BMC Biophysics

SOFTWARE Open Access

DelPhi: a comprehensive suite for DelPhi software and associated resources

Lin Li[1], Chuan Li[1], Subhra Sarkar[1,2], Jie Zhang[1,2], Shawn Witham[1], Zhe Zhang[1], Lin Wang[1], Nicholas Smith[1], Marharyta Petukh[1] and Emil Alexov[1*]

Abstract

Background: Accurate modeling of electrostatic potential and corresponding energies becomes increasingly important for understanding properties of biological macromolecules and their complexes. However, this is not an easy task due to the irregular shape of biological entities and the presence of water and mobile ions.

Results: Here we report a comprehensive suite for the well-known Poisson-Boltzmann solver, DelPhi, enriched with additional features to facilitate DelPhi usage. The suite allows for easy download of both DelPhi executable files and source code along with a makefile for local installations. The users can obtain the DelPhi manual and parameter files required for the corresponding investigation. Non-experienced researchers can download examples containing all necessary data to carry out DelPhi runs on a set of selected examples illustrating various DelPhi features and demonstrating DelPhi's accuracy against analytical solutions.

Conclusions: DelPhi suite offers not only the DelPhi executable and sources files, examples and parameter files, but also provides links to third party developed resources either utilizing DelPhi or providing plugins for DelPhi. In addition, the users and developers are offered a forum to share ideas, resolve issues, report bugs and seek help with respect to the DelPhi package. The resource is available free of charge for academic users from URL: http://compbio.clemson.edu/DelPhi.php.

Keywords: DelPhi, Poisson-Boltzmann equation, Implicit solvation model, Electrostatics, Biological macromolecules, Software

Background

Electrostatic interactions play an important role in biological systems [1-5] because biomolecules are composed of atoms carrying partial charges. Since the typical distances between atoms inside a biomolecule are on the order of several angstroms, the resulting electrostatic energy could be very large and be the major component of total energy [6-8]. Even more, at large distances, the electrostatic energy is the dominant component of the energy because all other components vanish. Since electrostatic interactions are the dominant factors for both inner- and inter- molecular interactions, accurate calculations of electrostatic potential and energies are crucial to reveal the mechanisms of many different biological phenomena, such as protein folding [9], protein-protein and protein-DNA binding [10-13], pKa shifts in proteins [14-17] and RNAs [18], and many others [19,20].

Biomolecules function in water, which makes the calculation of electrostatic potential a challenge due to the complexity of the water environment [21,22]. Various models have been developed to calculate electrostatic energy of biomolecules in the presence of surrounding water. These models can be categorized into two types [23-28]: explicit [29,30] and implicit [31-34] solvation models. Explicit solvation models treat the solvent as individual water molecules and are believed to be more accurate but time-consuming, and therefore, are usually suitable for systems involving small to medium size biomolecules. However, most biological systems are large and contain a huge amount of water molecules. To calculate electrostatics in such large systems, more computationally efficient algorithms are typically applied. These methods, called implicit solvation methods, treat the water phase as a continuum medium. The Poisson-

* Correspondence: ealexov@clemson.edu
[1]Physics Department, Computational Biophysics and Bioinformatics, Clemson University, Clemson, SC 29642, USA
Full list of author information is available at the end of the article

Boltzmann Equation (PBE) is one of the most successful implicit solvation models and was implemented in many well-known programs, such as DelPhi [35,36], APBS [37,38], MEAD [39], ZAP [40], PBEQ [41], MIBPB [42], UHBD [43], ITPACT [44], and several others. Among the above mentioned software, DelPhi has been proven to be among the best performers due to its unique features, such as capabilities of handling systems with multiple dielectric constants, modeling systems with multivalent ions, rapidly constructing molecular surfaces, calculating charged geometric object systems, and deriving ion concentration and dielectric maps.

In addition to the DelPhi source code and executable files which are free for academic users, several other resources are provided [45]. DelPhi's distribution is adapted for different operating systems and provides a series of examples along with the user manual. Parameter files for four of the most widely used force fields [46-51] are also provided on the DelPhi website, which gives the users more options to explore various scenarios and to port snap-shots from molecular dynamics simulations into DelPhi calculations. The DelPhi forum [52] is set up for users to exchange ideas, discuss and solve problems, post suggestions for further development, and other DelPhi related issues. Furthermore, the DelPhi web server is also developed [53], allowing non-experienced users to quickly perform calculations on their selected structures. The DelPhi suite offers various tools and plugins [54-56] developed by other researches which utilize DelPhi to address biological questions as well.

Implementation

The PBE model treats solvent as a continuum medium with high dielectric constant. Biomolecules are considered as low dielectric cavities made of charged atoms. Ions in the water phase are modeled as non-interacting point charges and their distribution obeys the Boltzmann law. Utilizing the Gauss-Seidel method, DelPhi solves both linear and nonlinear PBE in a cube of $N \times N \times N$ grid points [57,58].

The overall architecture of DelPhi is shown in the flowchart in Figure 1. The core of DelPhi are the subroutines described below. The DelPhi specific input parameters are provided in a parameter file and the input data is read from three input files: coordinates, charges and radii files. First of all, user-desired parameters are specified in a parameter file. This parameter file controls the initial set up of the run and provides the file names of the coordinate, charge and size files. One can specify the scale and filling percentage. DelPhi automatically calculates the necessary grid size, given the scale and filling percentage. According to the coordinate and size files, DelPhi generates the molecular surface by utilizing a rapid construction method. Dielectric constant values, which form a three dimensional dielectric constant map, are given at grid midpoints. Next, the charges are assigned to atoms and then distributed onto the grid points. Using distributed charges and the dielectric constant map, DelPhi initiates the iterations to solve the linear or nonlinear PBE. Iterations stop when the user specified tolerance or the maximal number of iterations is achieved. DelPhi then produces the three dimensional electrostatic potential map. Using this potential map, DelPhi can also generate the ion concentration map. If requested, the dielectric constant, electrostatic potential, and ion concentration maps can be saved into files and rendered by visualization software. Using the charge distribution and potential map, the Coulombic, grid and solvation energies are calculated.

In addition to the routines described above, DelPhi also has some unique functionalities, such as handling multiple dielectric constants and mixed multivalent ions, rapid constructing molecular surface, generating geometric objects and performing calculations. The multiple dielectric constant method divides the bio-molecular system into different parts, and assigns each part a specific dielectric constant allowing the difference in conformational flexibility to be modeled by different dielectric constants as illustrated in Ref. [59]. DelPhi can also model solvents with mixed ions, which may have different concentrations and valences. In order to speed up the process of generating a molecular surface, a rapid surface construction method has been developed in DelPhi, which implements the marching cube algorithm to construct the surface quickly and accurately. These features are described in detail in [35]. Four types of basic objects are now available in DelPhi package: sphere, cylinder, cone and box. Using the object functions, together with the multiple dielectric constants option, users can create complex geometric structures with different dielectric constants and shapes [35]. The DelPhi package is written in the FORTRAN and C language and can be compiled as single or double precision according to the practical usage. DelPhi version 5.1 is now available on different operating systems, including Windows, Linux and Mac. Although many useful functions and options have been implemented, DelPhi is still user friendly and compatible. Several other groups have developed third-party plugins and tools to utilize DelPhi on other software, such as UCSF Chimera [56], DelEnsembleElec (GUI and a plugin for VMD [55]), Biskit [54], and others [60].

