Ion Flux Through Membrane Channels—An Enhanced Algorithm for the Poisson-Nernst-Planck Model

WITOLD DYRKA,1 ANDY T. AUGUSSI,2 MALGORZATA KOTULSKA1
1Institute of Biomedical Engineering and Instrumentation, Wroclaw University of Technology, 50-370 Wroclaw, Poland
2School of Life Sciences, Kingston University, Penrhyn Rd, Surrey KT1 2EE, United Kingdom

Received 19 June 2007; Revised 23 December 2007; Accepted 13 January 2008
DOI 10.1002/jcc.20947
Published online 31 March 2008 in Wiley InterScience (www.interscience.wiley.com).

Abstract: A novel algorithmic scheme for numerical solution of the 3D Poisson-Nernst-Planck model is proposed. The algorithmic improvements are universal and independent of the detailed physical model. They include three major steps: an adjustable gradient-based step value, an adjustable relaxation coefficient, and an optimized segmentation of the modeled space. The enhanced algorithm significantly accelerates the speed of computation and reduces the computational demands. The theoretical model was tested on a regular artificial channel and validated on a real protein channel—α-hemolysin, proving its efficiency.


Key words: PNP; nanopore; algorithm optimization; relaxation technique; electrodiffusion

Introduction

The Poisson and Smoluchowski (Nernst-Planck) equations, called in short the PNP model, provide a basis for computing charge fluxes through narrow pores, especially biological ion channels. The PNP electrodiffusion theory can also be applied to artificial nanopores, such as an electronanopore1–3 or a nanopore in a polymer membrane.4 The PNP approach is an averaged theory, applicable under steady-state conditions for a large number of atoms over a long time scale relative to atomic and molecular processes. The theory is based entirely on the continuum approach to charge transport, which permits the inference of analytical laws relating the flux to pore structure.5 Importantly, it provides the fastest computational approach to transport phenomena on the nanometer and subnanometer scale. There are other alternative computational methods to the same problem, such as molecular dynamics (MD), which treats the pore and ions in a fully discrete way, and Brownian dynamics (BD), which treats the pore and the solute in a continuous manner and the ions discretely. The methods that utilize a discrete approach to the problem increase the computational accuracy. However, their main disadvantage is a very high computational cost, especially in MD, which greatly increases the computational time, making impossible the observation of phenomena above a time scale of the order of nanoseconds. This limitation hampers experimental validation of the results. On the other hand, extrapolating the solution to longer times or employing coarse-grained MD may affect the accuracy. Hence, although the accuracy of the PNP approach for channels of subnanometer size is limited, it is still the most efficient method of computation, especially for larger pores. Recently, the efforts of researchers have also been put into atom-scale supplements to the basic PNP theory.6,7

The PNP model is constituted of two equations, defining all intensive (size-independent) and extensive quantities. The flux density \( J \) is defined in the model by the Nernst-Planck equation:

\[
J = -D \nabla n + n \nabla \Phi \frac{ze}{kT}
\]  

where \( D \) is a diffusion constant, \( k \) is the Boltzmann constant, \( n \) is the ionic density, \( \nabla n \) is the ionic density gradient, \( \nabla \Phi \) is the gradient of the electric potential, \( z \) represents the ionic charge, \( e \) is the elementary charge, and \( T \) is the temperature in Kelvin. The electric potential in the pore, dependent on the accumulated charge, is defined by the Poisson equation as follows:

\[
\varepsilon_0 \nabla \cdot [\varepsilon(r) \nabla \Phi(r)] = -e \sum \nabla c \frac{z_i n_i(r) - \rho}{\varepsilon_0}
\]  

where \( \varepsilon_0 \) is the permittivity of free space, \( \varepsilon(r) \) is the permittivity of the medium, and \( \rho_{ex} \) represents external charges embedded in the pore walls. Summation over \( v \) takes into account the variety of ion species flowing through the pore.

Correspondence to: M. Kotulska; e-mail: kotulska@pwr.wroc.pl

© 2008 Wiley Periodicals, Inc.
Because of its nonlinearity, the PNP model has no general analytical solution, although under specific conditions some special cases have been solved. An example is the well-known Goldman-Hodgkin-Katz (GHK) equation, derived in the limited case of a constant electric field applied across a planar membrane, reducing the PNP model effectively to a single dimensional dependence.\textsuperscript{8,9} The same solution arises also in the limit of low ionic concentration or for a sufficiently short channel in an arbitrary electric field. The general PNP model can only be solved numerically, typically by finite difference or finite element methods.\textsuperscript{10}

The PNP model, although most efficient, is nonetheless still computationally demanding. There are two methods for improvement of such costly algorithms. The first approach involves modification of the physical model so that a new mathematical representation would reduce the computational complexity of its implementation. The second solution refers to purely algorithmic optimization, with no regard to the physical model itself. In the latter case, a model that is physically sufficiently accurate as a representation, and which is tried and proven, can be used; however it can be computationally enhanced. In this article, we adopt this latter approach and propose several algorithmic improvements to the numerical algorithm, which, when applied to the solution of the PNP model very significantly increase the computational effectiveness, without loss of accuracy of the solution. The method presented in this paper was applied to the standard 3D PNP model described by Corry et al.,\textsuperscript{11} however it can also be implemented in other, more complex, versions of the PNP theory. The accuracy of the computationally enhanced model was tested on a theoretical channel and a real α-hemolysin protein channel obtained from Protein Data Bank.

Methods

Basic Algorithm

The standard computational PNP algorithm is based on the discrete versions of eqs. (1) and (2), calculated over a system discretized into a three-dimensional rectangular grid of points over the pore, membrane, and reservoirs (Fig. 1). The grid defines cells of dimensions: \(h_x \times h_y \times h_z\). The Nernst-Planck equation in its discrete form reads as follows:

\[
J_i = -D \left( \frac{n_i - n_{i,old}}{h_j} + \frac{n_i + n_j}{2h_j} \cdot \frac{\Phi_i - \Phi_j}{h_j} \cdot \frac{ze}{kT} \right), \tag{3}
\]

where \(J_i, i = 1, \ldots , 6\) denotes the flux through each of the six cell walls. Equation (3) under steady-state conditions, i.e., \(\nabla J = 0\), which takes into account a position-dependent diffusion constant,\textsuperscript{12} gives a formula for the density \(n_i\) at the central grid point:

\[
n_i = \frac{\sum_{j=1}^{6} n_j \left[ \frac{6D}{h_j} \right] \left[ 1 + (ez/2kT)(\Phi_j - \Phi_i) \right]}{\sum_{j=1}^{6} \left[ \frac{6D}{h_j} \right] \left[ 1 - (ez/2kT)(\Phi_j - \Phi_i) \right]} \tag{4}
\]

Since there is no flux through the boundary, only the cells adjoining the pore are included in the calculations.

