**RT-Flux-PIHM   
User & Developer manual**

July 25th 2015

The Pennsylvania State University

University Park

Contact: [lxl35@eme.psu.edu](mailto:lxl35@eme.psu.edu)

[Cub200@psu.edu](mailto:Cub200@psu.edu)

Table of Contents

1 Introduction 4

2 Input file format 6

2.1 projectName.txt 6

2.2 \*.chem 6

2.2.1 TITLE 7

2.2.2 RUNTIME 7

2.2.3 GLOBAL 7

2.2.4 INITIAL\_CONDITIONS 8

2.2.5 OUTPUT 10

2.2.6 PUMP 10

2.2.7 Condition 11

2.2.8 SECONDARY\_SPECIES 11

2.2.9 MINERALS 12

2.2.10 PRECIPITATION 12

2.3 \*.cdbs 12

2.4 \*.cini 12

2.5 \*.prep 13

3 Output file format 14

3.1 \*.conc 14

3.2 \*.btcv and \*index.btcv files 14

3.3 \*.vol 15

4 Reactive transport fundamentals 16

4.1 Advection Diffusion Reaction Equation 16

4.2 Mass Action Law 16

4.3 Transition State Theory For Kinetic Reactions 18

4.4 Debye–Hückel Model For Species Activity 19

4.5 Reactive Transport in Unsaturated Zone 20

4.6 Sequential Non-Iterative Approach (SNIA) 20

4.7 Local Jacobian Matrix for Kinetic Reactions 22

4.8 Speciation For Equilibrium Controlled Reactions 24

4.9 Total Variation Diminishing 26

4.10 Influences of Temperature 27

4.11 Adaptive time stepping 28

5 RT code details 29

5.1 Source Code Structure 29

5.1.1 Code Coupling Structure Between RT and Flux-PIHM 29

5.1.2 Code Structure in RT 29

5.2 General introductions to RT source codes 31

5.2.1 Data Structure and Library <rt.h> 31

5.2.2 RT component main code <rt.c> 32

5.2.3 Transport functions <os3d.c> 33

5.2.4 Reaction Functions <react.c> 33

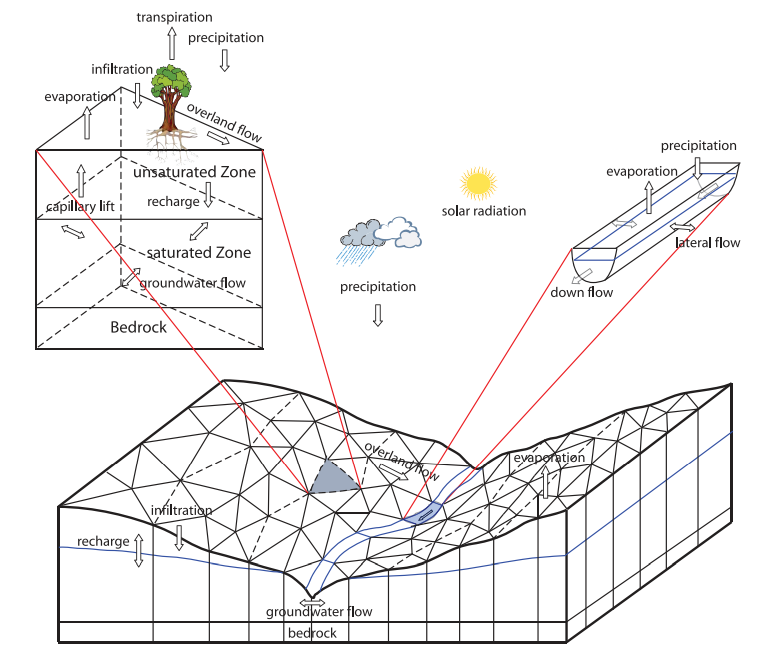
6 Flux-PIHM processes 35

7 Reference 37

8 38

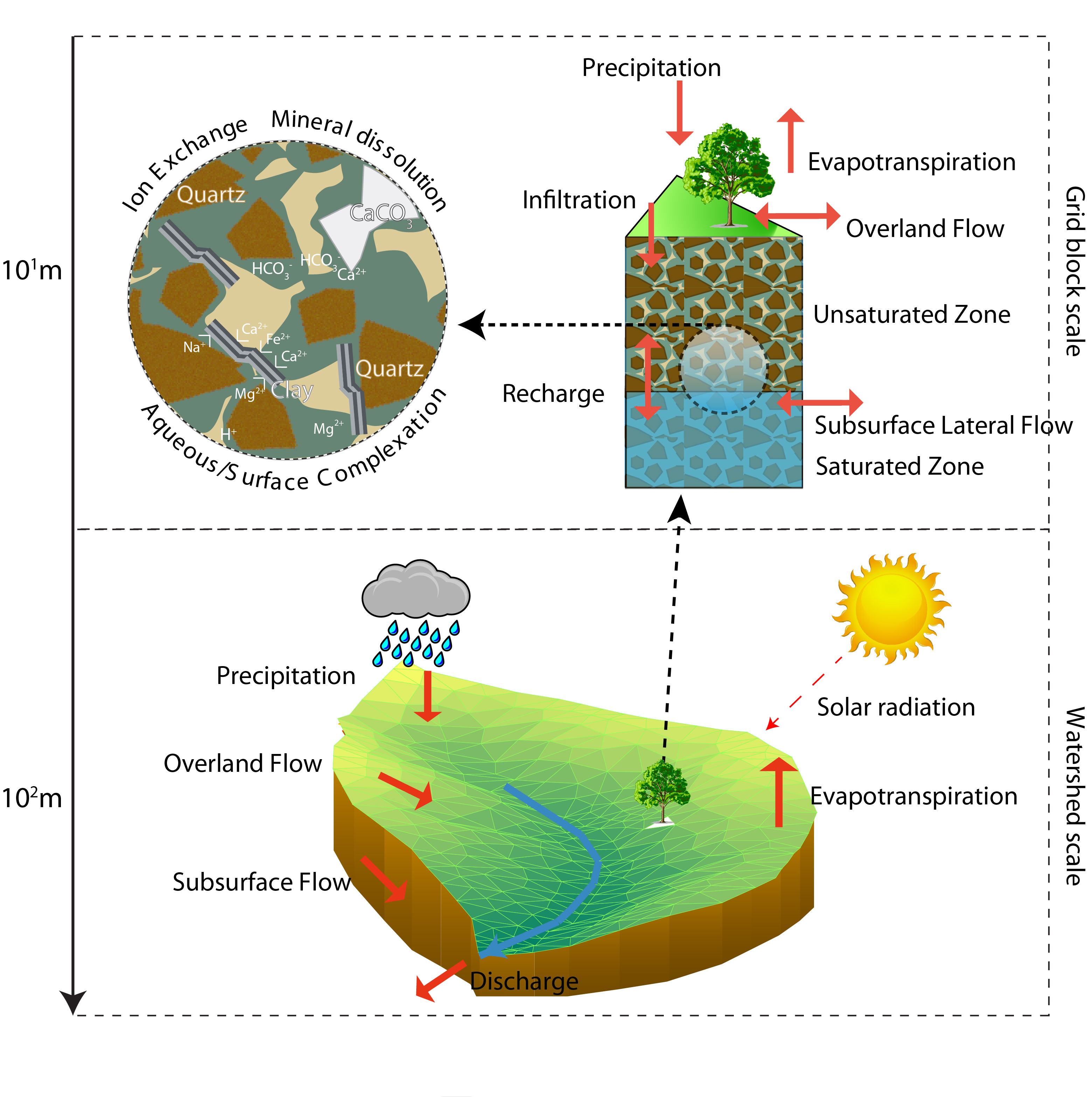
# Introduction

RT-Flux-PIHM is an integrated hydrogeocheical model that operates on the catchment scale. It is comprised of two components, a hydrological land surface model Flux-PIHM, and a reactive transport model RT. Flux-PIHM is a FVM based, semi discrete watershed simulator. For functionality, methodology and usage of Flux-PIHM, the users are recommended to refer to [*Qu and Duffy*, 2007; *Shi et al.*, 2013b]. The majority of the functions supported in Flux-PIHM is also supported in RT-Flux-PIHM.



