Numerical Solution of
Nonlinear Poisson Boltzmann Equation

Student: Jingzhen Hu, Southern Methodist University

Advisor: Professor Robert Krasny

1 A list of work done

- extended the numerical solution of nonlinear Poisson Boltzmann (PB) equation from 1D to 2D (radial symmetry) and 3D (spherical symmetry), applying the quasilinearization technique
- prepared a talk for the project of pKa computation
- read a list of papers in reading order
  - W. R. Bowen, A. O. Sharif, *Long-range electrostatic attraction between like-charge spheres in a charged pore*, Nature 393 (1998)
  - J. C. Neu, *Wall-Mediated Forces between Like-Charged Bodies in an Electrolyte*, Physical Review 82 (1999)
  - W. R. Bowen, A. O. Sharif, *Adaptive Finite-Element Solution of the Nonlinear Poisson-Boltzmann Equation: A Charged Spherical Particle at Various Distances from a Charged Cylindrical Pore in a Planar Surface*, Journal of Colloid Interface Science 187 (1997)
  - Z. Xu. *Electrostatic interaction in the presence of dielectric interfaces and polarization-induced like-charge attraction*, Physical Review E87, 013307 (2013)
  - W. Rocchia. *Polsson-Boltzmann Equation Boundary Conditions for Biological Applications*, Mathematical and computer modeling 41 (2005)
  - P. Li, H. Johnston, R. Krasny. *A Cartesian treecode for screened coulomb interactions*, Journal of Computational Physics 228 (2009)
  - F. H. Stillinger. *Interfacial solutions of the Poisson-Boltzmann Equation*, Journal of Chemical Physics 35 (1961)
- improved the pKa computation employing mean field & reduced site approximation, and Monte Carlo sampling

In this report, we will focus on the numerical method for solving nonlinear PB equation.

2 Introduction

A biomolecule in the body is naturally surrounded by biofuilds, which forms a biomolecule-solvent system. In biochemistry, interaction force, binding energies, and pKa computation are based on the electrostatic potential in the biomolecule-solvent system. The implicit Poisson-Boltzmann (PB) model has been developed to compute the potential, partitioning the 3D space into molecule part (Ω_m) and solvent part (Ω_s) by the molecular interface Γ with ion concentration denoted by C. Dielectric constants ϵ is a spaced-depended function which gives us different values between molecule (ϵ_m) and solvent (ϵ_s). The Figure 1 shows charges are distributed both in Ω_m and Ω_s. Charges in Ω_m are fixed, coming from partial charges allocated on the centers of atoms via force fields. Charges in Ω_s are mobile ions under the Boltzmann distribution. Given a position variable $x = (x_1, x_2, x_3)$ in 3D space, applying Gauss's law to the charge distribution both on the molecular and in the solvent (assuming the equal monovalent cation and anion) leads the PB model [1],

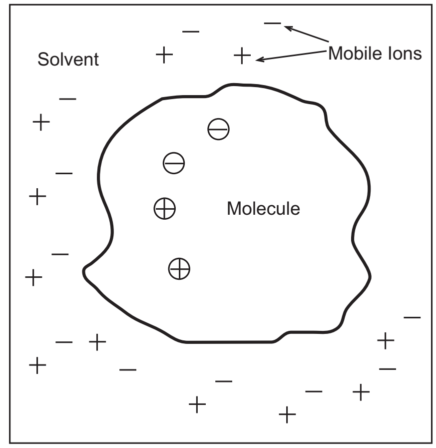

Figure 1: molecular-solvent system

$$-\epsilon_m \nabla^2 \phi_m(\boldsymbol{x}) = 4\pi C \sum_{k=1}^{N} q_k \delta(\boldsymbol{x} - \boldsymbol{y}_k) \qquad \boldsymbol{x} \in \Omega_m \tag{1}$$

$$-\epsilon_s \nabla^2 \phi_s(\boldsymbol{x}) + \sinh \kappa^2 \phi_2(\boldsymbol{x}) = 0 \qquad \boldsymbol{x} \in \Omega_s \tag{2}$$

with the continuous interface conditions of potentials and their normal derivatives (on Γ),

$$\phi_m(\boldsymbol{x}) = \phi_s(\boldsymbol{x}), \quad \epsilon_m \frac{\partial \phi_m(\boldsymbol{x})}{\partial \nu} = \epsilon_s \frac{\partial \phi_s(\boldsymbol{x})}{\partial \nu}, \tag{3}$$

where κ, named as inverse Debye screening length, measures the ionic strength. The electric charge density in molecular part is the summation of the charged distribution involving Delta function for N partial charges q_k located at y_k for k = 1,...,N. In Ω_s, the Boltzmann distribution introduces a nonlinear term $\sinh \kappa^2 \phi(\boldsymbol{x})$. Keep the first term in Tayler Series, it can be treated as a linear term $\kappa^2 \phi(\boldsymbol{x})$ when κ is a small value near zero, but the approximation makes the solution decay slower than the nonlinear one. Comparing the linear and nonlinear solution of 1D PBE, we also see the first derivative of potential, in physical representing the electric filed, forms a discrepancy when $x \to 0$. It is interesting to study the solution of nonlinear PBE and compare the difference between it and the linear one. We apply the quasilinearization technique to compute the electrostatic potential from 1D and 2D nonlinear PBE. The results shows the scheme converges at the rate of $O(N^{-2})$. Section 3 is mainly focus on the 1D case comparing with exact solution to test the solid second order accuracy convergency rate. Section 4 extends the quasilinearization technique into 2D case in polar coordinate with radius symmetry. Section 5 is 3D nonlinear PBE in spherical symmetry. In section 6, we plotted the1D, 2D, 3D results of the same boundary condition in one figure.

3 1D nonlinear Poisson-Boltzmann Equation (PBE)

3.1 solvent part

3.1.1 nondimensionalization

The 1D nonlinear PBE to study of electrostatics in salty solutions is showed [2]

$$\frac{d^2\phi(x)}{dx^2} = \frac{2Cq}{\epsilon_0\epsilon}\sinh\frac{q\phi(x)}{k_BT}, \tag{4}$$

where ϕ is electric potential; C is equal concentration of monovalent cations and anions; q is electric charge; T is temperature; ϵ_0 is electrical permittivity of free space; ϵ is the dielectric constant of solvent; k_B is Boltzmann constants. To reduce units, substitute $\hat{x}$ and $\hat{\phi}$ defined by,

$$\hat{x} = q\sqrt{\frac{C}{\epsilon_0\epsilon k_BT}}x, \quad \hat{\phi} = \frac{q}{k_BT}\phi, \tag{5}$$

the nondimensionalized equation is obtained as,

$$\hat{\phi}''(\hat{x}) = \sinh\hat{\phi}(\hat{x}). \tag{6}$$

3.1.2 analytical solution

Choose the domain $\hat{x} \in [x_0,\infty)$ and two boundary conditions as $\hat{\phi}(x_0) = \phi_0$, $\hat{\phi}(\infty) = 0$. Multipling both sides multiply by ϕ', equation (6) becomes,

$$\hat{\phi}'' \cdot \hat{\phi}' = \sinh\hat{\phi} \cdot \phi'. \tag{7}$$

Take the integral,

$$\frac{1}{2}(\hat{\phi}')^2 = \cosh\hat{\phi} + A. \tag{8}$$

Since $\hat{\phi}(\infty) = \hat{\phi}'(\infty) = 0$, $A = -1$. Taking a square root both side leads to,

$$\hat{\phi}' = \sqrt{2\cosh\hat{\phi} - 2} = \sqrt{4\cosh^2\frac{\hat{\phi}}{2} - 4} = -2\sinh\frac{\hat{\phi}}{2}. \tag{9}$$

In order to fit in the boundary condition chosen, keep the negative square root,

$$-\frac{d\hat{\phi}}{2\sinh\frac{\hat{\phi}}{2}} = d\hat{x}. \tag{10}$$

Substitute u and du,

$$u = \frac{1}{\cosh\frac{\hat{\phi}}{2}}, \quad du = -\frac{\sinh\frac{\hat{\phi}}{2}}{2\cosh^2\frac{\hat{\phi}}{2}}\,d\hat{\phi} = -\frac{u^2}{2}\sinh\frac{\hat{\phi}}{2}d\hat{\phi}. \tag{11}$$

Left hand side can be simplified as

$$-\frac{d\hat{\phi}}{2\sinh\frac{\hat{\phi}}{2}} = -\frac{\sinh\frac{\hat{\phi}}{2}\,d\hat{\phi}}{2\sinh^2\frac{\hat{\phi}}{2}} = -\frac{\sinh\frac{\hat{\phi}}{2}\,d\hat{\phi}}{2(\frac{1}{u^2}-1)} = \frac{du}{u^2(\frac{1}{u^2}-1)} = \frac{du}{1-u^2}. \tag{12}$$

