19}$$

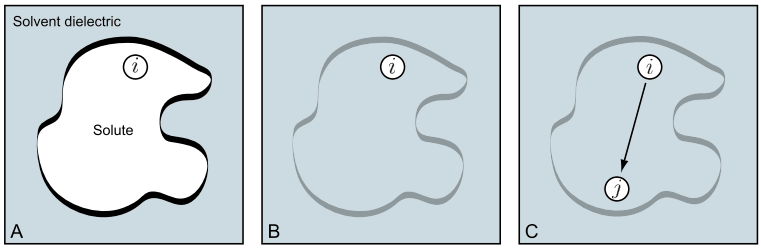

Figure 5: A: Whole solute. B: All atoms $j \neq i$ have been removed. C: Neutral "atom" j included, raising $G'_{\mathrm{pol},\,i}$.

3.3 Linear combination of pairwise overlaps

A sphere i, with a radius of R_i, has the surface area:

$$S_i = 4\pi R_i^2 \tag{20}$$

When this sphere is overlapping (only) with another sphere, j, the accessible surface area of sphere i, ASA_i, is simply:

$$\mathrm{ASA}_i = S_i - A_{ij} \tag{21}$$

where A_{ij} is the overlap of i with j, i.e. the area of i buried inside j. A_{ij} is given by:

$$A_{ij} = 2\pi R_i \left(R_i - \frac{r_{ij}}{2} - \frac{R_i^2 - R_j^2}{2r_{ij}} \right) \tag{22}$$

where r_{ij} is the Euclid distance between atoms i and j. Note that $A_{ij} \neq A_{ji}$ unless $R_i = R_j$.

When calculating the SASA in a model of a biomolecule, the number of atom spheres overlapping with atom i may be as many as 50 or more, many of them overlapping each other, and it is more or less impossible to analytically calculate the SASA without approximations. In this work, an analytical method called Linear Combination of Pairwise Overlaps (LCPO) [22] was used to calculate SASAs.

LCPO is force field independent in the sense that it has its own parameterization of atom types based on atom hybridization and number of bonded non-hydrogen atoms. Also, LCPO uses its own set of atomic radii, shown in table 1. Multiple overlaps are included by extending equation (21) with sums of pairwise overlaps, weighted with parameters for different atom types.

To begin with, define $N(i)$ as the neighbor list of atom i: $j \in N(i)$ if, and only if, i and j overlap. Then, the SASA of atom i, SASA_i, is calculated by LCPO as:

$$\text{SASA}_i = \beta_1 S_i + \beta_2 \sum_{j \in N(i)} A_{ij} + \beta_3 \sum_{\substack{j,\, k \in N(i) \\ k \in N(j) \\ k \neq j}} A_{jk} + \beta_4 \sum_{j \in N(i)} \left(A_{ij} \sum_{\substack{k \in N(i) \\ k \in N(j) \\ k \neq j}} A_{jk} \right) \quad (23)$$

In the LCPO equation (23), the first and second terms are simply equation (21) generalized for multiple atoms j overlapping with atom i, while the third and fourth terms are correction terms that take into account the overlaps of i's neighbors with each other. Each term is weighted by parameters β_1 to β_4, derived by applying multiple linear regression on data obtained from numerical calculations of the SASA for a large set of molecules.

Table 1: Atomic van der Waals radii in LCPO.

Element	Radius(Å)
C	1.70
N	1.65
O	1.60
P	1.90
S	1.90
Cl	1.80

3.4 Test compounds

The GB/SA method for calculating solvation energies was implemented in Q and the results from calculations on a set of small test molecules were compared to data from the similar implementation in Schrödinger's MacroModel® [23], to reported results for a similar implementations [5, 6], and to reported experimental values [6, 24]. The molecules are presented in table 2. These test molecules were simply chosen because there are both published G_{pol} (GB) and G_{sol} (GB/SA) results for them. A series of short MD simulations were performed on each molecule in Q to allow for relaxation ($\sim$ 360 fs) and conformational sampling ($\sim$ 5 ps). The SASA part of the implementation was also verified using a set of larger biomolecules, presented in table 3. These molecules were chosen as a subset of the molecules used in the original LCPO paper [22].

