Journal of Computational Physics 229 (2010) 4059–4074

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A new numerical method for nonlocal electrostatics in biomolecular simulations

S. Weggler [a,*], V. Rutka [b], A. Hildebrandt [a]

[a] Centrum for Bioinformatics, 66123 Saarbrücken, Germany
[b] Department of Mathematics and Physics, Universität Hannover, 30167 Hannover, Germany

ARTICLE INFO

Article history:
Received 4 May 2009
Received in revised form 22 December 2009
Accepted 31 January 2010
Available online 13 February 2010

Keywords:
Biomolecules
Nonlocal electrostatics
Transmission problem
Boundary element method
Immersed interface method

ABSTRACT

The electrostatic behavior of biomolecules solved in water can be described by an elliptic system of partial differential equations for the potential. In previous studies, this system has been solved by the Boundary Element Method (BEM).

In this paper, we apply the Explicit Jump Immersed Interface Method (EJIIM) as an alternative method to the BEM. Such a finite difference approach allows for a completely automatized software for analyzing biomolecules in their natural surrounding.

The new method shows excellent agreement with the BEM results and has good convergence properties and runtimes. In addition, in contrast to the BEM, where the fundamental solutions of operators are necessary, the EJIIM approach can be easily extended to more complex, especially nonlinear models.

1. Introduction

Protein–protein, protein–DNA, and protein–ligand interactions are fundamental to many processes in molecular biology. The latter is the basis for drug development and therefore, it is one of the most important, but also one of the most expensive, fields of research of our time. To reduce the costs as well as the time needed for testing novel drug candidates, effort is made to understand the basic functionalities of molecular interactions. Beside others, the electrostatic interaction energy of molecules is a crucial contribution to the total energy. Moreover, it is important to understand these interactions in the natural solvents of the biomolecules. Here, the basic question is: how does the solvent with its response to the biomolecule's charges influence the electric field and therefore the interaction of the biomolecules between each other?

By molecular dynamic simulations, solvent effects can be studied well, since the solvent molecules can be taken into account explicitly (see [27,48,41]). Another, less time consuming, approach is to develop a macroscopic model, indirectly incorporating the reaction of different media by the so called polarization field, $\boldsymbol{P}$:

$$\nabla \times \boldsymbol{E}(\boldsymbol{r}) = 0, \quad \boldsymbol{r} \in \mathcal{R}, \tag{1a}$$

$$\varepsilon_0 \nabla \boldsymbol{E}(\boldsymbol{r}) = \rho - \nabla \boldsymbol{P}(\boldsymbol{r}), \quad \boldsymbol{r} \in \mathcal{R}, \tag{1b}$$

where ε_0 is the dielectric constant of vacuum, see [16]. To account for the reaction of the medium, the *dielectric operator* ε is introduced. This operator acts on the electric field $\boldsymbol{E}$ as a filter, resulting in a *dielectric field* $\boldsymbol{D}$ caused by the external charges ρ only. Then, (1b) in a region $\mathcal{R}$ with dielectric response ε reads

* Corresponding author.
E-mail addresses: sophie@bioinf.uni-sb.de (S. Weggler), rutka@maphy.uni-hannover.de (V. Rutka), anhi@bioinf.uni-sb.de (A. Hildebrandt).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.01.040

$$\nabla \boldsymbol{D}(\boldsymbol{r}) = \rho, \quad \boldsymbol{r} \in \mathcal{R}, \tag{2a}$$

$$\boldsymbol{D}(\boldsymbol{r}) = \varepsilon_0 \boldsymbol{E}(\boldsymbol{r}) + \boldsymbol{P}(\boldsymbol{r}) := \varepsilon_0 \varepsilon(\boldsymbol{E}(\boldsymbol{r})), \quad \boldsymbol{r} \in \mathcal{R}. \tag{2b}$$

Please note that in order to solve the Maxwell equations suitable boundary conditions are required. The simplest linear operator usually taken to measure the dielectric response is a multiplicative factor

$$\boldsymbol{D}(\boldsymbol{r}) = \varepsilon_0 \varepsilon(\boldsymbol{r}) \boldsymbol{E}(\boldsymbol{r}), \quad \boldsymbol{r} \in \mathcal{R}. \tag{3}$$

In the following, this ansatz is called the *local* theory. Eq. (1a) can be directly fulfilled by a gradient ansatz, $\boldsymbol{E} = -\nabla\phi$, where the scalar field ϕ is the so called electrostatic potential.

Water is the main solvent for biomolecules and thus the solvent we will focus on. However, the local ansatz given in (3) does not suffice to describe this complex, dipolar fluid. One aspect neglected in (3) are nonlinearities in the dielectric response, which can easily become substantial, see [2,18,31]. A second effect neglected in (3) is the fact that, apart from the dipole-dipole correlations, the water molecules build up a network formed by their hydrogen bonds, i.e., the solvent molecules are trying to align with the neighbouring molecules as well as with the external charges.

In general, it is often argued that for proteins, the nonlinear effect should be comparatively small in magnitude, since the charge distribution usually consists of partial charges and is smeared out considerably. Thus we assume that a linear response of the medium is valid. In the following, we focus on the correlations of the water molecules by a linear operator ε: in [8], Dogonadze et al. denoted such a linear dielectric operator ε, whose reaction to the medium at position $\boldsymbol{r} \in \mathbb{R}^3$ depends *non-locally* on every other position $\boldsymbol{r}'$:

$$\boldsymbol{D}(\boldsymbol{r}) = \varepsilon_0 \int_{\mathbb{R}^3} \varepsilon(\boldsymbol{r}, \boldsymbol{r}') \boldsymbol{E}(\boldsymbol{r}') d\boldsymbol{r}'. \tag{4}$$

A well-founded model of the nonlocal dielectric operator is motivated in [8]. It reads

$$\varepsilon(\boldsymbol{r}, \boldsymbol{r}') = \varepsilon_\infty \delta(\boldsymbol{r} - \boldsymbol{r}') + \gamma^2 \frac{1}{4\pi} \frac{e^{-|\boldsymbol{r}-\boldsymbol{r}'|\kappa^2}}{|\boldsymbol{r} - \boldsymbol{r}'|}, \tag{5}$$

where $\gamma^2 = (\varepsilon_\Sigma - \varepsilon_\infty)\kappa^2 > 0$ and δ is the Dirac delta function.

In (5), several parameters are introduced to describe the nonlocal effect: the dimensionless parameter ε_∞ defines the dielectric response in the limit of increasing wavenumbers measuring the variations of the electric field, see [38,12]. The second material parameter is κ, where $1/\kappa$ defines the length scale at which solvent molecules interact with each other by their hydrogen bonds. The local dielectric model – a constant response of strength ε_Σ – is recovered in the limiting process $\kappa \to \infty$. More details on the validity can be found in [13].

In previous studies, first steps have been taken to model nonlocal electrostatics of biomolecules in water to get an idea of the relevance of this complex solvent for the functionality of biomolecules [38,40,7]. The resulting integro-differential equations can be solved [9], but they require extremely high computational resources, making an extensive research difficult.

Within certain approximations, Hildebrandt et al. have been able to turn the set of equations into a set of differential equations [14]. These approximations especially include a potential formulation of the dielectric field. We introduce the scalar dielectric potential field ψ and assume in the following

$$\boldsymbol{E} = -\nabla\phi \quad \text{and} \quad \boldsymbol{D} = -\nabla\psi.$$

It has already been demonstrated that the nonlocal model equations investigated in this paper successfully describe the solvation scenario of molecules in water [15,13] and they have been solved numerically using the Boundary Element Method (BEM). However, further model extensions (e.g. to include nonlinearities or to use different models for ε), require the availability of a method which is easily extendable and widely applicable especially for nonlinear equations.

A possibility which additionally acquires highly accurate results is offered by the Explicit Jump Immersed Interface Method (EJIIM) together with suitable conditions along the artificial exterior boundary [45,21,23]. In this work, we present the EJIIM for the nonlocal model equations and compare it with the BEM in terms of accuracy and performance. In addition, we exemplary apply the EJIIM solver to two biomolecules.

2. Equations for nonlocal electrostatics

We consider a molecule $\mathcal{M}$ to be a set of N spheres at fixed positions $\boldsymbol{r}_i$ with radii R_i, and fixed charges q_i, $i = 1, \ldots, N$ at its centers. The sum over all point charges forms the charge density

$$\rho(\boldsymbol{r}) = \sum_{i=1}^{N} q_i \delta(\boldsymbol{r} - \boldsymbol{r}_i), \quad \boldsymbol{r} \in \mathbb{R}^3. \tag{6}$$

With Ω we denote the domain within the so called solvent excluded surface of $\mathcal{M}$, see Section 4 for a description. With $\Sigma = \mathbb{R}^3 \setminus \bar{\Omega}$ we denote the volume occupied by the solvent. The surface Γ separates the solvent region Σ from Ω.

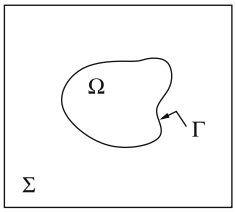

Fig. 1. Illustration of the geometrical setup.

The domain Σ is infinite in all directions. This property is easily realized with the BEM. For the finite difference approach, we restrict $\Sigma \cap \Omega$ to a sufficiently large box and provide exterior boundary conditions on Σ. The geometric setup is illustrated in Fig. 1.

The response of the inner medium is measured by a constant dielectric operator, i.e., inside, the local electrostatic equations are valid:

$$\phi_\Omega = \varepsilon_0 \varepsilon_\Omega \psi_\Omega. \tag{7}$$

In Σ we assume a nonlocal theory of electrostatics to account for the water network. Thus, the unknowns of our system are reduced to the set $\{\phi_\Omega, \phi_\Sigma, \psi_\Sigma\}$.