The discrete Poisson equation has the following form:

\[
e_0 \sum_{j=1}^{6} \Phi(r_j + h_j) - \Phi(r_i) \cdot \frac{V}{h_j} = -eV \sum_v z_v n_i(r_i) - \rho_v(r_i) \cdot \frac{v}{h_j} \tag{5}
\]

where \(r_i\) is a central grid point of the cell, \(\vec{r}\) is a unit vector normal to the plane dividing cells \(i\) and \(j\), \(v\) represents an ion species, and \(V\) denotes each cell volume. The potential \(\Phi_i (= \Phi(r_i))\) can be derived from [Eq. (5)], giving the following formula:

\[
\Phi_i = \frac{\sum_v e_{i\Phi} \cdot \Phi_i / h_j^2 + \sum_v z_v e_{n_i} / e_0 + q_i / (e_0 V)}{\sum_j e_{i\Phi} / h_j^2} \tag{6}
\]

Equations (4) and (6) are solved simultaneously, according to an interchangeable iterative scheme. The iteration starts with some initial guesses concerning the values of concentration and potential, and proceeds until both variables converge to stable values. In each iteration step, both variables are updated in the same manner as follows:

\[
n_{i,new} = w_a \cdot n_{i,old} + (1 - w_a) \cdot n_{i,calc} \tag{7a}
\]

\[
\Phi_{i,new} = w_\Phi \cdot \Phi_{i,old} + (1 - w_\Phi) \cdot \Phi_{i,calc} \tag{7b}
\]

Figure 1. The computational space including pore, membrane, and reservoirs discretized into a rectangular grid of points. Transverse section in the (a) \(xy\) plane at the midpoint. (b) \(yz\) plane at the midpoint. The pore is symmetric in the \(y\) and \(z\) directions.
where \( n_i \) denotes the concentration and \( \Phi_i \) denotes the potential in the \( i \)th cell, \( n_{\text{old}} \) and \( \Phi_{\text{old}} \) are values from the previous iteration step, \( n_{\text{calc}} \) and \( \Phi_{\text{calc}} \) are values derived from eqs. (4) and (6), and \( n_{\text{new}} \) and \( \Phi_{\text{new}} \) are values finally accepted for further calculation. A value of the convergence coefficient \( w_n \) is typically assumed as a constant value from the range \((0, 1) \) or \((-1, 1) \) if over-relaxation is acceptable. The iteration stops if both differences, \( n_{\text{diff}} \) and \( \Phi_{\text{diff}} \), between subsequent values of both variables, i.e., \( n_{\text{old}}, n_{\text{calc}} \) and \( \Phi_{\text{old}}, \Phi_{\text{calc}} \), respectively, change below the assumed maximal limit \( n_{\text{diffMAX}} \) (\( \Phi_{\text{diffMAX}} \)):

\[
\begin{align*}
    n_{\text{diff}} &= \max(|n_{\text{calc}} - n_{\text{old}}|) < n_{\text{diffMAX}} \quad (8) \\
    \Phi_{\text{diff}} &= \max(|\Phi_{\text{calc}} - \Phi_{\text{old}}|) < \Phi_{\text{diffMAX}} \quad (9)
\end{align*}
\]

Based on the standard algorithm described earlier, three steps of the algorithmic optimizations are proposed to be included into the iteration process: an adjustable gradient-based step value, an adjustable relaxation coefficient, and an optimized segmentation of the space. Each of the algorithmic improvements alone enhances the computational performance—their combination allows for a significant reduction of the computational demands.

**Adaptive Gradient-Based Optimization of Step-Size**

Super Relaxation

Analysis of the results obtained from the PNP model by the standard finite difference method, described in the previous section, reveals that values of the electrical potential and ionic concentration change monotonically (and often almost linearly) during the longest period of the convergence process (Fig. 2, solid line). This could be exploited in a gradient-based optimization of the algorithm, termed “Jp” here (short for “jumping”), because it extrapolates from the existing values and thereby “jumps” over some intervening iterations. Gradient-based algorithms take advantage of a monotonic convergence of the computational results by incorporating elements of prediction, based on the previous values. It is performed by introducing an adjustable variable, which is recurrently added to the computationally obtained results, enhancing the convergence of the algorithm.

In this implementation, the linear gradient is calculated for each point of the grid, after an explicitly chosen number of iterations \( M \), and then it is utilized to obtain a linear approximation for the concentration and potential values after a number of steps defined by a variable termed here as jump (the algorithm presented in Fig. 3, the result in Fig. 2 – dashed line). During the process, the jump parameter may be tuned to find an appropriate compromise between speed and stability. For example, as the monotonic character of the convergence is weak in the beginning of the computational process and then increases (Fig. 2), the jump value may also increase at the later stage. In our implementation, this correction is achieved by using an adaptive scheme accommodating to changing values of \( c_{\text{diff}} \) and \( \Phi_{\text{diff}} \) defined in eqs. (8) and (9). Ionic concentration \( c \) (in moles/liter) is related to the number density of ions \( n \) (in SI units) through the following equation:

\[
c = \frac{n}{1000 \cdot N_A}.
\]

To find a proper value of jump, \( c_{\text{diff}} \) and \( \Phi_{\text{diff}} \) values are processed in the following way. First, the value of \( c_{\text{diff}} \) is standardized:

\[
c_{\text{diff}}(N) = c_{\text{diff}}(N) \cdot \Phi_{\text{diffMAX}} / c_{\text{diffMAX}} \quad (11)
\]

Then, after a cycle of \( M \) iterations (Fig. 3-D), the following expressions are calculated:

\[
c_{\text{diff1}} = c_{\text{diff}}(N) - c_{\text{diff}}(N - M) \quad (12)
\]

\[
\Phi_{\text{diff1}} = \Phi_{\text{diff}}(N) - \Phi_{\text{diff}}(N - M) \quad (13)
\]

\[
c_{\text{diff2}} = c_{\text{diff}}(N - M + 1) - c_{\text{diff}}(N - M) \quad (14)
\]

\[
\Phi_{\text{diff2}} = \Phi_{\text{diff}}(N - M + 1) - \Phi_{\text{diff}}(N - M). \quad (15)
\]