Table 2: The set of small molecules used to test the GB/SA implementation for calculating solvation energies.

Name	Formula	Number of atoms
Acetamide	CH_3CONH_2	9
Acetone	CH_3COCH_3	10
Methanol	CH_3OH	6
Phenol	C_6H_5OH	13
Thiophenol	C_6H_5SH	13

Table 3: The set of larger biomolecules used to test the LCPO implementation for calculating SASAs.

Name	Abbreviation	Number of atoms
3-chloro-4-hydroxybenzoic acid	CHB	11
7-chlorotetracycline	CTC	33
Crambin	1CRN	327
Human lysozyme	1LZ1	1029

4 Results and discussion

The results from the implementation of LCPO in *Q* were compared to the LCPO and numerical SASAs reported in the original LCPO paper [22]. The results from the SASA calculations are presented in table 4 and figures 6 and 7.

Table 4: Comparison of the results from the implementation of LCPO in *Q* to reported LCPO and numerical SASAs.

Compound	*Q* LCPO SASA/Å^2	Publ. [22] LCPO SASA/Å^2	Publ. [22] Numerical SASA/Å^2
CHB	293.0	289.8	315.9
CTC	621.8	606.3	608.4
1CRN	3088.7	3064.9	2976.3
1LZ1	6679.3	6681.9	6739.9

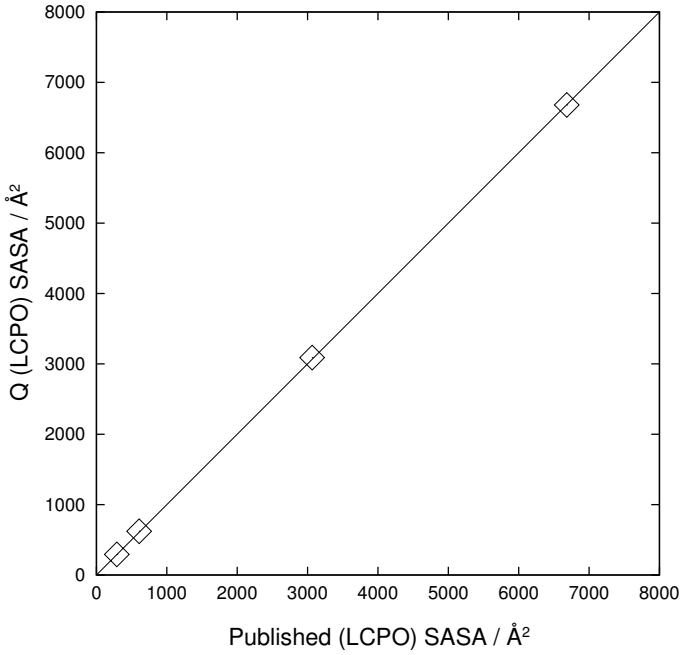

Figure 6: SASA values obtained from the the LCPO implementation in *Q* compared to published LCPO results [22].

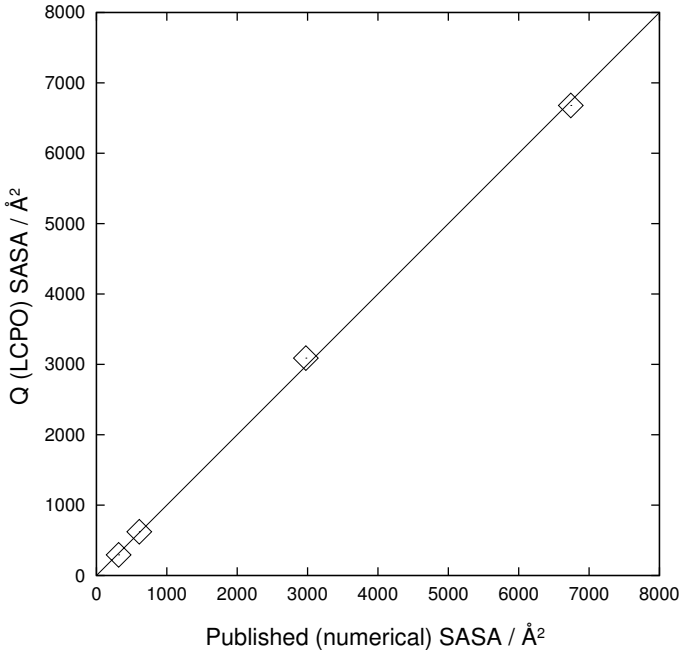

Figure 7: SASA values obtained from the the LCPO implementation in Q compared to published numerical results [22].