Results and discussion

There are several important characteristics used to classify methods and software packages: accuracy, rate of convergence and speed of calculations. In the next

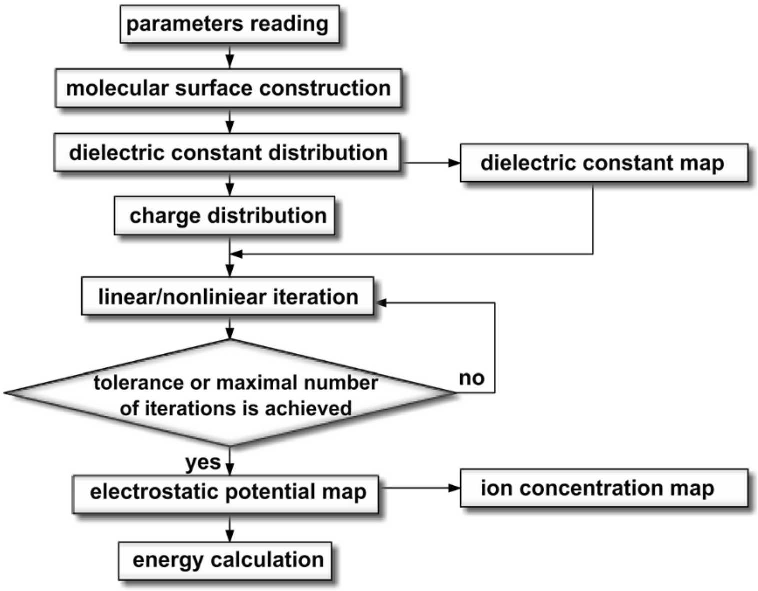
parameters reading
molecular surface construction
dielectric constant distribution
dielectric constant map
charge distribution
linear/nonliniear iteration
tolerance or maximal number of iterations is achieved
no
yes
electrostatic potential map
ion concentration map
energy calculation

Figure 1 Flowchart of the DelPhi program.

several subsections, we performed several tests on DelPhi with respect to these features.

Accuracy test

Accuracy is one major concern of any numerical solver. In order to measure the solver's accuracy and demonstrate that it solves the exact problem, the solver is usually tested on simple examples for which analytical solutions exist. In this subsection, three simple examples with regular geometry were selected. We compared their analytical solutions with the numerical ones obtained by DelPhi. Since the computational algorithm does not distinguish simple geometrical objects from real biological macromolecules with more complex shape, it indicates that DelPhi solves the PB equation and produces close numerical approximations to the real solutions. The following constants were fixed in all examples: elementary charge $e = 1.602176565 \times 10^{-19}C$, vacuum permittivity $\varepsilon_0 = 8.8541878176 \times 10^{-12}F/m$, Boltzmann constant $k = 1.38 \times 10^{-23}J/K$ and temperature $T = 297.33K$.

A sphere in water

The first example presents a charged atom with a lower dielectric constant ϵ_{int} immersed in a continuum media with a higher dielectric constant ϵ_{ext}. In this example, the electrostatic component of solvation energy ΔG^{sol} can be obtained by the Born formula and is explicitly given by

$$\Delta G^{sol} = -\frac{Q^2}{2\cdot 4\cdot \pi\cdot \varepsilon_0}\cdot\frac{1}{r}\left(\frac{1}{\varepsilon_{\text{int}}}-\frac{1}{\varepsilon_{ext}}\right), \qquad (1)$$

where Q and r are charge and radius of the charged atom, as shown in Figure 2A.

Setting $\varepsilon_{int} = 4.0$, $\varepsilon_{ext} = 80.0$ and $Q = 10.e$, values of ΔG^{sol} obtained by Equation (1) are −6673.71kT, -3336.86kT and −2224.57kT (rounded to two decimals) for radii $r = 1$ Å, 2 Å and 3 Å, respectively. These values were compared to those obtained by DelPhi at various scales (points/Å) and the results are shown in Figure 2B. It is clear that no visual difference can be observed when the scale is greater than 0.5 points/Å.

Two charges in a protein

The next example, shown in Figure 3A, describes a spherical protein with radius b locating inside a media without any ions. Dielectric constants in the interior and exterior of the protein are denoted by ε_{int} and ε_{ext} again. Two atoms with radii r_i and r_j are centered at points with polar coordinates (R_i, θ_i) and (R_j, θ_j). These two atoms are placed inside the protein and are assigned charges Q_i and Q_j, respectively. This example has been studied by Barry Honig and co-workers [1]. The analytical solution of the electrostatic component of solvation energy ΔG is composed of four terms:

$$\Delta G = \Delta G^{c}_{ij} + \Delta G^{pol}_{ij} + \Delta G^{self}_{ii} + \Delta G^{self}_{jj}, \qquad (2)$$

where ΔG^{c}_{ij} is the Coulombic energy of atom *i*, *j*, ΔG^{pol}_{ij} is the pairwise polarization interaction energy, ΔG^{self}_{ii} is the total self-energy of atom i and ΔG^{self}_{jj} is

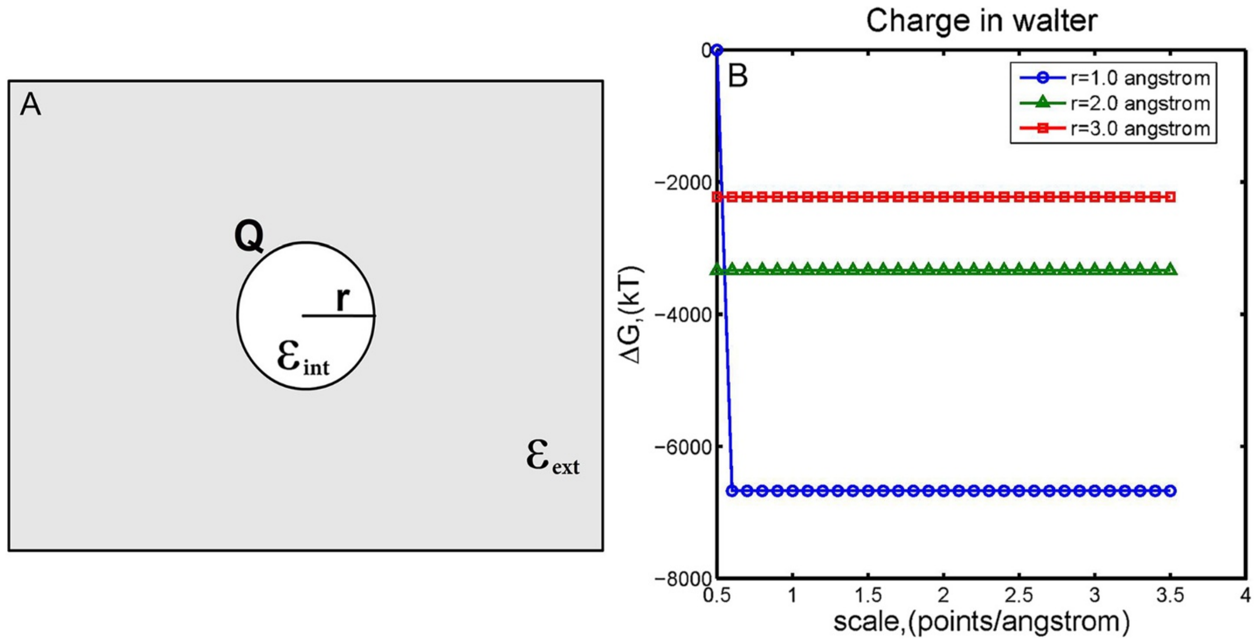

Figure 2 **(A) Schematic illustration of example 1: A charged sphere with low dielectric constant is inside a media with a high dielectric constant.** (**B**) Electrostatic component of the solvation energies calculated by DelPhi against the analytical solution.