Turn equation (10) into the form,

$$\frac{du}{1-u^2} = d\hat{x} \tag{13}$$

Take the integral,

$$\tanh^{-1}(u) + B = \hat{x} \tag{14}$$

As $\hat{\phi}(x_0) = \phi_0$, then $B = x_0 - \tanh^{-1}(\frac{\phi_0}{2})$ which leads to,

$$\tanh^{-1}(\frac{1}{\cosh\frac{\hat{\phi}}{2}}) - \tanh^{-1}(\frac{1}{\cosh\frac{\phi_0}{2}}) = \hat{x} - x_0, \tag{15}$$

$$\frac{1}{2}\ln(\frac{\cosh\frac{\hat{\phi}}{2}+1}{\cosh\frac{\hat{\phi}}{2}-1}) - \frac{1}{2}\ln(\frac{\cosh\frac{\phi_0}{2}+1}{\cosh\frac{\phi_0}{2}-1}) = \hat{x} - x_0, \tag{16}$$

$$(\frac{\cosh\frac{\hat{\phi}}{2}+1}{\cosh\frac{\hat{\phi}}{2}-1}\cdot\frac{\cosh\frac{\phi_0}{2}-1}{\cosh\frac{\phi_0}{2}+1})^{\frac{1}{2}} = e^{(\hat{x}-x_0)}. \tag{17}$$

Apply $\cosh v + 1 = 2\cosh^2(v/2)$, $\cosh v - 1 = 2\sinh^2(v/2)$ to simplify the equation,

$$\frac{\tanh\frac{\phi_0}{4}}{\tanh\frac{\hat{\phi}}{4}} = e^{(\hat{x}-x_0)}. \tag{18}$$

The solution is

$$\hat{\phi}(\hat{x}) = 4\tanh^{-1}(\tanh(\frac{\phi_0}{4})e^{-(\hat{x}-x_0)}). \tag{19}$$

If choosing the domain of x to be $(-\infty, x_N]$ and the boundary conditions $\phi(-\infty) = 0$, $\phi(x_N) = \phi_N$ and following similar steps, we will get,

$$\phi(x) = 4\tanh^{-1}(\tanh(\frac{\phi_N}{4})e^{(x-x_N)}). \tag{20}$$

3.2 Numerical solution

3.2.1 quasilinearization [3]

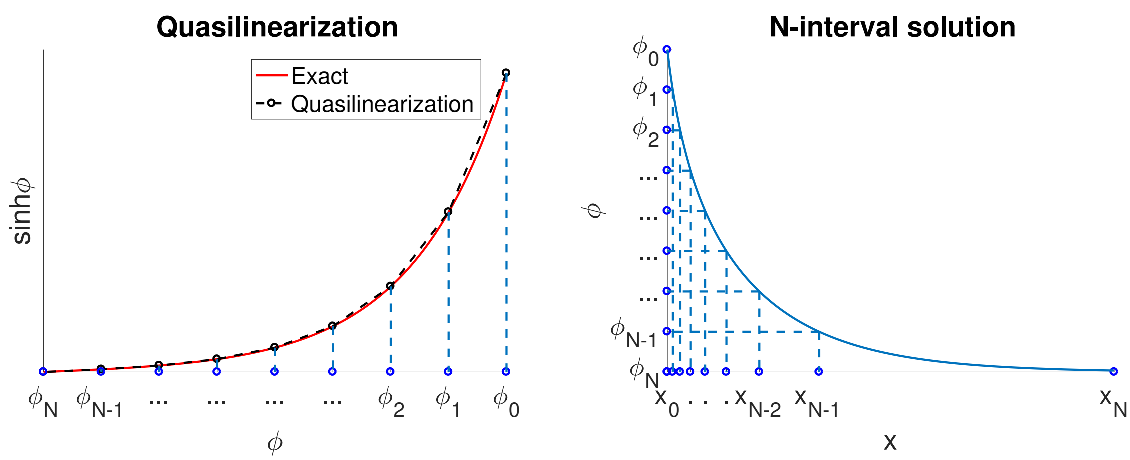

Figure 2: Discretization, uniform ϕ mesh points with corresponding projected x points

Since we are going to apply quasilinearization technique solves nonlinear protein-solvent schema, we can use the solvent part as a test case. It employs a piecewise linear approximation of the nonlinear term in the differential equation. The mesh points are chosen by projecting uniform values of the potential onto the domain, and matching conditions are enforced at the interior mesh points. The Figure 2 gives a N-interval solution and its approximation of the right hand side function.

Before looking the half-infinity domain problem, we solve for a two-point boundary value problem at first,

$$\phi''(x) = \sinh(\phi(x)), \tag{21}$$

$$\phi(a) = \psi, \quad \phi(b) = 4\tanh^{-1}(\tanh(\frac{\phi_0}{4})e^{-b+a}) \tag{22}$$

In this case, the right hand side function $f(\phi)$ is defined by,

$$f(\phi) = \sinh(\phi). \tag{23}$$

Apply the idea of quasilinearization, we replace the right hand side function, $\sinh(\phi)$, by its equally linear interpolation under a ϕ-mesh (ϕ_0, ϕ_1,..., ϕ_N) with corresponding x value (x_0, x_1, ..., x_N), where

$$a = x_0 < x_1 < x_2 < ... < x_N = b. \tag{24}$$

Define the ith interval by the boundary $x_{i-1} \leq x \leq x_i$, for $i = 1 : N$. In ith interval, the equation (21) and (22) turn to be,

$$\phi_i''(x) = \alpha_i^2(\phi_i(x) - \phi_{i-1}) + f(\phi_{i-1}), \tag{25}$$

$$\phi_i(x_{i-1}) = \phi_{i-1},\ \phi_i(x_i) = \phi_i, \tag{26}$$

where the

$$\alpha_i^2 = \frac{f(\phi_i) - f(\phi_{i-1})}{\phi_i - \phi_{i-1}}. \tag{27}$$

3.2.2 solution for each interval

Solution is a combination of homogeneous part and particular part. The homogeneous solution satisfies the boundary condition (26). The particular solution satisfy the zero boundary condition. First, find a combination of the fundamental solutions, $u_i(x)$ and $v_i(x)$ that satisfy the boundary condition,

$$u_i(x_{i-1}) = v_i(x_i) = 0, \qquad u_i(x_i) = v_i(x_{i-1}) = 1. \tag{28}$$

The $u_i(x)$ and $v_i(x)$ are given by,

$$u_i(x) = \frac{\sinh \alpha_i(x - x_{i-1})}{\sinh \alpha_i(x_i - x_{i-1})}, \quad v_i(x) = \frac{\sinh \alpha_i(x_i - x)}{\sinh \alpha_i(x_i - x_{i-1})}. \tag{29}$$

The solution is in the form,

$$\phi_i(x) = \phi_i u_i(x) + \phi_{i-1} v_i(x) + P_i(x). \tag{30}$$

where the $P_i(x)$ is the particular solution which satisfies the $P_i(x_{i-1}) = P_i(x_i) = 0$. Since the Green's function exists for the problem, the particular solution can be expressed into,

$$P_i(x) = \frac{v_i(x)}{c_i}\int_{x_{i-1}}^{x} u_i(\xi)(f(\phi_{i-1}) - \alpha_i^2\phi_{i-1})d\xi + \frac{u_i(x)}{c_i}\int_{x}^{x_i} v_i(\xi)(f(\phi_{i-1}) - \alpha_i^2\phi_{i-1})d\xi \tag{31}$$

where

$$c_i = \begin{vmatrix} u_i & v_i \\ u_i' & v_i' \end{vmatrix} = u_i v_i' - u_i' v_i = -\frac{\alpha_i}{\sinh \alpha_i (x_{i+1} - x_i)}. \tag{32}$$

After simplification, the particular solution is,

$$P_i(x) = (\phi_{i-1} - \beta_i)(1 - u_i(x) - v_i(x)), \tag{33}$$

where $\beta_i = \frac{f(\phi_{i-1})}{\alpha_i^2}$. Recalling the equation (30), the solution is,

$$\phi_i(x) = \phi_i u_i(x) + \phi_{i-1} v_i(x) + (\phi_{i-1} - \beta_i)(1 - u_i(x) - v_i(x)) \tag{34}$$

3.2.3 matching condition

We match up all the piecewise solutions by setting the derivative of the solutions equal at the interiors,

$$\phi_i'(x_i) = \phi_{i+1}'(x_i), \quad i = 1 : N-1. \tag{35}$$

The derivative form of the solution is,

$$\phi_i'(x) = \phi_i u_i'(x) + \phi_{i-1} v_i'(x) - (\phi_{i-1} - \beta_i)(u_i'(x) + v_i'(x)) \tag{36}$$

where

$$u_i'(x) = \frac{\alpha_i \cosh \alpha_i (x - x_{i-1})}{\sinh \alpha_i (x_i - x_{i-1})}, \quad v_i'(x) = -\frac{\alpha_i \cosh \alpha_i (x_i - x)}{\sinh \alpha_i (x_i - x_{i-1})}. \tag{37}$$

The boundary for derivative of $u_i(x)$ and $v_i(x)$ is,

$$u_i'(x_{i-1}) = \frac{\alpha_i}{\sinh \alpha_i (x_i - x_{i-1})}, \quad u_i'(x_i) = \frac{\alpha_i}{\tanh \alpha_i (x_i - x_{i-1})} \tag{38}$$

$$v_i'(x_{i-1}) = -\frac{\alpha_i}{\tanh \alpha_i (x_i - x_{i-1})}, \quad v_i'(x_i) = -\frac{\alpha_i}{\sinh \alpha_i (x_i - x_{i-1})} \tag{39}$$

For the x_i point, the equation (35), which is expressed by

$$\begin{aligned} &\phi_{i-1} v_i'(x_i) + \phi_i (u_i'(x_i) - v_{i+1}'(x_i)) - \phi_{i+1} u_{i+1}'(x_i) = \\ &\qquad (\phi_{i-1} - \beta_i)(u_i'(x_i) + v_i'(x_i)) - (\phi_i - \beta_{i+1})(u_{i+1}'(x_i) + v_{i+1}'(x_i)). \end{aligned} \tag{40}$$

For the 2-interval solution, only having one middle point to update, which can be solved using fixed-point iteration,

$$\phi_1 = \frac{-\beta_1 (u_1'(x_1) + v_1'(x_1)) + \beta_2 (u_2'(x_1) + v_2'(x_1)) + \phi_0 u_1'(x_1) + \phi_2 u_2'(x_1)}{u_1'(x_1) + u_2'(x_1)}. \tag{41}$$

Use the To form a row of a matrix, we cleaning the equation to a form,

$$a_i\phi_{i-1} + b_i\phi_i + c_i\phi_{i+1} = R_i, \qquad i = 1, 2, ..., N-1 \tag{42}$$

where

$$a_i = v_i'(x_i), \quad b_i = u_i'(x_i) - v_{i+1}'(x_i), \quad c_i = -u_{i+1}'(x_i), \tag{43}$$

$$r_i = (\phi_{i-1} - \beta_i)(u_i'(x_i) + v_i'(x_i)) - (\phi_i - \beta_{i+1})(u_{i+1}'(x_i) + v_{i+1}'(x_i)), \tag{44}$$

where A is a tridiagonal and symmetric matrix.

