MATLAB version of the APBS and its corresponding GUI components

(Dated: December 16, 2009)

Abstract

This solver uses the biconjugate gradient stabilized method and the inexact LU decomposition to numerically solve the linearized PB equation on a Cartesian 3D-grid. This version requires the shifted dielectric and the ion accessibility coefficient (kappa function) maps as generated by the APBS code as well as the corresponding pqr file generated by the pdb2pqr code. It uses standard three-linear splines (spl0) to spread the charge density along the nearest grid points if needed. It is able to solve the linearized PB eq with either Dirichlet or Focus boundary conditions. In the later case, this code solves the PB equation in a large (low resolution) domain and the resulting electrostatic potential solution is subsequently used to evaluate the Dirichlet boundary condition to solve the PB equation in a (higher resolution) sub-domain region. The resulting electrostatic potential and charge maps for both the coarse and target grids are saved in dx format in different folders. This version includes the option to calculate the energy for point-like charge systems. For visualization purpose, this code also generates two files (.fig and .tiff) corresponding to the graphical representation of the electrostatic potential surface. This version was used to solve the most of models provided by the APBS package in the Examples section. For the Dirichlet Boundary Condition, this version basically provides the same results than the previous one. The main difference is that the user doesn't have to edit the source files in this version but only have to provide the target input-file name and the corresponding full path as the only argument of the (new) Matlab function MAPBS (x). See the pka example and the MATLAB_PB_SOLVER_6 package for more details. In addition, we developed two useful GUI applications, named MATLAB_APBS.m and comparison_pot.m, which provide an easy and efficient way to generate the Matlab input files and to run MAPBS(x) without the need for a text editor. The former GUI helps the user to generate the proper Matlab input files using just a browser to search the required information on your computer. This would avoid errors in generating input files and make the user feels more comfortable using this code. Once the input files was generated by the GUI, the user may run the MAPBS code by simply clicking a push button. The other GUI helps the user to perform the comparison between MAPBS and APBS for reaction field potentials as well as for solvated or reference states. It brings basically the same features than the aforementioned GUI.

Description

This code is based on Michel Holst's thesis and Nathan Baker's APBS approach. The box-method is used to discretize the following (linearized) PB equation

$$-\nabla . \left(\epsilon\left(\mathbf{r}\right)\nabla u\left(\mathbf{r}\right)\right)+\bar{\kappa}\left(\mathbf{r}\right)^{2}u\left(\mathbf{r}\right)=magic\sum_{i=1}^{N}z_{i}\delta\left(\mathbf{r}-\mathbf{r}_{i}\right) \tag{1}$$

where $u\left(\mathbf{r}\right)=e_{c}\Phi\left(\mathbf{r}\right)/K_{B}T$ and $magic=4\pi e_{c}^{2}/K_{B}T$. For a diagonal dielectric tensor, the resulting discretized linear PB equations at the nodes $u_{ijk}=u\left(x_{i},y_{j},z_{k}\right)$ for $1\leq i\leq N_{x}$, $1\leq j\leq N_{y}$ and $1\leq k\leq N_{z}$ reads

$$\left[\epsilon^{x}_{i-1/2,j,k}\frac{\left(h_{j-1}+h_{j}\right)\left(h_{k-1}+h_{k}\right)}{4h_{i-1}}+\epsilon^{x}_{i+1/2,j,k}\frac{\left(h_{j-1}+h_{j}\right)\left(h_{k-1}+h_{k}\right)}{4h_{i}}+\right.$$

$$\epsilon^{y}_{i,j-1/2,k}\frac{\left(h_{i-1}+h_{i}\right)\left(h_{k-1}+h_{k}\right)}{4h_{j-1}}+\epsilon^{y}_{i,j+1/2,k}\frac{\left(h_{i-1}+h_{i}\right)\left(h_{k-1}+h_{k}\right)}{4h_{j}}+$$

$$\epsilon^{k}_{i,j,k-1/2}\frac{\left(h_{i-1}+h_{i}\right)\left(h_{j-1}+h_{j}\right)}{4h_{k-1}}+\epsilon^{k}_{i,j,k+1/2}\frac{\left(h_{i-1}+h_{i}\right)\left(h_{j-1}+h_{j}\right)}{4h_{k}}+$$

$$\left.\kappa_{ijk}\frac{\left(h_{i-1}+h_{i}\right)\left(h_{j-1}+h_{j}\right)\left(h_{k-1}+h_{k}\right)}{8}\right]u_{ijk}+$$

$$\left[-\epsilon^{x}_{i-1/2,j,k}\frac{\left(h_{j-1}+h_{j}\right)\left(h_{k-1}+h_{k}\right)}{4h_{i-1}}\right]u_{i-1jk}+\left[-\epsilon^{x}_{i+1/2,j,k}\frac{\left(h_{j-1}+h_{j}\right)\left(h_{k-1}+h_{k}\right)}{4h_{i}}\right]u_{i+1jk}+$$

$$\left[-\epsilon^{y}_{i,j-1/2,k}\frac{\left(h_{i-1}+h_{i}\right)\left(h_{k-1}+h_{k}\right)}{4h_{j-1}}\right]u_{ij-1k}+\left[-\epsilon^{y}_{i,j+1/2,k}\frac{\left(h_{i-1}+h_{i}\right)\left(h_{k-1}+h_{k}\right)}{4h_{j}}\right]u_{ij+1k}+$$

$$\left[-\epsilon^{k}_{i,j,k-1/2}\frac{\left(h_{i-1}+h_{i}\right)\left(h_{j-1}+h_{j}\right)}{4h_{k-1}}\right]u_{ijk-1}+\left[-\epsilon^{k}_{i,j,k+1/2}\frac{\left(h_{i-1}+h_{i}\right)\left(h_{j-1}+h_{j}\right)}{4h_{k}}\right]u_{ijk+1}=$$

$$magic\frac{\left(h_{i-1}+h_{i}\right)\left(h_{j-1}+h_{j}\right)\left(h_{k-1}+h_{k}\right)}{8}f_{ijk} \tag{2}$$

in which

$$h_{i}=x_{i+1}-x_{i},\ h_{j}=y_{j+1}-y_{j}\ h_{k}=z_{k+1}-z_{k}$$

The delta functions appearing in the right hand side of the starting equations are approximated with linear B-splines (spl0) which spread the point like charge along the nearest neighborhood. The resulting f_{ijk} represent the smearing of the point charges along the grid points.

For more details, including used unit system, please refer to the Michel Holst's thesis and the APBS user guide online. To visualize more clearly the problem, let's explicitly write the first equations for a cubic grid of 5x5x5 containing general coefficients

$$a_{222}u_{222} + a_{122}u_{122} + a_{322}u_{322} + a_{212}u_{212} + a_{232}u_{232} + a_{221}u_{221} + a_{223}u_{223} = f_{222}$$

$$a_{322}u_{322} + a_{222}u_{222} + a_{422}u_{422} + a_{312}u_{312} + a_{332}u_{332} + a_{321}u_{321} + a_{323}u_{323} = f_{322}$$

$$a_{422}u_{422} + a_{122}u_{122} + a_{322}u_{322} + a_{212}u_{212} + a_{232}u_{232} + a_{221}u_{221} + a_{223}u_{223} = f_{422}$$

$$a_{232}u_{232} + a_{132}u_{132} + a_{332}u_{332} + a_{222}u_{222} + a_{242}u_{242} + a_{231}u_{231} + a_{233}u_{233} = f_{232}$$

$$\ldots$$

in which the nodes are arranged using the natural ordering

$$U = [u_{111}, u_{211}, .., u_{N_x 11}, u_{121}, .., u_{221}, u_{321}, .., u_{N_x 21} ..., u_{N_x N_y N_z}]^T$$

Note that the prescribed values of nodes $u_{1jk}, u_{N_x,j,k}, u_{i,1,k}, u_{i,N_y,k}, u_{ij1}$ and u_{ijN_z} along the faces of the box coming from the Dirichlet boundary conditions will have their corresponding elements removed in such a way that only equations for the interior nodes remain. In other words, we will only consider the following set of unknown nodes

$$U = [u_{222}, u_{322}, .., u_{N_x-1,22}, u_{232}, .., u_{332}, u_{432}, .., u_{N_x-2,32} ..., u_{N_x-1,N_y-1,N_z-1}]^T$$

in such a way that the previous equations become

$$a_{222}u_{222} + a_{322}u_{322} + a_{232}u_{232} + a_{223}u_{223} = f_{222} - a_{122}u_{122} - a_{212}u_{212} - a_{221}u_{221} \equiv b_{222}$$

$$a_{322}u_{322} + a_{222}u_{222} + a_{422}u_{422} + a_{332}u_{332} + a_{323}u_{323} = f_{322} - a_{312}u_{312} - a_{321}u_{321} \equiv b_{322}$$

$$a_{422}u_{422} + a_{322}u_{322} + a_{232}u_{232} + a_{223}u_{223} = f_{422} - a_{122}u_{122} - a_{212}u_{212} - a_{221}u_{221} \equiv b_{422}$$

$$a_{232}u_{232} + a_{332}u_{332} + a_{222}u_{222} + a_{242}u_{242} + a_{233}u_{233} = f_{232} - a_{132}u_{132} - a_{231}u_{231} \equiv b_{232}$$

in which the boundary $u's$ are conveniently brought to the right-hand-side of the equations. The resulting left-hand side equations can be written in compact form in term of matrix vector product as follows

$$Au = b$$

in which

$$u\,(p) = u_{ijk}, \qquad b\,(p) = b_{ijk}, \qquad p = (k-2)(N_x-2)(N_y-2) + (j-2)(N_x-2) + i - 1$$

$$i = 2, .., N_x - 2, \quad j = 2, .., N_y - 2, \quad k = 2, .., N_z - 2$$

and A is a (seven banded block tri-diagonal form) $(N_x-2)(N_y-2)(N_z-2)$ by $(N_x-2)(N_y-2)(N_z-2)$ squared symmetric positive definite matrix containing the following nonzero elements (see figure 1):