*Figure 1‑1 Schematic plot of PIHM illustrating unstructured grids and processes within each grid (adapted after Qu and Duffy 2007).*

The reactive transport model RT is an add-on to Flux-PIHM, it uses the flow and thermal conditions that PIHM generated at each time step as drivers to update the changes in the water chemistry and mineralogy. For information and backgrounds on general purpose reactive transport model, the users are recommended to refer to [*Steefel*, 2009]. In order to run a simulation, information on the watershed chemistry, mineralogy, a priori knowledge of reaction thermodynamics and kinetics, boundary conditions of water chemistry are required. In the next section, we will explain the format of input files required by RT-Flux-PIHM. In the third section, we will cover the format and physical meanings of the output files generated by RT-Flux-PIHM in addition to Flux-PIHM. In the final section, a brief explanation of the original code is provided for future developers.



*Figure 1‑2 Schematic plot of reactive transport processes and reaction types occurring in field.*

# Input file format

|  |  |
| --- | --- |
| projectName.txt | Specifying the project name |
| \*.chem | Specifying the initial chemical condition and solving techniques |
| \*.cdbs | The geochemical database |
| \*.cini | Specifying the distribution of the initial geochemical condition in the field |
| \*.prep | The time series of the rainwater solute concentration |

## projectName.txt

This file is used to indicate the name of the project. All the input files for this project should have the same filename but with different suffix. For input files for Flux-PIHM, please refer to the Flux-PIHM input file format document.

## \*.chem

The asteroid here is the project name specified in projectName.txt. The structure of the chemical condition file “\*.chem” is very similar to the input file of the reactive transport code Crunchflow. Within the file, both “#” and “!” could be used to comment. It is recommended to use “#” to supply general comments on this text file interface and use “!” to comment out unwanted specifications. This file is divided into several blocks. Within each block, keywords are used to specify simulation options. Reserved block names are:

TITLE

RUNTIME

GLOBAL

INITIAL\_CONDITIONS

OUTPUT

PUMP

Condition

PRECIPITATION

SECONDARY\_SPECIES

MINERALS

### TITLE

Title or comments for the project, it will not be read by the RT simulator.

### RUNTIME

A full list of RT runtime variables is listed in the following table.

|  |  |  |  |
| --- | --- | --- | --- |
| Variables | Value Type | Value range | Remarks |
| tvd | Bool | True/false | Please refer to TVD section in code function part. Increase accuracy if enabled at the price of reducing speed. |
| output | Integer | >0 | Frequency of chemical output, in hours |
| activity | Integer | 0/1 | If enabled (1), use activity model |
| act\_coe\_delay | Integer | 0/1 | If enabled (1), activity coefficient is not updated in every solving iteration |
| thermo | Integer | 0/1 | If enabled (1), RT simulator will use temperature data to compute chemical equilibrium and kinetics |
| relmin | Integer | 0/1 | If enabled (1), mineral volume fraction will be relative (to total mineral volume fraction) |
| effads | Integer | 0/1 | If enabled (1), RT simulator will use effective adsorption mode (MIM) |
| transport\_only | Integer | 0/1 | If enabled (1), RT simulator will enter transport only mode. Reaction will be skipped. |
| precipitation | Integer | 0/1/2 | 0: no precipitation concentration will be specified (0 solute concentration in precipitation assumed)  1: precipitation concentration will be specified in this file follow PRECIPITATION keyword  2: precipitation concentration will be specified in \*.prep file as a time series |
| RT\_delay | Integer | >0 | Run RT simulator after certain days from the start of Flux-PIHM simulation. |
| Condensation | Double | >1 | The concentration ratio between throughfall and rainwater, generally 1.0~5.0 |
| AvgScl | Integer | >0 | Average scaling factor, in minutes. Generally 0~30. If runtime error is encountered, reduce this number until no error is reported. |

### GLOBAL

A full list of global variables used in RT simulator is listed in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| Variables | Value Type | Value range | Remarks |
| t\_species | Integer | >0 | Number of total primary species including minerals. An easy way to get this number to from the number of lines in each chemical condition block. |
| s\_species | Integer | >0 | Number of secondary species |
| minerals | Integer | >0 | Number of minerals |
| adsorption | Integer | >0 | Number of primary adsorption species |
| cation\_exchange | Integer | >0 | Number of primary cation exchange species |
| mineral\_kinetic | Integer | >0 | Number of kinetic mineral reactions |
| aqueous\_kinetic | Integer | >0 | Number of kinetic aqueous reactions |
| diffusion | Double | >0.0 | Fixed diffusion coefficient, in cm2/s |
| dispersion | Double | >0.0 | Fixed dispersivity, in m. |
| cementation | Double | >0 | Cementation factor |
| temperature | Double | >0 | Field temperature in oC, voided if thermo option is enabled. |

### INITIAL\_CONDITIONS

Chemical properties of soil and water are highly likely to be very heterogeneous. To account for this heterogeneity, we provide two ways to specify the spatial distribution of chemical conditions. The first method is to directly specify the chemical conditions for groups of finite elements in the field in this keyword block. The second method is to use \*.cini file to specify chemical condition for every finite elements one by one. To use the first option, the users can use syntax like:

CONDITION\_NAME 1-1070

This specification gives the chemical condition “CONDITION\_NAME” to elements 1 to 1070.

CONDITION\_NAME 1,2,3,4,5

This specification gives the chemical condition “CONDITION\_NAME” to elements 1, 2, 3, 4 and 5.

Multiple conditions could be specified at once:

CONDITION\_NAME1 1-1070

CONDITION\_NAME2 500-600

CONDITION\_NAME3 550-602

CONDITION\_NAME4 703,704

This specification assign “CONDITION\_NAME1” to elements 1 to 1070, and then assign “CONDITION\_NAME2” to elements 500 to 600. Later assignments will overwrite previous assignments in this case. Each of the CONDITION\_NAME appeared here should be defined in later sections of this input file using “Condition” keyword as a prefix.

To use the second option, the users should attach FILE keyword right after INITIAL\_CONDITIONS keyword in this block and then specify the filename of the initial condition distribution. E.g:

INITIAL\_CONDITIONS FILE shp.cini

END

FILE keyword has a higher priority than direct specifications. If FILE keyword appears after INITIAL\_CONDITIONS, the code will skip specification made in the first method. E.g.:

INITIAL\_CONDITIONS FILE shp.cini

CONDITION\_NAME1 1-1070

CONDITION\_NAME2 500-600

CONDITION\_NAME3 550-602

CONDITION\_NAME4 703,704

END

The code will read shp.cini regardless the specifications made within this block. Also note that the number of elements is doubled in RT compared to Flux-PIHM. Each Flux-PIHM element is further divided into the unsaturated zone element and the saturated zone element. Suppose there is Num\_Ele elements in Flux-PIHM, there will be 2\* Num\_Ele elements in RT. The first Num\_Ele elements are saturated zone elements and the Num\_Ele+1 to 2\* Num\_Ele elements are unsaturated zone elements.