Subsequently, expressions for electric potential and ionic concentration are averaged, resulting in two values, which are utilized to calculate the length of the next jump (Fig. 3-E):

\[
diff1 = (c_{\text{diff1}} + \Phi_{\text{diff1}})/2 \quad (16)
\]

\[
diff2 = (c_{\text{diff2}} + \Phi_{\text{diff2}})/2. \quad (17)
\]

After that, the value of the parameter jump is modified adaptively as presented in Figure 3; this process is described in more detail later. Finally, the gradient is calculated and used to extrapolate further jump steps, as illustrated below.
\[ c_{\text{JUMP}}(N) = c_i(N) + \text{jump} \left( \frac{c_i(N) - c_i(N-M)}{M} \right) - \text{for each ion species} \]  

\[ \Phi_{\text{JUMP}}(N) = \Phi_j(N) + \text{jump} \left( \frac{\Phi_j(N) - \Phi_j(N-M)}{M} \right) \]

An appropriate choice of the conditions controlling the jump length is a key factor for efficient and stable functioning of the algorithm. The first condition [eqs. (12), (13), (16) and Fig. 3-EF] states that jump can be increased (Fig. 3-HJ) only if the computational process seems to be converging, i.e., the average maximal difference of the potential and concentration values declines in relation to \( M \) iterations back; in other words, \( \text{diff}_i < 0 \) (Fig. 3-F). In the opposite case, jump length should be shortened (Fig. 3-GI). Lengths of single incrementations of the jump values are defined by \text{jumpUp} and \text{jumpDn} parameters, which should be adjusted to the maximal jump length to preserve both the speed and stability of the algorithm. However,
the first condition is not sufficient because it is only based on the change in the sign of the difference. In cases when a large increase in $\text{diff}_1$ is followed by its small decrease, after jump reduction, the process of convergence may become unstable. One possible solution is to consider not only the signs but also the values of $\text{diff}_1$, but such an approach demands a formula relating $\text{diff}_1$ to jump length. Instead, we can introduce a second condition for the jump increase [eqs. (14), (15), (17) and Fig. 3-EF].

It is fulfilled only if the average maximal differences of the potential and concentration values just after a jump decline in relation to the same quantity just before the jump. This condition seems to be quite tough but exhaustive analyses proved its benefits for the stability of the process and showed that the convergence did not deteriorate, in some cases being even faster. Another alternative way to minimize the danger of instability can be achieved by setting a value of the parameter jumpUp permanently lower than a value of jumpDn.

The simultaneous convergence progress of the mutually dependent electric potential and ionic concentration has to be well balanced; therefore we set a common jump length for both. The condition stating that jump can be increased only if both maximal differences $c_{\text{diff}}$ and $\Phi_{\text{diff}}$ are declining appears to be too conservative. The average value of the (standardized) differences is more suitable and can be recognized as a good indicator of the overall state of the convergence process.

Indeed, our observations confirm that the process is at a serious risk only if $c_{\text{diff}}$ and $\Phi_{\text{diff}}$ are both rapidly growing. Another simple but efficient protection against instability is an upper limit imposed on jump length. The variables affecting single step sizes, namely jumpUp and jumpDn, should be customized to this value.

**Adjustable Relaxation Coefficient**

The relaxation technique in a finite difference, PNP solver demands a careful choice of the coefficient $w$ [eq. (7)] and an adaptive scheme for adjustment of the relaxation coefficient (termed “Rx” here) can be introduced. A method similar to the gradient-based method could be applied (Fig. 3a). After standardization of the value of $c_{\text{diff}}$ [eq. (11)], expressions for the first convergence condition, which are identical to the Jp scheme, are calculated according to eqs. (12) and (13).

The main difference between the Jp and Rx methods appears in an additional condition for the relaxation coefficient $w$ correction (Fig. 3a-L). Equations (14, 15), used in the basic and gradient-based algorithm, are modified accordingly:

$$c_{\text{diff}2} = c_{\text{diff}}(N) - c_{\text{diff}}(N - M + 2) \tag{20}$$

$$\Phi_{\text{diff}2} = \Phi_{\text{diff}}(N) - \Phi_{\text{diff}}(N - M + 2) \tag{21}$$

After that, as in the case of the adaptive gradient-based optimization of the step, expressions for electric potential and ionic concentration maximal differences are averaged [eqs. (16) and (17)] and two values obtained, $\text{diff}_1$ and $\text{diff}_2$, are utilized to optimize the value of coefficient $w$ (Fig. 3a).

The first condition [eqs. (12), (13), (16) and Figure 3a-LM] has a function analogous to the function described in the previ-
ous section for the Jp method: \( w \) can be decreased (Fig. 3a-PQ) only if the average maximal difference of the potential and concentration values declines relative to \( M \) iterations back. In the opposite case \( w \) should be increased (Fig. 3a-NO).

The second condition (Fig. 3a-P) is designed to respond to changes in \( c_{\text{diff}} \) and \( F_{\text{diff}} \) occurring between jumps; therefore it enables the reduction of \( w \) only if its recent value led to a decline of the average maximal difference of the potential and concentration values in relation to the first PNP calculation after the last jump.

Although such conditions do not always prevent the process from unstable behavior, the instability can be avoided by setting up the single step up (\( w_{\text{Up}} \)) significantly smaller than the step down (\( w_{\text{Dn}} \)). It provides the algorithm with the capacity for a rapid response in the case of divergence.

Although allowing the relaxation parameter to change independently for the concentration and potential may appear desirable with regard to the optimal efficiency, in general this leads to instability and divergence (Fig. 4). This can be avoided if the relaxation parameter for both the concentration and potential is also modified here by averaging eqs. (16) and (17). It should be noted that the adaptive scheme described earlier has one significant drawback, i.e., it slows down the process of passing through intervals by temporarily increasing \( c_{\text{diff}} \) and \( F_{\text{diff}} \). A reasonable solution against this effect is to lower the upper limit for \( w \).

**Space Segmentation**

Dividing the space grid into segments is the third method of optimization applied to the PNP solver in this work. Faster convergence can be obtained due to an appropriately chosen grid precision and solver parameters specific for each segment. This approach might be especially useful for very asymmetric pores and for regions demanding very precise modeling (Fig. 5a). It also allows for application of different PNP solvers for each segment (e.g., constant or variable diffusion coefficient, solvation taken into account or not\(^{6,14}\)) as more sophisticated and hence more computationally expensive solvers are necessary only for very narrow or irregular domains.