No significant differences were observed when comparing the two LCPO implementations – the average relative error is around 1%. Using molecular dynamics to generate a large number of conformations of a given test molecule, the difference in SASA between the individual conformations is bigger than the difference between their mean value and the reported LCPO SASA.

There were slightly greater differences in the G_{pol} results, shown in table 5 and figures 8 and 9, although the values are still very reasonable. Combining the G_{pol} and SASA calculations, using equations (8) and (14), yields the G_{sol} values presented in table 6 and figures 10 and 11.

Table 5: G_{pol} values obtained from the the GB implementations in Q and Schrödinger's MacroModel® compared to published values obtained from FEP calculations.

	Q	MacroModel [23]	Publ. [5]	Publ. [5]
	GB	GB	GB	FEP
	G_{pol}	G_{pol}	G_{pol}	G_{pol}
Compound	/kcal mol^{-1}	/kcal mol^{-1}	/kcal mol^{-1}	/kcal mol^{-1}
Acetamide	-11.9	-12.1	-11.4	-10.8 ± 0.5
Acetone	-4.3	-4.3	-5.7	-7.1 ± 0.3
Methanol	-7.6	-8.1	-7.1	-7.1 ± 0.3
Phenol	-7.9	-8.4	-7.7	-8.0 ± 0.7
Thiophenol	-4.7	-4.5	n/a	n/a

Table 6: G_{sol} values obtained from the the GB/SA implementations in Q and Schrödinger's MacroModel® compared to published GB/SA and experimental values.

	Q	MacroModel [23]	Publ. [6]	Publ. [6]
	GB/SA	GB/SA	GB/SA	Experimental
	G_{sol}	G_{sol}	G_{sol}	G_{sol}
Compound	/kcal mol^{-1}	/kcal mol^{-1}	/kcal mol^{-1}	/kcal mol^{-1}
Acetamide	-11.2	-11.1	-11.2	-9.7
Acetone	-3.0	-2.7	-2.7	-3.9
Methanol	-6.8	-7.2	-7.3	-5.1
Phenol	-6.2	-7.0	-6.4	-6.6
Thiophenol	-3.1	-2.5	-2.8	-2.6

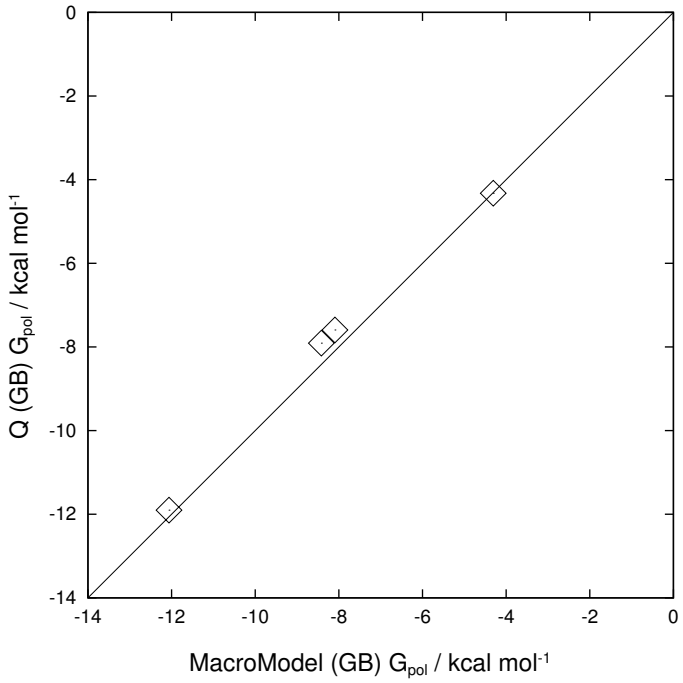

Figure 8: G_{pol} values obtained from the the GB implementation in Q compared to results from GB calculations in Schrödinger's MacroModel® [23].