### OUTPUT

This keyword is used to output chemical concentration time series in certain elements, followed by the total number of elements that the user wish to see the concentration evolution. E.g.:

OUTPUT 3

1100

500

100

END

This specification will instruct the code to output chemical concentration time series for elements 1100, 500 and 100. The output file generated in this manner will be named \*1100.btcv, \*500.btcv, \*100.btcv, respectively. The asteroid here is the project name.

### PUMP

This keyword is used to specify the location and the injection rate of injecting wells/ groundwater influxes, followed by the total number of injection/ influx elements. For example:

PUMP 1

1109 Mg++ 1000

END

Specifies a groundwater influx at element 1109 that will induce an inflow of Mg++ of 1000 mol/year. Note that substantially large injection rate relative to the soil water concentration will lead to non-convergence of code due to sudden jumps in the total concentration induced by injection.

### Condition

This keyword is used to specify the details of the chemical conditions specified in INITIAL\_CONDITIONS. Multiple total concentrations of primary species could be specified in a way that is very similar to CRUNCHFLOW. E.g.:

Condition MS

pH 4.48

HCO3- 8.0E-4

Mg++ 7.05E-5

Cl- 3.12E-5

Na+ 18.5E-6

H4SiO4 1.00E-8

K+ 1.53E-5

shillite 0.2690 -ssa 70.0

shchlorite 0.0669 -ssa 70.0

Kaolinite 0.0283 -ssa 70.0

Quartz 0.6358 -ssa 70.0

FeOOH 1.00E-5 -ssa 70.0

>SOH 1.00E-5

END

For method to specify chemical concentration, please refer to CRUNCHFLOW’s user manual.

### SECONDARY\_SPECIES

This block is used to specify the secondary species that the users would like to track. A complete list of the available secondary species resulted from the specified primary species could be generated via running a CRUNCHFLOW instance with the same primary species set using the database\_sweep option.

### MINERALS

The kinetic mineral reactions are specified here. This block follows a similar format as the corresponding block in CRUNCHFLOW input file.

### PRECIPITATION

This block is used to specify a static rainwater composition.

## \*.cdbs

The chemical database file \*.cdbs has the exactly same format as CRUNCHFLOW’s database files. In fact, the users can use RT’s database file and CRUNCHFLOW’s database file interchangeably. For a complete description on the CRUNCHFLOW’s database file, please refer to CRUNCHFLOW user’s manual.

## \*.cini

The initial chemical condition file \*.cini is used to specify the spatial distribution of chemical conditions in the field. This file is simply a full list of elements and the chemical condition they belong to.

File Structure:

|  |  |
| --- | --- |
| Index | Condition index |
| Repeating *NumEle* times | |

Description:

|  |  |  |  |
| --- | --- | --- | --- |
| Variables | Value Type | Value range | Remarks |
| Index | Integer | >0 | Index of element |
| Condition index | Integer | >0 | Index of chemical conditions in \*.chem |
| NumEle | Integer | >0 | Number of elements in simulation |

## \*.prep

RT simulator can utilize time series date in rainwater solute concentration. Users could use the syntax “precipitation 2” to enable this option.

File Structure:

|  |  |  |
| --- | --- | --- |
| Pconc | Num\_Pconc | Num\_Precipitation |
| Species\_index | *Repeat Num\_Pconc times* | |
| Time | Value |  |
| Repeat *Num\_Precipitation times* | | |

Description:

|  |  |  |  |
| --- | --- | --- | --- |
| Variables | Value Type | Value range | Remarks |
| Pconc | Keyword |  | Keyword indicate beginning of the file |
| Num\_Pconc | Integer | >0 | Number of species in rainwater |
| Num\_Precipitation | Integer | >0 | Number of rainwater concentration measurements |
| Time | Time string | time |  |
| Value | Double | >0.0 | Rainwater concentration of solutes, in M |

# Output file format

|  |  |
| --- | --- |
| \*.conc | Chemical concentration evolution in the field |
| \*.btcv | Chemical concentration evolution in the outlet of stream |
| \*index.btcv | Chemical concentration evolution in selected elements |
| \*.vol | Later saturated zone flow velocity |

## \*.conc

This file is the main output of RT simulator. It contains the concentration of all species in each element at specified output frequency (e.g. hourly, daily, etc.).

File Structure:

|  |
| --- |
| Time |
| Cell | Species\_Name | … *Num\_Species* |
| Index | Concentration | … *Num\_species* |
| Repeat 2\**Num\_Ele times* | | |

|  |
| --- |
| Time |
| Cell | Species\_Name | … *Num\_Species* |
| Index | Concentration | … *Num\_species* |
| Repeat 2\**Num\_Ele times* | | |

… Repeat Num\_output times

Description:

|  |  |  |  |
| --- | --- | --- | --- |
| Variables | Value Type | Value range | Remarks |
| Time | Time string | time |  |
| Species\_Name | String |  | Name of all the chemical species |
| Index | Integer | >0 | Index of elements |
| Concentration | Double |  | Logarithm of concentration in M |
| Num\_species | Integer | >0 | Number of total species simulated |
| Num\_Ele | Integer | >0 | Number of Flux-PIHM elements |
| Num\_output | Integer | >0 | Total number of output data points |

## \*.btcv and \*index.btcv files

The breakthrough curve files are auxiliary output files that provide the concentration evolution at selected elements.

File Structure:

|  |  |  |
| --- | --- | --- |
|  | Species\_Name | … *Num\_Species* |
| Time | Concentration | … *Num\_species* |
| Repeat *Num\_output times* | | |

Description:

|  |  |  |  |
| --- | --- | --- | --- |
| Variables | Value Type | Value range | Remarks |
| Time | Time string | time |  |
| Species\_Name | String |  | Name of all the chemical species |
| Concentration | Double |  | Logarithm of concentration in M |
| Num\_species | Integer | >0 | Number of total species simulated |
| Num\_output | Integer | >0 | Total number of output data points |

## \*.vol

This file contains the information on the velocity field in the watershed. Both the x component and the y component of the velocity vectors in each of the saturated zone elements are reported. In unsaturated zone, there is no lateral flow.

File Structure:

|  |  |  |
| --- | --- | --- |
| Time |  |  |
| Velocity\_x | … *Num\_Ele* |  |
| Velocity\_y | … *Num\_Ele* |  |
| Repeat *Num\_output times* | | |

Description:

|  |  |  |  |
| --- | --- | --- | --- |
| Variables | Value Type | Value range | Remarks |
| Time | Time string | time |  |
| Velocity\_x | Double |  | X component of the velocity vector |
| Velocity\_y | Double |  | Y component of the velocity vector |
| Num\_Ele | Integer | >0 | Number of Flux-PIHM elements |

# Reactive transport fundamentals

The essence of reactive transport model is to solve for the mass conservation of each chemical species within the system of study while considering mass transfer and chemical reactions [*Steefel and Lasaga*, 1994].

## Advection Diffusion Reaction Equation

As introduced in the first chapter, the governing equation of reactive transport model is the Advection Diffusion Reaction Equation (ADR) where chemical reaction is considered as an additional source/sink:

 *( 4‑1*

Where *φ* is the porosity in the control volume [m3/m3]; *Ci* is the concentration of chemical species *i* [mol/m3]; *V* is the volume of the control volume [m3]; *Θi* is the mobility indicator of the chemical species, which is 1 for aqueous mobile species and 0 for immobile species including minerals, adsorbed species, cation exchange sites, etc. ***D*** is the combined dispersion/diffusion tensor [m2/s] and ***u*** is the Darcian flux vector [m/s]. *Ri* is the total reaction rate of the chemical species *i* [mol/s]; *Si* is the additional source or sink [mol/s].