the total self-energy of atom *j*. Energies on the right-hand side of Equation (2) can be calculated by

$$\begin{aligned}
\Delta G_{ij}^{c} &= \frac{1}{4\cdot\pi\cdot\varepsilon_0}\cdot\frac{Q_i\cdot Q_j}{\varepsilon_{\text{int}}\cdot R_{ij}},\\
\Delta G_{ij}^{pol} &= \frac{1}{4\cdot\pi\cdot\varepsilon_0}\cdot\frac{Q_i\cdot Q_j}{\varepsilon_{\text{int}}}\cdot\sum_{n=0}^{\infty}B_{n,ij}\cdot P_n\left(\cos\left(\theta_i-\theta_j\right)\right),\\
\Delta G_{ii}^{self} &= \frac{1}{4\cdot\pi\cdot\varepsilon_0}\cdot\frac{Q_i^2}{2\cdot\varepsilon_{\text{int}}}\cdot\sum_{n=0}^{\infty}B_{n,ii},
\end{aligned}\tag{3}$$

where $B_{n,ij} = \frac{\left(R_i\cdot R_j\right)^n}{b^{2\cdot n+1}}\cdot\frac{(n+1)\cdot\left(\varepsilon_{\text{int}}-\varepsilon_{ext}\right)}{(n+1)\cdot\varepsilon_{ext}+n\cdot\varepsilon_{\text{int}}}$ and $P_n\left(\cos\left(\theta_i-\theta_j\right)\right)$ is the nth order Legendre polynomial.

Substituting $Q_i = Q_j = 10\cdot e$, $R_i = R_j = 5\sqrt{2}$ Å, $\theta_i = \pi/4$, $\theta_j = 3\pi/4$, $b = 10$ Å, $r = 1$ Å, $\varepsilon_{\text{int}} = 2.0$ and $\varepsilon_{ext} = 80.0$ into Equations (2) – (3) yields $\Delta G = -5083.19$ kT after rounding to two decimals. Numerical calculations were performed at grid size = 85, 125, and 165 and various scales. The numerical results, together with the value of ΔG, were compared and shown in Figure 3B. One can see that the numerical solutions converge to the real solution quickly as scale increases for all three tested grid sizes.

A sphere in semi-infinite dielectric region

The third example considers a space split into two regions with different dielectric constants, as shown in Figure 4A. Dielectric constant in the left region is ϵ_1 and that in the right region is ε_2 ($\varepsilon_2 > \varepsilon_1$). A sphere with radius r and dielectric constant ϵ_1 is initially positioned in the right region. The distance between the center of the sphere and the boundary of two regions is denoted by d. Let the sphere move towards the boundary and eventually get into the left region. We consider $d > 0$ when the center of the sphere is still in the right region and $d < 0$ when it is in the left region. The sign of d indicates the position of the sphere. During the moving process of the sphere, except the moment when the sphere intersects both regions (i.e., $|d| \leq r$), the electrostatic component of the solvation energy ΔG can be analytically expressed as a function of distance d

$$\Delta G = \begin{cases} \frac{1}{4\cdot\pi\cdot\varepsilon_0}\cdot\frac{Q^2}{2\cdot r}\cdot\left(\frac{1}{\varepsilon_2}-\frac{1}{\varepsilon_1}\right)+\frac{1}{4\cdot\pi\cdot\varepsilon_0}\cdot\frac{\varepsilon_2-\varepsilon_1}{\varepsilon_2+\varepsilon_1}\cdot\frac{Q^2}{4\cdot\varepsilon_2\cdot d}, & \text{when } d > r,\\ \frac{1}{4\cdot\pi\cdot\varepsilon_0}\cdot\frac{\varepsilon_2-\varepsilon_1}{\varepsilon_1+\varepsilon_2}\cdot\frac{Q^2}{4\cdot\varepsilon_1\cdot d}, & \text{when } d < -r. \end{cases}\tag{4}$$

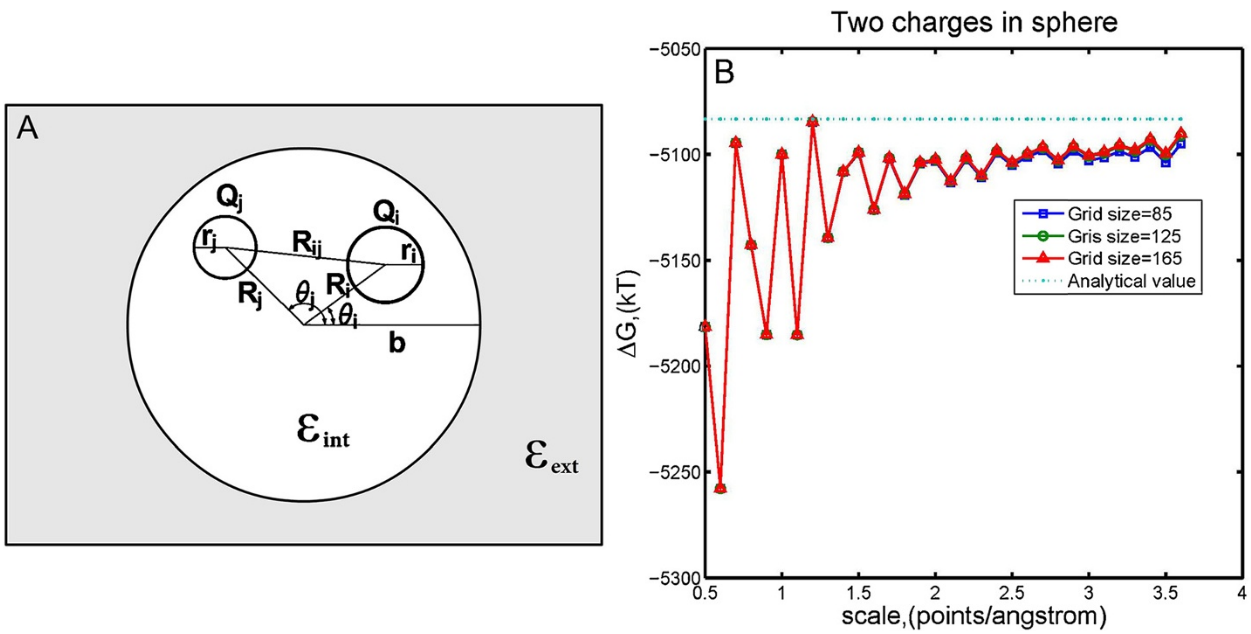

Figure 3 (A) Schematic illustration of example 2: a cavity with low dielectric constant is inside a media with high dielectric constant. Two charged atoms are located inside the cavity. (**B**) Electrostatic component of solvation energy obtained from DelPhi compared with analytical solution.