A key feature of the PNP calculations including segmentation is ensuring continuity between adjacent domains. This can be achieved by imposing new boundary conditions resulting from calculations in the neighboring segments. The simplest way is to solve a model for a coarser grid and then impose boundary conditions obtained in this manner for finer calculations in an inner region (Fig. 5a). More general solutions presume continuity after every calculation step for each segment. New boundary conditions are to be applied to all neighboring or only finer-precision domains (suitable for both cases in Fig. 5). An overall scheme for a solver with segmentation-based optimization is presented in Figure 6. In accordance with this scheme, we applied and tested three different versions of the procedure:

- **Inner-Outer Sequential (IOS):** an outer coarser segment is calculated, then boundary conditions for a finer inner segment are determined and the latter is calculated (grid partitioning as in Fig. 5a);
- **Inner-Outer Parallel (IOP):** outer and inner segments are calculated in parallel. After every calculation in the outer segment boundary conditions are imposed on the inner segment (grid partitioning as above);
- **Left-Right Parallel (LRP):** there are two neighboring segments of different grid precisions; the continuity condition is imposed after every calculation of the potential or concentration, and boundary conditions are updated for both segments (grid partitioning as in Fig. 5b).

**Results**

**Convergence Optimization**

In this section, the gradient-based optimization and the adaptive scheme for the relaxation coefficient will be examined and evaluated in terms of their efficiency, reliability, and mutual effect
when applied for a theoretical channel with high regularity and fixed diffusion coefficient. Table 1 summarizes the model and algorithm parameters. In Table 1, a 6 nm diameter pore (Fig. 7a) models a nanopore induced in an artificial membrane, e.g., by an electroporation technique, while 0.6 nm diameter pore (Fig. 7b) can be regarded as a very simplified model of a protein channel. A surface charge of $\pm 1e$ can be obtained for an artificial nanopore by appropriate choice of a solvent pH. $D = 2.0 \times 10^{-9}$ m$^2$/s is an approximate value of the diffusion coefficient of potassium and chloride ions in a free solvent. Assumed ion concentrations in reservoirs and the electric potential on the membrane are physiological values often applied in experiments with biological membranes. The results are summarized in Table 2.

**Efficiency**

The results shown in Table 2 indicate that the methods introduced lead to significant improvement of the PNP solver efficiency. In the best case, for the algorithm with both improvements—the gradient-based and relaxation optimization—the number of PNP calculations needed to fulfill the stop conditions was reduced fivefold. In many cases, this result could be easily further improved by lowering the upper limit for the relaxation coefficient.

**Robustness**

The ultimate evaluation of the optimization methods depends on their reliability, which is shown to be high. The adaptive relaxation introduced maximal deviations of the electric potential and ionic concentrations with reference to the basic algorithm at the level of the stop conditions ($1 \times 10^{-6}$ V and M). The application of the gradient-based procedure led to deviations not greater than 0.5% for the electric potential and 0.1% of the ionic concentrations (maximal deviations appeared for a combination of the two optimization techniques). As modeling of ionic currents is one of the most important aims of the PNP models and current values are most vulnerable to changes of the potential or concentration profile, we regard it as a good measure of the reliability of the method. Similarly to the potential and concentration profiles, deviations of ion currents were also smallest for the adaptive relaxation alone (data not shown). However, intro-
duction of the gradient-based jump scheme never led to differences higher than 0.6% (and usually lower than 0.1%) for the ionic current. This means, for example, that deviations for the fastest algorithm after all 462 steps (wide pore without surface charges) are 10-fold lower than those for the basic algorithm after 1000 steps (data not shown).

Co-operation

The tests conducted on the regular channel revealed that both optimization schemes—the gradient-based adaptive jump and the adaptive relaxation cooperate rather than compete with each other. For example, analysis of the wide channel (6 nm) showed that the relaxation coefficient for the algorithm utilizing both improvements was approximately twofold lower than that for the algorithm with the adaptive relaxation only. Also, moving from the gradient-based method to combined optimization did not affect jump lengths significantly. However, for more irregular channels the situation is different. Then, the adaptive jump often has a greater impact on improving the efficiency and in some conditions it is comparatively effective alone (see the section on α-hemolysin protein channel).

The co-operation of the schemes dealing with the danger of instability on a theoretical channel can be observed in Figure 8.

The analysis of the narrow channels confirms this finding. Figure 9 shows that application of the gradient-based jumps condenses the convergence curves rather than changes their shapes. However, this also demonstrates the requirement for any adaptive scheme relating to relaxation to be able to deal efficiently with nonmonotonic differences in consecutive iterations of the potentials and concentrations.

Table 1. Model and Algorithm Parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Pore 1</th>
<th>Pore 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pore and membrane</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>$L = 6.1$ nm</td>
<td>$L = 6.1$ nm</td>
</tr>
<tr>
<td>Smallest diameter</td>
<td>$\phi = 6.0$ nm</td>
<td>$\phi = 0.6$ nm</td>
</tr>
<tr>
<td>Surface charge</td>
<td>$q = 0$ or $+1e$/nm$^3$</td>
<td>$q_{\text{protein}} = 2$</td>
</tr>
<tr>
<td>Dielectric coefficient</td>
<td>$\varepsilon_{\text{membrane}} = \varepsilon_{\text{protein}} - 2$</td>
<td></td>
</tr>
<tr>
<td>Diffusion coefficient</td>
<td>$D = 2.0 \times 10^{-9}$ m$^2$/s (const)</td>
<td></td>
</tr>
<tr>
<td>Ion species</td>
<td>$+1$ and $-1$</td>
<td></td>
</tr>
<tr>
<td>Ion concentrations</td>
<td>$c_L = 0.1$ M; $c_R = 0.01$ M</td>
<td></td>
</tr>
<tr>
<td>Electric potential</td>
<td>$V = 100$ mV</td>
<td></td>
</tr>
<tr>
<td>Model</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Grid</td>
<td>$37 \times 40 \times 40$</td>
<td>$101 \times 39 \times 39$</td>
</tr>
<tr>
<td>Cell</td>
<td>$(0.5$ nm$)^3$</td>
<td>$(0.1$ nm$)^3$</td>
</tr>
<tr>
<td>Shape</td>
<td>Figure 7c</td>
<td>Figure 7b</td>
</tr>
<tr>
<td>Cross-section</td>
<td>Circular</td>
<td></td>
</tr>
<tr>
<td>Algorithm parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Steps to adaptive adjustment</td>
<td>$M = 3$</td>
<td></td>
</tr>
<tr>
<td>Relaxation coefficient range</td>
<td>$0.025 \leq w \leq 0.950$</td>
<td>$0.025 \leq w \leq 0.500$</td>
</tr>
<tr>
<td>Relaxation coefficient step</td>
<td>$w_{\text{Up}} = 0.025$; $w_{\text{Dn}} = 0.050$</td>
<td></td>
</tr>
<tr>
<td>Gradient jump range</td>
<td>$0 \leq \text{jump} \leq 10$</td>
<td></td>
</tr>
<tr>
<td>Gradient jump step</td>
<td>$\text{jump}<em>{\text{Up}} = \text{jump}</em>{\text{Dn}} = 1$</td>
<td></td>
</tr>
<tr>
<td>Stop condition</td>
<td>$c_{\text{diffMAX}} = 10^{-6}$ M; $\Phi_{\text{diffMAX}} = 10^{-6}$ V</td>
<td></td>
</tr>
</tbody>
</table>