## Mass Action Law

Although equation *4-1* holds true for any individual species that exists in the fluids, the number of total equations to be solved in each control volume could be reduced because of the “mass action law”.

For each individual species:

 *( 4‑2*

Where *Reqi* is the rate of reaction in chemical equilibrium and *Rkini* is the rate of kinetic reaction [mol/s]. In general, if a reaction reaches its equilibrium within a short time, it is classified as equilibrium-controlled reaction. For each of the equilibrium-controlled reactions simulated in the system, concentration of one chemical species in that reaction becomes a function of the concentrations of other species in that reaction and the chemical equilibrium constant. Therefore, because of equilibrium-controlled reactions, usually only a fraction of chemical species are independent variables. The system with *Ntot* chemical species could then be represented by *Nc* independent chemical species plus *Nx*=*Ntot - Nc* chemically dependent species [*Kirkner and Reeves*, 1988]. We would call the *Nc* independent chemical species as primary species, and other species secondary species. Partitioning between primary species and secondary species are not unique. We can have different set of primary species/ secondary species combination for the same problem.

If a secondary species is generated by equilibrium controlled reactions:

* ( 4‑3*

Where *Ai* and *Aj* is the chemical formula of the primary and secondary species in the reaction; *vij* is the stoichiometry coefficient of the species in this reaction. For each equilibrium-controlled reaction, we can then write the **Mass Action Law** that is

 *( 4‑4*

Where *Ki* is the equilibrium constant for the equilibrium controlled reaction, *γi* is the activity coefficient of the species *i*; *Xi* is the concentration of secondary species *i* [mol/m3].

The production rate of a primary species due to equilibrium-controlled reaction can be written in terms of the sum of production rates of the secondary species due to the same equilibrium-controlled reaction [*Kirkner and Reeves*, 1988].

 *( 4‑5*

Where *Reqi* is the production rate of secondary species *i* and *Reqj* is the production rate of primary species *j* in this equilibrium controlled reaction. This leads to a reduction of total numbers of ADR equations by:



*( 4‑6*

If we further define the **Total Analytic Concentration** for a given primary species as:

 *( 4‑7*

 *( 4‑8*

## Transition State Theory For Kinetic Reactions

For a lot of reactions, its reaction rate is not fast enough to reach chemical equilibrium within a short period of time. This is especially true for heterogeneous reactions where reactants are in different phases and the reaction therefore only happens on the interfaces of different phases. For example, the dissolution of calcite:



**Rate laws** are used to compute the kinetic rate of kinetic reactions. Transition State Theory (TST) is a prevalent rate law for precipitation/dissolution reactions [*Eyring*, 1935; *Lasaga*, 1998]. In RT-Flux-PIHM, TST is the default rate law [*Helgeson et al.*, 1984; *Lasaga*, 1984]:

* ( 4‑9*

Where *Am* is the bulk surface area in a control volume [m2]; *km* is the intrinsic rate constant in the unit of [mol/m2/s]; *Ea* is the activation energy [J/mol]; *R* is the gas constant [J/K/mol]; *T* is temperature [K]; *ai* is the activity of species *i*; *m1, m2* and *m3* are matching parameters and are 1 for most reactions if unspecified. Bulk surface area is calculated from:

 ( 4‑10

where *Aspecific* is the specific surface area [m2/g]; *φm* is the volume fraction of the mineral m; *MWm* is the molar mass of mineral m [g/mol];  is molar volume of mineral m [m3/mol].

*g* is defined as:

 *( 4‑11*

Where *Q* is the Ion Activity Product (IAP) and *Keq* is the equilibrium constant for this kinetic reaction.

## Debye–Hückel Model For Species Activity

Only a fraction of molecules of a chemical species will participate chemical reaction. The fraction that actively reacts is often referred activity [*Pitzer*, 1977]:

 ( 4‑12

We used an extended Debye–Hückel Equation to compute for the activity coefficient of chemical species:



*( 4‑13*

where *γi* is the activity coefficient of species *i*; *AT*, *BT* and *bT* is the temperature dependent Debye–Hückel parameters that can be looked up in the database; *Zi* is the charge of species *i*; I is the ionic strength of the solution:

 ( 4‑14

## Reactive Transport in Unsaturated Zone

Control volumes imported from Flux-PIHM is can be saturated with water or unsaturated with water. The unsaturated zones function differently from the saturated zone in many ways including transport phenomena [*Nielsen and Biggar*, 1986; *Šimůnek et al.*, 2003], heat conductivity [*Saito et al.*, 2006], and reactions [*Voss*, 1984]. Therefore, we added saturation into our governing equations to account for the aqueous phase volume.

 ( 4‑15

Therefore, the governing equation becomes:

 ( 4‑16

Where S*W* is the water saturation. And the reaction rate calculation, the bulk surface area that is reactive in unsaturated zone is calculated by

 ( 4‑17

where *Am,0* is the bulk surface area for mineral *m* if it is in the saturated zone.

## Sequential Non-Iterative Approach (SNIA)

The RT module follows a sequential non-iterative approach (SNIA) that has been widely applied [*Walsh et al.*, 1984; *Yeh and Tripathi*, 1989]. This approach is efficient in integrating subsurface flow simulation with reactive transport algorithms with some limited noted [*Jacques et al.*, 2006]. The SNIA is a derivative of the operator-splitting method that decouples ADR equation and solves transport and reaction steps separately [*Xu et al.*, 1999; *Zysset et al.*, 1994]. In the RT module, the transport step is solved using the Euler forward method while the reaction step is solved iteratively using the Crank–Nicolson and Newton Raphson method. In the reaction step, the local matrices accounting for the mass balance of all primary species in each finite volume are assembled and solved using a matrix solver in SUNDIALS [*Hindmarsh et al.*, 2005].

The accuracy of the SNIA approach depends on multiple factors including spatial discretization, time stepping, and flow conditions. To minimize numerical errors and improve convergence, we adopted the approach in Jacques et al. (2006). Performance index (PI) is used to optimize the time stepping. PI is set to be less than 0.05 for every shared surface between elements to minimize the operator splitting error associated with temporal discretization [*Jacques et al.*, 2006; *Perrochet and Bérod*, 1993]:

 ( 4‑18

Here  is the time duration in each step. The maximum time step is set to be 15 minutes [*Jacques et al.*, 2006]. The use of small PI and maximum time step requires a longer computational time; however, this improves the SNIA accuracy.

The numerical procedures carried out in RT between time *n* and time *n+1* could be summarized as:

 ( 4‑19

 ( 4‑20

Considering the fact that Flux-PIHM uses area normalized fluxes and water heights as variables, the above equations could be further written into:

 ( 4‑21

where *Dlk* is the dispersion tensor at the interface of element *l* and *k*; *Ak* is the base area of the prismatic element *k*; *qlk* is the area normalized flux from element *l* to *k*.

 ( 4‑22

We used Crank-Nicolson method to evaluate the exact value of *Rkinj* right after each iteration:

 ( 4‑23

where *(Rkinj)n,t+1* is the rate evaluated based on the concentrations of chemical species at the time *n* and iteration *t+1*.

which leads to

 ( 4‑24

Equation (4-22 and (4-24 are the two equations that are solved in RT component of RT-Flux-PIHM. The first equation is solved using an Euler forward method (os3d.c) and the second equation on chemical reaction is iteratively solved using Crank–Nicolson method and Newton Raphson method for each of the elements in the field (react.c).