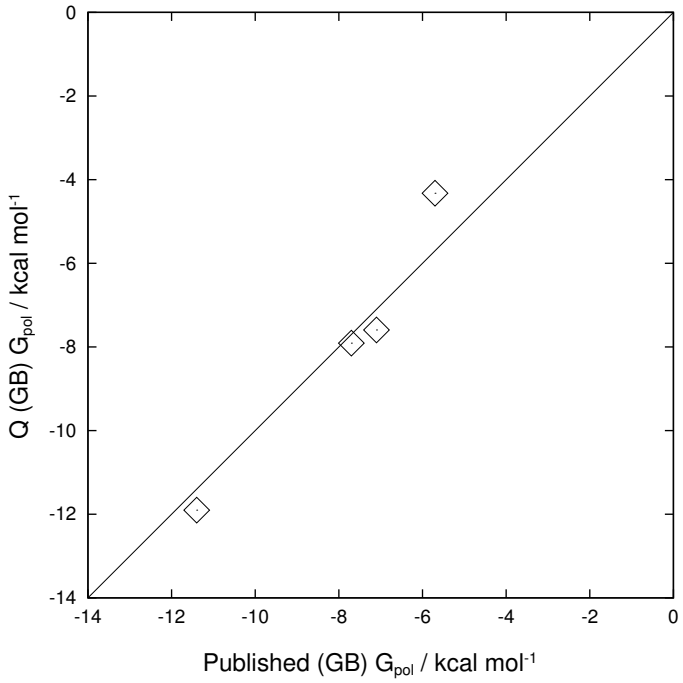

Figure 9: G_{pol} values obtained from the the GB implementation in Q compared to published GB results [5].

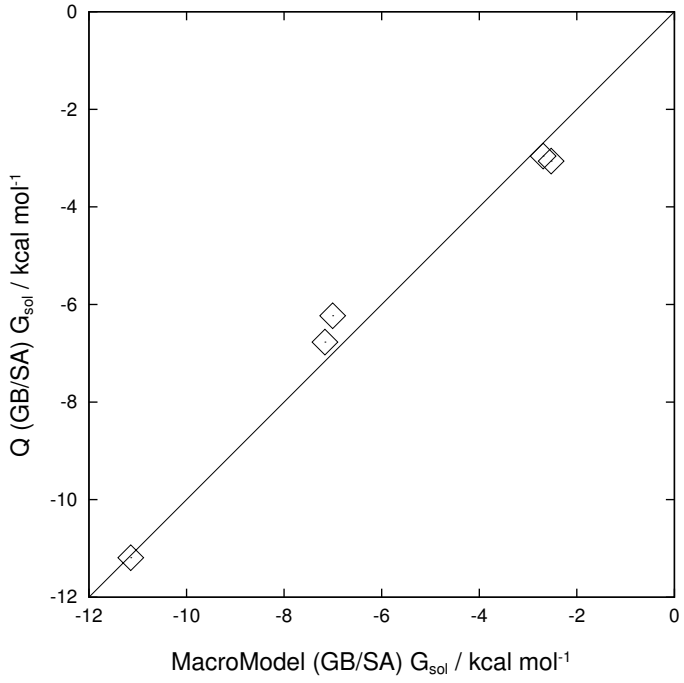

Figure 10: G_{sol} values obtained from the the GB/SA implementation in Q compared to results from GB/SA calculations in Schrödinger's MacroModel® [23].