Figure 7. Transverse sections (cy planes at midpoints) of the pore models utilized for tests described in this section. (a) 6 nm pore; computational cell of $(0.5$ nm$)^3$ and (b) 0.6 nm pore; computational cell of $(0.1$ nm$)^3$.
Space Segmentation

In this section, the segmentation-based optimization will be examined and evaluated in terms of its efficiency and reliability. An asymmetric pore of length 6.0 nm and with a smallest diameter of 6.0 nm was modeled on a grid of dimensions $36 \times 40 \times 40$ units and with computational cells of size $(0.5 \text{ nm})^3$. The concentrations of both ions ($\text{1e}$ and $\text{2e}$) were 0.1 and 0.01 M, respectively, and the electric transmembrane potential was 20.5 V. No surface charge was included. The relaxation coefficient $w$ ranged from 0.025 to 0.950. Other parameters were the same as for the wide channel in the previous section. A combined adaptive scheme was utilized in these calculations. Three versions of the segmentation described in the Methods section were tested, and the version with only one compartment (Basic) was used as a reference. In Inner-Outer versions, on the basis of the original model, the coarser outer grid with coordinates ($[1; 1; 1], [36; 40; 40]$ in reference to the original grid, Fig. 10) and cell size of 1 nm$^3$ and the finer inner grid with coordinates ($[9; 11; 11], [28; 30; 30]$ (IOSa and IOPa in the Table 3, Figs. 10a and 10b) or $[7; 7; 7], [30; 34; 34]$ (IOSb and IOPb)) and cell size (0.5 nm$^3$) were created. For Left-Right version, the original model was segmented into a coarser right domain with coordinates ($[17; 1; 1], [36; 40; 40]$ (LRPa, Figs. 10b and 10c) or $[15; 1; 1], [36; 40; 40]$ (LRPb)) and cell size of 1 nm$^3$ and a finer left domain with coordinates ($[1; 11; 11], [20; 30; 30]$ (LRPa, Figs. 10b and 10c) or $[1; 7; 7], [22; 34; 34]$ (LRPb) and cell size of (0.5 nm)$^3$. For the IOS version, there are two independent stages for calculations (for the inner and outer segment), therefore numbers of grid points and size $\times$ steps quantities in the Table 3 are summed for both of the stages. Ion currents deviations for IOS, IOP, and LRP are calculated with the reference to the basic algorithm. The results are summarized in Table 3.

Efficiency and Robustness

The results presented and other tests (not shown) indicate that the segmentation-based optimization can lead to a significant (up

Table 2. Comparison of the efficiency and reliability of optimization methods.

<table>
<thead>
<tr>
<th>Pore type</th>
<th>Basic</th>
<th>Jp</th>
<th>Rx</th>
<th>Jp+Rx</th>
<th>$\Delta V_{\text{MAX}}$ (M)</th>
<th>$\Delta I_{\text{MAX}}$ (V)</th>
<th>$\delta_{I+}$ (%)</th>
<th>$\delta_{I-}$ (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi = 6 \text{ nm}; q = 0$</td>
<td>2,061 (100%)</td>
<td>762 (37%)</td>
<td>1,299 (63%)</td>
<td>462 (22%)</td>
<td>$10^{-3}$ (&lt;1%)</td>
<td>$10^{-4}$ (&lt;0.5%)</td>
<td>0.57%</td>
<td>0.08%</td>
</tr>
<tr>
<td>$\phi = 6 \text{ nm}; q = 1e/\text{nm}^3$</td>
<td>2,088 (100%)</td>
<td>720 (34%)</td>
<td>1,488 (71%)</td>
<td>408 (20%)</td>
<td>$10^{-3}$ (&lt;0.1%)</td>
<td>$10^{-2}$ (0.1%)</td>
<td>&lt;0.01%</td>
<td>0.02%</td>
</tr>
<tr>
<td>$\phi = 0.6 \text{ nm}; q = 0$</td>
<td>9,162 (100%)</td>
<td>3,141 (34%)</td>
<td>4,836 (53%)</td>
<td>1,722 (19%)</td>
<td>$10^{-3}$ (&lt;0.1%)</td>
<td>$10^{-6}$ (&lt;0.1%)</td>
<td>0.04%</td>
<td>0.02%</td>
</tr>
<tr>
<td>$\phi = 0.6 \text{ nm}; q = 1e/\text{nm}^3$</td>
<td>9,858 (100%)</td>
<td>3,546 (36%)</td>
<td>5,319 (54%)</td>
<td>2,073 (21%)</td>
<td>$10^{-3}$ (&lt;0.1%)</td>
<td>$10^{-6}$ (&lt;0.1%)</td>
<td>0.03%</td>
<td>0.02%</td>
</tr>
</tbody>
</table>

Basic, basic algorithm; Jp, adaptive jump only; Rx, adaptive relaxation coefficient only; Jp+Rx, both. Maximal deviations of the electric potentials, ion concentrations, and maximal relative deviations of currents (referring to the basic algorithm) occurred for algorithm number 4. Deviations of $\varepsilon$ and $\Phi$ are coarse values. Ionic current calculated as the average over the whole pore length.

Figure 8. Convergence process for the wide channel without surface charges. The effect of the schemes dealing with the danger of instability is shown, i.e., as the differences of $c_{\text{diff}}$ increases, the values of jump and $w$ (common) change adaptively to continue to minimize the differences once more. In other words, nonmonotonicity of the profiles for $c_{\text{diff}}$ and $\Phi_{\text{diff}}$ is not an inherent difficulty, and does not necessarily lead to divergence.