As the charge density ρ causes discontinuities in the electric potential ϕ_Ω, we take the ansatz $\phi_\Omega = \phi_\Omega^{reac} + \phi_{mol}$. The field ϕ_{mol} accounts for all the charges, treating them as if they were solved in a medium with ε_Ω, whereas the so called reaction field potential, ϕ_Ω^{reac} measures the correction due to the dielectric boundary.

With (6) and the requirement, $\lim_{|\mathbf{r}|\to\infty} \phi_{mol} = 0$, we have (cf. (2.79) in [13])

$$\triangle \phi_{mol}(\mathbf{r}) = -\frac{1}{\varepsilon_0 \varepsilon_\Omega}\rho \;\Rightarrow\; \phi_{mol}(\mathbf{r}) = \frac{1}{4\pi\varepsilon_0\varepsilon_\Omega}\sum_i^N \frac{q_i}{|\mathbf{r}-\mathbf{r}_i|}.$$

Therefore, the unknown part of ϕ_Ω reduces to ϕ_Ω^{reac}. For simplicity, we will drop the superscript from now on and write ϕ_Ω instead of ϕ_Ω^{reac}.

The considered system consists of two domains with different dielectric response. It reads:

$$\Delta\phi_\Omega = 0 \quad \text{in } \Omega, \tag{8a}$$
$$\Delta\psi_\Sigma = 0 \quad \text{in } \Sigma, \tag{8b}$$
$$\phi_\Sigma - \phi_\Omega = p \quad \text{along } \Gamma, \tag{8c}$$
$$\alpha\partial_n\phi_\Omega - \partial_n\psi_\Sigma = g \quad \text{along } \Gamma, \tag{8d}$$
$$(\Delta - \eta^2)\phi_\Sigma + \sigma^2\psi_\Sigma = 0 \quad \text{in } \Sigma, \tag{8e}$$
$$\beta\partial_n\phi_\Sigma - \partial_n\psi_\Sigma = q \quad \text{along } \Gamma, \tag{8f}$$

(2a) yields (8a) in Ω due to (7), and we introduced ϕ_{mol} to account for the source term ρ.

As for the present the solvent is assumed to be free of charge, we get (8b) instead of (2a) in Σ. An extension to dilute salt solutions can be easily done with a finite difference solver. Eqs. (8c) and (8d) are the transmission conditions, forcing the electric potential itself and the normal derivative of the dielectric potential to be continuous, see [16].

The dielectric response in Σ is given by (2b) together with (5) and finally yields (8e). To get a unique system of equations, we need one further boundary condition. (8f) is an artificial boundary condition, which was proposed in [13].

The reference to the physical quantities is given in Table 1.

Before starting with the numerical treatment, we sum up the set of physically motivated parameters, used for all calculations presented in this paper

$$\varepsilon_\Omega = 2.0, \quad \varepsilon_\Sigma = 78.5, \quad \varepsilon_\infty = 1.8, \quad \kappa = 1/20\ \text{Å}^{-1}.$$

These values are taken from former successful studies of the solvation process, see [15,13,14].

3. Numerical methods

3.1. Boundary element method

We use BEM in this paper as the reference method. The main idea of BEM is to derive boundary integral equations whose solution is the full set of boundary conditions-also known as Cauchy data, see [37]. Then, using so called representation formulas, the fields in the regions of interest can be evaluated at any arbitrary point in a postprocessing step. Contrary to common discretization methods, like the finite difference or the finite element method, no artificial boundary condition is necessary for exterior boundary value problems.

Table 1
Notations.

η^2	σ^2	α	β	p	g	q
$\kappa^2\varepsilon_\Sigma/\varepsilon_\infty$	$\kappa^2/(\varepsilon_\infty\varepsilon_0)$	$\varepsilon_0\varepsilon_\Omega$	ε_∞	$-\phi_{mol}$	$-\varepsilon_0\varepsilon_\Omega\partial_n\phi_{mol}$	0

The required geometric input data is a numerical approximation of the molecular surface Γ – a triangulation of the surface. The generation of such triangulations has received considerable attention in the literature, and most surface definitions can today be triangulated with good quality (see, e.g., [34,4,47]). However, molecular surfaces tend to be very complex – much more complex than, e.g., typical machine parts in CAD applications – and a triangulation with guaranteed quality will quite often require tens or hundreds of millions of triangles. Thus, input meshes for BEM are often subjected to mesh coarsening algorithms (see, e.g., [32,6]). In this case, a manual inspection is usually performed to ensure topological correctness (for instance, charges located on small hydrogens might end up on the wrong side of the mesh after coarsening). In practice, even though there has been significant progress in input mesh generation for biomolecular BEM, typical application scenarios today still require manual intervention and curation, preventing a fully automated workflow [15]. This, however, will likely be overcome in the near future.

Apart from this technicality, the use of the boundary element method imposes a more serious restriction – the need for a fundamental solution of the differential operator. Due to this requirement, an extension of the fundamental equations of nonlocal electrostatics like the exchange of the dielectric operator or the inclusion of nonlinear counter ion terms, becomes a challenging task.

3.2. Explicit jump immersed interface method

The Explicit Jump Immersed Interface Method (EJIIM) is a finite difference method that works on an equidistant Cartesian grid. The method can handle non-grid aligned discontinuities in the equation and of the solutions: near discontinuities, the standard finite difference approximations are modified by adding correction terms, which involve jumps in the function and its derivatives.

For details on EJIIM, we refer the interested reader to the earlier works [43,46,33,28,30,29]. In this section, only the main idea of the method is recapitulated and necessary extensions that are needed for solving (8) are presented.

System (8) is a *two-phase* problem for ϕ and a *boundary value* problem for the variable ψ. As always done in the EJIIM approach, the system (8) is extended by an additional equation

$$\Delta\psi_\Omega = 0 \quad \text{in } \Omega, \quad \psi_\Omega = 0 \quad \text{along } \Gamma,$$

forcing ψ_Ω to be zero. Along the exterior boundary $\partial(\Sigma\cup\Omega\cup\Gamma)$, a Dirichlet boundary condition is given. This will be discussed in more detail in Section 6.

Remark 1. Requiring ψ_Ω to be zero is different from the local electrostatic condition (7). Note, however, that no changes are made with respect to the system (8) and the "correct" ψ_Ω can be easily obtained in a postprocessing step, if required.

For a piecewise continuous function $u:\Omega\cup\Sigma\to\mathbb{R}$ with a possible discontinuity across the interface Γ and $\alpha\in\Gamma$, we define the onesided values of u as $u_\Omega(\alpha) := \lim_{\Omega\ni\boldsymbol{r}\to\alpha} u(\boldsymbol{r})$, $u_\Sigma(\alpha) := \lim_{\Sigma\ni\boldsymbol{r}\to\alpha} u(\boldsymbol{r})$ and the *jump* $[u]_\alpha := u_\Sigma(\alpha) - u_\Omega(\alpha)$. Mostly, the index and argument α will be omitted for the jump condition, meaning that the relation holds everywhere along Γ.

3.2.1. Discretization of system (8)

3.2.1.1. Regular grid. A regular grid with mesh width h is imposed over $\Sigma\cup\Omega$. We will use the standard notation $u_{i,j,k} := u(x_i,y_j,z_k)$ for a grid point (x_i,y_j,z_k), $i\in\{1,\dots,n_x\}$, $j\in\{1,\dots,n_y\}$, $k\in\{1,\dots,n_z\}$. Points where the interface Γ is cut by the grid lines are called *intersections*.

A grid point is classified as *regular* if the standard central finite difference approximation of (8) is not influenced by the interface Γ. All other points are called *irregular*. To discretize the equations, standard finite differences are used at all regular points.

3.2.1.2. Corrections at irregular points. At all irregular points, jump dependent *correction terms* are introduced to achieve an $\mathcal{O}(h)$ approximation. In a planar situation as in Fig. 2, we would have

$$\partial_{xx}u(x_i,y_j) \approx \partial^h_{xx}u(x_i,y_j) - \frac{1}{h^2}\left([u]_{\alpha_2} + \tau_2[\partial_x u]_{\alpha_2} + \frac{\tau_2^2}{2}[\partial_{xx}u]_{\alpha_2}\right), \tag{9a}$$

$$\partial_{yy}u(x_i,y_j) \approx \partial^h_{yy}u(x_i,y_j) - \frac{1}{h^2}\left([u]_{\alpha_1} + \tau_1[\partial_y u]_{\alpha_1} + \frac{\tau_1^2}{2}[\partial_{yy}u]_{\alpha_1}\right), \tag{9b}$$

where $\partial^h_{xx}u$ and $\partial^h_{yy}u$ are the standard central finite difference operators [43].

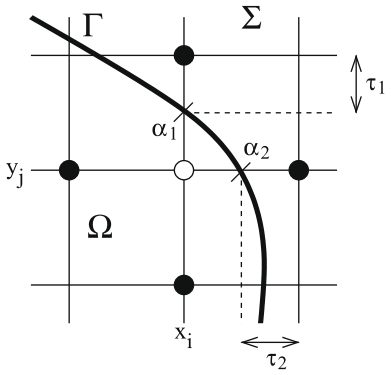

Fig. 2. Two dimensional illustration of a discretization at an irregular grid point (x_i, y_j) with two intersection points α_1 and α_2.

3.2.1.3. Jump variables. Note that the jumps in the Cartesian derivatives are not known explicitly. Instead, we introduce them as new *additional variables*.