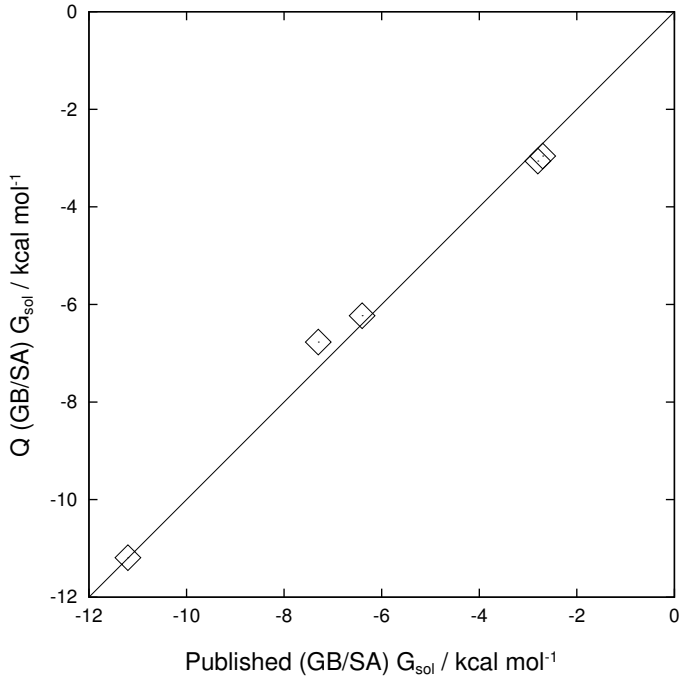

Figure 11: G_{sol} values obtained from the the GB/SA implementation in Q compared to published GB/SA results [6].

Overall, the results from Q are as good as the published GB/SA results and the results from Schrödinger's MacroModel [23]. The average unsigned error in G_{pol} (with the mean values of published FEP results [5] as reference) for Q was 1.12 kcal mol^{-1}, against 0.58 and 1.37 kcal mol^{-1} for the reported GB results [5] and MacroModel, respectively. For G_{sol}, the average unsigned error when comparing with the experimental values was 0.99 kcal mol^{-1} for Q versus 1.06 kcal mol^{-1} for the published GB/SA results [6] and 1.04 kcal mol^{-1} for MacroModel.

The differences in the results are likely due to small differences in the force field parameters used in the calculations. For example, there seems to be some confusion regarding which partial charges to assign the atoms. Some of the values used by default in MacroModel are for some reason not identical to the ones specified in the OPLS parameters file. Whenever there was a difference, the parameter values used by MacroModel were used in this work, since the data was readily available. It is reasonable to assume that the reported results [5, 6] originate from calculations in which some parameter values were different from those used in this work. Still, as illustrated by table 6 and figures 12 – 14, the results are very similar and the deviation from experimental G_{sol} values is about the same for the calculations in Q, MacroModel, and published results.

The relatively small number of molecules used to test the implementation in this work suggests that further studies may be needed to verify the robustness of the implementation. Also, the implementation will be adapted to be used in ligand binding simulations [8], as mentioned earlier. Hopefully, the method will provide fast, yet reasonably accurate, results in that application too.

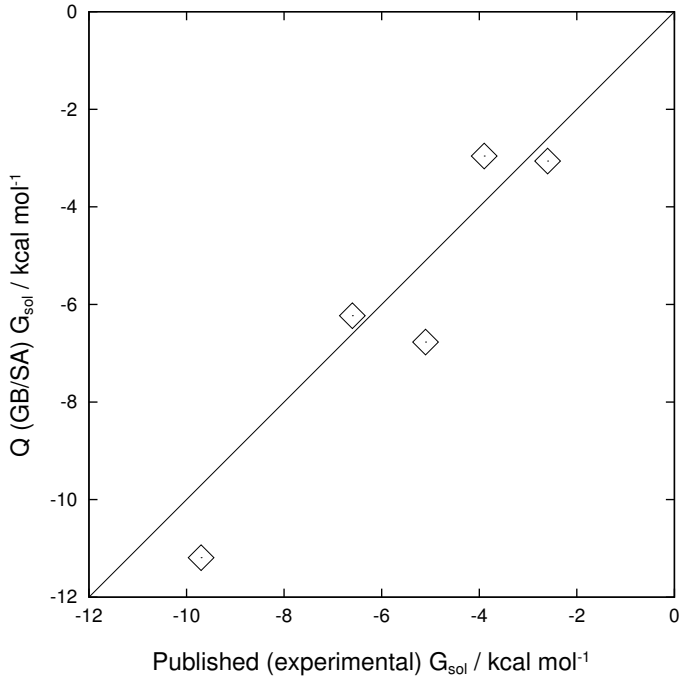

Figure 12: G_{sol} values obtained from the GB/SA implementation in Q compared to published experimental values [6].

