FEHM V3.3.1 Release Summary

FEHM Version 3.3.1 December 4 2015 to September 21 2017
LA-CC-2012-083 Export control on source code

This Version 3.3.1 was used to start a new repository at https://github.com/lanl/FEHM in preparation for Open Source distribution. This was moved from an internal server hosting a mercurial repository on https//fehm.lanl.gov that will be closed.

The major changes in the FEHM application from V3.3.0 (2015) to this release V3.3.1 are: further developments for macros SALT and RPL/RLPM. Improvements added related to TRXN, CDEN and CO2 coupling with tracers. GDKM input was modified to have a general formulation equivalent to DPDP and direction-specific fracture formulation. New humidity related boundary conditions were added to BOUN including HUF (flowing humidity), HU (fixed humidity), TH (humidity temperature), and PH (humidity pressure). Minor code modifications were made to improve the performance of FEHM simulations.

The User Manual FEHM V3.3.1 is updated to describe code enhancements and added capabilities. It now includes a short description on mesh quality and the possible impacts on FEHM solutions.

Known code bugs fixed:

- Fix the pressure BC issue (remember to use PA in BOUN)
- Fixed phase change guess with low water vapor pressure.
- Minor fixes to improve read/write of various files and the handling of input parameters.
- Change in startup.f resolved a simulation result discrepancy between AMANZI/analytical solution and FEHM. This fixes a code change between Nov 2014 to Aug 2016 and preserves the new codes added for salt related problems.
- All V3. * versions of FEHM will report negative coupling coefficients if they exist. The older V2 releases do not report on coupling coefficients. This difference in reporting is not a bug in either release, just a difference in the information that is written.

Known issues:

- Macro DVA has a forced explicit update to fix possible derivative problems, needs further checking.
- The vector files written for contouring are the control volume interface volumetric flux values mapped back on to the mesh nodes. The algorithm for this mapping will have errors most visible for non-orthogonal meshes.
- This mac test suite was created on Mavericks and some comparison binary files are no longer compatible with OS Yosemite or newer. For these tests, results need to be checked and compared individually. FEHM V3.1.1 results were checked and all tests passed.

FEHM V3.3.1 Executables:

FEHM_V3.3.1lnUbuntu16.28Feb18 (Linux Ubuntu 16) compiled with gfortran 5.4

FEHM_V3.3.1w64.28Feb18.exe (Windows) compiled with Intel Parallel Studio XE 2013

FEHM_V3.3.1macSierra.05Mar18 (Mac OSX Sierra 10.12.16) compiled with gfortran 7.1

Code changes from 12/04/2015 to 9/21/2017:

January, 2016

Modify co2h2o_combine.f: changed to avoid stack overflow crashes for large problems

Salt development:

airctr.f- Added coding to check for bad input temperatures for isothermal calculations.

allocmem.f - added allocation on new accumulation term storage (changes during phase change, eg deni_ch). algorithm not active yet

bnswer.f- added coding to support the normalization restart in gensl4.f

co2ctr.f - added coding for new ngas boundary conditions and merged with FEHM V3.3.0.

comai.f- added global integer variable that save the total number of gridblocks in each phase and the change for a timestep. added the variable to save the timestep restart criteria (mlz_save)

comci.f - added allocatable arrays for mass and energy storage terms (coordinated with those described in allocmem.f

comdi.f - added allocatable arrays for variable humidity air inflow in the boun macro (humid, phumid, thumid)

cplxcalc.f - dos2unix change from dos format to unix format

diskread.f- minor modification to improve code performance

dvacalc.f-forced an explicit update for dva(), the derivatives were not working (in GAZ opinion). Have not seen it go unstable yet but need to check. Also need to look harder at derivatives.

fehmn.f- coding changed slightly to facilitate restarted timesteps when normalization fails (only for ngas presently)

flow_boun.f- added coding for humidified air inflow

flow_boundary_conditions.f - added coding for humidified air inflow, minor changes to improve wtsi seepage face calcs.

flow_humidity_bc.f- New routine to calculate the inlet flowrates and enthalpy for dry air and water vapor given total flowrate of air(or ngas), T, P, and humidity.

geneqc.f- minor modification to improve code performance

gensl4.f-added coding so if there is a normalization failure, the timestep is restarted with a smaller timestep size (like when the iterations are exceeded)

headctr.f - minor modification to improve code performance

h2o_properties.f - fixed a typo in comment line

icectrco2.f- Changed some on the initial guessed parameters during phase change calculations. Removed most of the temperature guesses and changes the guess for fl().

initdata2.f - nothing substantive (different print out for errors)

model_setup.f- Added input for variable input of humidified air.

outbnd.f- Zeroed the out of bounds error code mlz.

porosi.f- Changed coding so that porosity models 6 and 7 continue to run no chemistry model. (The tracer array has been allocated so that the code does not crash).

psatl.f- Added coding for vapor pressure lowering with salt and capillary pressure. (small corrections if this was in an earlier version)

rlp_cap.f -Added coding to correct error with VG relative permeability models when the saturation approachies zero. Now identical with older formulation.