To write the corrected approximations in a matrix–vector form, we introduce discrete solution vectors Φ and Ψ containing values of ϕ and ψ at the grid points as well as vectors F_1 and F_2 storing the right hand side of (8a,b,e). In addition, we have vectors J^ϕ, J^ψ and $J = (J^\phi, J^\psi)$ containing all newly introduced jumps in ϕ, ψ, and their derivatives.

Then, the discrete form of (8) can be written as

$$\underbrace{\begin{pmatrix} \Delta^h - \eta^2 I & \sigma^2 I \\ 0 & \Delta^h \end{pmatrix}}_{=:A} \underbrace{\begin{pmatrix} \Phi \\ \Psi \end{pmatrix}}_{=:W} + \underbrace{\begin{pmatrix} C_{11} & 0 \\ 0 & C_{22} \end{pmatrix}}_{=:C} \underbrace{\begin{pmatrix} J^\phi \\ J^\psi \end{pmatrix}}_{=:J} = \underbrace{\begin{pmatrix} F_1 \\ F_2 \end{pmatrix}}_{=:F}, \tag{10}$$

where Δ^h is the discrete Laplace matrix (defined in $\Omega \cup \Sigma$) and I is a diagonal matrix with ones at the rows corresponding to the grid points in Σ and zeros otherwise. The term CJ codes the correction terms at the irregular grid points.

The additional equations for the jump variables are gained from the interface conditions in (8) and from onesided extrapolation.

3.2.2. Approximating the jump variables

3.2.2.1. Jump conditions. In order to apply the method in (10), we need the jumps in Cartesian derivatives of the functions ϕ and ψ, respectively.

- The 0th order jump for ϕ is given by (8c). For ψ, we set the trivial jump condition $[\psi] = \psi_\Sigma - \psi_\Omega = \psi_\Sigma$ as we need $\psi_\Omega = 0$.
- The 1st order jumps: Let t and s be two tangents of Γ such that (n, t, s) forms an orthonormal positively oriented system. Further, let $\mathbf{m} \in \mathbb{R}^{3\times 3}$ be the transformation matrix with columns given by vectors n, t, and s. Then, from system (8) we directly obtain

$$\begin{bmatrix} \partial_x \phi \\ \partial_y \phi \\ \partial_z \phi \end{bmatrix} = \mathbf{m} \begin{bmatrix} \partial_n \phi \\ \partial_t \phi \\ \partial_s \phi \end{bmatrix} = \mathbf{m} \begin{pmatrix} \frac{\alpha-\beta}{\beta} \partial_n \phi_\Omega + \frac{q-g}{\beta} \\ \partial_t p \\ \partial_s p \end{pmatrix}.$$

For ψ, we do not know $[\psi]$ analytically. Thus, we now use

$$\begin{bmatrix} \partial_x \psi \\ \partial_y \psi \\ \partial_z \psi \end{bmatrix} = \begin{pmatrix} \partial_x \psi_\Sigma \\ \partial_y \psi_\Sigma \\ \partial_z \psi_\Sigma \end{pmatrix} = \mathbf{m} \begin{pmatrix} \partial_n \psi_\Sigma \\ \partial_t \psi_\Sigma \\ \partial_s \psi_\Sigma \end{pmatrix} = \mathbf{m} \begin{pmatrix} \frac{1}{2}(\alpha \partial_n \phi_\Omega - g) + \frac{1}{2}(\beta \partial_n \phi_\Sigma - q) \\ \partial_t \psi_\Sigma \\ \partial_s \psi_\Sigma \end{pmatrix}.$$

- All necessary second order jumps are obtained by extrapolation:

$$[\partial^2 \phi] = \partial^2 \phi_\Sigma - \partial^2 \phi_\Omega, \quad [\partial^2 \psi] = \partial^2 \psi_\Sigma - \partial^2 \psi_\Omega = \partial^2 \psi_\Sigma.$$

The onesided values of $\phi_{\Omega,\Sigma}$, ψ_Σ and derivatives are obtained by extrapolation (of unknown functions) from the grid points to the interface.

3.2.2.2. Onesided extrapolations. In [29], it was observed for planar boundary value problems that an appropriate choice of onesided extrapolations can increase the efficiency of the method considerably by reducing the necessary iteration counts. Our extrapolation is built on a least squares based second order polynomial fit and is essentially the same as used in [33]. We

have an additional requirement that the extrapolations should fulfill also the Eqs. (8a,b,e). This forces to simultaneously extrapolate ϕ and ψ.

3.2.3. Solving the discrete system

3.2.3.1. Discrete system. Onesided extrapolation together with the jump condition can be written in the form $DW + J = \widetilde{F}$. Then, with (10) we obtain the full EJIIM discretization

$$\begin{pmatrix} A & C \\ D & \mathbb{I} \end{pmatrix} \begin{pmatrix} W \\ J \end{pmatrix} = \begin{pmatrix} F \\ \widetilde{F} \end{pmatrix}, \tag{11}$$

where $\mathbb{I}$ is the identity matrix. System (11) is solved iteratively using BiCGSTAB [17] by a Schur complement for the variable J:

$$(\mathbb{I} - DA^{-1}C)J = \widetilde{F} - DA^{-1}F. \tag{12}$$

Remark 2. Any Krylov space method that is applicable to nonsymmetric matrices could be used here. A popular choice is GMRES [17] due to its often good convergence properties. However, for three dimensional problems, storage becomes an essential issue and even restarted GMRES has higher memory requirements than BiCGSTAB.

Once the jump vector J has been found, W is obtained from $W = A^{-1}(F - CJ)$.

3.2.3.2. Fast solver. In each iteration, when solving (12), we need to apply the operator A^{-1}. For this the following discrete problem has to be solved

$$\begin{pmatrix} \Delta^h - \eta^2 I & \sigma^2 I \\ 0 & \Delta^h \end{pmatrix} \begin{pmatrix} \Phi \\ \Psi \end{pmatrix} = \begin{pmatrix} P_1 \\ P_2 \end{pmatrix},$$

where Φ and Ψ are discrete solution vectors at the grid points and $P_1,\ P_2$ are some right hand side vectors, changing in each iteration. Solving is done in two steps:

1. $\Psi = (\Delta^h)^{-1} P_2$
2. $\Phi = (\Delta^h - \eta^2 I)^{-1}(P_1 - \sigma^2 I \Psi)$
   This is solved by an additional (inner) BiCGSTAB iteration. Iteration counts in this formulation are bad (around 50–60 to reach 10^{-10} tolerance) and are improved considerably by using a discrete Laplace as preconditioner. That is, we solve instead

   $$(\Delta^h)^{-1}(\Delta^h - \eta^2 I)\Phi = (\Delta^h)^{-1}(P_1 - \sigma^2 I \Psi).$$

Using preconditioning, 5–6 iterations were typically necessary for convergence.

The operator $(\Delta^h)^{-1}$ is always applied by an FFT-based fast solver from [44].

4. Input generation for the explicit jump immersed interface method

The input information EJIIM requires is a surface description of the molecule in the grid. Since there is no canonical definition of a molecule's surface, three different surface models are commonly used, see Figs. 3 and 4.

1. *Van der Waals surface (VdW):* The union of the atoms the molecule $\mathcal{M}$ is composed of. The atoms are modelled by spheres.
2. *Solvent Accessible Surface (SAS):* Rolling a solvent molecule over the VdW surface, the SAS is the area traced out by the center of the solvent molecule during the rolling process. The solvent molecule is represented by a sphere, the so–called probe sphere.
3. *Solvent Excluded Surface (SES):* Rolling a solvent molecule over the VdW surface, the SES is the surface traced by the inward-facing surface of the probe sphere.

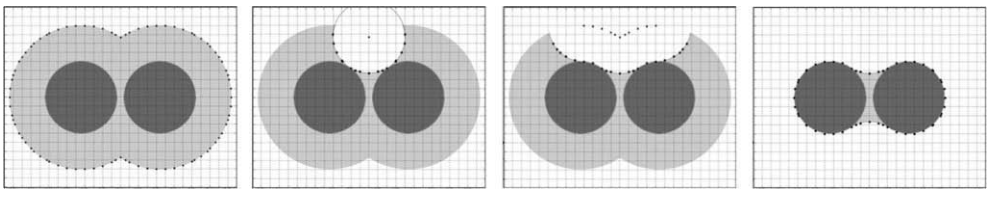

Fig. 3. Left: VdW (dark gray), SAS (light gray); middle: constructing SES:SAS intersection points that were already processed are additionally drawn; right: SES.

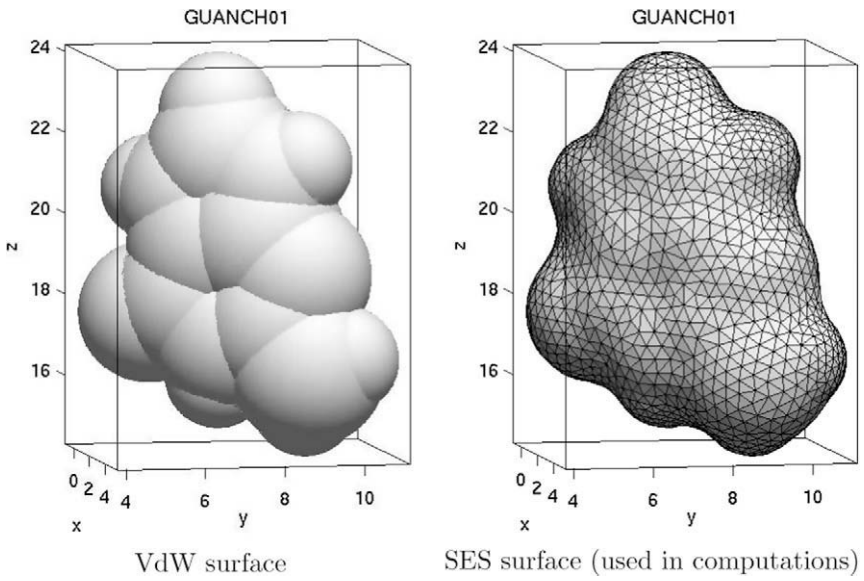

Fig. 4. GUANCH01 as a typical example of a molecule in the tested subset of the MMFF94 test data set. The surface mesh is the original triangulation used for the BEM calculations. Length units are given in Angstrom [Å].

