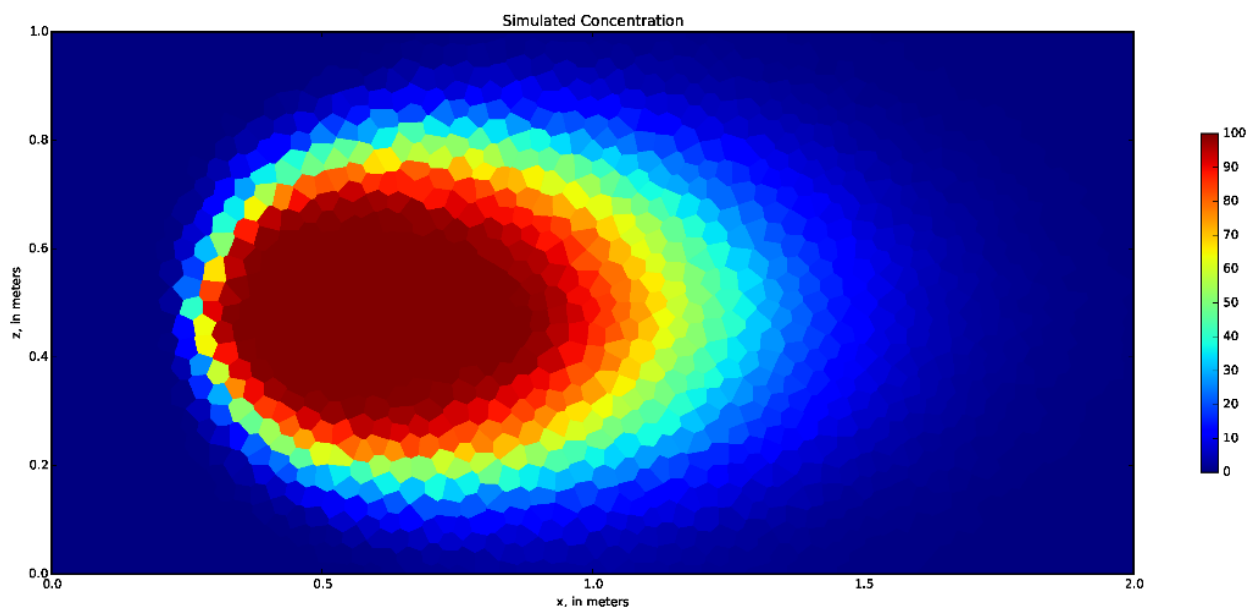


USG-TRANSPORT: TRANSPORT AND OTHER ENHANCEMENTS TO MODFLOW-USG

Version 2.8.0



Background Document

Author:

Sorab Panday, PhD

21 July, 2026

USG-Transport Version 2.8.0: Transport and other Enhancements to MODFLOW-USG

By
Sorab Panday

Suggested Citations:

Panday, S., 2026; USG-Transport Version 2.8.0: Transport and Other Enhancements to MODFLOW-USG, GSI Environmental, 2026
<http://www.gsi-net.com/en/software/free-software/USG-Transport.html>

Should be cited along with the original MODFLOW-USG document:

Panday, Sorab, Langevin, C.D., Niswonger, R.G., Ibaraki, Motomu, and Hughes, J.D., 2013, MODFLOW–USG version 1: An unstructured grid version of MODFLOW for simulating groundwater flow and tightly coupled processes using a control volume finite-difference formulation: U.S. Geological Survey Techniques and Methods, book 6, chap. A45, 66 p.

Cover: The cover image depicts results from a simulation of transport in a uniform flow field using an unstructured mesh. Courtesy Christian D. Langevin.



TABLE OF CONTENTS

Preface	1
Acknowledgments	3
Disclaimer	3
Abstract	5
Introduction	6
Governing Transport Equations and Formulation	9
Porosity Definitions	9
Transport Formulation for a 3-D Porous Medium	12
Mass Storage Term	13
Unconfined Transient Flow Conditions.....	14
Confined Transient Flow Conditions	15
Convertible Transient Flow Conditions.....	16
Storage Term on Soil	16
Solubility and Mass Precipitation	17
Dispersion	19
Dual Porosity Transport Formulation	22
Transport Formulation for CLN Cells	27
Immobile Component Species and equation ordering	28
Chain Decay	29
Generalized Multi-Species Kinetic Reactions	31
Bilinear Reaction Formulation	32
Multilinear Reaction Formulation.....	33
Mass – Mole Conversion	34
Density Dependent Flow and Transport Formulation	35
Heat Module of BTN Package	38
Introduction.....	38
Heat Module Formulation.....	38
Boundary Conditions for Heat Module	44
Matrix Diffusion Transport (MDT) Package	47
Formulation.....	47
Calculation of Matrix Diffusion Mass Flow.....	51