## Local Jacobian Matrix for Kinetic Reactions

To solve equation (4-24, Newton Raphson method is implemented. First, we define the evaluation functions of each of the total analytic concentrations:

 ( 4‑25

For a total of *Nx* species, we end up with *Nx* such evaluation functions:

 ( 4‑26

where **U** is the vector of total analytic concentration from species *1* to species *Nc*.

The corresponding Jacobian Matric for this evaluation function becomes:

 ( 4‑27

The exact values of the entries in **J** in each iterative step could be evaluated by using numerical perturbation method:

 ( 4‑28

In the chemical reaction system, it would be more convenient to mark the entries in the Jacobian matrix that will be always zeros in the initialization stage. Then in each solving process, those entries with a zero mark will not be evaluated to save CPU time:

 ( 4‑29

where *δij* is the dependency function of *fi* on *Uj.* If *fi* is dependent on *Uj*, *δij* =1, otherwise *δij* = 0. Entries with *δij* = 0 will not be evaluated in the iterative solver process, thusly saving CPU time.

The iterative solving process then is:

 ( 4‑30

The iterative process stops if the improvement between steps is smaller than a predefined threshold:

 ( 4‑31

where γ is the threshold of accepting a solution.

## Speciation For Equilibrium Controlled Reactions

In reactive transport model, only the total analytic concentrations of the independent species are solved. After reaching the solution for the total concentrations, additional steps are required to obtain the concentration of all the individual species. This process is often called “Speciation” in RTMs.

In this section, we will discuss the process of using Newton Raphson method to perform the speciation operation. Let us start from Mass Action Law discussed in earlier section. It will be simpler if we first express the concentration of secondary species into a logarithmic form:



 ( 4‑32

This could be also expressed in a matrix vector multiplication operation:



( 4‑33

where **X** is the vector of secondary species concentration. We use [***v***] to represent the matrix with entries *vij*. This matrix is also called dependency matrix, because the “dependencies” of secondary species on primary species are stored in this matrix.

If we also express the definition of total concentration:



in a matrix manner:



or

 ( 4‑34

It now becomes easier to see how the Newton Raphson method works for speciation process. Let us define evaluation function:

 ( 4‑35

Then the local Jacobian matrix for speciation purpose is:



( 4‑36

where *δij,g* is the dependency function of *gi* on *Cj.* If *gi* is dependent on *Cj*, *δij,g* =1, otherwise *δij,g* = 0. Entries with *δij,g* = 0 will not be evaluated in the iterative solver process, thusly saving CPU time.

The exact values of the entries in **J\**g*** in each iterative step could be evaluated by using numerical perturbation method:

 ( 4‑37

Finally, we solve for **C** iteratively:

 ( 4‑38

until the improvement satisfies

 ( 4‑39

## Total Variation Diminishing

Numerical dispersion is also a serious concern for large-scale transport simulations with large grid blocks and fast flow dominated by convection [*Li and Duffy*, 2012]. A total variation diminishing technique (TVD), which is third-order accurate in smooth regions, is used to reduce the extent of numerical dispersion [*Gupta et al.*, 1991]. This method addresses the numerical dispersion problem by using the concentration at the interface of the element under investigation and the upstream element, instead of the volume averaged concentration of the upstream element:

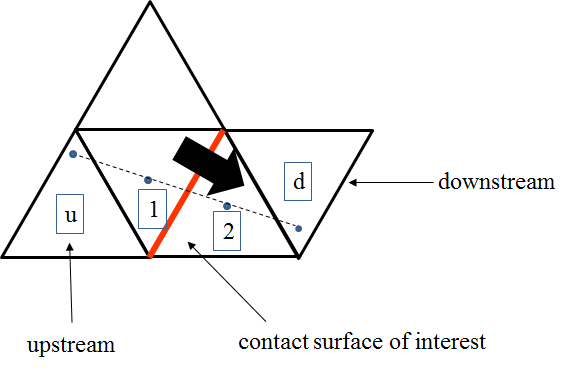


Figure 4‑1 Schematic plot of Gupta TVD for unstructured grid

Let us assume we are interested to calculate the concentration of species at the face between element 1 and element 2. The first step to perform the TVD is to find the (potential) upstream and downstream cell for the pair of element 1 and 2 by using a simple extrapolation of the centroid of element 1 and 2. This search is performed in the initialization stage of the code and is only performed once.

Then the concentration at the interface of element 1 and 2 is calculated as (assuming flow direction is from 1 to 2):

 ( 4‑40

Where

 ( 4‑41

And r is defined as:

 ( 4‑42

## Influences of Temperature

Flux-PIHM also simulates the temperature evolution in the subsurface. The kinetics of the chemical reactions is dependent on temperature. Therefore, the influence of temperature changes on kinetic reactions are also considered in RT-Flux-PIHM model:

 ( 4‑43

where *k* is the rate constant for kinetic reaction [mol/m2/s]; *kref* is the rate constant in the reference temperature, which is 298.15K in RT-Flux-PIHM; *Ea* is the activation energy [J/mol]; *R* is the gas constant [J/mol/K]; *T* is the soil temperature [K] and *Tref* is 298.15K.

## Adaptive time stepping

In the reaction step, the RT module uses concentrations from the previous time step as the initial guess to iteratively solve for the concentration of the current time step. The spatial heterogeneity in hydrological processes can cause significant non-convergence problems. For example, the flow rates between any two neighboring elements in Shale Hills vary by six orders of magnitude (from 0.01 m3/d for slowest surface flow to 4000 m3/d in stream flow during storms). When flow rate anywhere in the simulated domain is fast enough, time steps are reduced to facilitate convergence. Reducing the time steps based on the fastest flow, however, would essentially prevent the model from time marching. Therefore, sub time step interpolation is used in the fast-flow regime to obtain a closer initial guess when non-convergence is encountered.

# RT code details

## Source Code Structure

### Code Coupling Structure Between RT and Flux-PIHM

Based on the water distribution and flow rates calculated by Flux-PIHM, the RT module solves the ADR equations for the spatial and temporal evolution of aqueous and solid phase composition. The RT module is one-way coupled to Flux-PIHM, as shown in the following figure. That is, we have assumed that alteration in aqueous and mineralogical composition has negligible impact on the hydrological processes, which is reasonable at the time scale of months to years. Such a one-way coupling has been described by others as a “cascade” of models [*Goddéris and Brantley*, 2013].