In solvation studies, both, the SES and the SAS are reasonable surface definitions [20]. Since the SES really separates the molecule $\mathcal{M}$ from the solvent molecules, this definition is used in our study.

Since the input generation is not the focus of this paper, we only give a rough idea of the method: the core of the algorithm is the calculation of the intersection points of an arbitrary sphere with a regular, equidistant grid. For the SAS of a molecule, we generate this intersection information of every atom in a small grid and merge it consecutively with the final grid, which comprises all the information of already merged atoms. With the aid of different priority tables, the merging procedure can be adopted to a merging of atoms as well as of probe spheres. The latter is to simulate the solvent molecule rolling upon the VdW surface. This finally reveals the SES.

This procedure is sketched in Fig. 3: starting with the SAS of the molecule, which consists of two atoms, we merge probe grids that are centered on the discrete SAS intersection points. The algorithm provides the EJIIM input as it is described in [28] and seen in Fig. 2. It is used to generate the input data for all EJIIM calculations presented in the following sections.

The complexity of this simple approach is mainly determined by the merging procedures of all small probe grids, which yields $\mathcal{O}(M \times n^3)$ with M denoting the number of SAS intersection points and n^3 the probe grid dimension. For high resolution calculations and for molecules consisting of more than 100 atoms, this approach is no longer efficient. However, the idea is a good starting point for more sophisticated and more efficient methods currently under development.

5. Numerical comparison of EJIIM and BEM

Our goal is to develop an EJIIM solver that provides results with comparable or possibly better quality than the existing BEM code. For this, we will discuss several studies:

- we check the convergence order of both methods for spherical symmetry,
- we analyze the numerical convergence, i.e., how much does the EJIIM solution differ from the BEM result,
- we investigate different approximations of the boundary conditions.

5.1. Convergence towards analytical solution

The convergence studies are done with spherical geometry, as in this case, the analytical solution for model (8) is available. Additionally, the spherical geometry offers the possibility to generate "exact" input data for both methods.

Ω is a sphere with radius 1. We center the sphere at $(0.2, 0.3, 0.01)$ to avoid symmetry effects and locate a unit electron charge $q = 1$ at the sphere's center. For EJIIM, the computational domain is given by $\Sigma \cup \Omega = [-5,5] \times [-5,5] \times [-5,5]$ in all cases. The necessary Dirichlet condition along the exterior boundary is given by the analytical solution.

Convergence is tested with respect to the parameter h, which, in the case of EJIIM, is the usual mesh width. For a triangulation $\mathbb{T}$ with triangles $T \in \mathbb{T}$, h is defined as in [37], p. 213:

$$h^{bem} = \max_{T \in \mathbb{T}} h_T, \quad h_T = \sqrt{\text{area}(T)}. \tag{13}$$

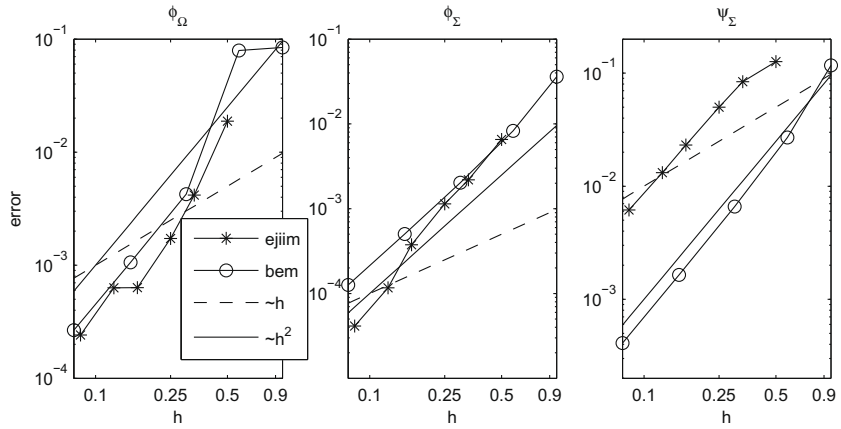

Fig. 5. Convergence of the absolute error for the nonlocal model (BEM, EJIIM). Please note Remark 3.

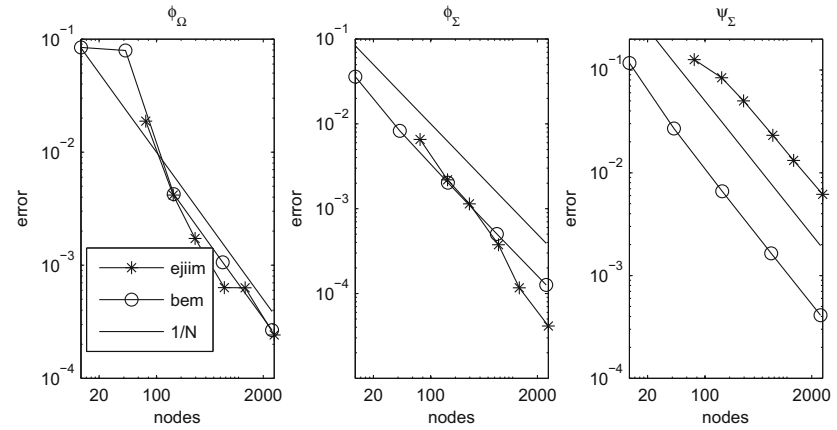

Fig. 6. Convergence of EJIIM and BEM with respect to the surface nodes (absolute error). Please note Remark 3.

For several resolutions (EJIIM) and triangulations (BEM), respectively, the absolute numerical error has been computed at randomly selected points (20 in Ω, 100 in Σ). Fig. 5 confirms that the expected second order convergence for both methods has been achieved.

An important observation can be made from Fig. 6, where we have plotted the error versus the number of intersection points (EJIIM) or nodes (BEM). For both methods, this is a quantity that is related to the surface resolution and can be directly compared. We see that for the potential ϕ, EJIIM and BEM need a similar number of surface nodes to achieve the same quality.

Remark 3. Figs. 5 and 6 confirm the theoretically expected second order convergence of EJIIM and BEM. Note, however, that the *quantitative* values for the dielectric potential ψ in the case of EJIIM are worse than for ϕ. This phenomenon can be explained by (9b). For the electrical potential ϕ, we have a pure transmission problem with a *known* jump $[\phi]$, whereas for $[\psi]$ it is numerically computed (second order polynomial fit as done in Section 3.2.2 leads to $\mathcal{O}(h^3)$ accuracy) and the correction term in (9b) has a coefficient of $\mathcal{O}(h^{-2})$. In our situation, this is not a real disadvantage, since the physically interesting quantity is ϕ.

The EJIIM code has also been tested with more general right hand sides and arbitrary functions in the transmission conditions, see Table 1. In all cases, the second order convergence was confirmed. Unfortunately, analogous runs with BEM were not possible due to built-in limitations of this method.

5.2. Comparison on molecular geometries

5.2.1. Test set

As test set, we have selected six molecules of the validation suite of MMFF94 [25], three of them with non-vanishing total charge (charged) and three with vanishing total charge (uncharged). Radii and charge assignments have been performed

Table 2
Summary data of the test set. The grid dimension is the same in all three directions.

Molecule	Grid dimension			Intersections			BEM elements		
	$h = 0.5$	$h = 0.25$	$h = 0.125$	$h = 0.5$	$h = 0.25$	$h = 0.125$	QECD	Original	BF
Charged									
AN05A	31	61	121	482	1904	7556	702	1404	5616
BRMW1	33	65	129	648	2586	10252	642	1286	5144
GUANCH01	39	77	153	1034	4046	16192	1646	3292	13168
Uncharged									
FUCMUL	35	69	137	906	3600	14440	1060	2122	8488
FUDPOJ	31	61	121	656	2636	10598	978	1956	7824
ZZZIZA01	39	77	153	1242	4874	19568	1436	2872	11488

with the MMFF94 implementation in BALL [19]. In Table 2, we give the numerically relevant parameters for the molecules. Further details are provided in the Supplementary material.

All EJIIM computations have been performed with three different grids, with mesh widths $h = 0.5$, $h = 0.25$, and $h = 0.125$. For BEM, three resolutions have been used: the original one, generated by [5], roughened by a Quadric Edge Collapse Decimation algorithm (QECD) and refined by a Butterfly Subdivision algorithm (BF) [6,10].

5.2.2. Error measures

Equipped with two different methods to solve our system of Eq. (8), the question arises how to measure their performance in real world problems, where no analytical solution is available. We decided to stochastically generate point sets, X_Ω and X_Σ, of 100 points inside and outside the molecule, respectively. X_Σ lies in the computational box of the EJIIM calculations. The deviation is measured in the following seminorms

$$|u|_\Omega := \max|u(X_\Omega)|, \quad |u|_\Sigma := \max|u(X_\Sigma)|.$$

Further, for the quantity u, we will use the notations u^h: the EJIIM solution with mesh width $h \in \{0.5, 0.25, 0.125\}$, u^{bem}: the BEM solution with the original surface triangulation, u^{qecd}: the BEM solution with QECD-coarsened and u^{bf}: the BEM solution with BF-refined surface.