Combining all the matching condition, the equation (42), we can form them into a matrix,

$$A(x,\phi)\phi = R(x,\phi), \tag{45}$$

$$\begin{pmatrix} b_1 & c_1 & 0 & 0 & \cdots & 0 \\ 0 & a_2 & b_2 & c_2 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \cdots & 0 \\ 0 & \cdots & 0 & a_{N-2} & b_{N-2} & c_{N-2} \\ 0 & \cdots & \cdots & 0 & a_{N-1} & b_{N-1} \end{pmatrix} \times \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{N-2} \\ \phi_{N-1} \end{pmatrix} = \begin{pmatrix} r_1 - a_1\phi(0) \\ r_2 \\ \vdots \\ r_{N-2} \\ r_{N-1} - c_{N-1}\phi(L) \end{pmatrix}. \tag{46}$$

If we choose a uniform mesh for ϕ, the error is bound by $O(N^{-2})$ caused by solving the system iteratively.

3.2.4 numerical implementation

To make sure the ϕ points are uniformly distributed in each iteration, we have an **inner** iteration with index k and an **outer** iteration with index j. The **inner** iteration is to update ϕ values with fixed x values to satisfy the matching condition,

$$A(x^j, \phi^k)\phi^{k+1} = R(x^j, \phi^k). \tag{47}$$

Given a convergence criteria ϵ_ϕ, the inner iteration stop when

$$|\phi^{k+1} - \phi^k| < \epsilon_\phi. \tag{48}$$

Since the result of the **inner** iteration, the ϕ value will not be uniformly distributed. To reach the optimal error, we project the evenly distributed ϕ value on to the latest numerical solution that formed by the updated ϕ value obtained by the **inner** iteration to get new x points.

Recalling the equation (34), we solve for the x value when we set the right hand side a uniform mesh of ϕ,

$$\phi^{k+1}(x^{j+1}) = \phi_e, \tag{49}$$

where ϕ_e is a uniform mesh of ϕ with N intervals. The explicit item in the ϕ_e is given by,

$$\phi_e^n = n\frac{\phi(0) - \phi(L)}{N}, \quad n = 1 : N - 1. \tag{50}$$

The **outer** iteration has a converge criteria ϵ_x. The whole iteration stop when,

$$|x^{j+1} - x^j| < \epsilon_x. \tag{51}$$

Each time at the end of the **outer** iteration, we can multiple the number of intervals, N, by 2. The result turns out to be $N = 1, 2, 4, ..., 2^m$. For each iteration, the initial guess come form the previous solution. For example, the initial guess for the two intervals iteration comes form the result of the one interval solution. The first initial guess can be obtained with the result of one interval, where the right hand side function is approximated by one straight line. We project new equispaced ϕ points onto x-axis to obtain the initial guess for new partitions. Then, we go into the **inner** iteration to updated ϕ values and then the **outer** iteration to update x values.

We can treat the outer iteration as a root finding problem and apply the numerical methods like bisection method, Newton's method to update new x points. The pseudocode is provided as follow.

1 Provide the boundary information, x_0, x_N, ϕ_0, ϕ_N. Let N be the current partition number of intervals. Initially set to one.

2 Project the uniformly distributed ϕ point(s), denoted as ϕ_{mid}, onto the N-interval solution to get the x_{mid}, which is the initial partition for the inner iteration to get a matching $2N$-interval solution. Before projection, we need first judge which interval those ϕ_{mid} lie in and apply the root-finding technique.

Note that, we skip the even point in projection, since it should stay the same as the result of previous N-interval solution.

4 Do the inner iteration, using the matching condition to update ϕ values with fixed x values. The matching condition applies the iterative method, the equation (47).

5 Update the x values by projecting the same euqispaced ϕ_{mid} onto the current 2N-interval solution. The 2N-interval solution is generated by the current x values and updated phi values from inner iteration.

6 Go back to the inner iteration (step 4), and repeat doing the step 4 and 5 until the x values converges. This is the outer iteration.

7 Update the N value to be $2N$, $N = 2N$.

3.2.5 domain extension from [0,L] to [0, ∞]

The last interval is treated as a special case to make the domain of x extends to [0,∞]. The boundary condition for the last interval becomes,

$$\phi_N(x_{N-1}) = \phi_{N-1}, \quad \phi_N(\infty) = 0. \tag{52}$$

Recall the equation (25) and apply it for the last interval,

$$\phi''_N(x) = \alpha_N^2(\phi_N(x) - \phi_{N-1}) + f(\phi_{N-1}), \tag{53}$$

Suppose the particular solution is a constant C, then make a substitution,

$$C = \phi_{N-1} - \beta_N. \tag{54}$$

The fundamental solutions are $e^{\alpha_N x}$ and $e^{-\alpha_N x}$. Then, the solution is given by,

$$\phi_N(x) = Ae^{\alpha_N x} + Be^{-\alpha_N x} + \phi_{N-1} - \beta_N. \tag{55}$$

Solving coefficients A and B to satisfy the boundary conditions in equation (52),

$$\phi_N(\infty) = 0 \Longrightarrow A = 0, \quad \phi_{N-1} - \beta_N = 0, \tag{56a}$$

$$\phi_N(x_{N-1}) = \phi_{N-1} \Longrightarrow B = \phi_{N-1}e^{\alpha_N x_{N-1}}. \tag{56b}$$

The solution, equation (55), and its derivative are as follows,

$$\phi_N(x) = \phi_{N-1}e^{-\alpha_N(x-x_{N-1})}, \quad \phi'_N(x) = -\alpha_N\phi_{N-1}e^{-\alpha_N(x-x_{N-1})}. \tag{57}$$

For the matching condition, **inner** iteration, the special case of the last interval turns to be,

$$\phi'_{N-1}(x_{N-1}) = \phi'_N(x_{N-1}). \tag{58}$$

Apply the equation (36) and (132) to expand the equation as follows,

$$\phi_{N-1}u'_{N-1}(x_{N-1}) + \phi_{N-2}v'_{N-1}(x_{N-1}) - (\phi_{N-2} - \beta_{N-1})(u'_{N-1}(x_{N-1}) + v'_{N-1}(x_{N-1})) = -\alpha_N\phi_{N-1}. \tag{59}$$

Then last row of the matrix that applies the matching condition is,

$$-\phi_{N-2}u'_{N-1}(x_{N-1}) + \phi_{N-1}(u'_{N-1}(x_{N-1}) + \alpha_N) = -\beta_{N-1}(u'_{N-1}(x_{N-1}) + v'_{N-1}(x_{N-1})). \quad (60)$$

Also, for the first ϕ midpoint, the explicit formula is given,

$$\phi_1 = \frac{-\beta_1(u'_1(x_1) + v'_1(x_1)) + \phi_0 u'_1(x_1)}{u'_1(x_1) + \alpha_2}. \quad (61)$$

For the **outer** iteration, the process is quite direct,

$$\phi_{N-1}e^{-\alpha_N(x-x_{N-1})} = \phi_{evenly} \Rightarrow x = x_{N-1} + \frac{1}{\alpha_N}\ln\frac{\phi_{N-1}}{\phi_{evenly}}. \quad (62)$$

Alternatively, we make a revise on the bisection method to apply in the **outer** iteration. First pick ϕ_{N-1} as the right endpoint and check whether the sign of function values at two endpoints are opposite. If opposite, then apply the normal bisection method, otherwise, double the right endpoint iteratively.
Very similar process for the domain of $(-\infty, 0]$ to treat the first interval as a special case,

$$\phi_1(x) = \phi_1 e^{\alpha_1(x-x_1)}, \quad \phi'_1(x) = \alpha_1\phi_1 e^{\alpha_1(x-x_1)}. \quad (63)$$

The **inner** iteration turns to be,

$$(u'_2(x_1) + \alpha_1)\phi_1 - u'_2(x_1)\phi_2 = \beta_2(u'_2(x_1) + v'_2(x_1)) \quad (64)$$

The **outer** iteration, it is,

$$\phi_1 e^{\alpha_N(x-x_1)} = \phi_{evenly} \Rightarrow x = x_1 + \frac{1}{\alpha_1}\ln\frac{\phi_{evenly}}{\phi_1}. \quad (65)$$

3.2.6 results

The Figure 3 shows how the numerical solutions gradually approach the exact solution up to the eight-interval solution. The graph contains a original plot of solutions and a magnification version of the plot near the middle point of ϕ.