The blue curve in Figure 4B represents the function $\Delta G(d)$(Equation 4), here we set $\varepsilon_2 = 80.0$, $\varepsilon_1 = 2.0$, $r = 2$ Å, $Q = 1 \cdot e$. Numerical results obtained by running DelPhi at a series of discrete d values are shown by red circles in Figure 4B and fit the curve very well. Our tests in this example indicate that DelPhi delivers accurate numerical approximations to the real solution.

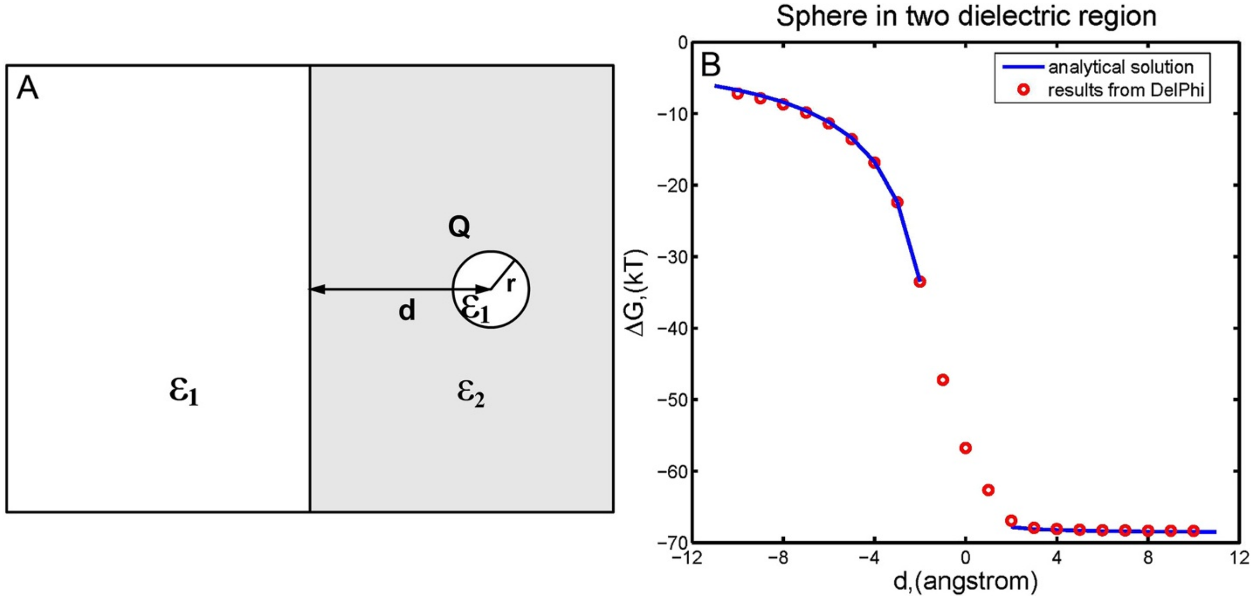

Figure 4 (A) Schematic illustration of example3: the space is divided by media with two different dielectric constants. A sphere with low dielectric constant ε_1 is initially positioned in region with high dielectric constant ε_2 and moves into the region with low dielectric constant ε_1. (**B**) Electrostatic component of solvation energy derived from DelPhi compared with analytical solutions. Blue solid lines represent analytical solutions, red circles represent numerical results from DelPhi.

Rate of convergence

The rate of convergence is another major concern from the numerical point of view. DelPhi utilizes the Gauss-Seidel iteration method, along with the optimized Successive Over-Relaxation method [58], to solve PBE in a cube. The solution is more accurate when the cube is discretized into finer grids. In order to determine the minimal requirement of computational time cost for DelPhi to achieve results within a desired accuracy, a series of tests were designed and implemented on a typical protein of medium size, namely the bovine alpha-chymotrypsin-eglin C complex [PDB:1ACB], to demonstrate the performance of DelPhi.

These tests were performed by varying the value of scale from 0.5 points/Å to 6.5 points/Å at step size 0.1 points/Å. Noticing that scale is the reciprocal of grid spacing, this means larger scale results in finer discretization of the cube. The filling percentage of the cube, *perfil* = 70%, was fixed in all tests. The resulting electrostatic component of solvation energy ΔG as a function of scale is shown in Figure 5.

The energy calculations on the structure of 1ACB show that the approximate scale threshold is 1 points/Å. At scale larger than the threshold, the calculated electrostatic component of solvation energy is almost scale-independent and reaches steady value of −28089 kT. Achieving such steady value at scale 1 – 2 points/Å demonstrates the robustness of the algorithm that calculates the electrostatic component of solvation energy, so termed the corrected reaction field energy method [35].

Speed of calculations

DelPhi utilizes various algorithms and modules to calculate electrostatic potential and energy. The basic modules include generating molecular surface, calculating electrostatic potential distribution and obtaining the corresponding electrostatic energy. The speed of calculations for each of these modules depends on various factors, such as scale, number of atoms/charges, shape/net charge of the molecule. In order to reveal their impact on the performance of DelPhi from the users' point of view, we first tested DelPhi on a particular protein complex with fixed filling of the cube and increasing scale, and next, tested DelPhi on multiple proteins with fixed scale. All calculations were performed on the same type of CPU, Intel Xeon E5410 (2.33 G Hz), on the Palmetto cluster [61] at Clemson University. Each run was repeated 5 times and the average is reported here in order to reduce unexpected fluctuations caused by system workload at run time. The resulting CPU time against scale and protein size, are reported as follows.

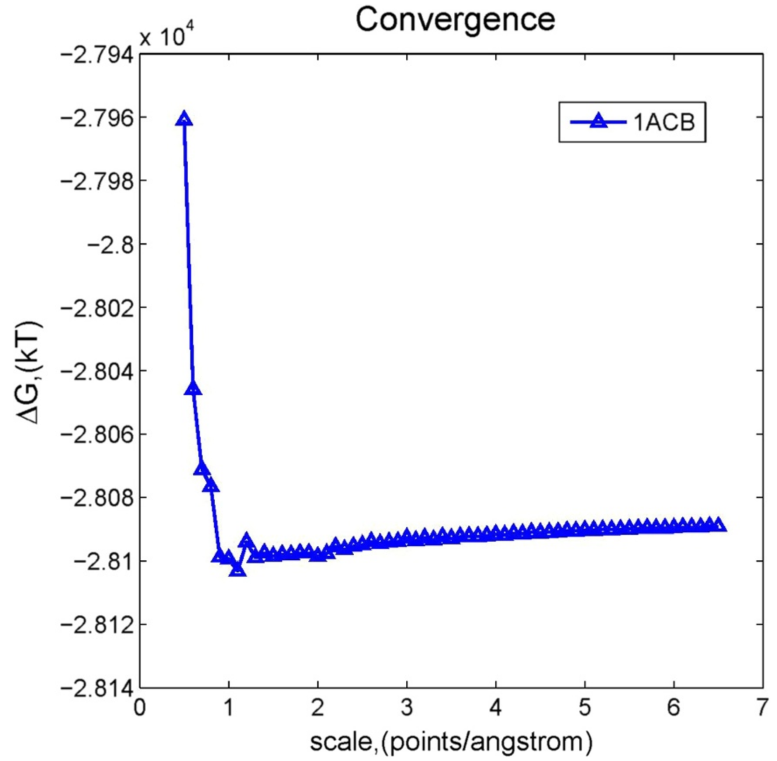

Figure 5 Electrostatic solvation energies of 1ACB obtained from DelPhi with respect to different scale values.