To compare quantities v and w we will use the function

$$e(v, w) := \frac{|v - w|_I}{|w|_I}, \quad I \in \{\Omega, \Sigma\}. \tag{14}$$

5.2.3. Results

First, we are interested in *numerical convergence* only, thus we use the $(\phi_\Sigma^{bf}, \psi_\Sigma^{bf})$ values as the exterior boundary condition for the EJIIM computation. Results computed with this boundary condition can be found in Tables 3 and 4. To save space, we report only the values for the physically relevant electric potential ϕ.

From the EJIIM as well as from the BEM results listed in Table 3, we can not directly deduce a specific convergence rate. This is due to the nontrivial input data. Additionally, the randomly chosen point sets on which the error function is evaluated

Table 3
Numerical convergence towards the fine grid solution.

Molecule	Charged			Not charged		
	AN05A	BRMW1	GUANCH01	FUCMUL	FUDPOJ	ZZZIZA01
$e\left(\phi_\Omega^{0.5}, \phi_\Omega^{0.125}\right)$	2.22e−2	1.58e−2	3.61e−2	5.47e−1	5.66e−1	2.10e−1
$e\left(\phi_\Omega^{0.25}, \phi_\Omega^{0.125}\right)$	2.65e−3	4.14e−3	5.70e−3	6.18e−2	5.31e−2	5.30e−2
Factor	8.39	3.82	6.33	8.87	10.66	3.95
$e\left(\phi_\Omega^{qecd}, \phi_\Omega^{bf}\right)$	6.28e−3	1.33e−2	9.62e−3	1.94e−1	2.50e−1	1.20e−1
$e\left(\phi_\Omega^{bem}, \phi_\Omega^{bf}\right)$	4.82e−3	4.97e−3	2.59e−3	6.02e−2	5.82e−2	5.90e−2
Factor	1.30	2.67	3.72	3.21	4.29	2.03
$e\left(\phi_\Sigma^{0.5}, \phi_\Sigma^{0.125}\right)$	9.23e−3	5.43e−3	1.69e−2	2.29e−2	9.43e−2	6.02e−2
$e\left(\phi_\Sigma^{0.25}, \phi_\Sigma^{0.125}\right)$	1.70e−3	1.10e−3	1.53e−3	1.67e−3	7.67e−3	2.61e−3
Factor	5.43	4.92	11.0	13.7	12.3	23.0
$e\left(\phi_\Sigma^{qecd}, \phi_\Sigma^{bf}\right)$	3.70e−3	1.18e−2	7.72e−3	4.80e−3	1.04e−2	9.58e−3
$e\left(\phi_\Sigma^{bem}, \phi_\Sigma^{bf}\right)$	2.97e−3	3.38e−3	1.01e−3	1.23e−3	9.75e−3	4.26e−3
Factor	1.25	3.50	7.62	3.92	1.06	2.25

Table 4
Convergence towards BEM results.

Molecule	Charged			Not charged		
	AN05A	BRMW1	GUANCH01	FUCMUL	FUDPOJ	ZZZIZA01
$e\left(\phi_\Omega^{0.5}, \phi_\Omega^{bf}\right)$	2.05e−2	1.35e−2	3.56e−2	5.75e−1	5.24e−1	2.05e−1
$e\left(\phi_\Omega^{0.25}, \phi_\Omega^{bf}\right)$	2.70e−3	3.75e−3	5.38e−3	8.71e−2	6.79e−2	4.77e−2
$e\left(\phi_\Omega^{0.125}, \phi_\Omega^{bf}\right)$	2.36e−3	4.26e−3	1.54e−3	3.91e−2	3.08e−2	1.87e−2
Error estimate	3e−3	5e−3	2e−3	4e−2	4e−2	2e−2
$e\left(\phi_\Sigma^{0.5}, \phi_\Sigma^{bf}\right)$	8.69e−3	7.68e−3	1.65e−2	2.29e−2	9.39e−2	6.02e−2
$e\left(\phi_\Sigma^{0.25}, \phi_\Sigma^{bf}\right)$	1.62e−3	3.63e−3	9.98e−4	1.67e−3	7.15e−3	2.64e−3
$e\left(\phi_\Sigma^{0.125}, \phi_\Sigma^{bf}\right)$	1.27e−3	2.67e−3	1.04e−3	1.30e−3	2.72e−3	1.06e−3
Error estimate	2e−3	3e−3	2e−3	2e−3	3e−3	2e−3

allow only a rough interpretation: The outer electric potential of uncharged molecules, ϕ_Σ, has higher absolute errors and faster EJIIM convergence rates than those of charged molecules on average. The reason lies in the fast decrease of the potential outside, which is better mapped for higher grid resolutions.

With Table 3, we demonstrate that both methods really converge to the fine grid solutions.

In Table 4, we study the EJIIM solution versus the BEM solution. The difference between the finest grid solutions, $\phi_{\Omega,\Sigma}^{0.125}$ for EJIIM and $\phi_{\Omega,\Sigma}^{bf}$ for BEM, rounded upwards to one significant digit, is used as a rough estimate for the numerical error later when approximating the boundary condition.

It is important to mention that in Table 4, we are not considering the numerical error, that is, the difference between the exact and numerical solutions. Instead, we are looking for the difference between two solutions, where $\phi_{\Omega,\Sigma}^{bf}$ has been used as reference. Thus, stagnating difference, like for the BRMW1 molecule, is legitimate.

6. Approximations of the exterior boundary condition

In order to assess the effect of different boundary value approximations, we propose different boundary conditions (b.c.) at the exterior boundary of the EJIIM computational domain:

- homogeneous exterior b.c., that is, zero Dirichlet boundary condition
- b.c. without inner medium (see Section 6.1)
- b.c. without inner medium and simplified charge distribution (see Section 6.1)

6.1. Boundary condition without inner medium and simplified charge distribution

Although we have to cope with a transmission problem, the behavior far away from the surface is mainly determined by the charge distribution and the dielectric response outside the molecule. This motivates the following approximation: we assume that the charge distribution exists in the outer medium only, skipping the boundaries and therefore the coupling of the two media.

The nonlocal electrostatic potentials of charge density ρ in a setting without inner medium, i.e., $\Sigma = \mathbb{R}^3$ and $\Omega = \emptyset$, read as follows

$$\psi_\Sigma(\boldsymbol{r}) = \frac{1}{4\pi} \int_{\mathbb{R}^3} \frac{\rho(\boldsymbol{r}')}{|\boldsymbol{r} - \boldsymbol{r}'|}\, d\boldsymbol{r}', \tag{15a}$$

$$\phi_\Sigma(\boldsymbol{r}) = \frac{1}{\varepsilon_\Sigma \varepsilon_0} \psi_\Sigma(\boldsymbol{r}) + \frac{1}{4\pi\varepsilon_0} \left(\frac{1}{\varepsilon_\infty} - \frac{1}{\varepsilon_\Sigma}\right) \int_{\mathbb{R}^3} \rho(\boldsymbol{r}') \frac{e^{-\eta|\boldsymbol{r}-\boldsymbol{r}'|}}{|\boldsymbol{r} - \boldsymbol{r}'|}\, d\boldsymbol{r}'. \tag{15b}$$

Assuming these analytical solutions with ρ defined in (6) as boundary value approximations is denoted *approximated* b.c. (approx).

To generate boundary values *efficiently*, we replace the charge distribution ρ defined in (6), by an arrangement with only two opposed charges. For index sets $I^+ := \{i \mid 1 \leqslant i \leqslant N,\ q_i > 0\}$ and $I^- := \{i \mid 1 \leqslant i \leqslant N,\ q_i \leqslant 0\}$ we define

$$Q^\pm := \sum_{i \in I^\pm} q_i, \quad \boldsymbol{r}^\pm := \frac{1}{Q^\pm} \sum_{i \in I^\pm} \boldsymbol{r}_i q_i.$$

Then,

$$\rho_{\mathit{eff}} := Q^+ \delta(\boldsymbol{r} - \boldsymbol{r}^+) + Q^- \delta(\boldsymbol{r} - \boldsymbol{r}^-).$$

Replacing the molecule's charge density by ρ_{eff} in (15b), we gain a fast boundary value approximation denoted *effective dipole* b.c. (edp). Note that, if one of the effective charges is zero, only the other is taken into account and we obtain an effective monopole.

6.2. Discussion of the boundary approximations

To check the quality of the proposed boundary conditions, we use again our test set. All calculations in this section are done on the finest mesh with $h = 0.125$. In addition, all calculations have been carried out on two different box dimensions with equal resolution in order to test the influence of the approximations: 4 Å and 6 Å from the molecular surface, respectively, see Fig. 7 for an illustration.

For the *reference solution* u^{ref}, where $u \in \{\phi_\Omega, \phi_\Sigma, \psi_\Sigma\}$, the exterior boundary condition is provided by the BEM calculation on the refined mesh. In Table 5 the values are computed as $\tilde{e}(u) := \|u - u^{ref}\|_\infty / \|u^{ref}\|_\infty$, $u \in \{\phi_\Omega, \phi_\Sigma\}$ with different boundary conditions for $(\phi_\Sigma, \psi_\Sigma)$.

6.2.1. Homogeneous, approximated, and effective dipole conditions

A closer look at Table 5 suggests that the approximated b.c. provides the best approximation: here, $\tilde{e}$, which accounts for the error made by the boundary value approximation, is always of the same order as the numerical error, which was extracted from Table 4.

However, also the effective dipole and even the homogeneous ansatz can reasonably reproduce the BEM results. Note that for all molecules of our test set with vanishing total charge, the effective dipole approximation does not lead to satisfying boundary condition. This highly depends on the spatial charge distribution: for vanishing effective monopole and dipole (mirror symmetry of the charge distribution for example) the effective dipole approximation is in fact a homogeneous boundary condition and therefore leads to the same accuracy. In contrast, for the charged molecules of our test set, the effective approximation is comparable with the approximated boundary condition when the 6 Å computational box is taken.