The Table 1 contains the error analysis under the error tolerance equals 1e-12. The first two columns are the number of intervals and the max error based on solution of such that number of intervals. The max error is calculated by the max of absolute differences of numerical results and exact solution on the mesh points. The third column indicates that the error is bound by $O(N^{-2})$. Note that each time increasing the number of interval, the $2N$-solution, which is closer to the exact solution, will be above the N-solution. Here is a simple proof.

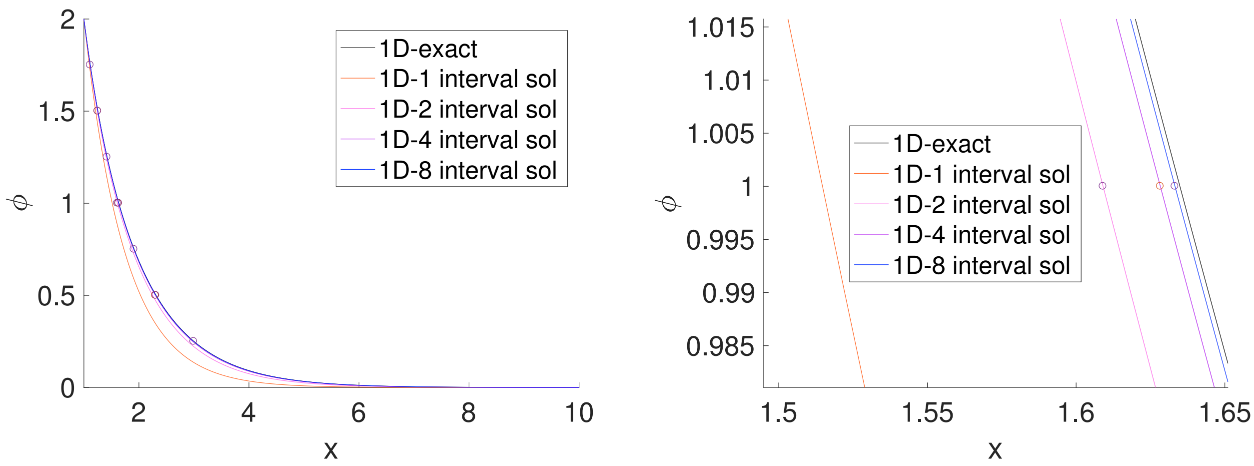

Figure 3: 1D numerical result: 8-Interval solution (left), its magnification around $\phi = 1$ (right); boundary condition: $\phi(0) = 2$, $\phi(\infty) = 0$

N	Max Error	Max Error$\cdot$ N^2
2	0.02688513	0.10754053
4	0.00688749	0.11019977
8	0.00180185	0.11531836
16	0.00045037	0.11529390
32	0.00011259	0.11529054
64	0.00002816	0.11534344

Table 1: Numerical results: errors analysis for 2^k number of intervals, k = 1 : 6; boundary condition : $\phi(0) = 2$, $\phi(\infty) = 0$

With the same boundary condition, we define the difference of the two solution,

$$u = \phi(x; N, \phi_0) - \phi(x; 2N, \phi_0), \quad u(0) = 0, \quad u(\infty) = 0 \tag{66}$$

Then, the second derivative of u is given,

$$u'' = \phi''(x; N, \phi_0) - \phi''(x; 2N, \phi_0) \geq 0 \tag{67}$$

With zero boundary condition and positive second derivative, the function $u(x)$ can be draw all below the x-axis, which means,

$$u \leq 0 \Rightarrow \phi(x; N, \phi_0) \leq \phi(x; 2N, \phi_0). \tag{68}$$

Under the same idea, we also can show that the nonlinear solution is always smaller than the linear solution because the $\forall\phi > 0$, $\sinh\phi > \phi$.

3.3 1D solvent-protein system

3.3.1 linear version: analytic solution

In 1D, the equation (1) and (2) can be linearized as,

$$-\epsilon_p\phi''(x) = q_m\delta(x-y) \qquad 0 \le x \le 1, 0 \le y \le 1, \tag{69a}$$

$$-\epsilon_s\phi''(x) + \kappa^2\phi(x) = 0 \qquad x \le 0 \text{ or x} \ge 1, \tag{69b}$$

with the boundary condition $\phi(\pm\infty) = 0$. The corresponding matching condition on the interface (Γ) is,

$$\phi(0^-) = \phi(0^+), \quad \phi(1^-) = \phi(1^+), \tag{70a}$$

$$\epsilon_s\phi'(0^-) = \epsilon_p\phi'(0^+), \quad \epsilon_p\phi'(1^-) = \epsilon_s\phi'(1^+). \tag{70b}$$

For the middle interval, the ϕ_2 is obtained by integration twice,

$$\phi_2(x) = -\frac{1}{\epsilon_p}\cdot[qH(x-y)(x-y) + Ax + B], \tag{71}$$

where A and B are two constants that will be determined by the matching condition later. For the two end intervals, let

$$\alpha = \frac{\kappa}{\sqrt{\epsilon_s}}. \tag{72}$$

Applying the boundary condition to each case, the ϕ_1 and ϕ_3 are in the form of

$$\phi_1(x) = Ce^{\alpha x} \quad \text{x} \le 0, \tag{73}$$

$$\phi_3(x) = De^{-\alpha x} \quad \text{x} \ge 1. \tag{74}$$

Applying the matching conditions in equation (70a) and (70b), we get a linear system

$$B + \epsilon_p C = 0 \tag{75a}$$

$$A + \epsilon_s\alpha C = 0 \tag{75b}$$

$$A + B + \epsilon_p De^{-\alpha} = -q(1-y) \tag{75c}$$

$$A - \epsilon_s\alpha De^{-\alpha} = -q \tag{75d}$$

The final solution is,

$$\phi_(x) = \begin{cases} Ce^{\alpha x} & \text{x} \leq 0 \\ -\frac{1}{\epsilon_p}(Ax+B) & 0 \leq \text{x} \leq \text{y} \\ -\frac{1}{\epsilon_p}(q(x-y)+Ax+B) & \text{y} \leq \text{x} \leq 1 \\ De^{-\alpha x} & \text{x} \geq 1, \end{cases} \tag{76}$$

where,

$$C = \frac{q[\epsilon_p + \epsilon_s \alpha(1-y)]}{\epsilon_s \alpha(2\epsilon_p + \epsilon_s \alpha)}, \quad A = -\epsilon_s \alpha C, \quad B = -\epsilon_p C, \quad D = \frac{(A+q)e^{\alpha}}{\epsilon_s \alpha}. \tag{77}$$

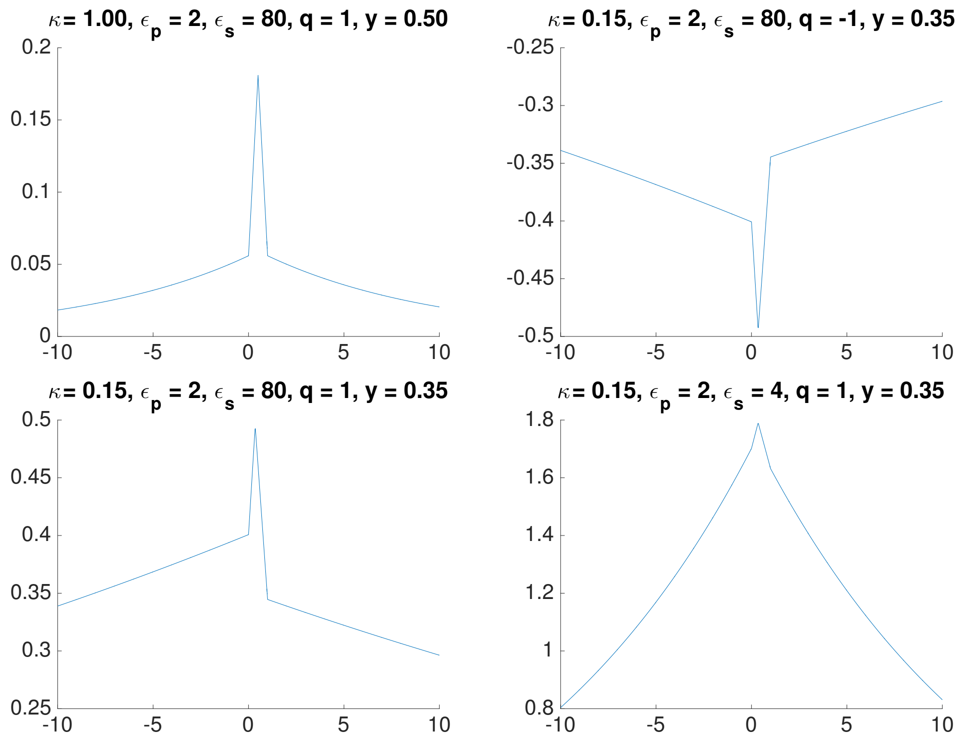

Figure 4: Analytic solution of linear protein-solvent system with different physical constants

The Figure 4 shows how the physical constant is related to solution. With a large κ value, the potential in the solvent will decrease a little bit slower. With a small difference of ϵ_s and ϵ_p, the solution is smoother. The sign of charge can makehe solution revseral. The 1D nonlinear solvent and protein system is set the same except for replacing the solvent part by,

$$\epsilon_s \phi''(x) = \sinh \kappa^2 \phi(x) \tag{78}$$

3.3.2 nonlinear version: numerical solution

Apply the quasilinearization to the solvent part and add the matching condition at the molecular surface into the whole matching condition.