Figure 9. Convergence process for the narrow channel with surface charges for the algorithm with the adaptive relaxation only (bold lines) and with both optimization schemes (thin lines). The gradient-based jumps condense the convergence curves.
to fourfold) decrease in computational expenses while still obtaining reliable results. The smallest deviations for cationic currents were observed for the algorithm in Inner-Outer Sequential version, while the same was true for anionic currents for the Inner-Outer Parallel version. In fact, deviations up to about 0.5% are at the same level as for the adaptive scheme (see Table 2), hence are insignificant. Although parallel calculations in segments did not prove their superiority over the algorithm in sequential version (IOS), further tests are necessary for more complex models to evaluate it. Relatively high deviations for the Left-Right Parallel version (two neighboring segments) could be reduced by increasing an overlapping region size (as shown in Fig. 5b), to avoid strong distortions stimulating each other at boundaries. It should be noted that although high deviations of a few percent for values of the ionic currents cannot be neglected, simplifications inherent to the PNP models, regarding structure and interactions, can lead to much higher deviations. They can be even 10-fold higher than those introduced by the algorithmic improvements introduced here.

**Model Validation by α-Hemolysin Protein Channel**

The data for testing of the computationally enhanced model on a real protein channel was obtained from Protein Data Bank. The α-hemolysin protein, which forms a heptameric channel (234 kDa), is available with the accuracy of 1.9 Å under the code 7AHL. The channel has a mushroom-like shape ~100 Å long, with the aqueous pore ranging from 15 to 46 Å in diameter. Such a wide pore makes it appropriate for PNP modeling. Moreover, due to its properties the channel is considered as a very good candidate for protein channel-based biosensors. Its conductivity has also been studied by molecular dynamics.

Our computational tests used α-hemolysin protein to validate the enhanced PNP model. The simulations were focused on testing a potential divergence between conductivity characteristics obtained from the standard PNP model and the computationally accelerated model. Additionally, the computational time necessary for the simulations on the real protein channel by the standard and enhanced algorithms has been recorded and compared.

Tests have been conducted on two modes of the histidine protonation. The first protonation mode was based on Noskov et al., in which His-35 and 259 were protonated on NE2, His-144 was protonated on ND1, and His-48 was protonated at both NE2 and ND1. The latter protonation mode was based on Aksimentiev and Schulten, in which all above histidines were protonated on NE2. Missing atomic coordinates of α-hemolysin were left out as located distantly from the pore region. The protein was embedded into the membrane. The atomic charges were obtained by means of the program PDB2PQR based on the force field CHARMM27. The channel profile with the charge distribution and preliminary determination of the space accessible for ions have been obtained by means of APBS program.

**Table 3. Comparison of Efficiency and Reliability of Segmentation Schemes.**

<table>
<thead>
<tr>
<th>Alg. version</th>
<th>No. of grid pts.</th>
<th>Steps to converge</th>
<th>Size × steps</th>
<th>$I_+$ (pA)</th>
<th>$\delta I_+$ (%)</th>
<th>$I_-$ (pA)</th>
<th>$\delta I_-$ (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic</td>
<td>57.6k</td>
<td>463</td>
<td>26.7 M (100%)</td>
<td>375.8</td>
<td>–</td>
<td>135.3</td>
<td>–</td>
</tr>
<tr>
<td>IOS a</td>
<td>15.2k</td>
<td>778+210*</td>
<td>7.3 M (27%)</td>
<td>371.9</td>
<td>−1.0%</td>
<td>136.0</td>
<td>0.5%</td>
</tr>
<tr>
<td>IOS b</td>
<td>26.0k</td>
<td>591+259*</td>
<td>9.1 M (34%)</td>
<td>374.0</td>
<td>−0.5%</td>
<td>136.1</td>
<td>0.6%</td>
</tr>
<tr>
<td>IOP a</td>
<td>15.2k</td>
<td>492</td>
<td>7.5 M (28%)</td>
<td>367.0</td>
<td>−2.3%</td>
<td>134.9</td>
<td>−0.3%</td>
</tr>
<tr>
<td>IOP b</td>
<td>26.0k</td>
<td>492</td>
<td>12.8 M (48%)</td>
<td>370.8</td>
<td>−1.3%</td>
<td>135.3</td>
<td>&lt;0.1%</td>
</tr>
<tr>
<td>LRP a</td>
<td>12.0k</td>
<td>521</td>
<td>6.3 M (24%)</td>
<td>405.0</td>
<td>7.8%</td>
<td>142.9</td>
<td>5.6%</td>
</tr>
<tr>
<td>LRP b</td>
<td>22.0k</td>
<td>548</td>
<td>12.1 M (45%)</td>
<td>389.3</td>
<td>3.6%</td>
<td>135.2</td>
<td>&lt;0.1%</td>
</tr>
</tbody>
</table>

*For IOS versions, the first figure refers to the number of steps to converge in outer segments, while the second figure refers to the same quantity for inner segments.

Figure 10. Illustration of a segmentation of the pore model. Areas covered by coarser grids are marked with dots, while areas covered by finer grids are marked with skewed lines. Transverse sections of (a) the IOSa and IOPa models in the $xy$ plane at the midpoint. (b) The IOSa, IOPa, and LRPa models in the $yz$ plane, as indicated in (a, c). (c) The LRPa model in the $xy$ plane at the midpoint.
Table 4. Comparison of the Efficiency and Reliability of Optimization Methods for the Real Protein Channel α-Hemolysin.