- The main diagonal elements

$$d_0(p) = \Bigg[\epsilon^{x}_{i-1/2,j,k}\frac{(h_{j-1}+h_j)\,(h_{k-1}+h_k)}{4h_{i-1}} + \epsilon^{x}_{i+1/2,j,k}\frac{(h_{j-1}+h_j)\,(h_{k-1}+h_k)}{4h_i} +$$

$$\epsilon^{y}_{i,j-1/2,k}\frac{(h_{i-1}+h_i)\,(h_{k-1}+h_k)}{4h_{j-1}} + \epsilon^{y}_{i,j+1/2,k}\frac{(h_{i-1}+h_i)\,(h_{k-1}+h_k)}{4h_j} +$$

$$\epsilon^{k}_{i,j,k-1/2}\frac{(h_{i-1}+h_i)\,(h_{j-1}+h_j)}{4h_{k-1}} + \epsilon^{k}_{i,j,k+1/2}\frac{(h_{i-1}+h_i)\,(h_{j-1}+h_j)}{4h_k} +$$

$$\kappa_{ijk}\frac{(h_{i-1}+h_i)\,(h_{j-1}+h_j)\,(h_{k-1}+h_k)}{8}\Bigg] \tag{3}$$

- The Next upper band diagonal, which is shifted in one column to the left from the first column, contains the following elements

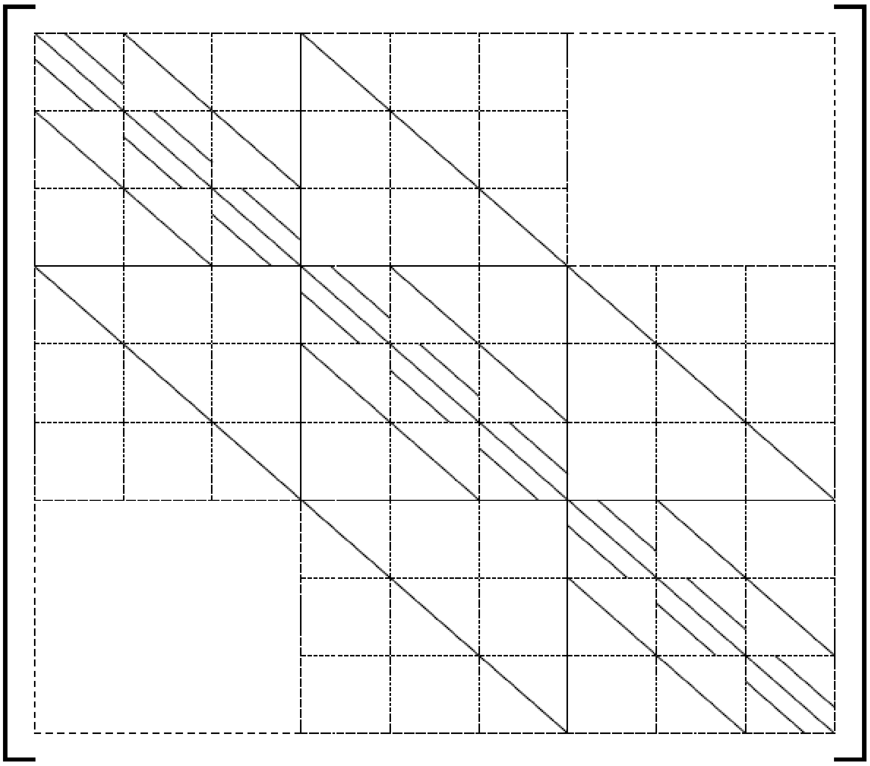


Figure 1: A Matrix representation

$$\left[d_1(p) = -\epsilon^{x}_{i+1/2,j,k} \frac{(h_{j-1} + h_j)\,(h_{k-1} + h_k)}{4h_i} \right] \tag{4}$$

- The second upper band diagonal which is shifted $N_x - 2$ columns from the first column

$$d_2(p) = \left[-\epsilon^{y}_{i,j+1/2,k} \frac{(h_{i-1} + h_i)\,(h_{k-1} + h_k)}{4h_j} \right] \tag{5}$$

- The third upper band diagonal which is shifted $(N_x - 2)(N_y - 2)$ columns from the first column

$$d_3(p) = \left[-\epsilon^{k}_{i,j,k+1/2} \frac{(h_{i-1} + h_i)\,(h_{j-1} + h_j)}{4h_k} \right] \tag{6}$$

The remaining elements of the upper triangular squared matrix A are set equal to zero. By symmetry we obtain the lower triagonal elements of the matrix A. Because the matrix A is sparse and large, we can implement efficient methods that optimally solve the linear system for U. Specifically, we use the biconjugate gradient stabilized method combined with the inexact LU decomposition of the matrix A. Having the numerical values for the nodes in the interior of the box, we finally add the previously removed prescribed values along the six faces to get the solution over the complete set of grid points.

Dirichlet Boundary Condition

The values of nodes $u_{1jk}, u_{N_x,j,k}, u_{i,1,k}, u_{i,N_y,k}, u_{ij1}$ and u_{ijN_z} along the six faces of the box is set to the values prescribed by a Debye-Hückel model for a multiple, non-interacting spheres with a point charges. The sphere radii are set to the atomic radii of the biomolecule and the sphere charges are set to the total charge of the protein.

Focus Boundary Condition

Our code uses linear interpolation to obtain the value of the potential at the six faces of the target box from the value of the potential obtained at larger domain.

Computational algorithm for Dirichlet boundary conditions

1. The code reads the (target) input file .inm to get the APBS input files (shifted dielectric coefficients, kappa function and pqr data file) as well as the number of (target) grid points, the box Lengths, temperature, and bulk properties (ionic strength and solvent dielectric coefficient) among other parameters.

2. The center of the grid is evaluated from the corresponding pqr file.

3. By using linear B-splines, the charge density is discretized to get f_{ijk} for $i = 1, .., N_x, \quad j = 1, .., N_y, \quad k = 1, .., N_z$. The Dirichlet boundary condition along the six faces of the box $u_{1jk}, u_{N_x,j,k}, u_{i,1,k}, u_{i,N_y,k}, u_{ij1}$ and u_{ijN_z} are calculated by using the temperature, the value of the bulk dielectric coefficient (usually water) and ionic strength.

4. The nonzero components of the matrix A, e.g., the diagonal elements $d_0(p), d_1(p), d_2(p)$, and $d_3(p)$, for $p = (k-2)(N_x-2)(N_y-2)+(j-2)(N_x-2)+i-1$ and $i = 2,..,N_x-1, \quad j = 2,..,N_y-1, \quad k = 2,..,N_z-1$ are evaluated by using the expressions (3), (4),(5), and (6). The values for the shifted dielectric coefficients and kappa function elements are obtained from the APBS input files. The values of the mesh size h_i, h_j and h_k are obtained from the number of grid points and the Length of the box. Next, the sparse upper triangular matrix A is constructed by filling with zeros the remaining elements of the matrix A. Next, the lower triangular elements of the matrix A are obtained by using the following symmetry property $A_{pq} = A_{qp}$ for $q = 1,..,(N_x-2)(N_y-2)(N_z-2)$ and $p = q,..,(N_x-2)(N_y-2)(N_z-2)$.

5. The elements of b_{ijk} are evaluated by using the values obtained for the discretized charge density f_{ijk} and the values of the Dirichlet boundary elements multiplied by the appropriate shifted dielectric coefficient values. The natural ordering $p = (k-2)(N_x-2)(N_y-2)+(j-2)(N_x-2)+i-1$ and $i = 2,..,N_x-1, \quad j = 2,..,N_y-1, \quad k = 2,..,N_z-1$ is used to construct the corresponding vector $b(p)$ (one index) from the data array structure (three indices) b_{ijk}.

6. The inexact LU decomposition of the matrix A is performed. The default tolerance value is set equal to 0.25 which provides a fast evaluation of the matrices L and U.

7. The resulting L and U matrices, the matrix A and the vector b are used to approximately solve $Au = b$ for the vector u using the biconjugate gradient stabilized method. The default accuracy is set equal to 10^-9 and the maximum number of iteration equal to 800.

8. The natural ordering relationship is used to convert the resulting vector $u(p)$ to data array structure to get the numerical solution for u_{ijk} for $i = 2,..,N_x-1, \quad j = 2,..,N_y-1, \quad k = 2,..,N_z-1$.