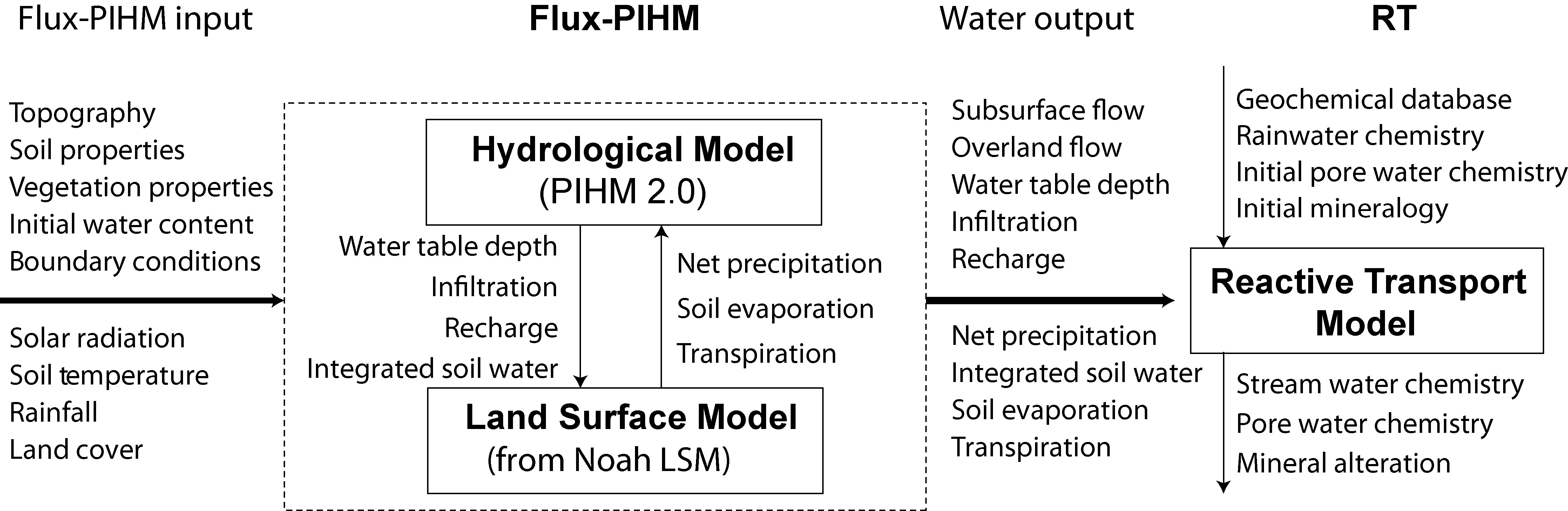


Figure 5‑1 Input, output, and process coupling within RT-Flux-PIHM. The Flux-PIHM module solves the hydrological and land surface dynamics. The RT module reads flow and water distributions from Flux-PIHM and specified initial chemistry of water and solid phases, and outputs the spatial and temporal evolution of aqueous and solid phase compositions based on the hydrological conditions and geochemical thermodynamics and kinetics.

### Code Structure in RT

In each time step, after Flux-PIHM finishes computation and reached a solution in the water distribution, RT will be called to perform reactive transport simulation based on the flow field information generated in Flux-PIHM. A flow chart of the RT component is provided in the following figure. The source code files responsible for the simulated processes are also indicated in different colors.

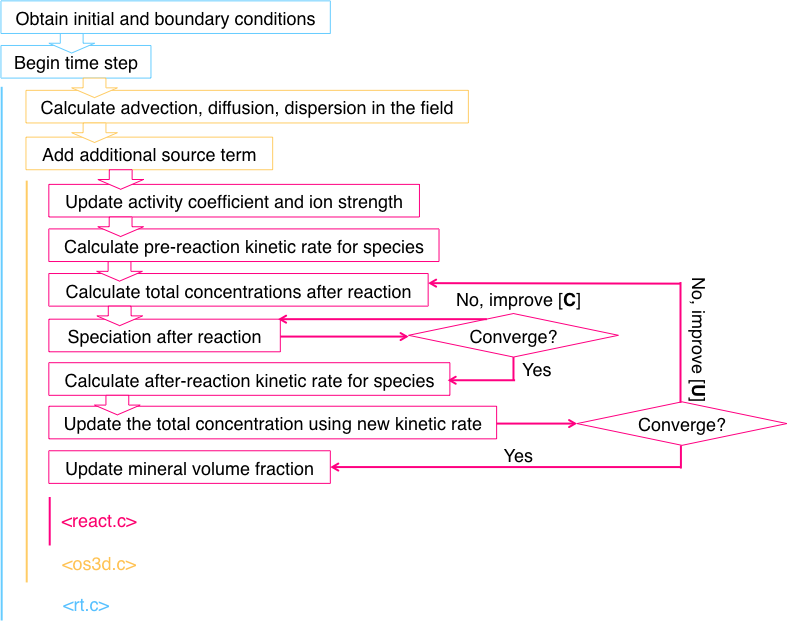


Figure ‑ Flow chart of RT for each time step. [**C**] is the vector of primary species concentration; [**U**] is the vector of total species concentration. A nested Newton Raphson iteration is performed to solve for concentration evolution resulted from reactions.

## General introductions to RT source codes

RT source codes includes four files:

|  |  |
| --- | --- |
| rt.h | Data structure and library for RT |
| rt.c | RT main code |
| os3d.c | Mass transfer functions |
| react.c | Chemical reaction functions |

The functions of each file will be discussed in the following sections.

### Data Structure and Library <rt.h>

This is the header file for the RT simulator. It contains the definition of the data structures used in other codes. We will introduce the data structures from a top-down perspective:

At the highest level, all the data are stored in a grand data structure that is called *chem\_data\_structure*. Within this data structure, runtime related and global variables are stored. These variables are generally specified in the beginning of the simulation and will stay constant throughout the simulation. These variables are used to control the simulator.

Since the simulation domain is discretized into hundreds of elements, the physical and chemical properties of each element are organized into a single class that is called *vol\_conc\_type* (volume/concentration data type). In this class, the volumetric properties such as bottom area, water height, porosity, saturation and temperature are stored, visited and updated in every time step. In general, RT simulator relies on Flux-PIHM to update these physical variables. Concentrations of chemical species are also stored here. In each time step, RT solves for the advection-diffusion-reaction equation and updates the concentrations accordingly.

RT uses a connection-based finite volume method to solve for the differential equations. A key component of this connection-based method is the definition of connection class *face\_type*. In *face\_type* instance, the information on the upstream, downstream nodes is recorded. RT also stores some dynamic variables, such as the flow velocity in this data type.

For the chemical reaction calculation, each of the chemical species is realized by using a *species\_type* data structure. Key chemical parameters such as the molar weight, charge, species names are stored in this data class. Additional data structure is also used to store the information on kinetic reactions (*Kinetic\_reaction\_structure*).

For details of the data structures, developers should refer to the source code and code comments. Most of the data types within data structures are quite self-explanatory or explained in the comments.

### RT component main code <rt.c>

This is the source code for data exchange between RT and Flux-PIHM, as well as many controlling functions:

*chem\_alloc:*

This is the function that allocation memory spaces for and initialize the chemical data structure, reading input files and look up chemical reaction network from the chemical database.

*fluxtrans*:

This is the function that import flow field information from Flux-PIHM at the end of every Flux-PIHM time step. The data be transferred include: surface flow, subsurface flow, water storage in the saturated and unsaturated zone, evapotranspiration, net precipitation, infiltration, recharge. This function also calls os3d for the advection diffusion dispersion calculation.

*Monitor*:

This function performs a mass balance check on the imported Flux-PIHM water distribution and flow rates.

*PrintChem*:

This function outputs the RT variables into output files at a predefined frequency.

*AdptTime*:

This function handles the adaptive time stepping in RT for the reaction step. Initially, the reaction time step is set to be equal to the time step of Flux-PIHM simulation. If a certain element failed to converge in the reaction step, its time step will be reduced by a half until it converges.

### Transport functions <os3d.c>

The source code os3d.c, as the name implies, utilizing the operator splitting (3D) method to solve for the mass transfer part of the advection diffusion equation [*Steefel et al.*, 2014; *Zysset et al.*, 1994]. Such a separation in solving process is made possible by carefully controlling the time step [*Jacques et al.*, 2006]. This function also handles the additional source/ sink of chemical species.

### Reaction Functions <react.c>

The react.c code contains mainly three functions: 1) *Lookup* function to help the code read the chemical database; 2) *Speciation* function to calculate the chemical equilibrium status from the given set of primary and secondary species and their corresponding total concentration; 3) *React* function that is responsible for calculation of kinetic reactions, which will be called at every RT-Flux-PIHM time step.

