MC_Diffusion manual

Peter M. Berger

GFZ German Research Centre for Geosciences

Telegrafenberg, 14473, Germany

peter.berger@gfz-potsdam.de

Software capabilities

The software is built to calculate multi-component diffusion of ions over a user defined domain based on the equations of previous workers which are summarized in Oelkers (1996). These equations account for charge balance and the derivative of the species activity coefficient with respect to changes in concentration. It uses a simple, fully explicit finite difference approximation to calculate diffusion profiles and has been compared to a previous benchmark used by other software packages (Rasouli et al., 2015).

The software is capable of simulating three different boundary conditions at the left boundary: closed, closed with a constant CO_2 concentration, and constant. The right boundary is always closed. One can also apply a temperature gradient over the domain, though the software has not been tested against a Soret effect benchmark.

Software compilation

The software is written in C++ and while it was originally developed using Visual Studio, it should compile with any suitable compiler. The software does use ChemPlugin to do the activity coefficient calculations. This is a third-party software library that must be purchased separately. The use of ChemPlugin by this software is in no way an endorsement, rather it is a question of personal preference. Regardless, the code does require ChemPlugin.h to compile and the associated library to link. No make file is provided.

Software alteration

The user is free to alter this software as they see fit. If they do not like using ChemPlugin, several other geochemical software libraries, such as PHREEQC, should be easy enough to swap out. It is also possible for the software to be extended to include mineral reactions and different boundary conditions. If you do this, please document your code as well any changes in the input file format.

Software use

The software is meant to be used with the user specifying an input file at the command line:

```
MultDiff myinput.dat
```

The input file should be a simple text file and can be named whatever the user likes. The software outputs three files:

output.dat – The concentration of each species over the first 10 time steps, then every 1,000 time steps and the last time step.

outputb.dat – The concentration of each basis species over the first 10 time steps, then every 1,000 time steps and the last time step.

outdelta.dat – The change in concentration over the first time step (useful when debugging).

Input file format

The proceeding instructions need to be given in the specified order. Keywords that are italicized are placeholders that merely serve to remind the user what needs to be put on that particular line. Blank spaces or comments are not allowed.

Nspecies number of species specified in the file

Nnode number of nodes in the simulation

Calc_act either a 1 to include the effects of activity coefficient derivatives in the calculations or a 0 to omit them

Spacing the spacing between the nodes

Time the length of time the software should simulate

Tconst either a 1 to have temperature to be the same throughout the domain or a 0 to apply a gradient.

Temp the temperature in °C. If Tconst was 0, this needs to be followed by the temperature difference between adjacent nodes.

Tmax the maximum allowable timestep.

Fixedend either a 1 to have a constant concentration boundary condition or a 0 to have a closed boundary condition on the left-hand side.

FixedCO2 if Fixedend is 0 and the left-hand side is closed, then this variable needs to be defined on the next line. It should be 1 if the left-hand CO_2 boundary condition should be constant and 0 if it should be closed.

Tor the tortuosity which will the diffusion coefficients will be multiplied by.

After the above input, there is a line for each of the species in the simulation with four parts:

Name – this should correspond to the name of the species in whatever thermodynamic database is being used

Ionic charge

The tracer conductivity of the ion at 25 °C

An integer encoding the type of ion

Monovalent cations	0
Divalent cations	1
Trivalent cations	2
Halogens	3
Monovalent oxy-anions	4
Divalent oxy-anions	5
Hydrogen	6
Hydroxyl	7
CO_2	8

After a line for each species comes the line:

Load_prev either a 1 to start by loading in an old output file as a starting point or a 0 to start from scratch.

After this comes the key line ”scope inlet”. All lines following this until the next key line are input into ChemPlugin to calculate the initial species distribution and activity of the node at the left most boundary.

The next key line is “scope domain”. All lines following this until the next key line are input into ChemPlugin to calculate the initial species distribution and activity of all nodes except for the far left boundary node.

The last key line is “end”, which closes the input file.

An example input file follows:

```
nspecies 5
nnode 100
calc_act 0
spacing 0.0001
time 1000
Tconst 1
temp 25
Tmax 0.01
fixedend 1
tor 1
H+ 1 349.65 6
NO3- -1 71.42 4
Na+ 1 50.08 1
Cl- -1 76.31 3
OH- -1 198.0 7
load_prev 0
scope inlet
pH 6.001
Na+ = -3.9999999 log mol
Cl- = -3.9999999 log mol
NO3- = 0.001 mmol
scope domain
pH 4.007
Na+ = -3.9999999 log mol
Cl- = -3.9999999 log mol
NO3- = -3.9999999 log mol
end
```

Software license

Cite as:

Berger, Peter (2024): MC_Diffusion - a C++ program to calculate multi-component diffusion. V. 1.0. GFZ Data Services. https://doi.org/10.5880/GFZ.3.1.2024.005

References

Oelkers, E.H., 1996, Chapter 3. PHYSICAL AND CHEMICAL PROPERTIES OF ROCKS AND FLUIDS FOR CHEMICAL MASS TRANSPORT CALCULATIONS, *in* Lichtner, P.C., Steefel, C.I., and Oelkers, E.H. eds., Reactive Transport in Porous Media, Berlin, Boston, De Gruyter, p. 131–192, doi:10.1515/9781501509797-006.

Rasouli, P., Steefel, C.I., Mayer, K.U., and Rolle, M., 2015, Benchmarks for multicomponent diffusion and electrochemical migration: Computational Geosciences, v. 19, p. 523–533, doi:10.1007/s10596-015-9481-z.