When the number of interval, $N = 1$, the matching condition at the 0 and 1 is treated as a special case. The one-interval solution equals to the linear solution of PBE. To enforce the matching condition, we need first get the solution of the protein part and its derivative at two endpoints. Solving by the homogeneous solution plus particular solution, the $\phi_2(x)$ is given,

$$\phi_2(x) = \phi_0^+ x + \phi_N^-(1-x) - \frac{q}{\epsilon_p}(x-y)H(x-y) + \frac{q}{\epsilon_p}(1-y)x, \tag{79}$$

and its derivation,

$$\phi_2'(x) = \phi_0^+ - \phi_N^- - \frac{q}{\epsilon_p}H(x-y) + \frac{q}{\epsilon_p}(1-y). \tag{80}$$

The derivative att two endpoints,

$$\epsilon_p\phi_2'(0) = \epsilon_s(\phi_0^+ - \phi_N^-) + q(1-y), \quad \epsilon_p\phi_2'(1) = \epsilon_s(\phi_0^+ - \phi_N^-) - qy. \tag{81}$$

Recalling to equations (132), (63), and (81), we can derive,

$$\epsilon_s\alpha_1^-\phi_1^- = \epsilon_p(\phi_0^+ - \phi_1^-) + q(1-y) \tag{82a}$$

$$-\epsilon_s\alpha_1^+\phi_0^+ = \epsilon_p(\phi_0^+ - \phi_1^-) - qy \tag{82b}$$

Write the two equations into a linear system, $A\phi = b$, where A is symmetric,

$$\begin{pmatrix} \epsilon_s\alpha_1^- + \epsilon_p & -\epsilon_p \\ -\epsilon_p & \epsilon_s\alpha_1^+ + \epsilon_p \end{pmatrix} \begin{pmatrix} \phi_1^- \\ \phi_0^+ \end{pmatrix} = \begin{pmatrix} q(1-y) \\ qy \end{pmatrix}. \tag{83}$$

As for $N \geq 2$, recalling the equation (36) the matching condition at the 0 and 1 turns to be,

$$\epsilon_s(\phi_N^- u_N^{'-}(0) + \phi_{N-1}^- v_N^{'-}(0) - (\phi_{N-1}^- - \beta_N^-)(u_N^{'-}(0) + v_N^{'-}(0))) = \epsilon_p(\phi_0^+ - \phi_N^-) + q(1-y) \tag{84}$$

$$\epsilon_s(\phi_1^+ u_1^{'+}(1) + \phi_0^+ v_1^{'+}(1) - (\phi_0^+ - \beta_0^+)(u_1^{'+}(1) + v_1^{'+}(1))) = \epsilon_p(\phi_0^+ - \phi_N^-) - qy \tag{85}$$

Put the two equations into the whole matching conditions. Since not involving normal middle matching condition, also treat $N = 2$ as a special case as follows,

$$\begin{pmatrix} u_2^{'-}(x_1^-)+\alpha_1^+ & -u_2^{'-}(x_1^-) & 0 & 0 \\ \epsilon_s v_N^{'-}(0) & \epsilon_s u_N^{'-}(0)+\epsilon_p & -\epsilon_p & 0 \\ 0 & -\epsilon_p & -\epsilon_s v_1^{'+}(1)+\epsilon_p & -\epsilon_s u_1^{'+}(1) \\ 0 & 0 & -u_1^{'+}(x_1^+) & u_1^{'+}(x_1^+)+\alpha_2^- \end{pmatrix} \begin{pmatrix} \phi_1^- \\ \phi_2^- \\ \phi_0^+ \\ \phi_1^+ \end{pmatrix}$$
$$= \begin{pmatrix} \beta_2^-(u_2^{'-}(x_1^-)+v_2^{'-}(x_1^-)) \\ \epsilon_s(\phi'_{N-1}-\beta_N^-)(u_N^{'-}(0)+v_N^{'-}(0))+q(1-y) \\ -\epsilon_s(\phi_0^+-\beta_0^+)(u_1^{'+}(1)+v_1^{'+}(1))+qy \\ -\beta_1^+(u_1^{'+}(x_1^+)+v_1^{'+}(x_1^+)) \end{pmatrix}. \tag{86}$$

When $N \geq 4$, the matching conditions are $Ax = b$, where

$$\begin{pmatrix} b_1 & c_1 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & a_2 & b_2 & c_2 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & a_{N-1} & b_{N-1} & c_{N-1} & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & 0 & a_N & b_N & c_N & 0 & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & 0 & a_{N+1} & b_{N+1} & c_{N+1} & 0 & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & a_{N+2} & b_{N+2} & c_{N+2} & \cdots & 0 \\ 0 & \cdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & a_{2N-1} & b_{2N-1} & c_{2N-1} & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & a_{2N} & b_{2N} \end{pmatrix} \begin{pmatrix} \phi_1^- \\ \phi_2^- \\ \vdots \\ \phi_{N-1}^- \\ \phi_N^- \\ \phi_0^+ \\ \phi_1^+ \\ \vdots \\ \phi_{N-2}^+ \\ \phi_{N-1}^+ \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_{N-1} \\ r_N \\ r_{N+1} \\ r_{N+2} \\ \vdots \\ r_{2N-1} \\ r_{2N} \end{pmatrix}, \tag{87}$$

$$\begin{aligned}
&b_1 = u_2^{'-}(x_1^-)+\alpha_1^+, \quad c_1 = -u_2^{'-}(x_1^-), \\
&a_2 = v_2^{'-}(x_2^-), \quad b_2 = u_2^{'-}(x_2^-) - v_3^{'-}(x_2^-), \quad c_2 = -u_3^{'-}(x_2^-), \\
&a_{N-1} = v_{N-1}^{'-}(x_{N-1}^-), \quad b_{N-1} = u_{N-1}^{'-}(x_{N-1}^-) - v_N^{'-}(x_{N-1}^-), \quad c_{N-1} = -u_N^{'-}(x_{N-1}^-), \\
&a_N = \epsilon_s v_N^{'-}(0), \quad b_N = \epsilon_s u_N^{'-}(0)+\epsilon_p, \quad c_N = -\epsilon_p, \\
&a_{N+1} = -\epsilon_p, \quad b_{N+1} = -\epsilon_s v_1^{'+}(1)+\epsilon_p, \quad c_{N+1} = -\epsilon_s u_1^{'+}(1), \\
&a_{N+2} = v_1^{'+}(x_1^+), \quad b_{N+2} = u_1^{'+}(x_1^+) - v_2^{'+}(x_1^+), \quad c_{N+2} = -u_2^{'+}(x_1^+), \\
&a_{2N-1} = v_{N-2}^{'+}(x_{N-2}^+), \quad b_{2N-1} = u_{N-2}^{'+}(x_{N-2}^+) - v_{N-1}^{'+}(x_{N-2}^+), \quad c_{2N-1} = -u_{N-1}^{'+}(x_{N-2}^+), \\
&a_{2N} = -u_1^{'+}(x_1^+), \quad b_{2N} = u_1^{'+}(x_1^+)+\alpha_2^-,
\end{aligned} \tag{88}$$

$$
\begin{aligned}
r_1 &= \beta_2^-(u_2'^-(x_1^-) + v_2'^-(x_1^-)),\\
r_2 &= (\phi_1^- - \beta_2^-)(u_2'^-(x_2^-) + v_2'^-(x_2^-)) - (\phi_2^- - \beta_3^-)(u_3'^-(x_2^-) + v_3'^-(x_2^-)),\\
r_{N-1} &= (\phi_{N-2}^- - \beta_{N-1}^-)(u_{N-1}'^-(x_{N-1}^-) + v_{N-1}'^-(x_{N-1}^-)) - (\phi_{N-1}^- - \beta_N^-)(u_N'^-(x_{N-1}^-) + v_N'^-(x_{N-1}^-)),\\
r_N &= \epsilon_s(\phi_{N-1}' - \beta_N^-)(u_N'^-(0) + v_N'^-(0)) + q(1-y),\\
r_{N+1} &= -\epsilon_s(\phi_0^+ - \beta_0^+)(u_1'^+(1) + v_1'^+(1)) + qy,\\
r_{N+2} &= (\phi_0^+ - \beta_1^+)(u_1'^+(x_1^+) + v_1'^+(x_1^+)) - (\phi_1^- - \beta_2^+)(u_2'^+(x_1^+) + v_2'^+(x_1^+)),\\
r_{2N-1} &= (\phi_{N-3}^+ - \beta_{N-2}^+)(u_{N-2}'^+(x_{N-2}^+) + v_{N-2}'^+(x_{N-2}^+)) - (\phi_{N-2}^+ - \beta_{N-1}^+)(u_{N-1}'^+(x_{N-2}^+) + v_{N-1}'^+(x_{N-2}^+)),\\
r_{2N} &= -\beta_{N-1}^+(u_{N-1}'^+(x_{N-1}^+) + v_{N-1}'^+(x_{N-1}^+)).
\end{aligned}
\tag{89}
$$

In Figure 5, the numerical results are obtained with input test physical constants,