*Lookup:*

When the user specifies a chemical species in the input file, it usually only have two components: the name of the chemical species and the total concentration. It becomes the code’s job to find out the right chemical parameters for this given chemical species. Therefore, the code searches for the name of the chemical species in the appropriate section of the database file. *Lookup* function is designed for this purpose.

*Speciation:*

The speciation function is used to calculate the concentrations of individual species from the total concentrations and thermodynamic parameters. The calculation follows an iterative manner as detailed in section 3.8.

RT-Flux-PIHM currently supports two modes of speciation: 1) speciation from total concentrations and 2) speciation from total concentration and pH. The method of the first mode could be found in section 3.8. In the second mode, total concentrations for all the other species are given except for hydrogen. Instead of specifying the total concentration of hydrogen ion, pH is specified, which is actually the activity of hydrogen ion. This specification actually reduces the number of total unknowns in the system from *Nc* to *Nc*-1. Therefore, we are only required to solve for this system with *Nc*-1 total concentrations. Once this system is solved and the concentrations for all other primary species are obtained, we calculate back the total concentration of hydrogen ion and store it.

*React:*

In each time step, the reaction rate of the kinetic reaction is calculated based on TST rate law and the current concentration of all species. Based on this reaction rate, an increase in product concentration is derived so as a reduction in the reactant concentration. Note that the concentrations derived here is still total concentrations. After this step, the system use speciation method to determine the exact concentrations of primary and secondary species. With the new concentrations of all species, the kinetic reaction rate is calculated again. This process could also be conceptualized in such a way: in every incremental time step, kinetic reactions will only generate total concentration for primary species. Then the primary species will automatically equilibrate with the secondary species.

# Flux-PIHM processes

The reactive transport code RT depends on the field flow information calculated in Flux-PIHM, a hydrological land surface simulator. Dr. Duffy’s research group in the Department of Civil and Environmental Engineering has developed this simulator in Penn State. Since the flow field used in the RT simulation is generated in Flux-PIHM, it will be beneficial to introduce the governing equations of Flux-PIHM for the flow field before we proceed to the details of RT itself.

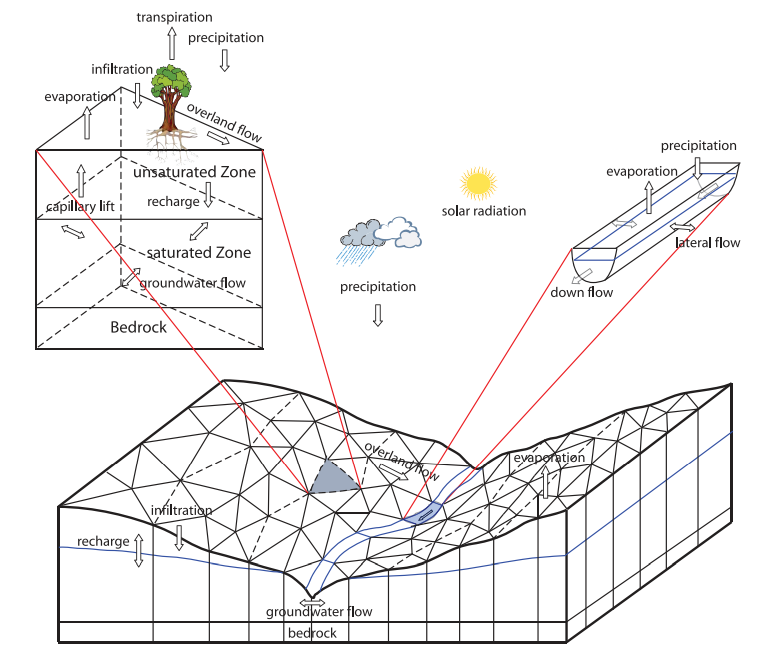


Figure 6‑1 Schematic plot of PIHM illustrating unstructured grids and processes within each grid (adapted after Qu and Duffy 2007).

In Flux-PIHM, several state variables are used as the principal unknowns to describe the water distribution in each prismatic element. They include water storage at surface *hs* [m3/m2] (water that builds upon land surface and forms overland flow), water storage in the unsaturated zone *hu*[m3/m2], and water storage in the saturated zone *hg* [m3/m2], where each storage volume is normalized by the base area of the prismatic element. The surface water storage in element *i* depends on precipitation, infiltration, evaporation, and overland flow rates in and out of the element *i*:

 ( 6‑1

where *p,* *qinfiltration, e*, and *qs,ij* are the rates of throughfall precipitation, infiltration from the water on land surface to the unsaturated zone, evapotranspiration, and lateral surface flow from element *i* to *j*, respectively, all normalized by the base area of the finite element [m3/(m2 s)]. *Ni,1*~3 is the index of the neighboring elements of element *i*. The flux rates *p, qinfiltration*, and *e* are computed in the land surface module Flux while *qs,ij* is calculated based on a diffusion wave approximation of the 2D St. Venant equation [*Gottardi and Venutelli*, 1993]. The water storage on surface governs the amount of overland flow.

Water height in the unsaturated and saturated zones of element *i* are each calculated as follows:

 ( 6‑2

and

 ( 6‑3

where *θ* is the porosity [m3pore space /m3 porous medium]; *qrecharge* is the area-normalized recharge rate into the saturated zone [m3/(m2 s)]; *qg,ij* is the saturated-zone-area normalized lateral flow rate [m3/(m2 s)] from element *i* to *j* in the saturated zone, which is based on Darcy’s law:

 ( 6‑4

where *Keff* is the effective hydraulic conductivity between element *i* and *j* [m/s]; *Bij* is the length of the common boundary; *Hg* is the hydraulic head which equals to *hg + z,* where *z* is the elevation of the bedrock; *Ai* is the base area of the prismatic element.

In Flux-PIHM, lateral flow rates are assumed to be zero in the unsaturated zone so that only vertical flow occurs and equation (2) only has the infiltration and recharge terms. Within each element, the saturated and unsaturated zones are divided by the (perched) water table in the soil layer. Because the position of the water table is variable in time and space, so are the thicknesses of the saturated and unsaturated zones. Flux-PIHM assumes a zero flux boundary condition at the top of the “bedrock”. Recharge is calculated based on integration over the unsaturated zone using the Van Genuchten equation [*Qu and Duffy*, 2007; *Van Genuchten*, 1980]. Macro pore flow is also simulated whenever the perched water table is higher than a predefined macro pore depth [*Beven and Germann*, 1982; *Shi et al.*, 2013b].

Within the river channel, the 1D St. Venant equation is applied to calculate the exchange between river elements. The water storage in the river elements is given by:

 ( 6‑5

where *hc,i* is the water storage in river channel; *qcin* and *qcout* are the area normalized flux rate from upstream element and to downstream element [m3/m2/s]; *qs,l* and *qg,l* are the area normalized flux rate from surface and aquifer of each side of the river channel.

If present, vegetation will intercept a fraction of rainfall. The storage of water there is given by:

 ( 6‑6

where *hv,i* is the water storage in the vegetation; *pv* is the total water equivalent precipitation; *ev* is the evapotranspiration within the surface vegetation. Moreover, for the calculation of net precipitation, soil evaporation and evapotranspiration in the system, surface and subsurface energy conservation are also considered in a land surface component adapted from Noah Land Surface Model [*Shi et al.*, 2013b].