Speed of calculations as a function of scale

In this case, we calculated the energy of *barstar* [PDB:1A19] using DelPhi, with scale values increasing from 0.5 to 10 points/Å at step size 0.1 points/Å. The *perfil* value was set to 70% regardless of the changing scales to keep the filling of the cube fixed. The resulting CPU time, plotted as a function of scale, is shown in Figure 6. It can be seen that the computational time rapidly increases with the scale, because the corresponding grid size increases as well. However, at scale of 2–4 points/Å. DelPhi is still very fast, resulting in runs of about a second to several seconds.

Speed of calculations as a function of protein size

To evaluate the influence of protein size, 200 proteins from Zhang's benchmark [62] were selected and tested using DelPhi. The number of atoms in these proteins range from 639 to 16361. For each protein, the scale was set to be 2 points/Å and *perfil* was 70%, which are reasonable and common values for calculations on real biomolecules. The resulting calculation time is plotted as a function of protein size in Figure 7A. One can see that the resulting computational time, in general, increases with the size of the protein. However, the energy of the largest protein in the dataset, composed of more than 16,000 atoms, was calculated in less than 120 s, almost the same time needed for small proteins made of about 8,000 atoms. This indicates that other factors, such as the number of charged groups, may play important roles as well. To test such a possibility, the number of charged residues for each protein was obtained and the calculation time is plotted against it (Figure 7B). The resulting plot is not much different from Figure 7A and the calculation time for the protein with the largest number of charged groups is not necessarily the longest one. This illustrates that the computation time is a complex function depending on the combination of protein size, number of charges, shape and many others.

Electrostatic energy calculations on large biomolecules usually cost more CPU time primarily due to two factors: Firstly, large biomolecules need a large cube and consequently more grids to be represented. Secondly, larger biomolecules contain more charged atoms and require more time to calculate the energy terms. However, the curve of the 200 proteins is not smooth, because there are several other factors which influence the calculation time. The size of the modeling cube depends not only on the atom number, but also on the molecule's shape. A narrow and long molecule may need a larger cube than a spherical molecule even if their atom numbers are the same. The irregularity of molecular surface

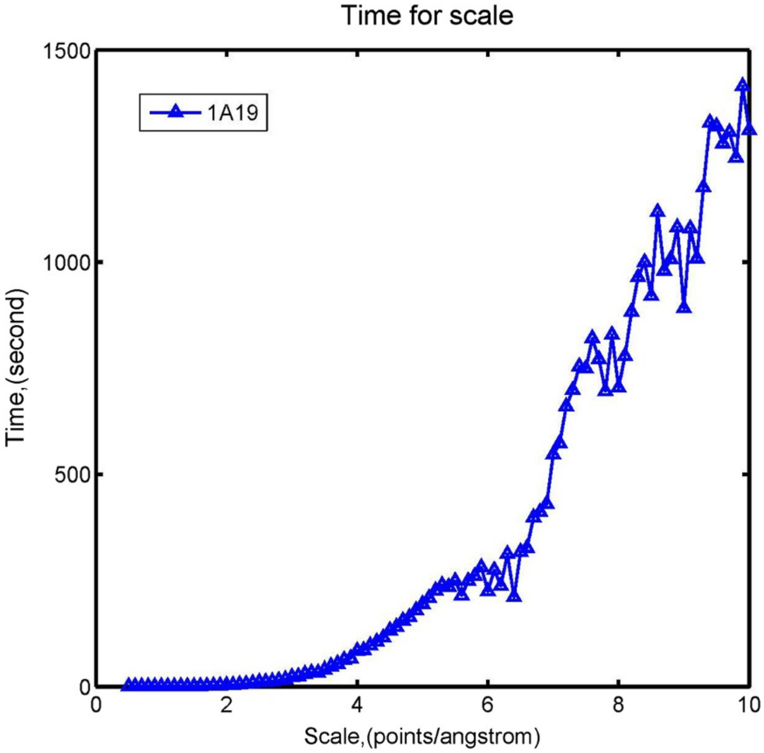

Figure 6 Electrostatic solvation energies calculation time of 1A19 with respect to different scale values.

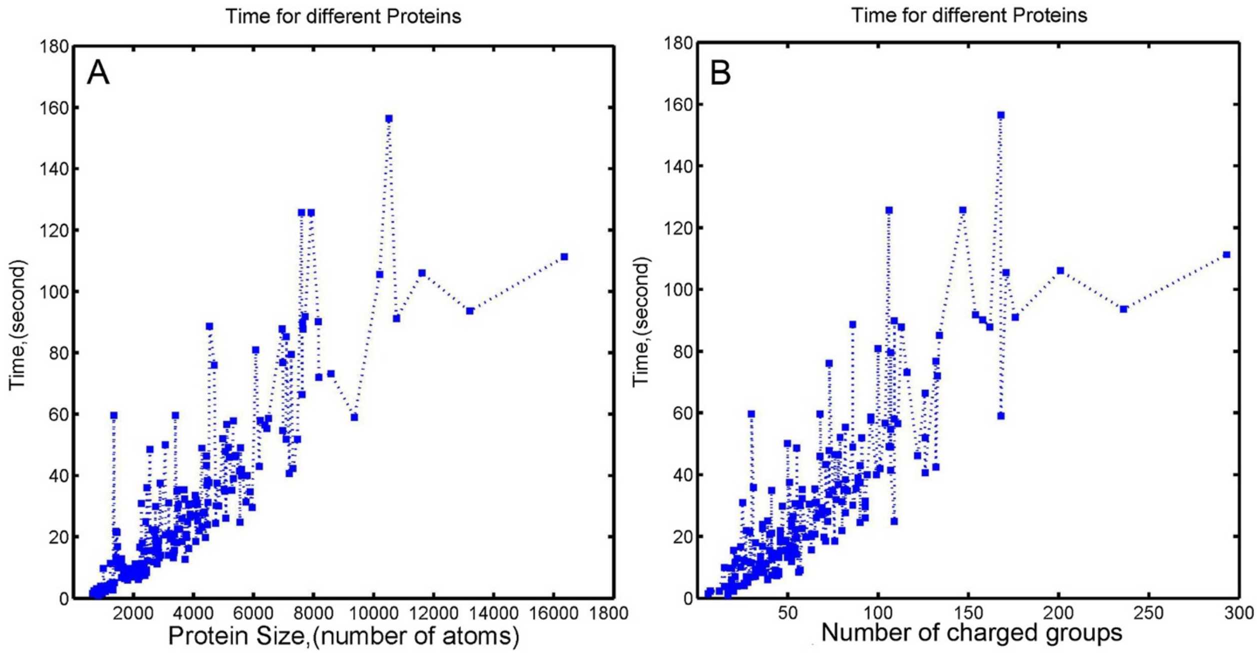

Figure 7 Electrostatic solvation energies calculation time of 200 proteins with respect to (A) number of atoms and (B) number of charged groups of each protein.

also affects the iteration time. A molecule with an irregular surface requires more iterations to converge than a molecule with a regular, smooth surface. Finally, a molecule with a higher charge needs more iterations than a molecule with a lower charge. Due to above reasons, larger molecules usually (but not necessarily) cost more time than smaller molecules to calculate the corresponding potential and energies.