9. Finally the previously removed values of the nodes at the faces of the box are used to obtain the solution for the nodes u_{ijk} over the complete set of grid points, namely for $i = 1,..,N_x, \quad j = 1,..,N_y, \quad k = 1,..,N_z$.

10. The electrostatic potential u_{ijk} and the charge f_{ijk} maps are saved in dx format files.

11. If required, it calculates the energy using linear interpolation to evaluate the solution obtained in the grid at the exact location of the point-like charges.

12. The electrostatic potential surface $u_{ij(N_z+1)/2}$ is saved in tiff and fig format files for visualization purpose.

Computational algorithm for Focus boundary conditions

The algorithm reads the target input file finding that the boundary condition line says "focusname.inm" instead of "sdh". Then the matlab code automatically first reads that input file "focusname.inm" to solve the PB equation in the specified coarse grid using Dirichlet boundary condition as explained previously. It saves the resulting electrostatic potential solution in a temporary dx formatted file and then perform the following steps;

1. The code reads the (target) input file .inm to get the APBS input files (shifted dielectric coefficients, kappa function and pqr data file) as well as the number of (target) grid points, the box Lengths, temperature, and bulk properties (ionic strength and solvent dielectric coefficient) among other parameters.

2. The center of the grid is evaluated from the corresponding pqr file.

3. By using linear B-splines, the charge density is discretized to get f_{ijk} for $i = 1,..,N_x, \quad j = 1,..,N_y, \quad k = 1,..,N_z$. The Dirichlet boundary condition along the six faces of the target (smaller) box $u_{1jk}, u_{N_x,j,k}, u_{i,1,k}, u_{i,N_y,k}, u_{ij1}$ and u_{ijN_z} are calculated by using a three-linear interpolation for the electrostatic potential solution obtained previously at larger boxsides (Focus boundary Condition).

4. The nonzero components of the matrix A, e.g., the diagonal elements $d_0(p), d_1(p), d_2(p)$, and $d_3(p)$, for $p = (k-2)(N_x-2)(N_y-2)+(j-2)(N_x-2)+i-1$ and $i = 2,..,N_x-1, \quad j = 2,..,N_y-1, \quad k = 2,..,N_z-1$ are evaluated by using the expressions (3), (4),(5), and (6). The values for the shifted dielectric coefficients and kappa function elements are obtained from the APBS input files. The values of the mesh size h_i, h_j and h_k are obtained from the number of grid points and the Length of the box. Next, the sparse upper triangular matrix A is constructed by filling with zeros the remaining elements of the matrix A. Next, the lower triangular elements of

the matrix A are obtained by using the following symmetry property $A_{pq} = A_{qp}$ for $q = 1, .., (N_x - 2)(N_y - 2)(N_z - 2)$ and $p = q, .., (N_x - 2)(N_y - 2)(N_z - 2)$.

5. The elements of b_{ijk} are evaluated by using the values obtained for the discretized charge density f_{ijk} and the values of the Dirichlet boundary elements multiplied by the appropriate shifted dielectric coefficient values. The natural ordering $p = (k-2)(N_x - 2)(N_y - 2) + (j-2)(N_x - 2) + i - 1$ and $i = 2, .., N_x - 1, \quad j = 2, .., N_y - 1, \quad k = 2, .., N_z - 1$ is used to construct the corresponding vector $b(p)$ (one index) from the data array structure (three indices) b_{ijk}.

6. The inexact LU decomposition of the matrix A is performed. The default tolerance value is set equal to 0.25 which provides a fast evaluation of the matrices L and U.

7. The resulting L and U matrices, the matrix A and the vector b are used to approximately solve $Au = b$ for the vector u using the biconjugate gradient stabilized method. The default accuracy is set equal to 10^-9 and the maximum number of iteration equal to 800.

8. The natural ordering relationship is used to convert the resulting vector $u(p)$ to data array structure to get the numerical solution for u_{ijk} for $i = 2, .., N_x - 1, \quad j = 2, .., N_y - 1, \quad k = 2, .., N_z - 1$.

9. Finally the previously removed values of the nodes at the faces of the box are used to obtain the solution for the nodes u_{ijk} over the complete set of grid points, namely for $i = 1, .., N_x, \quad j = 1, .., N_y, \quad k = 1, .., N_z$.

10. The electrostatic potential u_{ijk} and the charge f_{ijk} maps are saved in dx format files.

11. If required, it calculates the energy using linear interpolation to evaluate the solution obtained in the grid at the exact location of the point-like charges.

12. The electrostatic potential surface $u_{ij(N_z+1)/2}$ is saved in tiff and fig format files for visualization purpose.

Comment1: Note that in this case the user have to provide two inm.files and the corresponding dx and pqr files for both the coarse and target grids.

Comments2: In this version the user have to provide two pqr files, one representing the molecule by which the PB eq is solved and, the second one to define the center of the grid. It may be the same than the first one, but in general, for complex systems they are not.

Comments3: Finally, the user have to provide both directories for the input and output files respectively. In this way the user doesn't have to edit the source files at all. Just need to provide the name of the input file and the full path as the only argument in the Matlab function MAPBS (x).

Input File structure

The .inm file parsing is strict. Input must contain the value of these parameters in exactly this order in a column:

dime

glen

T

bulk

bc

digpres

dielx_str

diely_str

dielz_str

kappa_str

pqr_str

pqr_cent_str

energy

in_name_str

name_str

dime

This line is left to specify the number of grid points. The values for this keyword are:
nx ny nz

the (integer) number of grid points in the x-, y-, and z-directions, respectively.

glen

This line is left to specify the mesh domain lengths; this may be different in each direction. The values for this keyword are:

xlen ylen zlen

the (floating point) grid lengths in the x-, y-, and z-directions (respectively) in Å.

T

This line is left to specify the temperature of the system in Kelvin. The value for this keyword is:

T

bulk

This line is left to specify the bulk properties. The values for this keyword are:

I Solv-epsilon

where **I** is the ionic strength defined by $I = 0.5 \sum_i c_i z_i^2$ where the dummy sum is over all different ionic species, z_i and $c_i$are the valence and ionic concentration in moles respectively. The other parameter "**Solv-epsilon**" represents the value of the solvent dielectric coefficient (usually equal to 78.54 for water).

bc

This line is left to specify the boundary condition to be used to solve the linear PB equation. The words for this keyword can be either

sdh

if the user requires Dirichlet boundary condition, or

focusname.inm

if the user requires focus boundary condition.

digpres

This line is left to specify the number of digits of precision in the residual error obtained in the solution of the linear eq generated by the biconjugate gradient method. The value for this keyword is:

N

where usually this number is set equal to 6.

dielx_ str

This line is left to specify the name of the shifted x-component of the dielectric coefficients in dx format as generated by APBS. For instances

xfilename.dx

diely_ str

This line is left to specify the name of the shifted y-component of the dielectric coefficients in dx format as generated by APBS. For instances

yfilename.dx

dielz_ str

This line is left to specify the name of the shifted z-component of the dielectric coefficients in dx format as generated by APBS. For instances

zfilename.dx

kappa_ str

This line is left to specify the name of the ionic accessibility coefficients kappa in dx format (.dx extension) as generated by APBS. For instances

kappafilename.dx

pqr_str

This line is left to specify the name of the pqr file (.pqr extension) generated by pdb2pqr determining the target molecule by which the PB eq will be solved. For instances

moleculetarget.pqr

pqr_cent_str

This line is left to specify the name of the pqr file (.pqr extension) generated by pdb2pqr determining the molecule defining the center of grid. It may be the same than pqr_str if there is only one molecule. In complex systems having more than one molecular species it uses to define one of them as the molecule reference. In such cases the center of grid would be defined by the coordinates of such molecule for all the molecular species contained in the complex system. For instances

moleculereference.pqr

In some cases the user prefers to define the three coordinates (in A units) of the center of grid explicitly, namely xcent ycent zcent, instead of calculating these coordinates from the location of the atoms defined in an specific molecule pqr file. Because this line only reads pqr file, the easy way to do this is by writing a pqr file for only one atom having as coordinates those defined by the center of grid. For instances, if the user want to define the center of grid at the origin, namely at 0 0 0, then the user should write a pqr file like this one

ATOM 1 I ION 1 0.000 0.000 0.000 1.00 1.00

or in general like this

ATOM 1 I ION 1 xcent ycent zcent 1.00 1.00

energy

This line is left to specify the calculation of the energy. The words for this line can be either

calceneryes

if the user requires the calculation of the energy, or any other word, for instances

calcenerno

if the user doesn't want to.

in_name_str

This line is left to specify the full path to the input file directory containing all the dx and pqr files as well as focusname.inm if focus boundary condition is required. For instances

C:\User\myname\matlabworkspace\Input_Files

for Windows users or

/Users/myname/matlabworkspace/Input_Files

for Linux users.

name_str

This line is left to specify the full path to the output file directory that will contain the resulting dx, fig and jpg files. For instances

C:\User\myname\matlabworkspace\systemname

for Windows users or

/Users/myname/matlabworkspace/systemname

for Linux users.