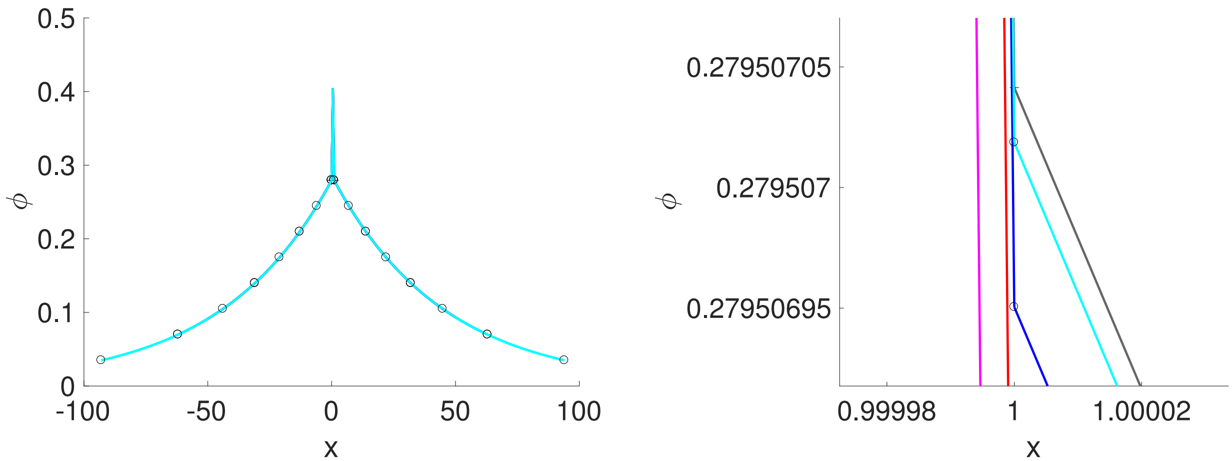

Figure 5: Numerical results: 8-Interval solution (left), magnification around ϕ_0^+ point (right)

$$
\epsilon_s = 80,\ \epsilon_p = 2,\ q = 1,\ \kappa = 0.15,\ y = 0.5, \tag{90}
$$

and the error tolerance is set as 1e-12. The left graph in Figure 5 is the magnification around the ϕ_0^+ point, which only left with the exact solution with the plus symbol and the 8-interval solution with a circle symbol. There are two curves on the right of the 8-interval solution are 4-interval solution and 2-intervals solution. The 4-interval solution is closer to the 8-interval solution which means the algorithm converges monotonically.

The error analysis matches our expectation of $O(N^{-2})$ in Table 2. Besides, the third column in Table 2 also shows that the converged constant calculated by max error times N^2 is a quite small number under 1e-6 scale. Later, we will change different physical constants and see how the max error changes.

N	Max Error	Max Error$\cdot$ N^2
1	0.0000089288	0.0000089288
2	0.0000021887	0.0000087550
4	0.0000005461	0.0000087380
8	0.0000001364	0.0000087349
16	0.0000000341	0.0000087342
32	0.0000000085	0.0000087339
64	0.0000000021	0.0000087334

Table 2: Numerical results: errors analysis for 2^k number of intervals, k = 0 : 6

4 2D polar coordinate with radial symmetry

4.1 solvent part finite domain [a,b]

In 2D polar coordinate, the Laplacian becomes $\nabla^2 = \frac{1}{r}\frac{\partial}{\partial r} + \frac{\partial^2}{\partial r^2} + \frac{1}{r^2}\frac{\partial^2}{\partial \theta^2}$. With radius symmetry, the nonlinear PBE in solvent part becomes,

$$\phi'' + \frac{1}{r}\phi' = \sinh\phi, \tag{91}$$

with arbitrary boundary conditions $\phi(r_0) = \phi_0, \phi(r_N) = \phi_N$, where $r_0 \neq 0$, $r_N \neq 0$. Assume a general right hand side function $f(\phi) = \sinh\phi$ and follow the steps in section 3.2.1, we solve the linearized problem in each sub-interval,

$$\phi_i''(r) + \frac{1}{r}\phi_i'(r) = \alpha_i^2(\phi_i(r) - \phi_{i-1}) + f(\phi_{i-1}), \tag{92}$$

with boundary condition,

$$\phi_i(r_{i-1}) = \phi_{i-1},\ \phi_i(r_i) = \phi_i, \tag{93}$$

where the

$$\alpha_i^2 = \frac{f(\phi_i) - f(\phi_{i-1})}{\phi_i - \phi_{i-1}}. \tag{94}$$

To find the fundamental solution of the homogeneous equation $\phi_i''(r) + \frac{1}{r}\phi_i'(r) - \alpha_i^2\phi_i(r) = 0$ and apply the Theorem 5.5 in Folland's book [4]. Choose $p = q = 1, a = -\alpha_i^2 < 0, b = 0$, we can obtain the general solution,

$$\phi(r) = c_1 I_0(\alpha_i r) + c_2 K_0(\alpha_i r) \tag{95}$$

Then, find a combination of fundamental solutions $u_i(r), v_i(r)$ satisfy the boundary condition,

$$u_i(r_{i-1}) = v_i(r_i) = 0, \qquad u_i(r_i) = v_i(r_{i-1}) = 1. \tag{96}$$

Define

$$Q_i = \frac{1}{I_0(\alpha_i r_{i-1})K_0(\alpha_i r_i) - I_0(\alpha_i r_i)K_0(\alpha_i r_{i-1})} \tag{97}$$

The $u_i(r), v_i(r)$ are given by,

$$u_i(r) = \frac{-I_0(\alpha_i r)K_0(\alpha_i r_{i-1}) + I_0(\alpha_i r_{i-1})K_0(\alpha_i r)}{Q_i}, \tag{98}$$

$$v_i(r) = \frac{I_0(\alpha_i r)K_0(\alpha_i r_i) - I_0(\alpha_i r_i)K_0(\alpha_i r)}{Q_i}. \tag{99}$$

With the same principle in section 3.2.1, we can get the particular solution $P_i(r) = (\phi_{i-1} - \beta_i)(1 - u_i(r) - v_i(r))$ satisfying zero boundary condition $P_i(r_{i-1}) = P_i(r_i) = 0$. Then the general solution in each sub-interval is,

$$\phi_i(r) = \phi_i u_i(r) + \phi_{i-1} v_i(r) + (\phi_{i-1} - \beta_i)(1 - u_i(r) - v_i(r)) \tag{100}$$

The derivative form of the solution is,

$$\phi_i'(r) = \phi_i u_i'(r) + \phi_{i-1} v_i'(r) - (\phi_{i-1} - \beta_i)(u_i'(r) + v_i'(r)) \tag{101}$$

where

$$u_i'(r) = -\frac{\alpha_i}{Q_i}[I_1(\alpha_i r)K_0(\alpha_i r_{i-1}) + I_0(\alpha_i r_{i-1})K_1(\alpha_i r)], \tag{102}$$

$$v_i'(r) = \frac{\alpha_i}{Q_i}[I_1(\alpha_i r)K_0(\alpha_i r_i) + I_0(\alpha_i r_i)K_1(\alpha_i r)]. \tag{103}$$

For the r_i point, applying the matching condition (35) to the derivative of two adjacent piecewise solutions,

$$\begin{aligned}\phi_i u_i'(r_i) + \phi_{i-1} v_i'(r_i) - (\phi_{i-1} - \beta_i)(u_i'(r_i) + v_i'(r_i)) = \\ \phi_{i+1} u_{i+1}'(r_i) + \phi_i v_{i+1}'(r_i) - (\phi_i - \beta_{i+1})(u_{i+1}'(r_i) + v_{i+1}'(r_i)).\end{aligned} \tag{104}$$

Then, follow the steps in section 3.2.1, we can solve the finite domain problem. For the **outer** iteration, we update x values by implementing bisection method in projection.

4.2 half-infinite domain [a,∞]

Follow the similar idea in section 3.2.5, but replace the fundamental solution with modified bessel functions,

$$\phi_N(r) = AI_0(\alpha_N r) + BK_0(\alpha_N r) + \phi_{N-1} - \beta_N, \tag{105}$$

where A and B are unknown coefficients determined by the boundary conditions. With the boundary condition,

$$\phi_N(x_{N-1}) = \phi_{N-1}, \quad \phi_N(\infty) = 0. \tag{106}$$

obtain the solution and its derivative for the last interval,

$$\phi_N(r) = \frac{\phi_{N-1}}{K_0(\alpha_N r_{N-1})} K_0(\alpha_N r), \quad \phi_N'(r) = -\frac{\alpha_N \phi_{N-1}}{K_0(\alpha_N r_{N-1})} K_1(\alpha_N r). \tag{107}$$

The **inner** iteration for the last interior point comes,

$$\phi_{N-1} u_{N-1}'(r_{N-1}) + \phi_{N-2} v_{N-1}'(r_{N-1}) - (\phi_{N-2} - \beta_{N-1})(u_{N-1}'(r_{N-1}) + v_{N-1}'(r_{N-1})) = - \frac{\alpha_N \phi_{N-1}}{K_0(\alpha_N r_{N-1})} K_1(\alpha_N r_{N-1}). \tag{108}$$

After simplification, the last row of the matrix is,

$$-\phi_{N-2} u_{N-1}'(r_{N-1}) + \phi_{N-1}[u_{N-1}'(r_{N-1}) + \alpha_N \frac{K_0(\alpha_N r_{N-1})}{K_1(\alpha_N r_{N-1})}] = -\beta_{N-1}(u_{N-1}'(r_{N-1}) + v_{N-1}'(r_{N-1})). \tag{109}$$

For the one-interval, we treat as a special fixed-point iteration,

$$\phi_1 = \frac{-\beta_1(u_1'(r_1) + v_1'(r_1)) + \phi_0 u_1'(r_1)}{u_1'(r_1) + \alpha_2 \frac{K_0(\alpha_2 r_1)}{K_1(\alpha_2 r_1)}}. \tag{110}$$

while the **outer** iteration turns to be,

$$\phi_N(r) = \frac{\phi_{N-1}}{K_0(\alpha_N r_{N-1})} K_0(\alpha_N r) = \phi_e \tag{111}$$

It can be solved directly but the explicit formula does not work in general. Thus, we choose the revised bisection method (continuously choose twice of the right point if the two end points cannot satisfy the initial condition for bisection method).