Effect of force field parameters (Charmm, Amber, OPLS and Parse)

It is well known that different force fields perform differently in protein folding [63-65]. It was also illustrated that the electrostatic component of binding free energy is very sensitive to force fields [66]. Because of that, it is desirable that DelPhi handles electrostatic calculations with different force fields on 3D structures obtained from the corresponding MD simulations. Currently, four widely used force fields are available in DelPhi package: AMBER98 [46], CHARMM22 [47], OPLS [48-50] and PARSE [51]. Here we calculated the electrostatic component of solvation energy of HIV-1 protease [PDB:1HVC] by using the above mentioned force fields (Figure 8). The *perfil* was set to be 70%, probe radius was 1.4 Å, the dielectric constants were set as 4.0 inside the protein and as 80.0 in the water, and the scales varied from 0.5 to 6.0 points/Å.

Results of the calculated electrostatic energies on 1HVC are shown in Figure 8, using AMBER, CHARMM, OPLS, and PARSE. The results reaffirm the previously made observations that calculations at very small scales are not accurate. However, once the scale is larger than 1 point/Å, DelPhi achieves convergence quickly and results are almost scale independent. There is a slight tendency that CHARMM and PARSE converge faster than other force fields, but the difference is small. At the same time, the electrostatic energies calculated using different force fields are quite different. When scale reaches 6 points/Å, the calculated energies are: -25109.58, -21474.99, -20471.78, -19234.74 kT for AMBER, CHARMM, OPLS, and PARSE, respectively. The largest difference in calculated energy is obtained by AMBER force field parameters versus others. Such a large difference should not be surprising since the force field parameters are developed with respect to the total energy, not just the electrostatic component. However, several studies [16,67,68] indicate that the energy difference remains even in the calculations of total energy, although the differences are smaller compared to the differences in the electrostatic component. The same is valid for calculations involving the difference of energies, as for example the electrostatic component of the binding energy [66]. It was shown [66] that the difference could be larger than 50 kcal/mol. These observations and the results presented in this work indicate the sensitivity of calculations with respect to the force field parameters and suggest that the outcome of the modeling should be tested with this regard.

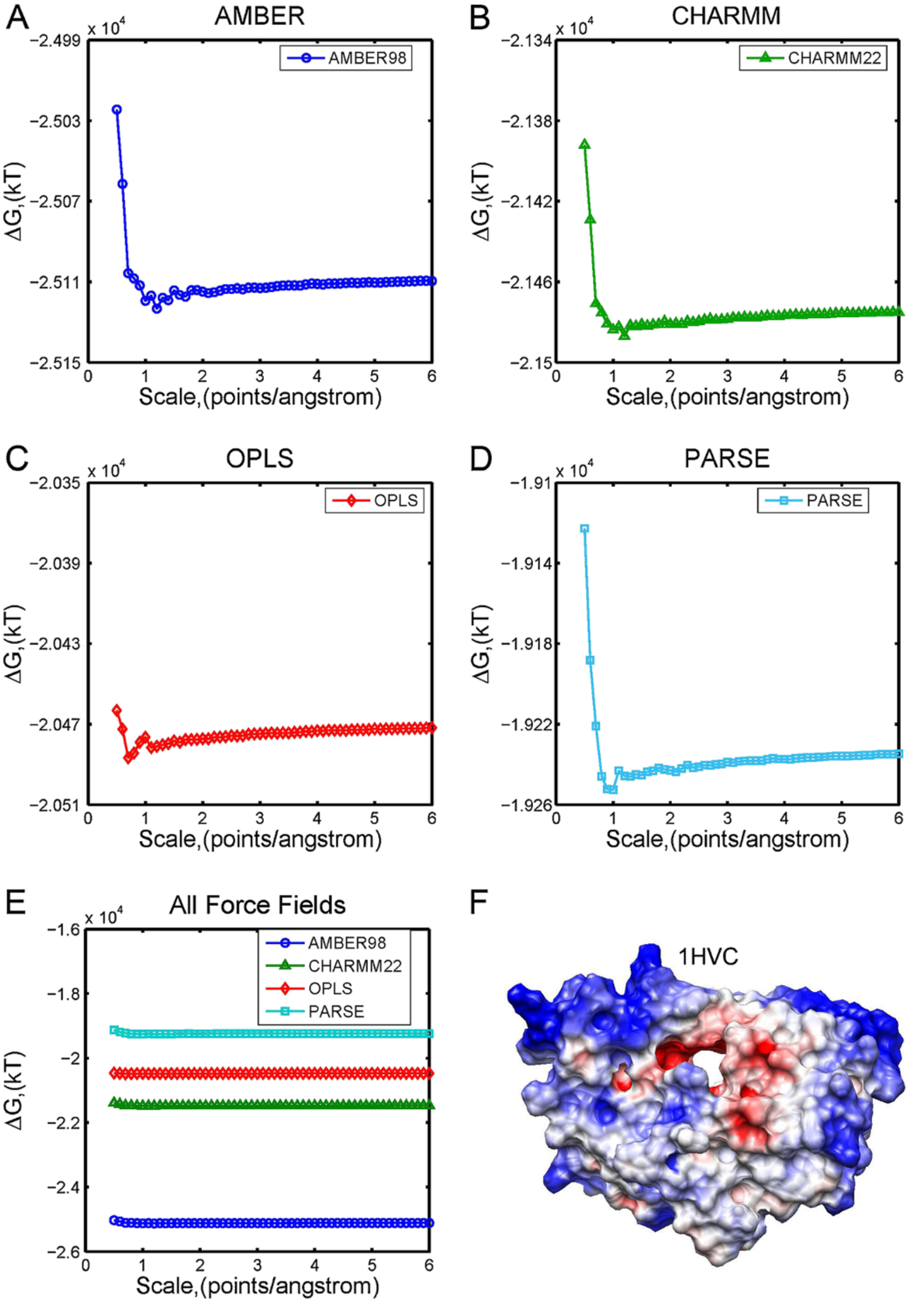

Figure 8 Electrostatic solvation energy calculation of 1HVC using 4 different force fields. (**A**)-(**D**) Calculated electrostatic solvation energies using AMBER, CHARMM, OPLS, and PARSE force fields. (**E**) All of the 4 results are shown in one figure to show the differences. (**F**) Electrostatic potential surface of HIV-1 protease [PDB:1HVC], generated by using AMBER force field.

Conclusions

In this work, we described the DelPhi package and associated resources. DelPhi is a comprehensive suite including DelPhi website, web server, forum, DelPhi software and other tools. Several tests were performed on DelPhi in this work to demonstrate DelPhi's capabilities in terms of accuracy, rate of convergence and speed of calculations. It was shown that DelPhi is a robust solver and capable of solving various biological applications. The benchmarks confirmed that DelPhi delivers energies that are almost grid independent, reaches convergence at scales equal to or larger than 1–2 grids/Å, and the speed of calculations is impressively fast. Finally, as shown in comparison with analytical solutions, the algorithm is, most importantly, capable of providing accurate energy calculations.