Input file examples

Just one inm. file is required if the user is not using focus boundary condition as you can see in the following example:

%|–example solvated-born.inm file using Dirichlet Boundary Condition–|

65 65 65

12 12 12

298.15

0.0 78.54

sdh

6

solvated-born-dielx.dx

```
solvated-born-diely.dx
solvated-born-dielz.dx
solvated-born-kappa.dx
born-ion.pqr
born-ion.pqr
calceneryes
C:\Users\Marce\Matlab_work_space\Input_Files
C:\Users\Marce\Matlab_work_space\born_model
%|————
```

Otherwise, two inm. files are required if the user use the focus boundary condition as you can see in the following example:

```
%|–example solvated-born.inm file using focus boundary condition–|
65 65 65
12 12 12
298.15
0.0 78.54
focusname.inm
6
solvated-born-dielx.dx
target-solvated-born-diely.dx
target-solvated-born-dielz.dx
target-solvated-born-kappa.dx
born-ion.pqr
born-ion.pqr
calcenerno
C:\Users\Marce\Matlab_work_space\Input_Files
C:\Users\Marce\Matlab_work_space\target_born_model
%|——————-
%|–example focusname.inm file for the calculation of the elect pot in the coarse grained calculation
65 65 65
50 50 50
```

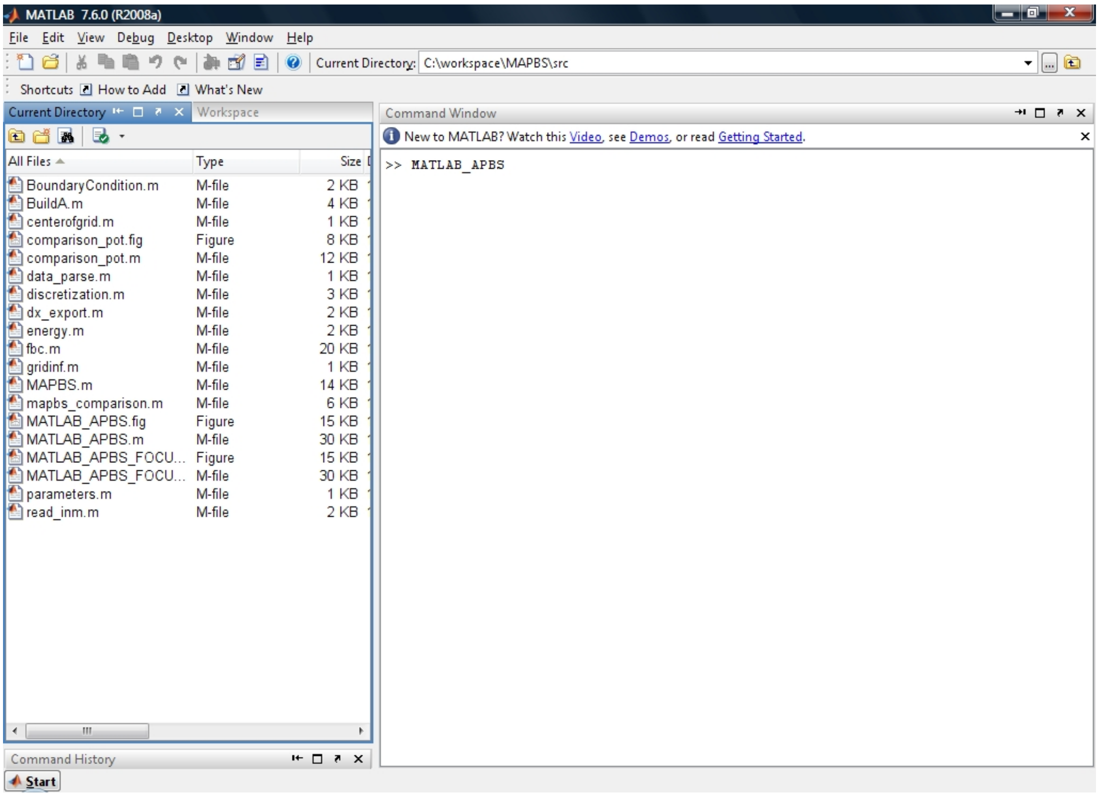
```
298.15
0.0 78.54
sdh
6
coarse-born-dielx.dx
coarse-born-diely.dx
coarse-born-dielz.dx
coarse-born-kappa.dx
born-ion.pqr
complex.pqr
calceneryes
C:\Users\Marce\Matlab_work_space\Input_Files
C:\Users\Marce\Matlab_work_space\coarse_born_model
%|————————|
```

GUI applications

Using MATLAB_APBS (this explanation is based on DOS platform but it can used on others)

We assume all the source files are located in the same directory, including MATLAB_APBS.m, MATLAB_APBS.fig, MAPBS.m. In our example, it is C:\workspace\MATLAB_APBS\src. We also assume you saved all the input files in the same directory. In our example, it is C:\workspace\MATLAB_APBS\Input_Files.

There are several ways to run this GUI. The simplest one consists in making you current directory the one in which the source files are located by using the search bar on the top of the Matlab window. Then just type >> MATLAB_APBS and press Enter as it is shown below

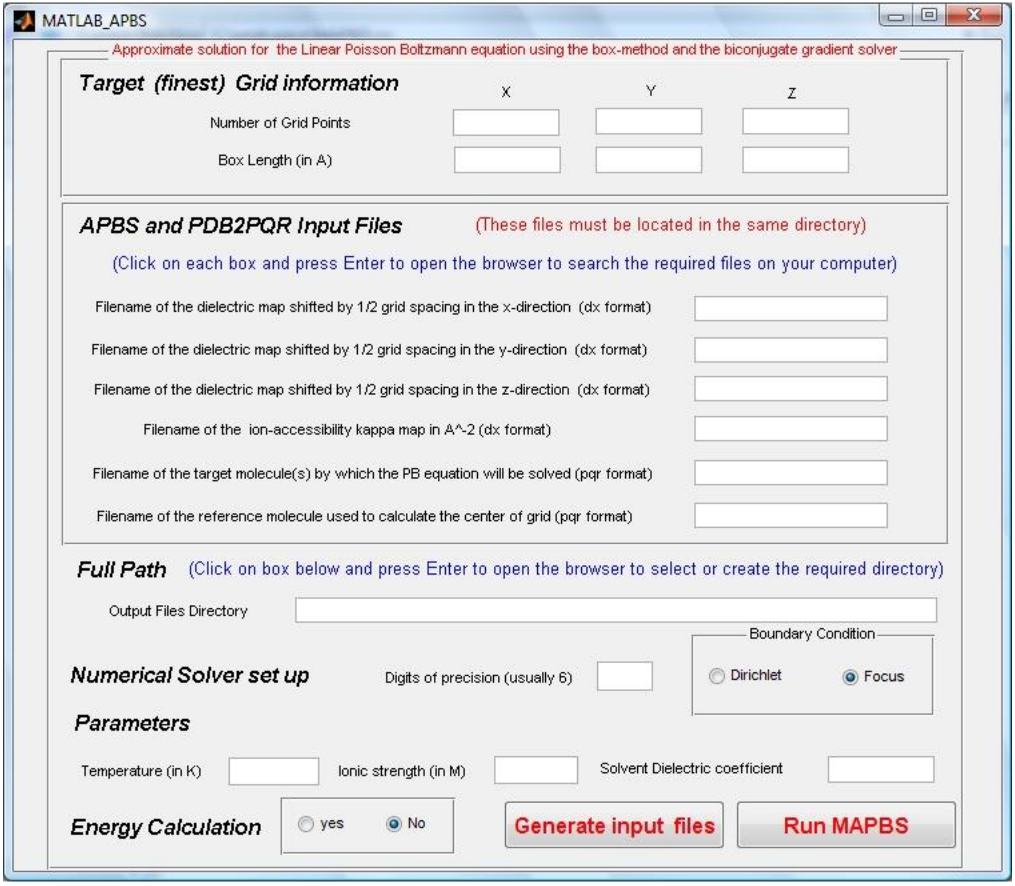


Current Directory: C:\workspace\MAPBS\src

All Files	Type	Size
BoundaryCondition.m	M-file	2 KB
BuildA.m	M-file	4 KB
centerofgrid.m	M-file	1 KB
comparison_pot.fig	Figure	8 KB
comparison_pot.m	M-file	12 KB
data_parse.m	M-file	1 KB
discretization.m	M-file	3 KB
dx_export.m	M-file	2 KB
energy.m	M-file	2 KB
fbc.m	M-file	20 KB
gridinf.m	M-file	1 KB
MAPBS.m	M-file	14 KB
mapbs_comparison.m	M-file	6 KB
MATLAB_APBS.fig	Figure	15 KB
MATLAB_APBS.m	M-file	30 KB
MATLAB_APBS_FOCU...	Figure	15 KB
MATLAB_APBS_FOCU...	M-file	30 KB
parameters.m	M-file	1 KB
read_inm.m	M-file	2 KB

```
>> MATLAB_APBS
```

Then, the graphical interface will open in a new window as follows



Approximate solution for the Linear Poisson Boltzmann equation using the box-method and the biconjugate gradient solver

Target (finest) Grid information

Number of Grid Points

Box Length (in A)

APBS and PDB2PQR Input Files (These files must be located in the same directory)

(Click on each box and press Enter to open the browser to search the required files on your computer)

Filename of the dielectric map shifted by 1/2 grid spacing in the x-direction (dx format)