In Fig. 7, we illustrate the behavior of the approximated boundary conditions for an (un)charged molecule, (FUDPOJ) GUANCH01. The difference between the solution of (8) and (15b), evaluated on Σ, is additionally plotted in Fig. 7 and supports our assumption that the influence of the dielectric boundary is negligible for the outer field solutions a few Å from the molecule.

6.2.2. Focusing condition

The focusing b.c. means to compute a larger but less dense grid and to take its result for initializing the original dense grid. For the rough calculation, we recommend boundary conditions, which are fast to generate. In the calculations, the effective b.c. turns out to be a good choice. Overall, the focusing method generates roughly the same accuracy as the dipole b.c. when the 4 Å computational box is taken, thus we dispense with the tables.

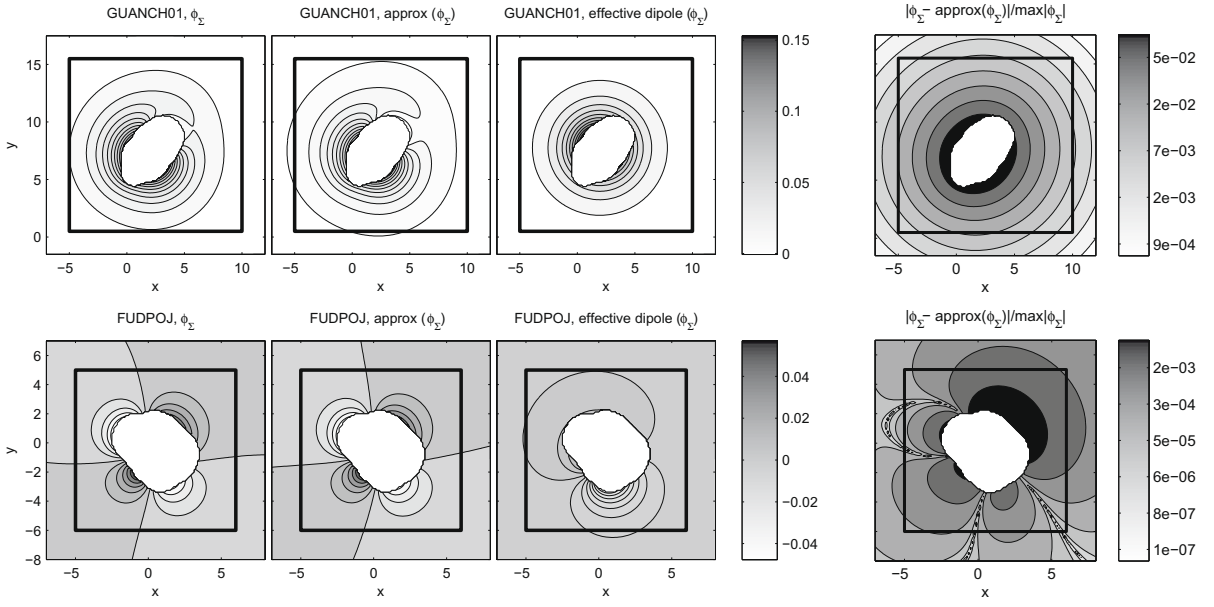

Fig. 7. Reference solution ϕ_Σ, approximated and effective dipole potentials. Cuts along the plane $z = z^*$, where (x^*, y^*, z^*) is the center of the computational box. The black border shows the 4 Å box (see Section 6.2). Contour lines of the difference are selected logarithmically to visualize also the small values.

Table 5
Approximation of the exterior b.c.; hom: homogeneous, edp: effective dipole, approx: approximated, e: estimate of the numerical error.

Molecule	4[A] Box			6[A] Box			e
	hom	edp	approx	hom	edp	approx	
$\tilde{e}(\phi_\Omega)$							
AN05A	1.69e−1	2.59e−2	2.51e−2	4.95e−2	5.54e−3	5.43e−3	3e−3
BRMW1	2.25e−1	6.73e−2	4.04e−2	5.76e−2	1.19e−2	7.92e−3	5e−3
GUANCH01	1.01e−1	5.86e−2	1.29e−2	3.19e−2	9.79e−3	2.63e−3	2e−3
FUCMUL	7.74e−1	7.74e−1	6.38e−2	1.10e−1	1.10e−1	1.09e−2	4e−2
FUDPOJ	3.98e−1	2.97e−1	4.36e−2	6.16e−2	3.63e−2	8.35e−3	4e−2
ZZZIZA01	9.27e−1	9.27e−1	8.28e−2	1.35e−1	1.35e−1	1.15e−2	2e−2
$\tilde{e}(\phi_\Sigma)$							
AN05A	1.90e−1	3.17e−2	2.95e−2	7.54e−2	1.16e−2	1.09e−2	2e−3
BRMW1	3.04e−1	1.13e−1	5.74e−2	1.16e−1	3.63e−2	2.07e−2	3e−3
GUANCH01	1.38e−1	1.06e−1	1.96e−2	5.77e−2	3.56e−2	7.51e−3	2e−3
FUCMUL	9.50e−2	9.50e−2	6.35e−3	2.86e−2	2.86e−2	2.06e−3	2e−3
FUDPOJ	7.84e−2	8.07e−2	5.91e−3	2.46e−2	2.16e−2	2.02e−3	3e−3
ZZZIZA01	2.25e−1	2.25e−1	2.10e−2	7.05e−2	7.05e−2	6.41e−3	2e−3

We conclude that the derived boundary value approximations do not reduce the quality of the solution strongly. The special choice of the b.c. depends on the demand on accuracy and efficiency. Thus, the alternative to the BEM finally consists of the EJIIM coupled with an appropriate approximation of the exterior boundary values.

7. Computational costs

The EJIIM approach was implemented using Matlab version R2007b, the BEM was implemented in C++ and uses the ATLAS library [42], for efficient solution of the linear system of equations.

To give a rough impression on the computational costs, we compared both methods on a machine with eight Intel(R) Xeon(R) CPU X5355 @ 2.66 GHz processors and 8 GB RAM. In Table 6, we list the runtimes necessary for the EJIIM calculation of AN05A (Small Box) with homogeneous b.c. and those of the BEM calculation to generate the Cauchy data.

The runtimes and iteration counts for other molecules are similar. In order to relate the results to the numerical quality and to the surface information, we refer to Tables 5 and 2, respectively.

In Table 6 we see that runtimes of both methods are of approximately the same order. Of course, this comparison does not allow us to answer the question if one or another method is faster or requires less memory. These factors depend heavily on the programming language, particular implementation, and the computational system. In addition, the output of EJIIM (full grid solution) differs from the BEM result (certain data along the surface). However, we see that both methods can really compete.

8. Application to biomolecules

In this section we exemplarily study the effect of the nonlocal theory on two biomolecules, the enzyme trypsin and the protein ovalbumin (PDB-entries 2PTC [24] and 1OVA [36], respectively). A sketch of the surfaces is shown in Fig. 8.

8.1. Trypsin

With its 3223 atoms, trypsin is a small protein. Trypsin is known for its functionality as a digestion enzyme and for its electrostatic to play a crucial role in the binding process of the inhibitor BPTI, [20]. In Fig. 8(a) the binding pocket of trypsin can be located in the upper left.

Table 6
Computational costs for AN05A.

EJIIM					BEM	
Type	Runtime (sec) matrix setup	Runtime (sec) solving	Outer BiCGSTAB iterations (10^{-8} stopping tolerance)	Inner BiCGSTAB iterations (10^{-10} stopping tolerance)	Type	Runtime (sec) solving Cauchy Data
0.5	5	7	6.5	4	*qecd*	3
0.25	41	78	7.5	4	*bem*	20
0.125	1023	1009	8.5	4	*bf*	821

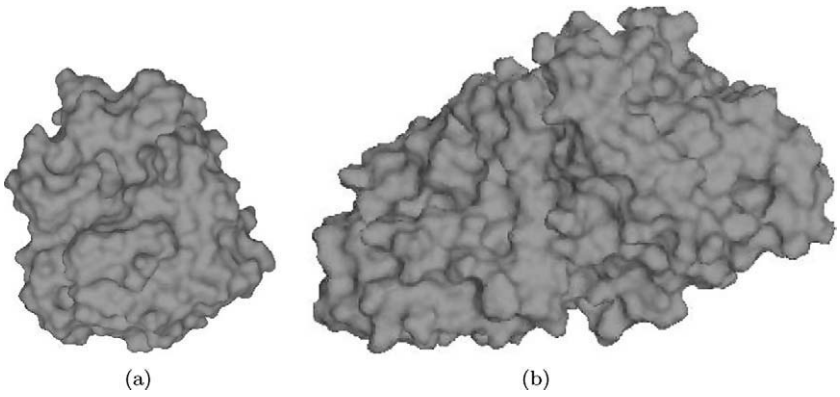

Fig. 8. Protein surfaces for (a) trypsin and (b) ovalbumin.

For trypsin we solved the nonlocal electrostatic equations with BEM and EJIIM to compare the influence of the boundary conditions and the overall accuracy of the solution. Additionally, we calculated the local electrostatic solution with the existing BEM solver, see [3] for the local model equations.

For the EJIIM calculation we took a mesh width of 0.5 Å and box dimensions [88 Å × 88 Å × 88 Å]. The boundary values have been calculated with the approximate boundary condition defined in Section 6.1. The BEM calculation is based on a surface triangulation with about 20 000 elements, for which all the available working storage was used.