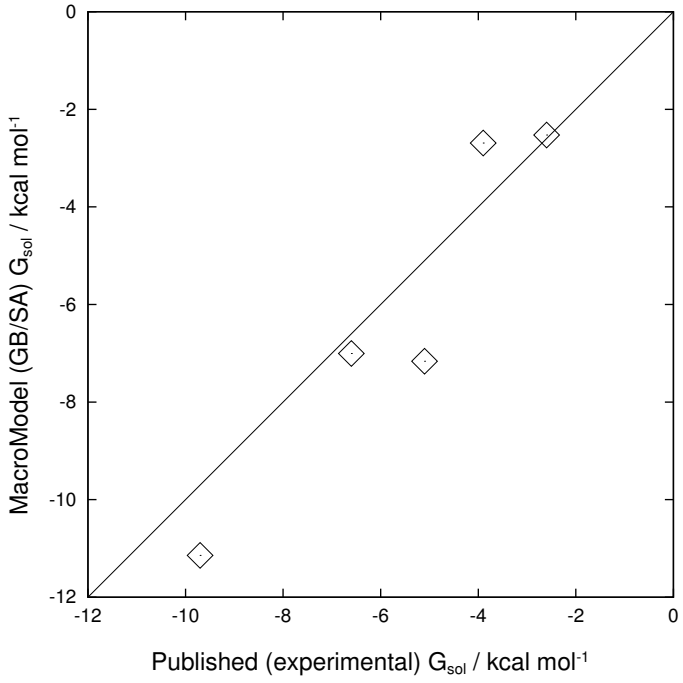

Figure 13: G_{sol} values obtained from the GB/SA implementation in Schrödinger's MacroModel® [23] compared to published experimental values [6].

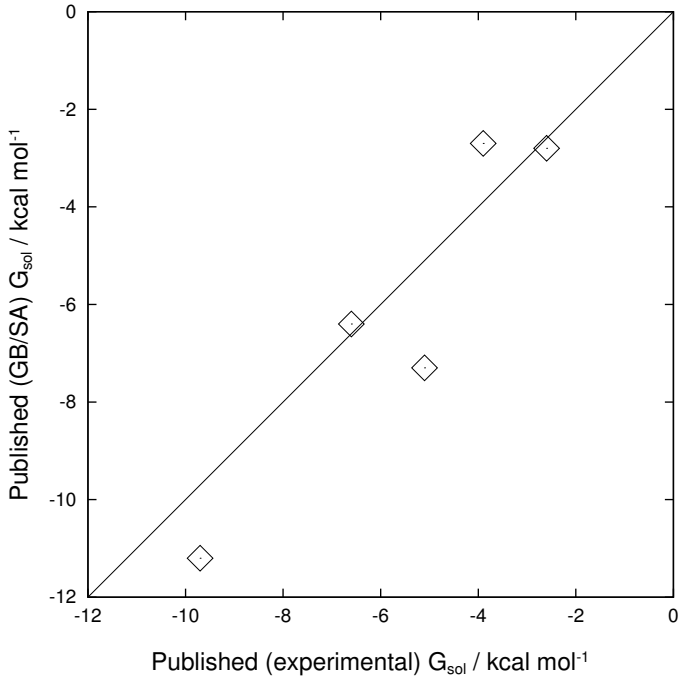

Figure 14: G_{sol} values obtained from a reported GB/SA implementation [6] compared to published experimental values [6].

5 Acknowledgements

First of all, I want to thank Johan Åqvist for giving me the opportunity to work in this great group, and for supervising this project. I also want to thank everybody in the Åqvist group for all the help, and all the fun. Thanks also to Anders Karlén for the help with MacroModel and for being the scientific reviewer for this project, and to Jörg Weiser and Peter Shenkin for the help with the LCPO implementation.

Last but not least, Celina, thank you for dragging me out of bed every morning. Without you, this project would have never been finished.

References

[1] D. Bashford and D. A. Case. Generalized Born models of macromolecular solvation effects. *Annu. Rev. Phys. Chem.*, 51:129–152, 2000.

[2] V. Tsui and D. A. Case. Theory and applications of the generalized Born solvation model in macromolecular simulations. *Biopolymers*, 56:275–291, 2001.

[3] J. Marelius, K. Kolmodin, I. Feierberg, and J. Åqvist. Q: A molecular dynamics program for free energy calculations and empirical valence bond simulations in biomolecular systems. *J. Mol. Graph. Mod.*, 16:213–225, 1998.