Filename of the dielectric map shifted by 1/2 grid spacing in the y-direction (dx format)

Filename of the dielectric map shifted by 1/2 grid spacing in the z-direction (dx format)

Filename of the ion-accessibility kappa map in A^-2 (dx format)

Filename of the target molecule(s) by which the PB equation will be solved (pqr format)

Filename of the reference molecule used to calculate the center of grid (pqr format)

Full Path (Click on box below and press Enter to open the browser to select or create the required directory)

Output Files Directory

Numerical Solver set up

Digits of precision (usually 6)

Boundary Condition: ○ Dirichlet ● Focus

Parameters

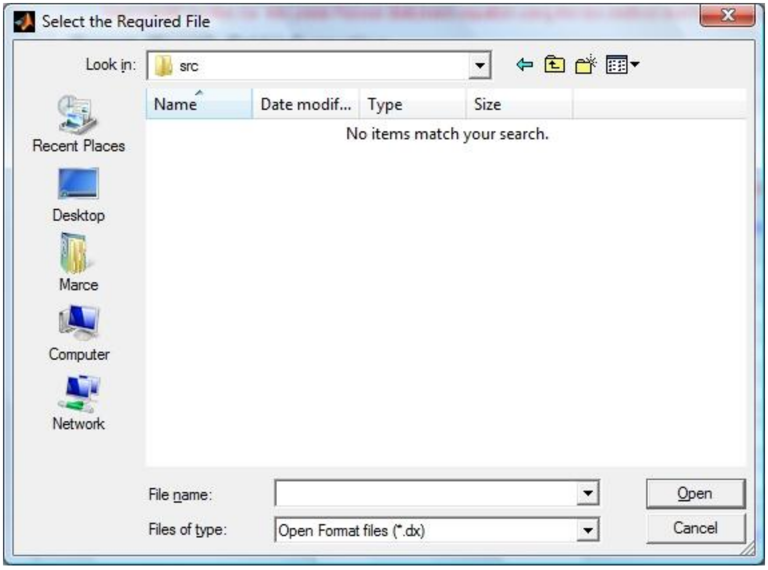
Temperature (in K)

Ionic strength (in M)

Solvent Dielectric coefficient

Energy Calculation: ○ yes ● No

By reading the instructions on this window, you are able to fill out the grid information typing the number of grid and the box length. Then, you may go to the next panel and click on the first box and press Enter as required. The following window will appear



Select the Required File

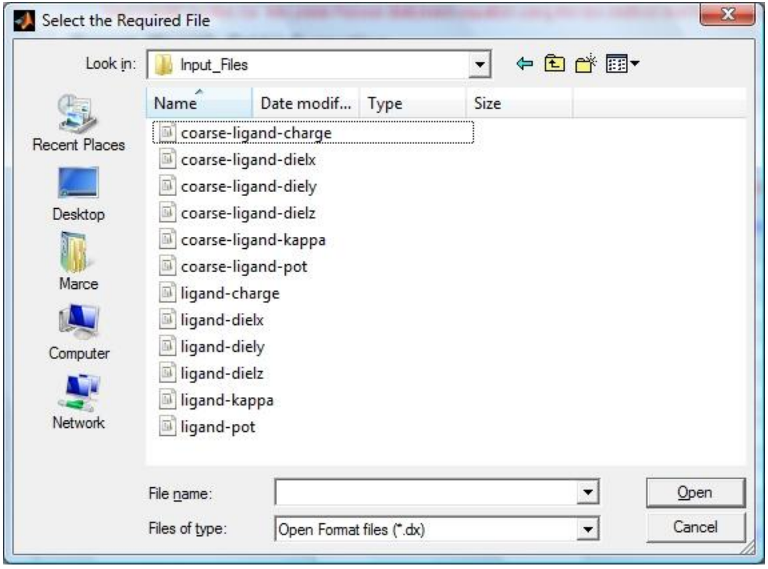
Look in: src

No items match your search.

File name:

Files of type: Open Format files (*.dx)

By default, the browser will open in the current directory. You have to change it to the folder in which you saved your input files. In our case it is



Select the Required File

Look in: Input_Files

coarse-ligand-charge
coarse-ligand-dielx
coarse-ligand-diely
coarse-ligand-dielz
coarse-ligand-kappa
coarse-ligand-pot
ligand-charge
ligand-dielx
ligand-diely
ligand-dielz
ligand-kappa
ligand-pot

File name:

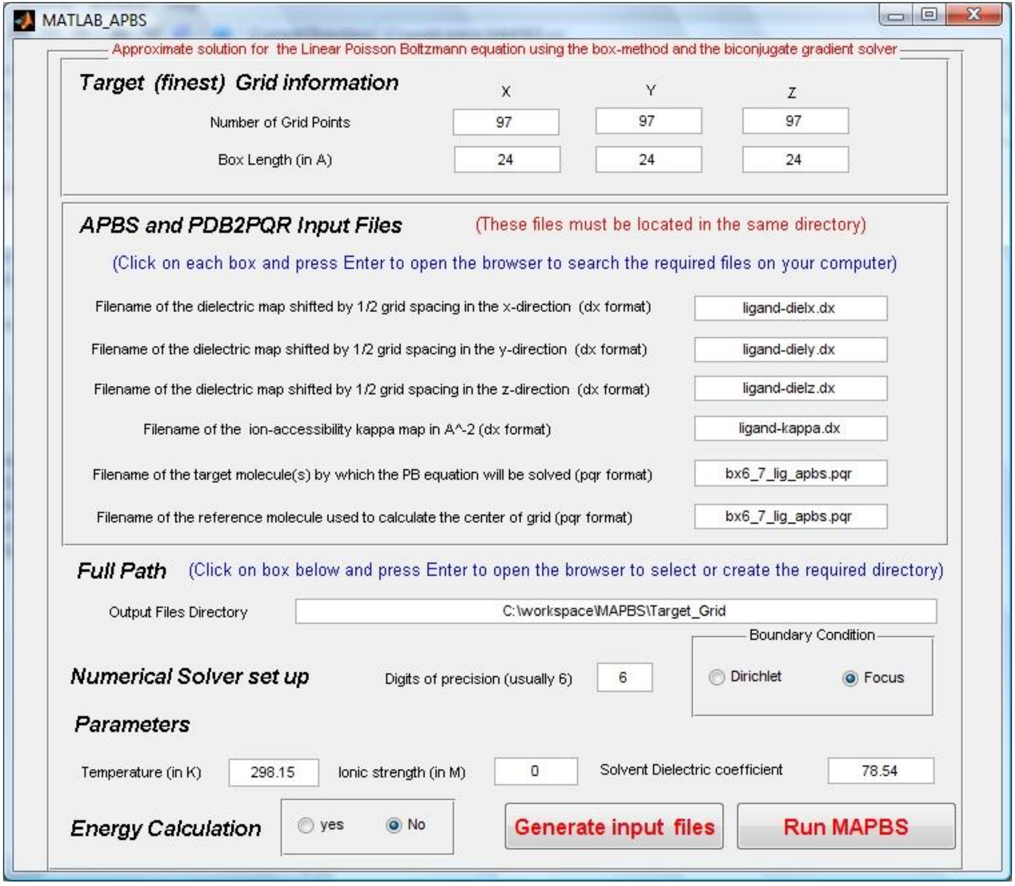
Files of type: Open Format files (*.dx)

To help to find the proper files there is a filter which only shows files having the right extension. Just select the required file and the press the “Open” button.

Repeat this procedure for each of the other boxes in the APBS and PDB2PQR panel. Because we assume all of the input files are located in the same directory, this GUI will subsequently open the browser in same input file directory to facilitate and make faster the user finds the required files.

Next, you have either to select or create the outputfile directory. This is similar to the previous procedure.

Finally, you may fill out the remaining information, namely the numerical solver set up and thermodynamic parameters to yield something like this



MATLAB_APBS

Approximate solution for the Linear Poisson Boltzmann equation using the box-method and the biconjugate gradient solver

Target (finest) Grid information

	X	Y	Z
Number of Grid Points	97	97	97
Box Length (in A)	24	24	24

APBS and PDB2PQR Input Files (These files must be located in the same directory)

(Click on each box and press Enter to open the browser to search the required files on your computer)

Filename of the dielectric map shifted by 1/2 grid spacing in the x-direction (dx format)	ligand-dielx.dx
Filename of the dielectric map shifted by 1/2 grid spacing in the y-direction (dx format)	ligand-diely.dx
Filename of the dielectric map shifted by 1/2 grid spacing in the z-direction (dx format)	ligand-dielz.dx
Filename of the ion-accessibility kappa map in A^-2 (dx format)	ligand-kappa.dx
Filename of the target molecule(s) by which the PB equation will be solved (pqr format)	bx6_7_lig_apbs.pqr
Filename of the reference molecule used to calculate the center of grid (pqr format)	bx6_7_lig_apbs.pqr

Full Path (Click on box below and press Enter to open the browser to select or create the required directory)