saltctr.f- Minor additions to functionality.

scanin.f- Added code to scan input files for humidity boundary conditiona, modified to search input files for new boun keywords (hu-humidified air inflow).

setparams.f - initialed parameters relating to humidified air input

startup.f - Added call to airctr. for checking temperatures.

thermw.f-nothing substantive (spaces only)

ther_co2_h2o.f- added an explanation to the code

thrair.f - Added coding improved the consistency and performance for seepage face models with wtsi (simplified water table).

thrmwc.f-modifed coding for very low water vapor that is useful when dealing with very low water vapor pressure

user_ymp.f- Added some code for fitting solubility parameters and vapor pressure parameters.

vaporl.f-Appended routine that calculates vapor pressure lowering from salt. (GAZ fitted curve that replaces Sparrow's curve) varchk.f- fixed phase change guess with low water vapor pressure

varchk_part.f- Used for primitive parallel processing-some minor changes to keep this routine consistent with varchk.f (but still not up-to-date)

wrtout.f- Added printout for number of gridblocks in each phase state.

February, 2016

Changes for Olivella functions:

vcon.f - Corrected existing model to implement Olivella thermal conductivity function.

> (still some question as to Temp units K or C (Olivella,2011 seems to be C)
>
> Can't find a reference temperature.

comai.f - added global parameter 'iaprf' to manage Olivella intrinsic permeability reduction function

comdi.f - added global parameters k0f, bkf, por0f for the Olivella intrinsic permeability reduction function

data.f - zeroed parameter 'iaprf'

porosi.f - Added coding to enable call to Olivella permeability function (including perm initialization)

saltctr.f - Added input coding for Olivella thermal conductivity (and warning in model 4 or 5 are not used with salt). Also added input for Olivella intrinsic perm function. Appended small subroutine perm_olivella to the end of saltctr.f

startup.f- changed comment only near call to porosi()

flow_boundary_conditions.f - added code so that a specified saturation entered in the boun macro works with Richards' equation

April, 2016

Following 4 routines are modified so that the vapor flowrates for Richard's Eq. outputs "0." for all vapor flowrates in the restart file:

startup.f - turned off vapor flow rate flag for flxz keyword based calculations for Richard's Eq. (keyword rich)

diskread.f- allowed reading of vapor flow rate information with Richards Eq. (fin file maybe from a 2phase problem) diskwrite.f- outputted 0. for vapor flow rate for Richards Eq (fin file may be used for a 2phase problem)

diskwrite_new.f - same change as to diskwrite.f

Modify air_combine.f - to remove the stack overflow

Modify thrair.f - modified for the free drainage. Free drainage flowrate should always be positive (or outflow).

May, 2016

Salt development modification:

Found a small error when a gridblock that is all gas (S = 0) and it is re-wetting from a flow from another gridblock that is above or below it. In thermal problems, the liquid density is not defined for a s= 0 gridblock. This causes problems for the gravity term which is always equally weighted between gridblocks. Fixed by taking the defined liquid density from the inflowing gridblock and not averaging. Idea motivated by discussion with David Dempsey and John OSullivan (Auckland University).

comji.f - added grav_wgt(:) to change the weighting for the gravity part on the flow equation

geneq1.f - modified gravity weighting term as described above (heat and mass pure h20)

geneqc.f - modified gravity weighting term as described above (heat and mass air-water-heat)

varchk.f - modified some tolerances for phase changes because with corrections above the phase changes are smoother

flow_boundary_conditions.f - corrected error for accessing memory in an array that was larger or smaller than the allocated amount.

June, 2016

Modify psatl.f - modified vapor pressure to use Sparrow for salt repository

Modify vaporl.f - added coding to include sparrows 2003 equation

Fix bug for hist macro:

inhist.f: change line 547 from "ishise = ishis + 150" to “ishisef = ishis + 155"

July, 2016

Modify ther_co2_h2o.f: added energy output for constant water production rate boundary condition for co2 storage problems.

Modify following five routines: Added temperature boundary condition for CO2 injection using boun macro.

model_setup.f

comdi.f

flow_boun.f

flow_boundary_condition.f

scanin.f

August, 2016

New humidity BC development: most of the files were changed to support developments in humidity initiaization, output and boundary conditions. Use 'humidity' for the relative humidity fraction.

model_setup.f - added boun sub keywords 'huf' (flowing humidity),'hu' (fixed humidity),

th(humidity temperature), ph (humidity pressure)

co2ctr.f - set relative humidity fraction at input temperature for gas phase, print out relative humidity fraction

csolve.f - modified to printout SIA iterations

flow_humidity_bc.f (new) - calculates the air and water mass fractions of an inflowing gas flow so that the inflow has a specified relative humidity.

scanin.f - added coding to identify new humidity related keywords in the boun macro.

thrmwc.f - added coding to implement new humidity related boundary conditions.

wrtcon.f - added coding to output more detail on SIA iterations

flow_boun.f - added coding for new humidity boundary conditions.