[4] M. Karplus and G. A. Petsko. Molecular dynamics simulations in biology. *Nature*, 347:631–639, 1990.

[5] W. C. Still, A. Tempczyk, R. C. Hawley, and T. Hendrickson. Semianalytical treatment of solvation for molecular mechanics and dynamics. *J. Am. Chem. Soc.*, 112:6127–6129, 1990.

[6] D. Qui, P. S. Shenkin, F. P. Hollinger, and W. C. Still. The GB/SA continuum model for solvation. A fast analytical method for the calculation of approximate Born radii. *J. Phys. Chem. A*, 101:3005–3014, 1997.

[7] T. P. Straatsma and McCammon J. A. Computational alchemy. *Annu. Rev. Phys. Chem.*, 43:407–435, 1992.

[8] R. Zhou, R. A. Fiesner, A. Ghosh, R. C. Rizzo, W. L. Jorgensen, and R. M. Levy. New linear interaction method for binding affinity calculations using a continuum solvent model. *J. Phys. Chem.*, 105:10388–10397, 2001.

[9] J. Åqvist, C. Medina, and J.E. Samuelsson. A new method for predicting binding affinity in computer-aided drug design. *Protein Eng.*, 7:385–391, 1994.

[10] R. W. Hockney. The potential calculation and some applications. *Meth. Comput. Phys.*, 9:136–211, 1970.

[11] G. D. Hawkins, C. J. Cramer, and D. G. Truhlar. Pairwise solute descreening of solute charges from a dielectric medium. *Chem. Phys. Lett.*, 246:122–129, 1995.

[12] G. D. Hawkins, C. J. Cramer, and D. G. Truhlar. Parameterized models of aqueous free energies of solvation based on pairwise descrening of solute atomic charges from a dielectric medium. *J. Phys. Chem*, 100:19824–19839, 1996.

[13] R. Constanciel and R. Contreras. Self consistent field theory of solvent effects representation by continuum models: Introduction of desolvation contribution. *Theor. Chim. Acta.*, 65:1–11, 1984.

[14] M. Z. Born. Volumen und Hydratationswarme der Ionen. *Z. Physik*, 1:45–48, 1920.

[15] B. Lee and F. M. Richards. The interpretation of protein structures: Estimation of static accessibility. *J. Mol. Biol.*, 55:379–400, 1971.

[16] R. B. Hermann. Theory of hydrophobic binding. II. The correlation of hydrocarbon solubility in water with solvent cavity surface area. *J. Phys. Chem.*, 76:2754–2759, 1972.

[17] G. L. Amidon, S. H. Yalkowsky, S. T. Anik, and S. C. Valvani. Solubility of nonelectrolytes in polar solvents. V. Estimation of the solubility of aliphatic monofunctional compounds in water using a molecular surface area approach. *J. Phys. Chem.*, 79:2239–2246, 1975.

[18] D. Eisenberg and A. D. McLachlan. Solvation energy in protein folding and binding. *Nature*, 319:199–203, 1986.

[19] http://xray.bmc.uu.se/∼aqwww/q/default.html. March 23, 2004.

[20] W. L. Jorgensen and J. Tirado-Rives. The OPLS potential functions for proteins. Energy minimizations for crystals of cyclic peptides and crambin. *J. Am. Chem. Soc.*, 110:1657–1666, 1988.

[21] G. Kaminski, E. M. Duffy, T. Matsui, and W. L. Jorgensen. Free energies of hydration and pure liquid properties of hydrocarbons from the OPLS all-atom model. *J. Phys. Chem.*, 98:13077–13082, 1994.

[22] J. Weiser, P. S. Shenkin, and W. C. Still. Approximate atomic surfaces from linear combination of pairwise overlaps (LCPO). *J. Comput. Chem.*, 20:217–230, 1999.

[23] http://www.schrodinger.com/Products/macromodel.html. March 23, 2004.

[24] S. Cabani, P. Gianni, V. Mollica, and L. Lepori. Group contributions to the thermodynamic properties of non-ionic organic solutes in dilute aqueous solution. *Solution Chem.*, 10:563–598, 1981.