Output Files Directory C:\workspace\MAPBS\Target_Grid

Numerical Solver set up

Digits of precision (usually 6) 6

Boundary Condition ○ Dirichlet ● Focus

Parameters

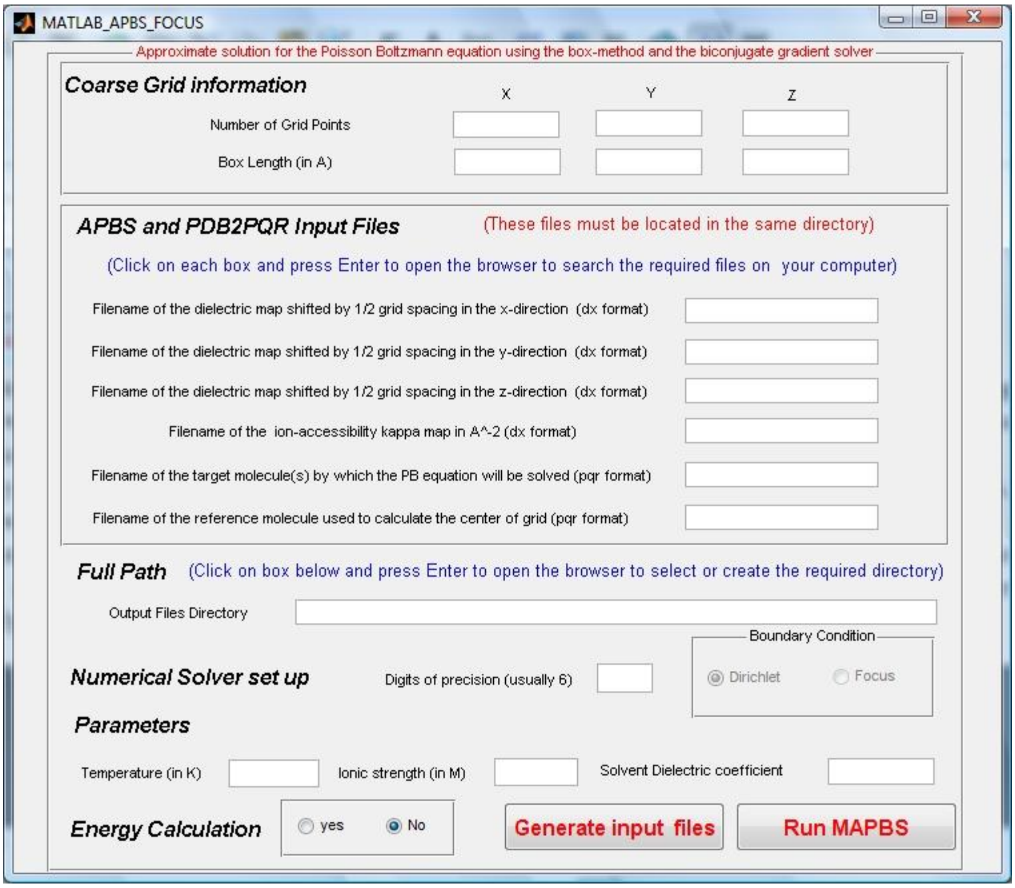
Temperature (in K) 298.15 Ionic strength (in M) 0 Solvent Dielectric coefficient 78.54

Energy Calculation ○ yes ● No

Now the user is ready to generate the Matlab input file (inputfile.inm) by clicking the proper red push button. A warning message will appear in a new window asking the user to check all the provided information. After the Matlab input file is generated, there are two options.

If the Dirichlet boundary condition is required, the user is able to run MAPBS by clicking on the proper red push button. Then, the current GUI window will close and the MAPBS will run on the Matlab command window.

Otherwise, if Focus boundary condition is required as shown in our example, the user will have to subsequently provide the corresponding information about the coarse grid calculation in a new window generating a second input file (focusname.inm). Specifically, the following window will open



MATLAB_APBS_FOCUS

Approximate solution for the Poisson Boltzmann equation using the box-method and the biconjugate gradient solver

Coarse Grid information

	X	Y	Z
Number of Grid Points			
Box Length (in A)			

APBS and PDB2PQR Input Files (These files must be located in the same directory)

(Click on each box and press Enter to open the browser to search the required files on your computer)

Filename of the dielectric map shifted by 1/2 grid spacing in the x-direction (dx format)

Filename of the dielectric map shifted by 1/2 grid spacing in the y-direction (dx format)

Filename of the dielectric map shifted by 1/2 grid spacing in the z-direction (dx format)

Filename of the ion-accessibility kappa map in A^-2 (dx format)

Filename of the target molecule(s) by which the PB equation will be solved (pqr format)

Filename of the reference molecule used to calculate the center of grid (pqr format)

Full Path (Click on box below and press Enter to open the browser to select or create the required directory)

Output Files Directory

Numerical Solver set up

Digits of precision (usually 6)

Boundary Condition: ● Dirichlet ○ Focus

Parameters

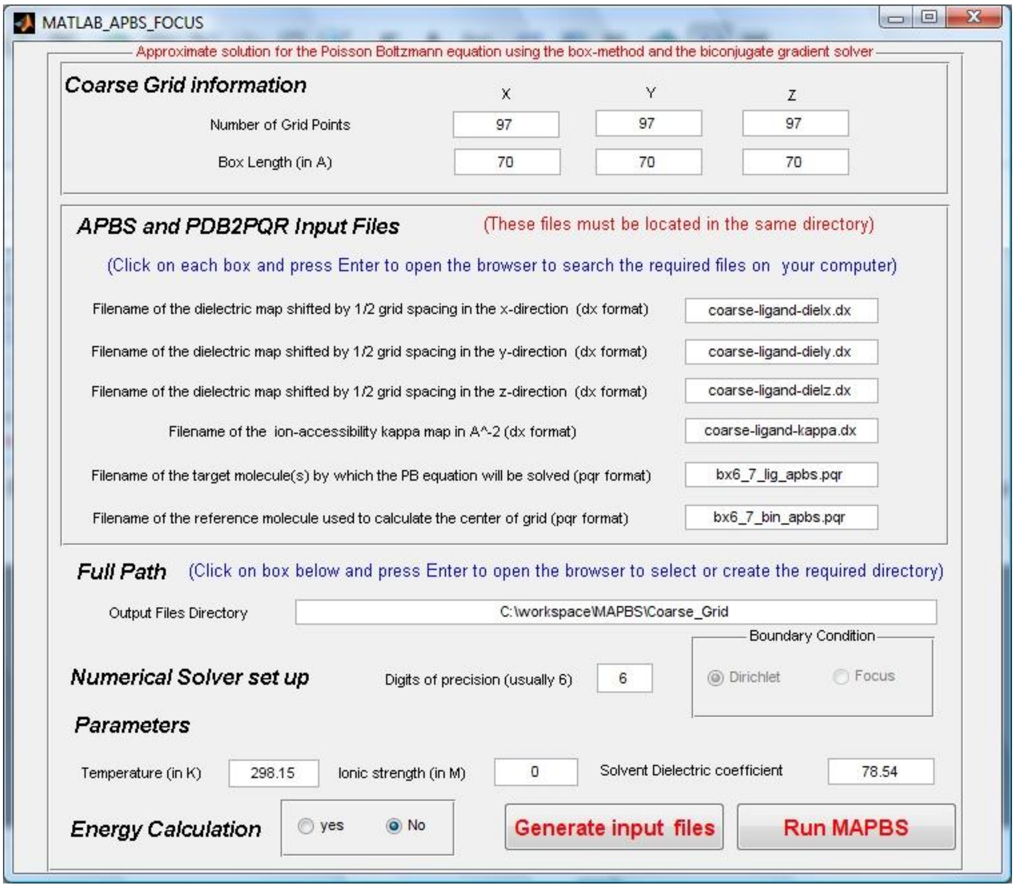
Temperature (in K)

Ionic strength (in M)

Solvent Dielectric coefficient

Energy Calculation ○ yes ● No

which is very similar to the previous one. You have to fill out all the information as explained for the target grid to yield something like this



Approximate solution for the Poisson Boltzmann equation using the box-method and the biconjugate gradient solver

Coarse Grid information

	X	Y	Z
Number of Grid Points	97	97	97
Box Length (in A)	70	70	70

APBS and PDB2PQR Input Files (These files must be located in the same directory)

(Click on each box and press Enter to open the browser to search the required files on your computer)

Filename of the dielectric map shifted by 1/2 grid spacing in the x-direction (dx format): coarse-ligand-dielx.dx

Filename of the dielectric map shifted by 1/2 grid spacing in the y-direction (dx format): coarse-ligand-diely.dx

Filename of the dielectric map shifted by 1/2 grid spacing in the z-direction (dx format): coarse-ligand-dielz.dx

Filename of the ion-accessibility kappa map in A^-2 (dx format): coarse-ligand-kappa.dx

Filename of the target molecule(s) by which the PB equation will be solved (pqr format): bx6_7_lig_apbs.pqr

Filename of the reference molecule used to calculate the center of grid (pqr format): bx6_7_bin_apbs.pqr

Full Path (Click on box below and press Enter to open the browser to select or create the required directory)