Availability and requirements

Project name: DelPhi
Project home page: e.g. http://compbio.clemson.edu/delphi.php
Operating system(s): Linux, Mac, Windows
Programming language: Fortran and C
Other requirements: no
License: free of charge license is required
Any restrictions to use by non-academics: Commercial users should contact Accelrys Inc.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

LL analyzed the data, drafted the manuscript and maintains DelPhi package. CL maintains the DelPhi package and helped writing the manuscript. SS, JZ, SW, ZZ, LW and MP developed and maintain DelPhi web server and website and associated resources. EA supervised DelPhi development and maintenance and finally draft the manuscript. All authors read and approved the final manuscript.

Acknowledgements

The work was supported by a grant from the Institute of General Medical Sciences, National Institutes of Health, award number 1R01GM093937.

Author details

[1]Physics Department, Computational Biophysics and Bioinformatics, Clemson University, Clemson, SC 29642, USA. [2]Department of Computer Science, Clemson University, Clemson, SC 29642, USA.

Received: 10 February 2012 Accepted: 17 April 2012
Published: 14 May 2012

References

1. Gilson MK, Rashin A, Fine R, Honig B: **On the calculation of electrostatic interactions in proteins.** *J Mol Biol* 1985, **184:**503–516.
2. Honig B, Nicholls A: **Classical electrostatics in biology and chemistry.** *Science* 1995, **268:**1144.
3. Russell S, Warshel A: **Calculations of electrostatic energies in proteins* 1: The energetics of ionized groups in bovine pancreatic trypsin inhibitor.** *J Mol Biol* 1985, **185:**389–404.
4. Zhang Z, Witham S, Alexov E: **On the role of electrostatics in protein–protein interactions.** *Phys Biol* 2011, **8:**035001.
5. Sharp KA, Honig B: **Electrostatic interactions in macromolecules: theory and applications.** *Annu Rev Biophys Biophys Chem* 1990, **19:**301–332.
6. Guest WC, Cashman NR, Plotkin SS: **Electrostatics in the stability and misfolding of the prion protein: salt bridges, self energy, and solvation.** *Biochem Cell Biol* 2010, **88:**371–381.
7. Laederach A, Shcherbakova I, Jonikas MA, Altman RB, Brenowitz M: **Distinct contribution of electrostatics, initial conformational ensemble, and macromolecular stability in RNA folding.** *Proc Natl Acad Sci U S A* 2007, **104:**7045–7050.
8. Avbelj F, Fele L: **Role of main-chain electrostatics, hydrophobic effect and side-chain conformational entropy in determining the secondary structure of proteins.** *J Mol Biol* 1998, **279:**665–684.
9. Yang AS, Honig B: **On the pH dependence of protein stability.** *J Mol Biol* 1993, **231:**459–474.
10. Bertonati C, Honig B, Alexov E: **Poisson-Boltzmann calculations of nonspecific salt effects on protein-protein binding free energies.** *Biophys J* 2007, **92:**1891–1899.
11. Jensen JH: **Calculating pH and salt dependence of protein-protein binding.** *Curr Pharm Biotechnol* 2008, **9:**96–102.
12. Spencer DS, Xu K, Logan TM, Zhou HX: **Effects of pH, salt, and macromolecular crowding on the stability of FK506-binding protein: an integrated experimental and theoretical study.** *J Mol Biol* 2005, **351:**219–232.
13. Talley K, Kundrotas P, Alexov E: **Modeling salt dependence of protein-protein association: Linear vs non-linear Poisson-Boltzmann equation.** *Commun Comput Phys* 2008, **3:**1071–1086.
14. Yang AS, Gunner MR, Sampogna R, Sharp K, Honig B: **On the calculation of pKas in proteins.** *Proteins* 1993, **15:**252–265.
15. Georgescu RE, Alexov EG, Gunner MR: **Combining conformational flexibility and continuum electrostatics for calculating pK(a)s in proteins.** *Biophys J* 2002, **83:**1731–1748.
16. Zhang Z, Teng S, Wang L, Schwartz CE, Alexov E: **Computational analysis of missense mutations causing Snyder-Robinson syndrome.** *Hum Mutat* 2010, **31:**1043–1049.
17. Witham S, Talley K, Wang L, Zhang Z, Sarkar S, Gao D, Yang W, Alexov E: **Developing hybrid approaches to predict pKa values of ionizable groups.** *Proteins: Structure, Function, and Bioinformatics* 2011, **79:**3260–3275.
18. Tang CL, Alexov E, Pyle AM, Honig B: **Calculation of pK(a)s in RNA: On the structural origins and functional roles of protonated nucleotides.** *J Mol Biol* 2007, **366:**1475–1496.
19. Mitra RC, Zhang Z, Alexov E: **In silico modeling of pH-optimum of protein-protein binding.** *Proteins-Structure Function and Bioinformatics* 2011, **79:**925–936.
20. Alexov E: **Numerical calculations of the pH of maximal protein stability. The effect of the sequence composition and three-dimensional structure.** *Eur J Biochem* 2004, **271:**173–185.
21. Harvey SC: **Treatment of electrostatic effects in macromolecular modeling.** *Proteins* 1989, **5:**78–92.
22. Lebard DN, Matyushov DV: **Protein-water electrostatics and principles of bioenergetics.** *Phys Chem Chem Phys* 2010, **12:**15335–15348.
23. Ma B, Nussinov R: **Explicit and implicit water simulations of a beta-hairpin peptide.** *Proteins* 1999, **37:**73–87.
24. Zhou R: **Free energy landscape of protein folding in water: explicit vs. implicit solvent.** *Proteins* 2003, **53:**148–161.
25. Spaeth JR, Kevrekidis IG, Panagiotopoulos AZ: **A comparison of implicit- and explicit-solvent simulations of self-assembly in block copolymer and solute systems.** *J Chem Phys* 2011, **134:**164902.
26. Tan C, Yang L, Luo R: **How well does Poisson-Boltzmann implicit solvent agree with explicit solvent? A quantitative analysis.** *Journal of Physical Chemistry B* 2006, **110:**18680–18687.
27. Rod TH, Rydberg P, Ryde U: **Implicit versus explicit solvent in free energy calculations of enzyme catalysis: Methyl transfer catalyzed by catechol O-methyltransferase.** *J Chem Phys* 2006, **124:**174503.
28. Pham TT, Schiller UD, Prakash JR, Dunweg B: **Implicit and explicit solvent models for the simulation of a single polymer chain in solution: Lattice Boltzmann versus Brownian dynamics.** *J Chem Phys* 2009, **131:**164114.
29. Druchok M, Vlachy V, Dill KA: **Explicit-water molecular dynamics study of a short-chain 3,3 ionene in solutions with sodium halides.** *J Chem Phys* 2009, **130:**134903.
30. Kony DB, Damm W, Stoll S, van Gunsteren WF, Hunenberger PH: **Explicit-solvent molecular dynamics simulations of the polysaccharide schizophyllan in water.** *Biophys J* 2007, **93:**442–455.
31. Baker NA: **Poisson-Boltzmann methods for biomolecular electrostatics.** *Methods Enzymol* 2004, **383:**94–118.
32. Gilson MK, Honig B: **Calculation of the total electrostatic energy of a macromolecular system: solvation energies, binding energies, and conformational analysis.** *Proteins* 1988, **4:**7–18.
33. Lee MC, Yang R, Duan Y: **Comparison between Generalized-Born and Poisson-Boltzmann methods in physics-based scoring functions for protein structure prediction.** *J Mol Model* 2005, **12:**101–110.
34. Grochowski P, Trylska J: **Continuum molecular electrostatics, salt effects, and counterion binding–a review of the Poisson-Boltzmann theory and its modifications.** *Biopolymers* 2008, **89:**93–113.
35. Rocchia W, Sridharan S, Nicholls A, Alexov E, Chiabrera A, Honig B: **Rapid grid-based construction of the molecular surface and the use of induced surface charge to calculate reaction field energies: Applications to the molecular systems and geometric objects.** *J Comput Chem* 2002, **23:**128–137.
36. Rocchia W, Alexov E, Honig B: **Extending the applicability of the nonlinear Poisson-Boltzmann equation: Multiple dielectric constants and multivalent ions.** *Journal of Physical Chemistry B* 2001, **105:**6507–6514.
37. Holst M, Baker N, Wang F: **Adaptive multilevel finite element solution of the Poisson–Boltzmann equation I. Algorithms and examples.** *J Comput Chem* 2000, **21:**1319–1342.