First, we analyse the accuracy of both numerical methods, where an appropriate measure for the deviation was defined in Section 5.2.2. To have representative point sets in (14) we stochastically choose 1000 points in Ω and Σ, respectively. The relative difference inside the molecule, $e\left(\phi_{\Omega}^{ejiim}, \phi_{\Omega}^{bem}\right) = 1.23\mathrm{e}-2$, whereas in Σ, $e\left(\phi_{\Sigma}^{ejiim}, \phi_{\Sigma}^{bem}\right) = 2.5\mathrm{e}-3$. Therefore, we observe the same order of magnitude of the numerical error as for the molecules discussed in Section 5. From this we can conclude that the EJIIM method can handle complex and nontrivial molecule geometries.

From the results in Section 6.2 we found that (15b) is not only suitable for an approximation of the boundary values, but gives an impression of the electrostatic potential in the whole exterior region. In Fig. 9 we compare the approximated potential to the numerical solution of (8) in terms of (14). As can be seen, the approximated boundary condition can reproduce the numerical solution not only for simply shaped, small molecules, it gives an adequate estimation of the potentials for complex geometries as well. This implies that we can reduce the box dimensions needed in the EJIIM calculations and therefore make the method more efficient. Moreover, it is a fast and very precise mean for a first estimation of the potential, at least several Å apart from the molecule.

Fig. 10 illustrates the influence of the water network by comparing the nonlocal to the corresponding local potential. Here, a significant difference in the strength of the electric potential can be observed: for the local model, Fig. 10(b) the potential reaches a value of -0.2 V only deep inside the binding pocket, whereas the nonlocal framework yields an interesting structure of the potential in the vicinity of the pocket and a far-reaching positive potential further away. In the local theory, the water shields the electrostatic potential with about 78, but this is overestimated because the water molecules try to align

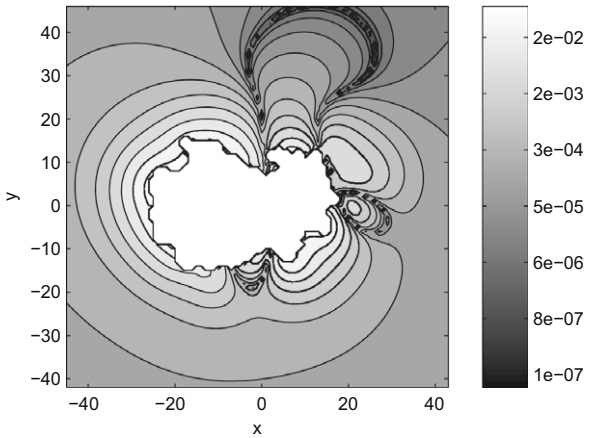

Fig. 9. The difference of ϕ_{Σ} and (15b) evaluated in Σ for an arbitrary cut through the molecule.

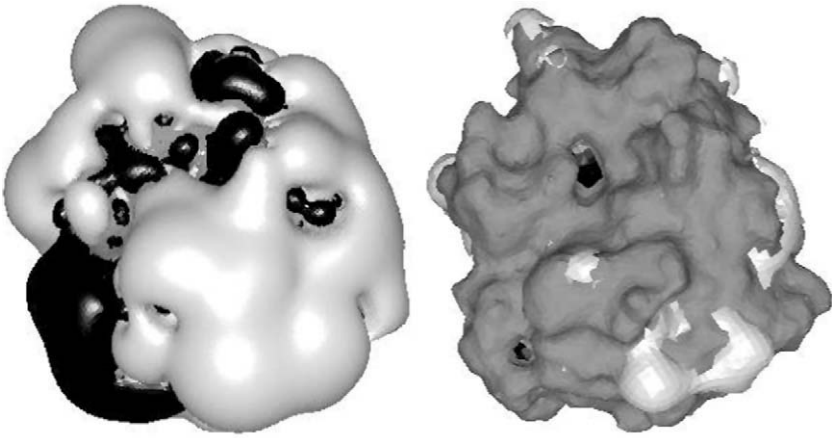

Fig. 10. Isosurfaces of trypsin for $\phi_\Sigma = 0.2\,\mathrm{V}$ (white) and $\phi_\Sigma = -0.2\,\mathrm{V}$ (black).

with their next neighbors and therefore reduce the shielding. The resulting long-range character can have an important influence on the binding process of ligands: on the one hand the ligand "feels" the force farther away. On the other hand ligands can be bound tightly to the binding pocket, since near the surface the nonlocal potential takes values up to one order higher than the local potential.

8.2. Ovalbumin

As a second application we choose the protein ovalbumin, which is used in different areas of research such as immunization and biochemical studies. With its 6000 atoms, Ovalbumin forms a complex surface shape, see Fig. 8(b). A qualitative surface triangulation would have up to 50000 elements. Such a calculation would exceed the memory requirements needed in our BEM implementation preventing its solution. However, with EJIIM we can solve (8): for this we use a grid resolution of 0.5 Å and box dimensions [93 Å × 93 Å × 93 Å]. The boundary values have been calculated with the approximate boundary condition, see Section 6.1.

In Fig. 11 we exemplarily present the isosurfaces of the nonlocal potential at $\pm 0.1\,\mathrm{V}$. Because of the large amount of partial charges on the surface, the potential exhibits a complex, diversified behaviour. Altogether, the positive isosurface is closer to the surface. The molecule has a total charge of $-6e$, and this is why the negative contribution dominates the potential when moving away from the molecule.

The application of nonlocal electrostatics using EJIIM clearly demonstrates that on the one hand the qualitative comparison of the electrostatic potential, based on the different solvent models is worthwhile since the dielectric response influences the characteristics of the system. On the other hand the results underline that an application to biomolecules of arbitrary size and shape is possible with EJIIM. The input generator together with the EJIIM solver can be used to automatically generate the physical potentials. This is an essential step to further studies, especially to the comparison of experimental data.

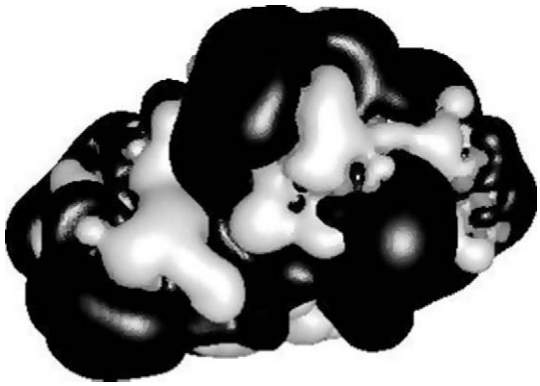

Fig. 11. Isosurfaces of ovalbumin for $\phi_\Sigma = 0.1\,\mathrm{V}$ (white) and $\phi_\Sigma = -0.1\,\mathrm{V}$ (black).

9. Summary and outlook

In this work, we have proposed the Explicit Jump Immersed Interface Method as a suitable alternative to the Boundary Element Method for solving the nonlocal electrostatic equations. The exterior boundary condition necessary for the EJIIM is provided by approximations considering the dielectric response of the solvent only. As our numerical results have shown, these boundary conditions do not spoil the accuracy of the solution. Indeed, this study reveals that the evaluation of the boundary approximation in the whole outer region yields an excellent estimation of the numerical solution.

It is often desirable to have several different methods that solve the same set of equations. On the one hand, the correctness and the accuracy of the methods can be double checked. This is successfully done in Sections 5 and 6. On the other hand, as the methodology of BEM and EJIIM is different, the method of choice depends on the focus of a particular study. If one is interested in the solvation energies of a molecule only, BEM is preferable as the evaluation of the representation formulas have to be done at a few discrete points only – the atom centers. On the other hand, to study dynamics of a molecule driven by electrostatic forces, e.g., we require knowledge about the whole scalar potential. Therefore, a method that solves for the fields in a grid directly would be a good choice. Here, recent progress in BEM has to be mentioned: with an adaptive multipole BEM, Lu et al. succeed in reducing the computational complexity of potential evaluation in a full grid of dimension n^3 (naive implementation: $\mathcal{O}(n^3 \times \text{BEM Elements})$) to an order $\mathcal{O}(n^3 + \text{BEM Elements})$, [22].

Besides the study in terms of accuracy and convergence for a set of small molecules, we solved the nonlocal electrostatic equations for a typical protein and an enzyme, see Section 8. A comparison of the nonlocal and the local electrostatic potentials reveals the "frustrated" character of the water network, that finally causes a higher electrostatic potential on and near the surface. This will certainly have an influence on all physical quantities.

Here, a future challenge is the modelling of nonlocal electrostatics within the nonlinear Poisson Boltzmann theory. Such an extension prevents the use of BEM even when applied to the local framework. Furthermore an intense study of the ion-water correlation that certainly influences the correlation length of the water network might reveal more insight in the diversity of interaction effects which are qualitatively beyond Poisson Boltzmann. This can already be seen in (8), where the water network obtains a similar role as ions in the Poisson Boltzmann theory when screening the electrostatic field.

In addition, one future project is to develop a fully automatic, robust solver for various (non)local electrostatic models freely available in the biomolecular framework BALL [19]. This is necessary to make the nonlocal model useful for the scientific community. An additional task which is essential for further application to biomolecular research, such as solvation studies of biomolecules in their natural surrounding, is the design of a parameter set for the atom type radii optimized for the nonlocal response. Commonly used force fields, [26,35,11,1,39], have parameter sets which work well using a local electrostatic contribution. With a "nonlocal parameter set" at hand physically meaningful quantities such as the solvation energy of molecules but also the binding affinities of small ligands to the protein's binding pocket can be studied and compared to available experimental data.