Output Files Directory: C:\workspace\MAPBS\Coarse_Grid

Numerical Solver set up

Digits of precision (usually 6): 6

Boundary Condition: ● Dirichlet ○ Focus

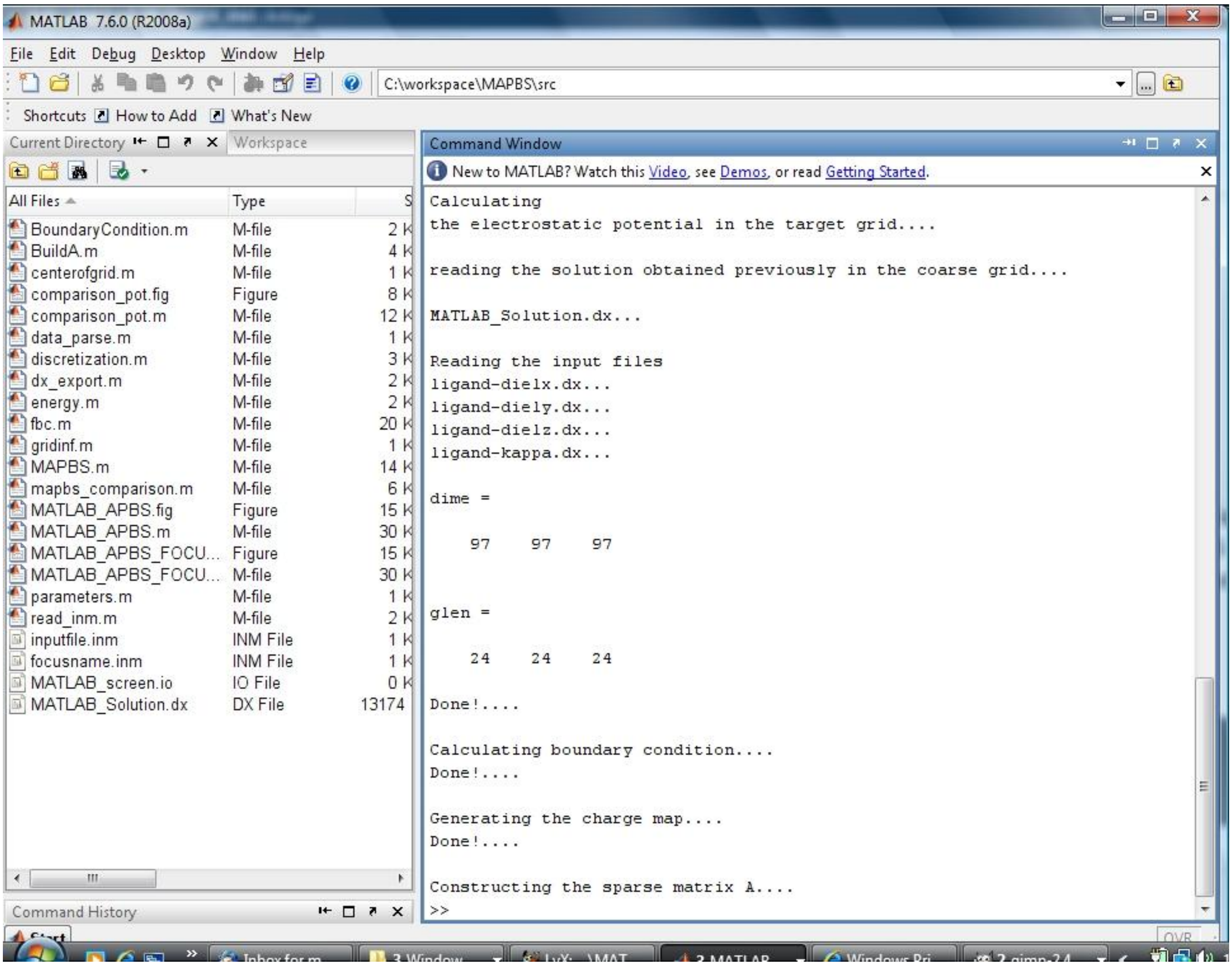
Parameters

Temperature (in K): 298.15 Ionic strength (in M): 0 Solvent Dielectric coefficient: 78.54

Energy Calculation ○ yes ● No

Now the user is ready to generate the second input file (focusname.inm) by clicking the proper red push button. Again, a warning message will alert the user to check all the provided information. Finally, the user is able to run MAPBS by clicking on the proper red push button. Then, the current GUI window will close and the MAPBS will run on the Matlab command window.

For both cases, the resulting screen on the Matlab command window running the MAPBS would look like this


MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
C:\workspace\MAPBS\src
Shortcuts How to Add What's New
Current Directory
Workspace
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.
Command History

All Files	Type	S
BoundaryCondition.m	M-file	2 K
BuildA.m	M-file	4 K
centerofgrid.m	M-file	1 K
comparison_pot.fig	Figure	8 K
comparison_pot.m	M-file	12 K
data_parse.m	M-file	1 K
discretization.m	M-file	3 K
dx_export.m	M-file	2 K
energy.m	M-file	2 K
fbc.m	M-file	20 K
gridinf.m	M-file	1 K
MAPBS.m	M-file	14 K
mapbs_comparison.m	M-file	6 K
MATLAB_APBS.fig	Figure	15 K
MATLAB_APBS.m	M-file	30 K
MATLAB_APBS_FOCU...	Figure	15 K
MATLAB_APBS_FOCU...	M-file	30 K
parameters.m	M-file	1 K
read_inm.m	M-file	2 K
inputfile.inm	INM File	1 K
focusname.inm	INM File	1 K
MATLAB_screen.io	IO File	0 K
MATLAB_Solution.dx	DX File	13174

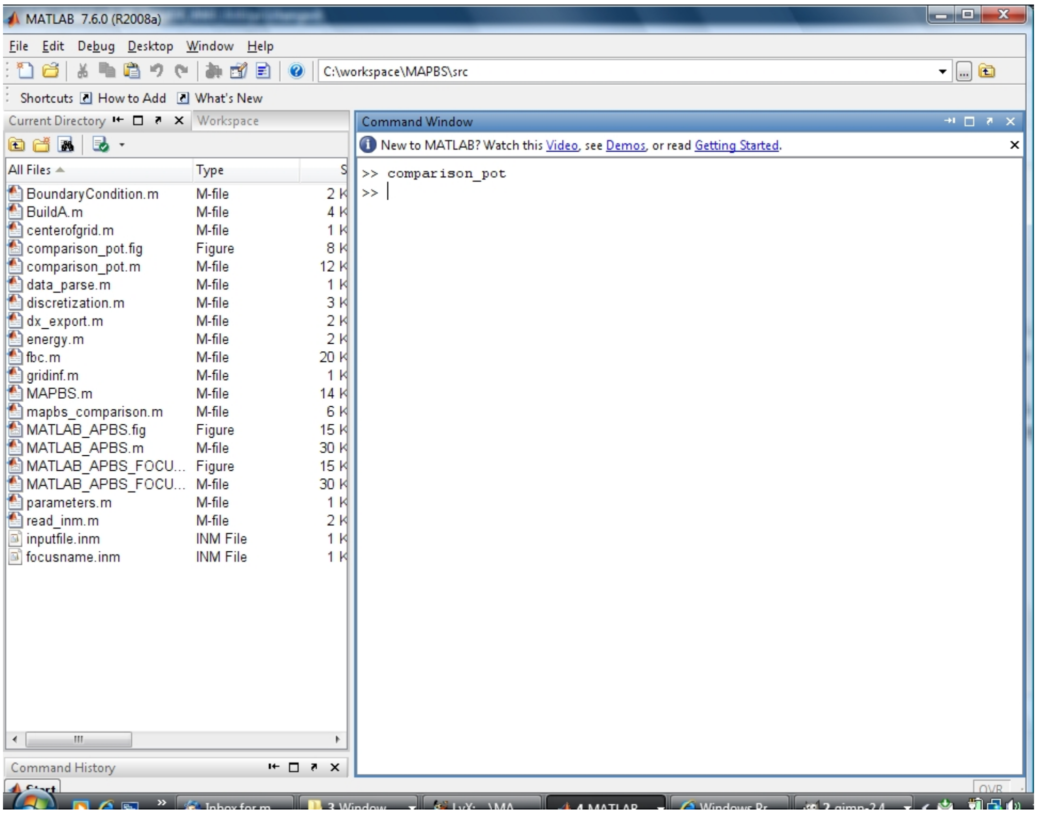
```
Calculating
the electrostatic potential in the target grid....

reading the solution obtained previously in the coarse grid....

MATLAB_Solution.dx...

Reading the input files
ligand-dielx.dx...
ligand-diely.dx...
ligand-dielz.dx...
ligand-kappa.dx...

dime =

    97    97    97


glen =

    24    24    24

Done!....

Calculating boundary condition....
Done!....

Generating the charge map....
Done!....

Constructing the sparse matrix A....
>>
```

In our example, the resulting files will be saved in the “outpufiles” subfolder located in both the “Target_Grid” and “Coarse_Grid” directories. In the former, you will also find the MATLAB_screen.io file which contains all the information printed on the screen in the Matlab command window during the execution of MAPBS.

Using comparison_pot

Let compare, for instances, the previously obtained coarse grain Matlab solution with the one coming from APBS.