38. Baker N, Holst M, Wang F: Adaptive multilevel finite element solution of the Poisson–Boltzmann equation II. Refinement at solvent-accessible surfaces in biomolecular systems. *J Comput Chem* 2000, **21**:1343–1352.
39. Bashford D: An object-oriented programming suite for electrostatic effects in biological molecules An experience report on the MEAD project. *Springer* 1997, **233**:240.
40. Grant JA, Pickup BT, Nicholls A: A smooth permittivity function for Poisson–Boltzmann solvation methods. *J Comput Chem* 2001, **22**:608–640.
41. Banavali NK, Roux B: Atomic radii for continuum electrostatics calculations on nucleic acids. *The Journal of Physical Chemistry B* 2002, **106**:11026–11035.
42. Zhou Y, Feig M, Wei G: Highly accurate biomolecular electrostatics in continuum dielectric environments. *J Comput Chem* 2008, **29**:87–97.
43. Davis ME, McCammon JA: Solving the finite difference linearized Poisson-Boltzmann equation: A comparison of relaxation and conjugate gradient methods. *J Comput Chem* 1989, **10**:386–391.
44. Cortis CM, Friesner RA: Numerical solution of the Poisson-Boltzmann equation using tetrahedral finite-element meshes. *J Comput Chem* 1997, **18**:1591–1608.
45. DelPhi Website, http://compbio.clemson.edu/delphi.php.
46. Ponder JW, Case DA: Force fields for protein simulations. *Advances in protein chemistry* 2003, **66**:27–85.
47. Brooks BR, Brooks C III, Mackerell A Jr, Nilsson L, Petrella R, Roux B, Won Y, Archontis G, Bartels C, Boresch S: CHARMM: the biomolecular simulation program. *J Comput Chem* 2009, **30**:1545–1614.
48. Kahn K, Bruice TC: Parameterization of OPLS–AA force field for the conformational analysis of macrocyclic polyketides. *J Comput Chem* 2002, **23**:977–996.
49. Kony D, Damm W, Stoll S, Van Gunsteren W: An improved OPLS–AA force field for carbohydrates. *J Comput Chem* 2002, **23**:1416–1429.
50. Xu Z, Luo HH, Tieleman DP: Modifying the OPLS-AA force field to improve hydration free energies for several amino acid side chains using new atomic charges and an off-plane charge model for aromatic residues. *J Comput Chem* 2007, **28**:689–697.
51. Sitkoff D, Lockhart DJ, Sharp KA, Honig B: Calculation of electrostatic effects at the amino terminus of an alpha helix. *Biophys J* 1994, **67**:2251–2260.
52. DelPhi Forum, http://compbio.clemson.edu/forum/index.php.
53. DelPhi Web Server, http://compbio.clemson.edu/sapp/delphi_webserver/.
54. Grünberg R, Nilges M, Leckner J: Biskit—a software platform for structural bioinformatics. *Bioinformatics* 2007, **23**:769.
55. Humphrey W, Dalke A, Schulten K: VMD: visual molecular dynamics. *J Mol Graph* 1996, **14**:33–38.
56. Pettersen EF, Goddard TD, Huang CC, Couch GS, Greenblatt DM, Meng EC, Ferrin TE: UCSF Chimera—a visualization system for exploratory research and analysis. *J Comput Chem* 2004, **25**:1605–1612.
57. Klapper I, Hagstrom R, Fine R, Sharp K, Honig B: Focusing of electric fields in the active site of Cu-Zn superoxide dismutase: Effects of ionic strength and amino-acid modification. *Proteins: Structure, Function, and Bioinformatics* 1986, **1**:47–59.
58. Nicholls A, Honig B: A rapid finite difference algorithm, utilizing successive over-relaxation to solve the Poisson–Boltzmann equation. *J Comput Chem* 1991, **12**:435–445.
59. Wang L, Zhang Z, Rocchia W, Alexov E: Using DelPhi Capabilities to Mimic Protein's Conformational Reorganization with Amino Acid Specific Dielectric Constants. *Comm Comp Phys* 2012, in press.
60. DelPhi Tools, http://compbio.clemson.edu/delphi_tools.php.
61. Palmettol Cluster, http://citi.clemson.edu/training_palm.
62. Zhang Y, Skolnick J: TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res* 2005, **33**:2302–2309.
63. Patapati KK, Glykos NM: Three Force Fields' Views of the 310 Helix. *Biophys J* 2011, **101**:1766–1771.
64. Yoda T, Sugita Y, Okamoto Y: Comparisons of force fields for proteins by generalized-ensemble simulations. *Chem Phys Lett* 2004, **386**:460–467.
65. Matthes D, De Groot BL: Secondary structure propensities in peptide folding simulations: a systematic comparison of molecular mechanics interaction schemes. *Biophys J* 2009, **97**:599–608.
66. Talley K, Ng C, Shoppell M, Kundrotas P, Alexov E: On the electrostatic component of protein-protein binding free energy. *PMC Biophys* 2008, **1**:2.
67. Zhang Z, Norris J, Schwartz C, Alexov E: In silico and in vitro investigations of the mutability of disease-causing missense mutation sites in spermine synthase. *PLoS One* 2011, **6**:e20373.
68. Witham S, Takano K, Schwartz C, Alexov E: A missense mutation in CLIC2 associated with intellectual disability is predicted by in silico modeling to affect protein stability and dynamics. *Proteins: Structure, Function, and Bioinformatics* 2011, **79**:2444–2454.

doi:10.1186/2046-1682-5-9
Cite this article as: Li *et al.*: DelPhi: a comprehensive suite for DelPhi software and associated resources. *BMC Biophysics* 2012 **5**:9.