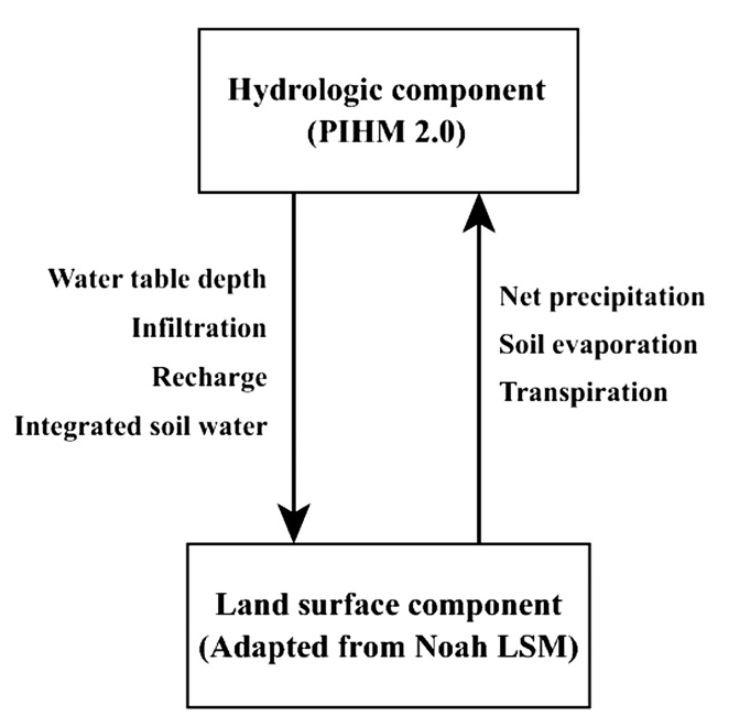


Figure 6‑2 Coupling structure in Flux-PIHM between PIHM and land surface model adapted from Noah Land Surface Model.

More details in governing equations and processes have been elaborated in two introductory papers for PIHM and Flux-PIHM [*Qu and Duffy*, 2007; *Shi et al.*, 2013a].

# Reference

Beven, K., and P. Germann (1982), Macropores and water flow in soils, *Water resources research*, *18*(5), 1311-1325.

Eyring, H. (1935), The activated complex in chemical reactions, *The Journal of Chemical Physics*, *3*, 107.

Goddéris, Y., and S. L. Brantley (2013), Earthcasting the future Critical Zone, *Elem. Sci. Anth.*, *1*(1), 000019.

Gottardi, G., and M. Venutelli (1993), A control-volume finite-element model for two-dimensional overland flow, *Advances in water resources*, *16*(5), 277-284.

Gupta, A. D., L. Lake, G. Pope, K. Sepehrnoori, and M. King (1991), High-resolution monotonic schemes for reservoir fluid flow simulation, *In Situ;(United States)*, *15*(3).

Helgeson, H. C., W. M. Murphy, and P. Aagaard (1984), Thermodynamic and kinetic constraints on reaction rates among minerals and aqueous solutions. II. Rate constants, effective surface area, and the hydrolysis of feldspar, *Geochimica et Cosmochimica Acta*, *48*(12), 2405-2432.

Hindmarsh, A. C., P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward (2005), SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers, *ACM Transactions on Mathematical Software (TOMS)*, *31*(3), 363-396.

Jacques, D., J. Šimůnek, D. Mallants, and M. T. van Genuchten (2006), Operator-splitting errors in coupled reactive transport codes for transient variably saturated flow and contaminant transport in layered soil profiles, *Journal of contaminant hydrology*, *88*(3), 197-218.

Kirkner, D. J., and H. Reeves (1988), Multicomponent mass transport with homogeneous and heterogeneous chemical reactions: effect of the chemistry on the choice of numerical algorithm: 1. Theory, *Water Resources Research*, *24*(10), 1719-1729.

Lasaga, A. C. (1984), Chemical kinetics of water-rock interactions, *Journal of Geophysical Research*, *89*(B6), 4009-4025.

Lasaga, A. C. (1998), *Kinetic theory in the earth sciences*, 811 pp., Princeton University Press, Princeton.

Li, S., and C. J. Duffy (2012), Fully-Coupled Modeling of Shallow Water Flow and Pollutant Transport on Unstructured Grids, *Procedia Environmental Sciences*, *13*(0), 2098-2121.

Nielsen, D., and J. Biggar (1986), Water flow and solute transport processes in the unsaturated zone, *Water resources research*, *22*(9S), 89S-108S.

Perrochet, P., and D. Bérod (1993), Stability of the standard Crank-Nicolson-Galerkin Scheme applied to the diffusion-convection equation: Some new insights, *Water Resources Research*, *29*(9), 3291-3297.

Pitzer, K. S. (1977), Electrolyte theory-improvements since Debye and Hückel, *Accounts of Chemical Research*, *10*(10), 371-377.

Qu, Y., and C. J. Duffy (2007), A semidiscrete finite volume formulation for multiprocess watershed simulation, *Water Resources Research*, *43*(8), W08419.

Saito, H., J. Šimůnek, and B. P. Mohanty (2006), Numerical analysis of coupled water, vapor, and heat transport in the vadose zone, *Vadose Zone Journal*, *5*(2), 784-800.

Shi, Y., K. J. Davis, F. Zhang, and C. J. Duffy (2013a), Evaluation of the Parameter Sensitivities of a Coupled Land Surface Hydrologic Model at a Critical Zone Observatory, *Journal of Hydrometeorology*(2013).

Shi, Y., K. J. Davis, C. J. Duffy, and X. Yu (2013b), Development of a Coupled Land Surface Hydrologic Model and Evaluation at a Critical Zone Observatory, *Journal of Hydrometeorology*, *14*(5).

Šimůnek, J., N. J. Jarvis, M. T. Van Genuchten, and A. Gärdenäs (2003), Review and comparison of models for describing non-equilibrium and preferential flow and transport in the vadose zone, *Journal of Hydrology*, *272*(1), 14-35.

Steefel, C. (2009), CrunchFlow software for modeling multicomponent reactive flow and transport. User’s manual, *Earth Sciences Division. Lawrence Berkeley, National Laboratory, Berkeley, CA. October*, 12-91.

Steefel, C. I., and A. C. Lasaga (1994), A Coupled Model for Transport of Multiple Chemical-Species and Kinetic Precipitation Dissolution Reactions with Application to Reactive Flow in Single-Phase Hydrothermal Systems, *Am J Sci*, *294*(5), 529-592.

Steefel, C. I., et al. (2014), Reactive transport codes for subsurface environmental simulation, *Computational Geosciences*, 1-34.

Van Genuchten, M. T. (1980), A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, *Soil science society of America journal*, *44*(5), 892-898.

Voss, C. I. (1984), *A finite-element simulation model for saturated-unsaturated, fluid-density-dependent ground-water flow with energy transport or chemically-reactive single-species solute transport*, US Geological Survey.

Walsh, M., S. Bryant, R. Schechter, and L. Lake (1984), Precipitation and dissolution of solids attending flow through porous media, *AIChE Journal*, *30*(2), 317-328.

Xu, T., J. Samper, C. Ayora, M. Manzano, and E. Custodio (1999), Modeling of non-isothermal multi-component reactive transport in field scale porous media flow systems, *Journal of Hydrology*, *214*(1), 144-164.

Yeh, G., and V. Tripathi (1989), A critical evaluation of recent developments in hydrogeochemical transport models of reactive multichemical components, *Water resources research*, *25*(1), 93-108.

Zysset, A., F. Stauffer, and T. Dracos (1994), Modeling of chemically reactive groundwater transport, *Water Resources Research*, *30*(7), 2217-2228.

# 