To use the comparison_pot GUI,and following the previous instructions, we have to type >> comparison_pot in the Matlab command window and press Enter as it is shown below

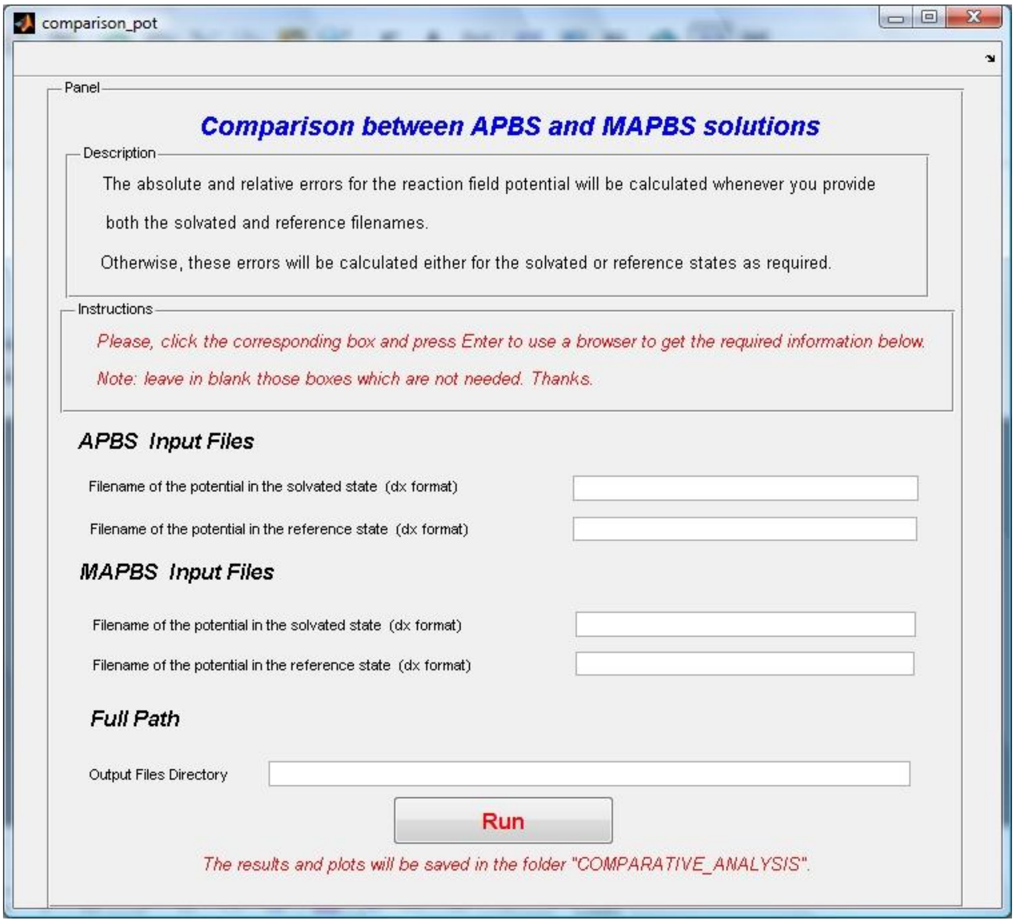


All Files	Type	S
BoundaryCondition.m	M-file	2 K
BuildA.m	M-file	4 K
centerofgrid.m	M-file	1 K
comparison_pot.fig	Figure	8 K
comparison_pot.m	M-file	12 K
data_parse.m	M-file	1 K
discretization.m	M-file	3 K
dx_export.m	M-file	2 K
energy.m	M-file	2 K
fbc.m	M-file	20 K
gridinf.m	M-file	1 K
MAPBS.m	M-file	14 K
mapbs_comparison.m	M-file	6 K
MATLAB_APBS.fig	Figure	15 K
MATLAB_APBS.m	M-file	30 K
MATLAB_APBS_FOCU...	Figure	15 K
MATLAB_APBS_FOCU...	M-file	30 K
parameters.m	M-file	1 K
read_inm.m	M-file	2 K
inputfile.inm	INM File	1 K
focusname.inm	INM File	1 K

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
>> comparison_pot
>> |
```

then, the graphical interface will open in a new window as follows



Comparison between APBS and MAPBS solutions

Description

The absolute and relative errors for the reaction field potential will be calculated whenever you provide both the solvated and reference filenames.

Otherwise, these errors will be calculated either for the solvated or reference states as required.

Instructions

Please, click the corresponding box and press Enter to use a browser to get the required information below.

Note: leave in blank those boxes which are not needed. Thanks.

APBS Input Files

Filename of the potential in the solvated state (dx format)

Filename of the potential in the reference state (dx format)

MAPBS Input Files

Filename of the potential in the solvated state (dx format)

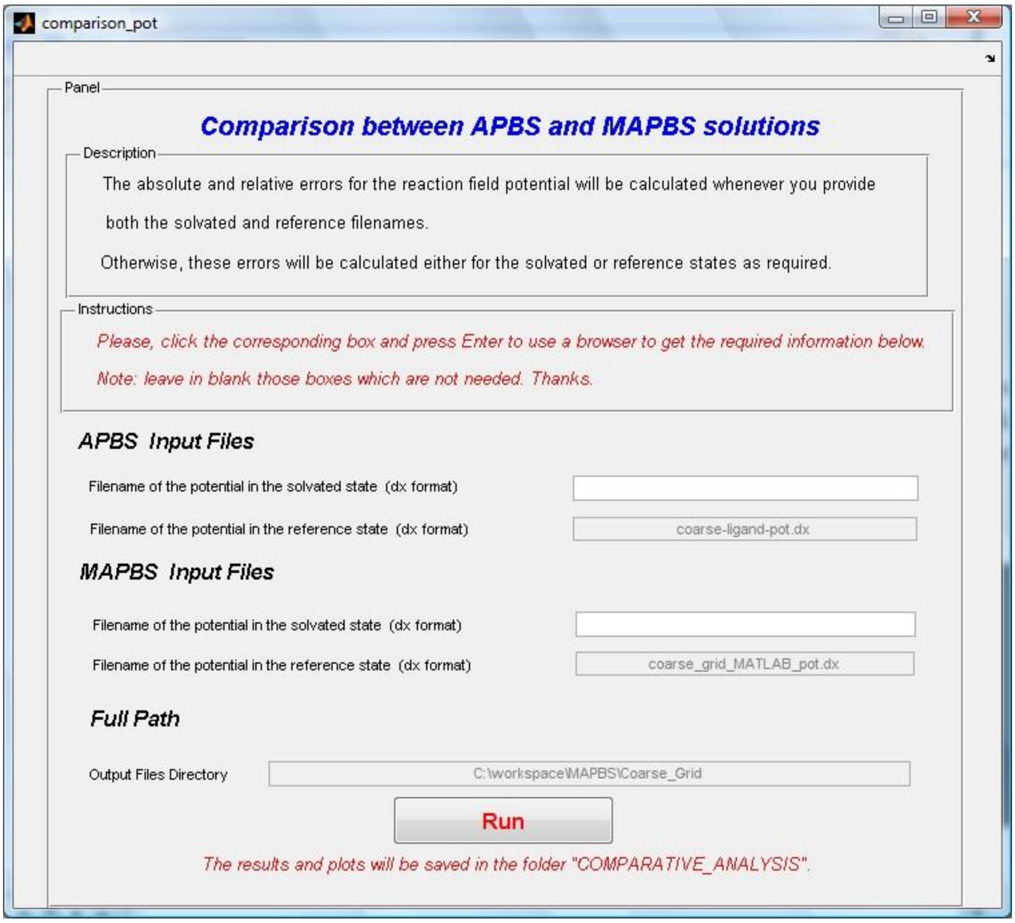
Filename of the potential in the reference state (dx format)

Full Path

Output Files Directory

The results and plots will be saved in the folder "COMPARATIVE_ANALYSIS".

It looks very similar to the MATLAB_APBS GUI. Following the instructions on this window, you may fill out each boxes as explained earlier. Because we will compare solvated states, we only have to fill out the first and third boxes use the browser as explained earlier, leaving in blank the other two corresponding to the reference states. In our example, the APBS and the MATLAB input files are located in C:\workspace\MATLAB_APBS\Input_Files and C:\workspace\MATLAB_APBS\Coarse_Grain\outputfiles directories, respectively. Finally, we have to either select or create the outputfiles directory to yield



comparison_pot

Panel

Comparison between APBS and MAPBS solutions

Description

The absolute and relative errors for the reaction field potential will be calculated whenever you provide both the solvated and reference filenames.

Otherwise, these errors will be calculated either for the solvated or reference states as required.

Instructions

Please, click the corresponding box and press Enter to use a browser to get the required information below.

Note: leave in blank those boxes which are not needed. Thanks.

APBS Input Files

Filename of the potential in the solvated state (dx format)

Filename of the potential in the reference state (dx format) coarse-ligand-pot.dx

MAPBS Input Files

Filename of the potential in the solvated state (dx format)

Filename of the potential in the reference state (dx format) coarse_grid_MATLAB_pot.dx

Full Path

Output Files Directory C:\workspace\MAPBS\Coarse_Grid

The results and plots will be saved in the folder "COMPARATIVE_ANALYSIS".

Now the user is ready to press the “run” red push button. The current GUI will close and the ’mapbs_comparison.m’ Matlab code will run on the Matlab command window. The resulting files will be saved in the subfolder “COMPARATIVE_ANALYSIS” within the required outputfiles directory.