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Basic</th>
<th>Jp</th>
<th>Rx</th>
<th>Jp+Rx</th>
<th>I+ (pA)</th>
<th>$\delta I_+ (%)$</th>
<th>I− (pA)</th>
<th>$\delta I_− (%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>−1.200 mV; 1.0 M</td>
<td>9,053</td>
<td>4,515</td>
<td>7,309</td>
<td>4,630</td>
<td>−565</td>
<td>&lt;0.01%</td>
<td>−559</td>
<td>&lt;0.01%</td>
</tr>
<tr>
<td>−150 mV; 1.0 M</td>
<td>12,008</td>
<td>6,025</td>
<td>9,657</td>
<td>5,960</td>
<td>−82</td>
<td>0.012%</td>
<td>−96</td>
<td>&lt;0.01%</td>
</tr>
<tr>
<td>−100 mV; 1.0 M</td>
<td>11,904</td>
<td>6,001</td>
<td>9,630</td>
<td>5,722</td>
<td>−57</td>
<td>0.019%</td>
<td>−68</td>
<td>0.025%</td>
</tr>
<tr>
<td>−50 mV; 1.0 M</td>
<td>11,755</td>
<td>6,037</td>
<td>9,499</td>
<td>5,756</td>
<td>−30</td>
<td>0.044%</td>
<td>−37</td>
<td>0.023%</td>
</tr>
<tr>
<td>50 mV; 1.0 M</td>
<td>11,444</td>
<td>5,887</td>
<td>9,252</td>
<td>5,849</td>
<td>29</td>
<td>0.028%</td>
<td>31</td>
<td>0.028%</td>
</tr>
<tr>
<td>100 mV; 1.0 M</td>
<td>11,773</td>
<td>6,211</td>
<td>9,518</td>
<td>6,775</td>
<td>62</td>
<td>&lt;0.01%</td>
<td>69</td>
<td>&lt;0.01%</td>
</tr>
<tr>
<td>150 mV; 1.0 M</td>
<td>12,300</td>
<td>6,380</td>
<td>9,802</td>
<td>6,490</td>
<td>97</td>
<td>0.012%</td>
<td>107</td>
<td>0.016%</td>
</tr>
<tr>
<td>1,200 mV; 1.0 M</td>
<td>11,700</td>
<td>5,734</td>
<td>10,114</td>
<td>5,775</td>
<td>1,379</td>
<td>&lt;0.01%</td>
<td>1,351</td>
<td>&lt;0.01%</td>
</tr>
<tr>
<td>−120 mV; 0.1 M</td>
<td>12,286</td>
<td>5,022</td>
<td>9,622</td>
<td>4,822</td>
<td>1.9</td>
<td>&lt;0.01%</td>
<td>6.0</td>
<td>&lt;0.01%</td>
</tr>
<tr>
<td>120 mV; 0.1 M</td>
<td>14,438</td>
<td>6,538</td>
<td>11,715</td>
<td>6,640</td>
<td>6.2</td>
<td>0.012%</td>
<td>10.5</td>
<td>0.014%</td>
</tr>
</tbody>
</table>

Maximal deviations $\delta I_+$ and $\delta I_−$ of the ionic currents (referring to the basic algorithm) occurred typically for algorithm Jp. The algorithm parameters were set to: $M = 3$, $w < 0.025, 0.500>$, $w_{up} = 0.025$, $w_{dn} = 0.050$, jump $\varepsilon <0, 10>$, $jump_{up} = jump_{dn} = 1$, and $\epsilon_{diff\text{MAX}} = 10^{-5}$ M, $\Phi_{diff\text{MAX}} = 10^{-7}$ V.

with the grid dimension of 1 Å. At this stage, the calculation method for the surface charge used only linear approximation without smoothing. The data obtained by this method was further optimized to eliminate charges mistakenly located by the previous method inside the channel pore and, afterward, the grid dimension was enlarged to 3 Å with the charge averaged. Eventually, the charge was smoothed by applying a Gaussian distribution with a standard deviation of 3 Å. This procedure allowed obtaining the most physical and realistic channel profile (Fig. 11a). The smoothing stage is also important with regard to higher stability of the PNP modeling. A similar procedure produced the coarser grid of resolution 6 Å utilized for validation of the segmentation-based optimization method. The diffusion constant for each ion is position-dependent (Fig. 11b) and it was calculated according to,

$$D_{ion}(r) = \frac{D_{ion}}{A + B \cdot \exp(\beta/C) + D \cdot \exp(\beta/E)},$$

where $A = 0.64309$, $B = 0.0044$, $C = 0.06894$, $D = 0.35647$, and $E = 0.19409$.

The simulations were carried out for two possible protonation states, in various physicochemical conditions, i.e., transmembrane potentials ranging from −1.2 to +1.2 V and ionic concentrations ranging from 0.1 to 2 M. The simulations on the α-hemolysin protein channel, including acceleration from the gradient-based and adjustable relaxation coefficient techniques, showed a very good overlap between the characteristics obtained by means of the standard PNP model and the accelerated model (Table 4) in all tested conditions. However, caution was exercised with regard to the values of the relaxation coefficients and the jump parameters so as not to endanger instability of the algorithm. The simulations on the irregular channel showed worse efficiency than on the simplified model, however still an improvement of about 50% was obtained, providing the identical conductivity characteristics for the standard and accelerated models. For the irregular channel, however, the adaptive relaxation method did not always combine very well with the adaptive jump method, and the results of the latter technique alone could produce a comparatively fast algorithm. On the other hand, the deviation of the ionic current for the adaptive jump method (~0.01%) was one or two orders greater in comparison with the deviation for the adaptive relaxation method, and usually slightly greater than the deviation for the combined method.

Analogically to the regular channel case, for the α-hemolysin channel three versions of the segmentation, described in the Methods section, were also tested. Again, the version with only one compartment (Basic) with the gradient-based optimization and the adaptive relaxation coefficient was used as a reference. In Inner-Outer versions, the coarser outer grid with resolution 6 Å covered the whole protein, while the finer inner grid of the resolution 3 Å covered the narrow stem of the channel with its neighborhood. For the Left-Right version, the right, finer domain was the same as above, but the coarser grid covered the left, wider side of the channel, with a reasonable overlap with the finer grid.

The results are presented in Table 5, where the basic segmentation is compared with three other segmentation algorithms. All the simulations use the gradient-based and the relaxation coefficient optimizations Rx + Jp. For the majority of the tested conditions, the best improvement was obtained for the IOS algorithm, reducing the computational time more than fivefold. Two other algorithms, i.e., IOP and LRP, had slightly worse performance. However, at low concentration the efficiency drops down and one of the studied cases (0.1 M, −120 mV) showed an increase in the computational time and a significant current deviation, probably because of higher sensitivity to precision in the charge distribution. Generally, the error introduced by the segmentation-based optimization depends on the experimental conditions, and the segmentation algorithm (Table 5). It is always significantly greater than the error associated with the gradient-based and the relaxation coefficient optimizations but it is usu-
ally within acceptable limits. The optimization conditions in the gradient-based and the relaxation coefficient methods do not seem to affect the improvement rate of the segmentation alone. However, they do affect the stability of the combined algorithm. Totally, the segmentation-based improvement combined with the gradient-based and the relaxation coefficient optimizations, in relation to the basic algorithm on the basic grid, can shorten the computational time up to 10-fold on a real irregular protein channel such as \( \alpha \)-hemolysin.