4.3 results

The Figure 6 shows the numerical solutions converge monotonically up to the eight-interval solution. The graph contains a original plot of solutions and a magnification version of the plot near the middle point of ϕ.

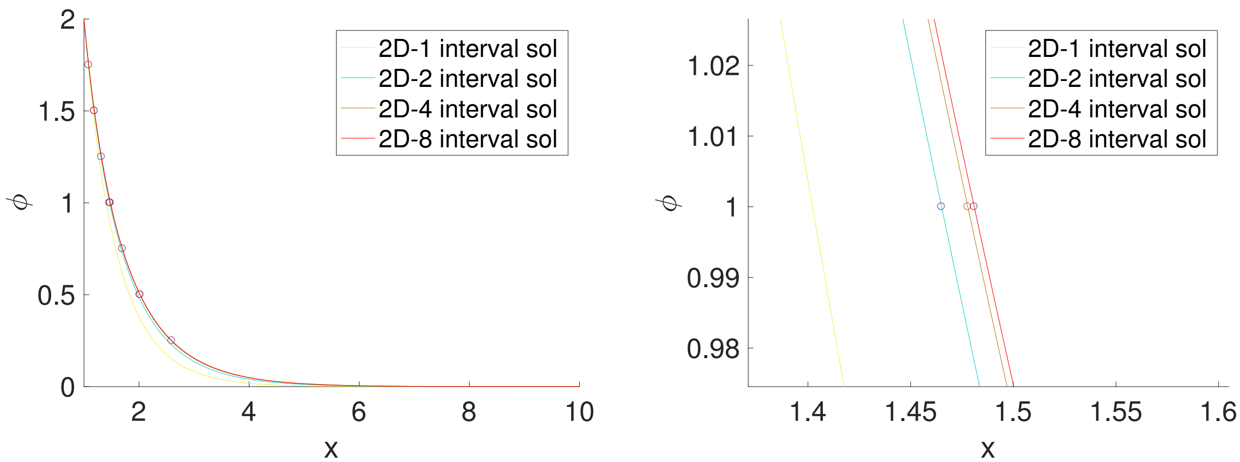

Figure 6: 2D numerical result: 8-Interval solution (left), its magnification around $\phi = 1$ (right); boundary condition: $\phi(0) = 2$, $\phi(\infty) = 0$

N	Max Error	Max Error· N^2
2	0.02296504	0.09186017
4	0.00582557	0.09320905
8	0.00147031	0.09409996
16	0.00036417	0.09322820
32	0.00008989	0.09204906
64	0.00002140	0.08764091

Table 3: Numerical results: errors analysis for 2^k number of intervals, k = 1 : 6; boundary condition : $\phi(1) = 2$, $\phi(\infty) = 0$

The Table 4 contains the error analysis under the error tolerance equals 1e-10. Assume the 128-interval solution is a enough-accurate solution, we compute the max error by taking the max of difference between current interval solution and 128-interval solution on x-mesh points. The first two columns are the number of intervals and the max error based on solution of such that number of intervals. The third and forth column indicates that the error is bound by $O(N^{-2})$ for the first several rows. Although the last row the error bound is invalid since the 64-interval solution is too closer to 128-interval solution, it is reasonable to see the quadratic convergence in 2D case.

4.4 include zero in the domain

Similarly as section 4.2, the general solution in the first interval is given,

$$\phi_1(r) = c_1 I_0(\alpha_1 r) + c_2 K_0(\alpha_1 r) + \phi_0 - \beta_1, \tag{112}$$

where $\beta_1 = \frac{\sinh\phi_0}{\alpha_1^2}$ we treat the first interval as a special case with the specified boundary condition and solve for the two unknown coefficients in the solution,

$$\phi_1'(0) = 0, \quad \phi_1'(L) = \phi_L' \text{ [6]}. \tag{113}$$

With $c_1 = \frac{\phi_L'}{\alpha_1 I_1(\alpha_1 L)}$, $c_2 = 0$, we can represent the solution as a function of the two endpoints ϕ_0 and ϕ_L. Then, we solve for ϕ_0 and ϕ_L by enforcing the boundary condition itself $\phi_1(0) = \phi_0$, $\phi_1(L) = \phi_L$ using fixed-point iteration before going into regular **inner** and **outer** iteration.

$$\phi_0^{i+1} = \frac{\phi_L'}{\alpha_1^i I_1(\alpha_1^i L)} + \phi_0^i - \beta_1^i \tag{114}$$

$$\phi_L^{i+1} = \frac{\phi_L' I_0(\alpha_1^i L)}{\alpha_1^i I_1(\alpha_1^i L)} + \phi_0^i - \beta_1^i \tag{115}$$

where index i means the current step and $i+1$ represents the updated values.

5 3D spherical coordinate with shpecial symmetry

5.1 finite domain

In 3D spherical coordinate, the Laplacian becomes

$$\nabla^2 = \frac{1}{\rho^2}\frac{\partial}{\partial\rho}(\rho^2\frac{\partial}{\partial\rho}) + \frac{1}{\rho^2\sin\theta}\frac{\partial}{\partial\theta}(\sin\theta\frac{\partial}{\partial\theta}) + \frac{1}{\rho^2\sin^2\theta}\frac{\partial^2}{\partial\psi^2}. \tag{116}$$

Under spherical symmetry, the nonlinear PBE is given,

$$\phi'' + \frac{2}{\rho}\phi' = \sinh\phi \text{ [5]}, \quad \text{with boundary condition : } \phi(\rho_0) = \phi_0,\ \phi(\rho_N) = \phi_N, \tag{117}$$

where $\rho_0 \neq 0$ and $\rho_N \neq 0$. Following the same quasilinearization technique, we first solve the linearized problem in each sub-interval,

$$\phi_i''(\rho) + \frac{2}{\rho}\phi_i'(\rho) = \alpha_i^2(\phi_i(\rho) - \phi_{i-1}) + f(\phi_{i-1}), \tag{118}$$

with boundary condition,

$$\phi_i(\rho_{i-1}) = \phi_{i-1},\ \phi_i(\rho_i) = \phi_i, \tag{119}$$

where the

$$\alpha_i^2 = \frac{f(\phi_i) - f(\phi_{i-1})}{\phi_i - \phi_{i-1}}. \tag{120}$$

First, to find the fundamental solution of homogeneous equation

$$\phi_i''(\rho) + \frac{2}{\rho}\phi_i'(\rho) - \alpha_i^2\phi_i(\rho) = 0, \tag{121}$$

we introduce a new function $g(\rho)$ such that

$$\phi = \frac{g}{\rho},\ \phi' = \frac{1}{\rho}g' - \frac{g}{\rho^2},\ \phi'' = \frac{2}{\rho^3}g + \frac{1}{\rho}g'' - \frac{2}{\rho^2}g', \tag{122}$$

and substitute into equation (121) to obtain,

$$g'' - \alpha_i^2 g = 0. \tag{123}$$

Back solve to get the two fundamental solutions of equation (121) : $\frac{e^{-\alpha_i\rho}}{\rho}$ and $\frac{e^{\alpha_i\rho}}{\rho}$. Then, find a combination of fundamental solutions,

$$u_i(\rho) = \frac{\rho_i \sinh\alpha_i(\rho - \rho_{i-1})}{\rho\sinh\alpha_i(\rho_i - \rho_{i-1})}, \qquad v_i(\rho) = \frac{\rho_{i-1}\sinh\alpha_i(\rho_i - \rho)}{\rho\sinh\alpha_i(\rho_i - \rho_{i-1})}. \tag{124}$$

satisfing the boundary condition,

$$u_i(\rho_{i-1}) = v_i(\rho_i) = 0, \qquad u_i(\rho_i) = v_i(\rho_{i-1}) = 1. \tag{125}$$

Combine the particular solution satisfying zero boundary condition (same principle in section 3.2.1), the general solution in linearized sub-interval is given,

$$\phi_i(\rho) = \phi_i u_i(\rho) + \phi_{i-1}v_i(\rho) + (\phi_{i-1} - \beta_i)(1 - u_i(\rho) - v_i(\rho)) \tag{126}$$

The derivative form of the solution is,

$$\phi_i'(\rho) = \phi_i u_i'(\rho) + \phi_{i-1}v_i'(\rho) - (\phi_{i-1} - \beta_i)(u_i'(\rho) + v_i'(\rho)), \tag{127}$$

where

$$u_i'(\rho) = \frac{\rho_i\alpha_i\cosh\alpha_i(\rho - \rho_{i-1})}{\rho\sinh\alpha_i(\rho_i - \rho_{i-1})} - \frac{\rho_i\sinh\alpha_i(\rho - \rho_{i-1})}{\rho^2\sinh\alpha_i(\rho_i - \rho_{i-1})}, \tag{128}$$

$$v_i'(\rho) = -\frac{\rho_{i-1}\alpha_i\cosh\alpha_i(\rho_i - \rho)}{\rho\sinh\alpha_i(\rho_i - \rho_{i-1})} - \frac{\rho_{i-1}\sinh\alpha_i(\rho_i - \rho)}{\rho^2\sinh\alpha_i(\rho_i - \rho_{i-1})}. \tag{129}$$

Then, implement the **inner** and **outer** similarly.