Changes for Olivella functions:

porosi.f - Added coding to enable call to Olivella permeability function (including perm initialization). Corrected the number of loop calls and improved cpu.

saltctr.f - Added conditional to Olivella perm function to enable calculation of a single gridblock to be consistent with loop on porosi.

September, 2016

Modify startup.f: replaced line 1319 "c   psini = ps" to be “if (isalt.eq.0) psini=ps”

This change resolved one simulation scenario results discrepancy between AMANZI/analytical solution and FEHM for code dated between Nov 13, 2014 to Aug 2, 2016; and meanwhile still preserve salt development for salt related problems.

Modified diskread.f: comment out line 645 and 1216 "psini = ps". This change allow the model to use restart fin file with or without porosity in all simulation scenarios.

Modify write_avs_node_mat.f: fixed bug when writing material output files during a gdkm run, so that gdkm contour output will work again with geo and material keywords; Another small change to get tecplot output working when using gdkm

Modify csolve.f: concentrations are output at flow time steps now

Modify start_macro.f: zone or zonn file can be used with zone or zonn macro in the input deck, input deck macro name will determine which macro is used

October-November, 2016

Development related to trxn, cden and co2 coupling with tracers:

allocmem.f - increased size of cpntprt array

cden_cor.f90 - added indexing variables so that multiple tracer species for density calculation can be specified

comchem.f - added improvements to reporting of tracer names

comco2.f - added variables required for coupling of CO2 and carbon tracer

concen.f - adjusted logic to allow for more cden options

diskread.f - overwrite initial dissolved carbon concentrations with tracer concentration read in from restart file if cfrac flag=2 is specified

initchem.f - added logic for coupling co2 with carbon tracer

read_rxn.f - fixed problem with tracer name reporting, added co2 coupling

rdcon.f - fixed cden species indexing

rdtr.f90 -

fixed bugs related to indexing species during read of trxn macro added logic related to co2/carbon tracer coupling

fixed bug related to selecting which species to print, including the 'all' option

improved error reporting during assign keyword read

scanin.f - improved species name reporting

ss_trans.f - added logic for coupling co2 with carbon tracer

trxnvars.f90 - some of this code has been moved to lookupvars

wrtcon.f - improved species name reporting

ther_co2_h2o.f - modified following lines so code work with rlpm new development,

replace

```
        dqh(mi) = qdis*denwp
        deqh(mi) = qdis*denwta
```

with

```
        dqh(mi)  =  qdis*denwp + enw*dq(mi)
        deqh(mi) = qdis*denwt + enw*dqt(mi)
```

January, 2017

Modify flow_boundary_conditions_gaz.f: fix the pressure BC issue (remember to use pa in boun). Modified the GDKM input slightly to have a general formulation that is equivalent to dpdp (completed) and direction-specific fracture formulation.

Modify start_macro.f: increase read in file name character length, file names using 'file' option in macros can now be 200 characters long

Modify icectrco2.f: corrected enthalpy calculation for constant rate of CO2 release

February, 2017

Modify thrmwc.f: initializing variable "htc", uninitialized htc cause PC debug executable fail, and cause linux executable giving wrong results in salt study simulations

Modify chemod.f: fixed Henry's constant correction in decay chain reaction

July-August, 2017

rlp and rlpm related development:

rlperm_co2.f - call to subroutine vgrlps modified to be consistent with updated version of vgrlps.f

brooks_corey_rlp.f - equations are now consistent with Miller et al. Adv. Water Resourc. (1998), Table 4

corey.f - added comment lines

brooks_corey_cap.f - modified comment lines

scanin.f - corrected problem with reading long lines (increased size of dumstring) and with unallocated species array

vcap-ek.f - added comment lines, re-arranged code so that it's more modular.

check_rlp_carb.f - updated headings for relperm/cap pressure table, corrected bugs

rlp_cap_table.f - added comments

exponential.f - expanded functionality to include non-wetting phase calculation

phase_name.f90 - new routine, new code to calculate unique index for each phase couple

vgrlps.f - star is now calculated inside the subroutine, not passed in as a parameter, for consistency with all other relative perm subroutines

rlp_frac2.f - star is now calculated inside the subroutine, not passed in as a parameter, for consistency with all other relative perm subroutines

vgcap_fit3.f - new routine, update spline parameter calculation to be consistent with rlpm variable definitions

thrmwc.f - additional call to rlp_cap for consistency with new rlpm implementation

inrlp.f90 - replace inrlp.f, new implementation of rlpm input format

cappr2.f90 - new routine, new variable definitions to support new implementation of rlpm

rlp_cap.f90 - replace rlp_cap.f, new implementation of rlpm

check_rlp.f - changed a few parameter names to make code more readable, improved headers to make output table more readable

comrlp.f - added new variables to support new implementation of rlpm

brooks_corey_rlp3.f - fixed bug in parameter passing

linear.f - very minor improvements to precision

September, 2017

Modify user_ymp.f: changed the control to turn off CO2 injector for constant pressure boundary condition

End Developer Notes 12/04/2015 to 9/21/2017 --------------------------