In summary, with the development of the presented finite difference solver we pave the way for a broader application of nonlocal electrostatics. This study offers new possibilities on the research of biomolecules solved in water: with EJIIM we have the ability to easily extend and change the solvent model in a straightforward way and therefore to advance the knowledge about molecular interaction.

Acknowledgments

SW and AH have been supported by Deutsche Forschungsgemeinschaft (DFG) grants BIZ1/1-4. The authors would like to thank the group of Prof. Louis, University of Saarland, for kindly providing compute resources.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.jcp.2010.01.040.

References

[1] A.D. Mackerell, Empirical force fields for biological macromolecules: overview and issues, Journal of Computational Chemistry 25 (13) (2004) 1584–1604.
[2] C. Azuara, H. Orland, M. Bon, P. Koehl, M. Delarue, Incorporating dipolar solvents with variable density in Poisson–Boltzmann electrostatics, Biophysical Journal 95 (12) (2008) 5587–5605.
[3] Ralf Blossey, Computational biology, first ed., Mathematical and Computational Biology Series, Chapman & Hall/CRC, 2006.
[4] H.-L. Cheng, X. Shi, Quality mesh generation for molecular skin surfaces using restricted union of balls, Computational Geometry 42 (3) (2009) 196–206.
[5] H.L. Cheng, H. Edelsbrunner, Skin.exe. http://biogeometry.duke.edu/software/skin/index.html.
[6] P. Cignoni, Meshlab. http://meshlab.sourceforge.net.
[7] R.R. Dogonadze, A.A. Kornyshev, A.M. Kuznetsov, Phenomenological description of polar media on the basis of an effective hamiltonian, Teoreticheskaya i Matematicheskaya Fizika 15 (1973) 127.
[8] Revaz R. Dogonadze, Erika Kálmán, Alexei A. Kornyshev, Jens Ulstrup, The Chemical Physics of Solvation, Part A: Theory of Solvation, Elsevier Science Ltd., 1985.
[9] C. Fasel, S. Rjasanow, O. Steinbach, A boundary integral formulation for nonlocal electrostatics, in: Numerical Mathematics and Advanced Applications – Proceedings of ENUMATH 2007, Springer, 2008, pp. 117–124.

[10] M. Garland and P. Heckbert. Surface simplification using quadric error metrics, in: SIGGRAPH, 1997.
[11] T.A. Halgren, Merck molecular force field i–v, Journal of Computational Chemistry 17 (1996) 490–640.
[12] J.B. Hasted, Aqueous Dielectrics, Chapman and Hall, London, 1973.
[13] A. Hildebrandt, Biomolecules in a Structured Solvent, Rhombos-Verlag, 2005.
[14] A. Hildebrandt, R. Blossey, S. Rjasanow, O. Kohlbacher, H.-P. Lenhof, Novel formulation of nonlocal electrostatics, Physical Review Letters 93 (10) (2004) 108104.
[15] A. Hildebrandt, R. Blossey, S. Rjasanow, O. Kohlbacher, H.-P. Lenhof, Electrostatic potentials of proteins in water: a structured continuum approach, Bioinformatics 23 (2) (2007) e99–103.
[16] John David Jackson, Classical Electrodynamics, third ed., John Wiley & Sons, Inc., 1998.
[17] C.T. Kelley, Iterative Methods for Linear and Nonlinear Equations, Society for Industrial and Applied Mathematics, Philadelphia, 1995.
[18] Patrice Koehl, Henri Orland, Marc Delarue, Beyond the Poisson–Boltzmann model: modeling biomolecule–water and water–water interactions, Physical Review Letters 102 (8) (2009) 087801.
[19] O. Kohlbacher, H.P. Lenhof, et al., Ball–Biochemical Algorithms Library. http://www.bioinf.uni-sb.de/OK/BALL.
[20] A. Leach, Molecular Modelling: Principles and Applications, Pearson Education, 2001.
[21] Z. Li, The Immersed Interface Method – A Numerical Approach for Partial Differential Equations with Interfaces. Ph.D. Thesis, University of Washington, 1994.
[22] B.Z. Lu, X. Cheng, J. Huang, J.A. McCammon, An adaptive fast multipole boundary element method for Poisson–Boltzmann electrostatics, Journal of Chemical Theory and Computation 5 (6) (2009) 1692–1699.
[23] B.Z. Lu, Y.C. Zhou, M.J. Holst, J.A. McCammon, Recent progress in numerical methods for the Poisson–Boltzmann equation in biophysical applications, Communications in Computational Physics 3 (5) (2008) 973–1009.
[24] M. Marquart, J. Walter, J. Deisenhofer, W. Bode, R. Huber, The geometry of the reactive site and of the peptide groups in trypsin, trypsinogen and its complexes with inhibitors, Acta Crystallographica Section B 39 (4) (1983) 480–490.
[25] Mmff94 validation suite, Merck and Inc., Co., http://ftp.ccl.net/cca/data/MMFF94/index.shtml.
[26] D.A. Pearlman, D.A. Case, J.W. Caldwell, W.S. Ross, T.E. Cheatham, S.E. DeBolt, D.M. Ferguson, G.L. Seibel, P.A. Kollman, Amber, a package of computer programs for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to simulate the structural and energetic properties of molecules, Computation Physics Letters (1995).
[27] Claudia Rocchi, Anna Rita Bizzarri, Salvatore Cannistraro, Water dynamical anomalies evidenced by molecular-dynamics simulations at the solvent-protein interface, Physical Review E 57 (3) (1998) 3315–3325.
[28] V. Rutka, Immersed Interface Methods for Elliptic Boundary Value Problems. Ph.D. Thesis, TU Kaiserslautern, 2005.
[29] V. Rutka, A staggered grid based explicit jump immersed interface method for two-dimensional stokes flows, International Journal for Numerical Methods in Fluids 57 (2008) 1527–1543.
[30] V. Rutka, A. Wiegmann, Explicit jump immersed interface method for virtual material design of the effective elastic moduli of composite materials, Numerical Algorithm 43 (2006) 309-–330.
[31] L. Sandberg, R. Casemyr, O. Edholm, Calculated hydration free energies of small organic molecules using a nonlinear dielectric continuum model, The Journal of Physical Chemistry B 106 (32) (2002) 7889–7897.
[32] J. Schöberl, H. Gerstmayr, R. Gaisbauer, Netgen – automatic mesh generator. http://www.hpfem.jku.at/netgen/.
[33] J.A. Sethian, A. Wiegmann, Structural boundary design via level set and explicit jump immersed interface methods, Journal of Computation Physics 163 (2) (2000) 489–528.
[34] X. Shi, P. Koehl, The geometry behind numerical solvers of the Poisson–Boltzmann equation, Communications in Computational Physics 3 (5) (2008) 1032–1050.
[35] Doree Sitkoff, Kim A. Sharp, Barry Honig, Accurate calculation of hydration free energies using macroscopic solvent models, Journal of Physical Chemistry 98 (1994) 1978–1988.
[36] P.E. Stein, A.G. Leslie, J.T. Finch, R.W. Carrell, Crystal structure of uncleaved ovalbumin at 1.95 A resolution, Journal of Molecular Biology 222 (3) (1991) 941–959.
[37] Olaf Steinbach, Numerische Näherungsverfahren für elliptische Randwertprobleme – Finite Elemente und Randelemente, Advances in Numerical Mathematics, Teubner Verlag/GWV Fachverlage GmbH, Wiesbaden, 2003 (in German).
[38] Godehard Sutmann, Die nichtlokale dielektrische Funktion von Wasser. Ph.D. Thesis, Forschungszentrum Jülich, Institut für Werkstoffe und Verfahren der Energietechnik, 1999.
[39] J. Swanson, S. Adcock, J.A. McCammon, Optimized radii for Poisson–Boltzmann calculations with the amber force field, Journal of Chemical Theory and Computation 1 (3) (2005) 484–493.
[40] M.A. Vorotynsev, A.A. Kornyshev, Electrostatics of Media with Spatial Dispersion, Nauka, Moscow, 1993.
[41] J. Wang, C. Tan, Y.-H. Tan, Q. Lu, R. Luo, Poisson–Boltzmann solvents in molecular dynamics simulations, Communications in Computational Physics 3 (5) (2008) 1010–1031.
[42] R. Clint Whaley, Antoine Petitet, Jack J. Dongarra, Automated empirical optimization of software and the ATLAS project, Parallel Computing 27 (1–2) (2001) 3–35. Also available as University of Tennessee LAPACK Working Note#147, UT-CS-00-448, 2000 %3cwww.netlib.org/lapack/lawns/lawn147.ps%3e.
[43] A. Wiegmann, The Explicit–Jump Immersed Interface Method and Interface Problems for Differential Equations. Ph.D. Thesis, University of Washington, 1998.
[44] A. Wiegmann, Fast Poisson, Fast Helmholtz and Fast Linear Elastostatic Solvers on Rectangular Parallelepipeds. Technical Report LBNL-43565, Lawrence Berkeley National Laboratory, MS 50A-1148, One Cyclotron Rd, Berkeley CA 94720, 1999.
[45] A. Wiegmann, K.P. Bube, The immersed interface method for nonlinear differential equations with discontinuous coefficients and singular sources, SIAM Journal on Numerical Analasys 35 (1) (1998) 177–200.
[46] A. Wiegmann, K.P. Bube, The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions, SIAM Journal on Numerical Analasys 37 (3) (2000) 827–862.
[47] R.J. Zauhar, Smart: a solvent–accessible triangulated surface generator for molecular graphics and boundary element applications, Journal of Computer Aided Molecular Design 9 (1995) 149–159.
[48] Ruhong Zhou, Free energy landscape of protein folding in water: explicit vs. implicit solvent, Proteins: Structure Function and Genetics 53 (2) (2003) 148–161.