5.2 semi-infinite domain

Under similar idea, give the general solution of the special last interval and its corresponding boundary conditions to solve unknown coefficients,

$$\phi_N(x) = A\frac{e^{\alpha_N \rho}}{\rho} + B\frac{e^{-\alpha_N \rho}}{\rho} + \phi_{N-1} - \beta_N. \tag{130}$$

$$\phi_N(\rho_{N-1}) = \phi_{N-1}, \quad \phi_N(\infty) = 0. \tag{131}$$

The special solution an its derivative for the last interval,

$$\phi_N(\rho) = \phi_{N-1}\rho_{N-1}\frac{e^{-\alpha_N(\rho-\rho_{N-1})}}{\rho}, \quad \phi_N'(\rho) = \phi_{N-1}\rho_{N-1}(-\frac{\alpha_N}{\rho} - \frac{1}{\rho^2})e^{-\alpha_N(\rho-\rho_{N-1})}. \tag{132}$$

For the matching condition, **inner** iteration, the special case of the last interval turns to be,

$$\phi_{N-1}'(x_{N-1}) = \phi_N'(x_{N-1}) \tag{133}$$

$$\begin{aligned} \Rightarrow \phi_{N-1}u_{N-1}'(x_{N-1}) + \phi_{N-2}v_{N-1}'(x_{N-1}) - (\phi_{N-2} - \beta_{N-1})(u_{N-1}'(x_{N-1}) + v_{N-1}'(x_{N-1})) \\ = (-\alpha_N - \frac{1}{\rho_{N-1}})\phi_{N-1}. \end{aligned} \tag{134}$$

Then last row of the matrix that applies the matching condition is,

$$-\phi_{N-2}u_{N-1}'(x_{N-1}) + \phi_{N-1}(u_{N-1}'(x_{N-1}) + \alpha_N + \frac{1}{\rho_{N-1}}) = -\beta_{N-1}(u_{N-1}'(x_{N-1}) + v_{N-1}'(x_{N-1})). \tag{135}$$

Also, for the first ϕ midpoint, the explicit formula is given,

$$\phi_1 = \frac{-\beta_1(u_1'(x_1) + v_1'(x_1)) + \phi_0 u_1'(x_1)}{u_1'(x_1) + \alpha_2 + \frac{1}{\rho_1}}. \tag{136}$$

5.3 results

The Figure 7 shows the numerical solutions converge monotonically up to the eight-interval solution. The graph contains a original plot of solutions and a magnification version of the plot near the middle point of ϕ.

The Table 4 contains the error analysis under the error tolerance equals 1e-10. Assume the 128-interval solution is a enough-accurate solution, we compute the max error by taking the max of difference between current interval solution and 128-interval solution on x-mesh points. The first two columns are the number of intervals and the max error based on solution of such

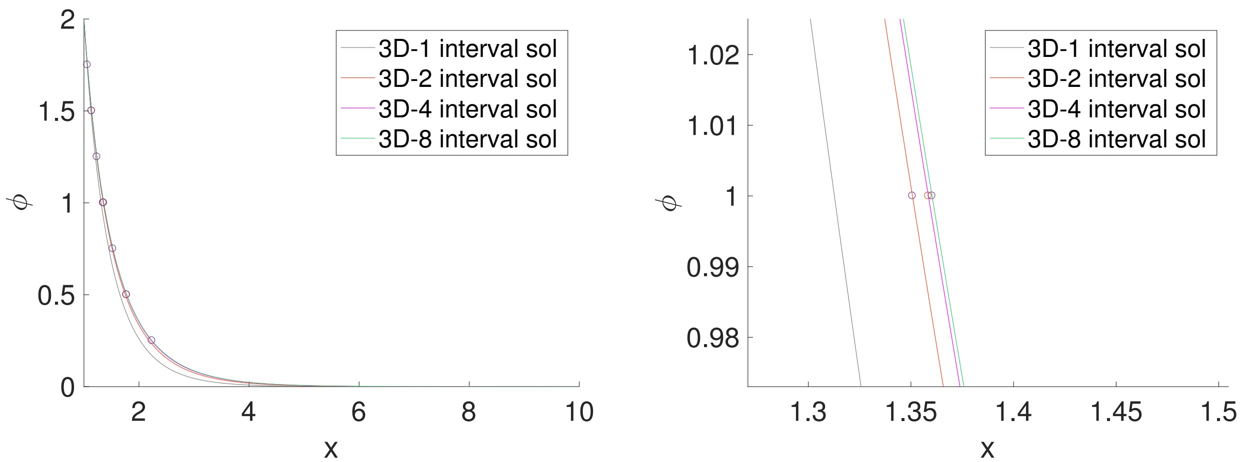

Figure 7: 3D Numerical result: 8-Interval solution (left), its magnification around $\phi = 1$ (right); boundary condition: $\phi(1) = 2$, $\phi(\infty) = 0$

N	Max Error	Max Error$\cdot$ N^2
2	0.01809091	0.07236365
4	0.00458612	0.07337786
8	0.00110604	0.07078673
16	0.00027151	0.06950628
32	0.00006680	0.06840531
64	0.00001587	0.06500015

Table 4: Numerical results: errors analysis for 2^k number of intervals, k = 1 : 6; boundary condition : $\phi(1) = 2$, $\phi(\infty) = 0$

that number of intervals. The third and forth column indicates that the error is bound by $O(N^{-2})$ for the first several rows. Although the last several rows the error bound is invalid since the 64-interval solution is too closer to 128-interval solution, it is reasonable to see the quadratic convergence in 3D case.

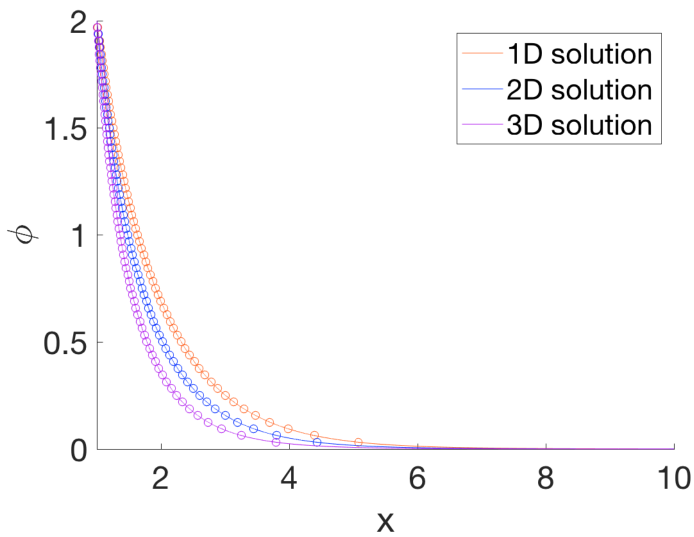

Figure 8: Comparison between 1D, 2D and 3D 64-interval numerically convergent solutions with same boundary condition : $\phi(1) = 2$, $\phi(\infty) = 0$

6 comparison of 1D, 2D, 3D cases

As we can see in the Figure 8, the solutions of 1D, 2D, and 3D nonlinear PBE under the same boundary condition $\phi(1) = 2$, $\phi(\infty) = 0$ have an order relation such that,

$$\phi_{3D} \leq \phi_{2D} \leq \phi_{1D}. \tag{137}$$

It will be interesting to prove for the process.

7 Conclusion

In the report, we focus on solving the 1D, 2D, and 3D nonlinear PBE. Quasilinearization, a numerical technique, is applied to the solve the nonlinear PBE which approximates the nonlinear term by piecewise interpolated lines. The numerical result showing that the scheme converges at the rate of $O(N^{-2})$, where N is the number of the intervals in the approximation. We also compared the solution of 1D, 2D, and 3D nonlinear PBE but leave the analysis for the next. Currently, we are applying the quasilinearization technique to the 2D solvent/protein/solvent model and do the corresponding analysis. We are also exploring to compare our numerical method and Runge-Kutta shooting method on the test of charged surface modified PBE schema. In the future, we will see how to extend the numerical method to get non-monotonical solution, especially figure out a robust way to figure out intervals and do the projection in **outer** iteration.

References

[1] W. Geng and R. Krasny., *A treecode-accelerated boundary integral Poisson-Boltzmann solver for electrostatics of solvated biomolecules.*, J. Comput. Physics, **247**, 62-79 (2013).

[2] R K. Hobbie, B. J. Roth., *Intermediate Physics for Medicine Biology*, Springer, 242,262-263, (2015).

[3] R. C. Y. Chin and R. Krasny, *A hybrid asymptotic-finite method for stiff two-point boundary value problem*, SIAM J. Sci. Stat. Comput., **4**, 229-243 (1983).

[4] Gerald B. Folland., *Fourier Analysis and its Application*, Wadsworth & Brooks/Cole Advanced Books & Software, 161-162, (1992).

[5] W. Rocchia, *Poisson-Boltzmann Equation Boundary Conditions for Biological Applications*, Math. and Comp. Modeling, **41**, 1109-1118 (2005).

[6] K. Bohinc, A. Igic, T. Slivnik, *Linearized Poisson Boltzmann theory in cylindrical geometry*, Electrotechnical Review, **75**, 82-84 (2008).