Our results have been compared with characteristics obtained by Noskov et al.\(^{13}\) (Fig. 12) and others\(^{15,17}\) (data not shown), which confirmed the resemblance in the most typical experimental conditions. The variation of characteristics obtained by various authors may be attributed to differences in preprocessing of the charge distribution, ion accessibility channel profile, the profile of diffusion coefficients and, finally, a different modeling method.\(^{17}\)

**Discussion and Conclusions**

Three methods for algorithmic improvement in the Poisson-Nernst-Planck model have been proposed and tested: the adapt-

---

### Table 5. Comparison of Efficiency and Reliability of Segmentation Schemes for \( \alpha \)-Hemolysin Channel.

| Cond. V, c | Alg. version | No. of grid pts. | Steps to converge | Size \( \times \) steps | \( I_+ \) (pA) | \( \Delta I_+ \) (%) | \( I_- \) (pA) | \( \Delta I_- \) (%) |
|-----------|--------------|-----------------|------------------|-------------------------|---------------|----------------|---------------|----------------|}
| -50 mV; 1.0 M Basic (Jp+Rx) | 24.8k | 5,756 | 143 M (100%) | -30 | - | -37 | - |
| IOS | 3.5k + 7.2k | 2,308 + 1,384 | 18.1 M (13%) | -32 | -6.7% | -40 | -8.1% |
| IOP | 10.7k | 2,799 | 30.0 M (21%) | -32 | -6.7% | -40 | -8.1% |
| LRP | 9.5k | 2,150 | 20.3 M (14%) | -31 | -3.3% | -38 | -2.7% |
| 150 mV; 1.0 M Basic (Jp+Rx) | 24.8k | 6,490 | 161 M (100%) | 97 | - | 107 | - |
| IOS | 3.5k + 7.2k | 2,344 + 1,690 | 20.4 M (13%) | 100 | 3.1% | 110 | 2.8% |
| IOP | 10.7k | 2,857 | 30.6 M (19%) | 100 | 3.1% | 110 | 2.8% |
| LRP | 9.5k | 3,164 | 29.9 M (19%) | 102 | 5.2% | 104 | 2.8% |
| 1,200 V; 1.0 M Basic (Jp+Rx) | 24.8k | 5,775 | 143 M (100%) | 1,379 | - | 1,351 | - |
| IOS | 3.5k + 16.1k | 4,263 + 2,362 | 49.0 M (34%) | 1,468 | 6.5% | 1,409 | 4.3% |
| IOP | 19.7k | 4,263 | 83.9 M (59%) | 1,463 | 6.1% | 1,450 | 7.3% |
| LRP | 18.7k | 6,026 | 113 M (79%) | 1,816 | 31.7% | 1,840 | 36.2% |
| -120 mV; 0.1 M Basic (Jp+Rx) | 24.8k | 4,822 | 119 M (100%) | 1.89 | - | 6.0 | - |
| IOS | 3.5k + 16.1k | 4,229 + 9,850 | 174 M (146%) | 0.03 | -98% | 5.3 | -11.7% |
| IOP | 19.7k | 9,973 | 196 M (165%) | 0.02 | -99% | 6.2 | 3.3% |
| LRP | 18.7k | 4,423 | 218 M (183%) | 0.03 | -98% | 5.4 | -10% |
| 120 mV; 0.1 M Basic (Jp+Rx) | 24.8k | 6,640 | 164 M (100%) | 6.2 | - | 10.5 | - |
| IOS | 3.5k + 16.1k | 4,351 + 5,914 | 111 M (68%) | 5.6 | -9.7% | 13.1 | 24.8% |
| IOP | 19.7k | 6,598 | 130 M (79%) | 5.4 | -12.9% | 14.6 | 39.0% |
| LRP | 18.7k | 8,065 | 151 M (92%) | 5.1 | -17.7% | 12.8 | 21.9% |

Basic (Jp+Rx) refers here to the optimized version (Jp+Rx) of the algorithm with one segment of resolution 3 Å. All segmentation-based simulations also use Jp+Rx optimization.

---

**Figure 11.** The computational model of \( \alpha \)-hemolysin channel. (a) Transverse section in \( xy \) plane at the midpoint. (b) Diffusion coefficient profiles for \( K^+ \) (dashed line) and \( Cl^- \) (dotted line) ions.

**Figure 12.** Current–voltage dependency for the \( \alpha \)-hemolysin protein channel from the PNP model in 1.0 M KCl solution (results from basic and enhanced (Jp+Rx) algorithms overlap in the plot, empty circles). The cationic current is denoted by “+” and anionic current by “x.” The characteristics compared with results from another PNP modeling (filled circles) by Noskov et al.\(^{13}\)
itive gradient-based method, the adjustable relaxation coefficient technique, and the space segmentation method. Their combined implementation substantially reduces the computational cost of the PNP without significant loss of accuracy.

The adaptive gradient-based optimization technique, similarly to the other methods introduced, has been demonstrated to be effective when applied alone, as well as when the techniques are combined together. We consider that this technique is analogous to an extension of the over-relaxation technique, and we term this here as an example of a super-relaxation technique. It appears to exhibit greater stability than the standard over-relaxation technique with a choice of the weighting parameter significantly less than −1, probably because our technique is based on a mean of several iteratively derived values, rather than a simple projection from a single value.

The adjustable relaxation coefficient technique has also proven to be more efficient than the basic algorithm, although less efficient than the super-relaxation technique discussed earlier, probably because it is more susceptible to high-frequency fluctuations in the iteration profile (see e.g., Fig. 2). The combination of these two techniques provides a powerful improvement in iteration efficiency, typically reducing the number of iteration steps by at least 50% depending on the conditions and the channel regularity.

The space segmentation technique is similarly effective for a regular channel, reducing the number of iterations by ~50–75%, with good accuracy in achieving the final value. Moreover, the simulations on the z-hemolysin, showed that the segmentation-based improvement, combined with the gradient-based and the relaxation coefficient optimizations, can be especially justified on irregular channels. This optimization seems inappropriate at low ionic concentration. However, in advantageous conditions, the use of the combined method involving segmentation optimization with the adjustable relaxation coefficient and the gradient-based optimization could shorten the computational time more than 10-fold (in relation to the basic algorithm on the basic grid). Our results show that applying one or a combination of the optimization methods can significantly reduce the computational demands of the PNP model without changing the physical model.

Acknowledgments

The authors would like to acknowledge the award of an Erasmus travel grant to WD that in part supported this work. Part of the calculations has been carried out in Wroclaw Centre for Networking and Supercomputing (http://www.wcss.wroc.pl).

References