Air-Water Interface Adsorption Package for PFAS in the Unsaturated Zone.....	53
Introduction to PFAS Behavior.....	53
Adding PFAS simulation capabilities to USG-Transport.....	54
Implementation into MODFLOW USG-Transport.....	57
Numerical Treatment for Transport Equations	62
Numerical Expansion of the Transport Equation for an Unstructured Grid	62
Expansion of the Advection Term	64
Expansion of the Dispersion Term.....	65
Numerical Treatment of Dual Porosity Transport (DPT) Package	74
Numerical Treatment of Density Dependent Flow (DDF) Package.....	74
Numerical Treatment for CLN Transport	75
Numerical Treatment of Matrix Diffusion Transport Package	75
Output Files from MDT Package.....	77
Starting an MDT Run with Non-zero Matrix Concentrations	77
Numerical Treatment of Localized Flow Errors in Transport Solution	78
Transport Implementation and Program Design.....	79
Program Structure	79
Internal Array Storage and Precision of Variables.....	82
Model Input and Output	83
Input of Transport Boundary Conditions.....	83
Transport Output Files	84
Guidance for Using the BCT Process.....	85
General Considerations for Grid Design and Temporal Discretization.....	85
Transport Solution Options	87
Additional Considerations for Transport of Multiple Components	90
Other Enhancements in USG-Transport.....	90
Richards' Equation for Unsaturated Flow	91
Evapotranspiration Formulation for Solving Richards' Equation	94
CLN Package Modifications – Turbulent Flow; Well Efficiency.....	98
CLN Flow with Turbulent Flow Option.....	98
Formulation of Flow within the CLN Domain	103
Laminar Flow within the CLN domain.....	104
Turbulent Flow with the Darcy-Weisbach Equation	105
Turbulent Flow with the Hazen-Williams Equation	106
Turbulent Flow within the CLN Domain using the Manning's Equation.....	107
Numerical formulation of the laminar and turbulent CLN-CLN flux	107
Formulation of Flow between the CLN and GWF Domains	114



MODFLOW-USG-Transport

Sorab Panday

Confined Option for CLN-CLN Flow	117
Flow to Convertible Cells for CLN-CLN Flow.....	117
Cylindrical Conduit Geometry Type.....	120
Newton Raphson Linearization of CLN flow terms	122
Relation between Skin Friction and Well Efficiency.....	126
Time-Variant Materials Package (TVM v2) for MODFLOW-USG	129
Introduction and Formulation	129
Compatibility of TVM2 with Other Packages	131
Extension of TVM2 for Dual Domain Flow.....	131
Extension of TVM2 for Solute Transport Simulations	132
TVM2 Adjustments to Storage Formulation.....	132
TVM2 Adjustments to Transport Storage Formulation.....	135
Transient IBOUND Package to Simulate Excavation/Reclamation, or Well Drilling/Plugging.....	137
Solute Transport in LAK Package	137
Input for LAK package for solute transport	138
Output from LAK package for solute transport	138
Recharge (RCH) Package Modifications.....	138
Evapotranspiration (EVT) Package Modifications	139
Segmented Evapotranspiration (ETS) Package Modifications	139
DRT Package with Multiple Return Locations, GHB-RT and Transport.....	140
Sink with Return Flow (QRT) Package.....	141
Flow and Head Boundary (FHB) Package Modification.....	142
Discretization file extension	143
Sparse Matrix Solver (SMS) Package.....	144
The Truncated Newton Method.....	145
River (RIV) Package Modifications.....	147
Specified Gradient Boundary (SGB) Package.....	148
WEL Package Modification	149
IHM Integration with USG-T	150
Introduction.....	150
Capabilities and Functionality	150
Testing and History	153
IHM Availability, Details, Training.....	154
References	154
Output Control Package for Simulating Flow and Transport: Input Instructions.....	155
Introduction to Output Control Options.....	155
Adaptive time stepping.....	155
Output control flexibility.....	156
Adaptive solver settings through stress periods	156



Fast-forwarding files.....	157
Bootstrapping.....	157
Example Problems	159
Advection, Dispersion and Decay in a One-Dimensional Uniform Flow Field	159
Advection and Dispersion in a Two-Dimensional Confined Radial Flow Field	161
Transport of Solute through a Conduit within a Multi-Aquifer System.....	163
Dual Domain Transport in a One-Dimensional, Uniform Flow Field.....	165
Henry Problem for Density-Dependent Flow and Transport	166
Verification Example 1 for Heat Module	167
Verification Example 2 for Heat Module	169
Example Problems for Matrix Diffusion Transport Package	170
Example MD1: Comparison with Semi-Analytical Solution	171
Example MD2: Comparison with Embedded Low Permeability Zones	173
Example MD3: Demonstration of PCE Decay	176
Example Problem for Solute Transport in LAKE Package Verification	180
Example Problem for PFAS Adsorption on Air-Water Interface in the Unsaturated Zone Verification.....	181
Summary	184
References Cited.....	185
List of Symbols	188

Tables

Table 1. Convective Heat Transfer Coefficient for various pipe diameters and flow rates.....	43
Table 2. Parameter values for thermal or equivalent solute simulations.....	168
Table 3. Model input parameters for the simulation of matrix diffusion through a single fracture.....	171
Table 4. Model input parameters for the simulation of the sand tank.....	175
Table 5. Model input parameters for the simulation of sequential decay.....	178

Figures

Figure 1. Porosity Definitions	10
Figure 2. Dual Porosity Definitions.....	11
Figure 3. Function correlating mass precipitated out into other phases with solute concentration.....	18
Figure 4. Porosity conceptualization of a dual porosity model.....	23
Figure 5. Definition of Geometrical Properties for Various Grid Types.....	66
Figure 6. Angle between x-coordinate direction and the perpendicular to face nm.....	67
Figure 7. Schematic for computing the average x and y- direction velocities of a cell.....	68
Figure 8. Principal dispersion components of cell m across face nm.....	70
Figure 9. Cross dispersion components across face nm.....	72
Figure 10. dc/dx and dc/dy for each face.....	72
Figure 11. Flow chart showing the MDT subroutine implementation in MODFLOW-USG.....	76



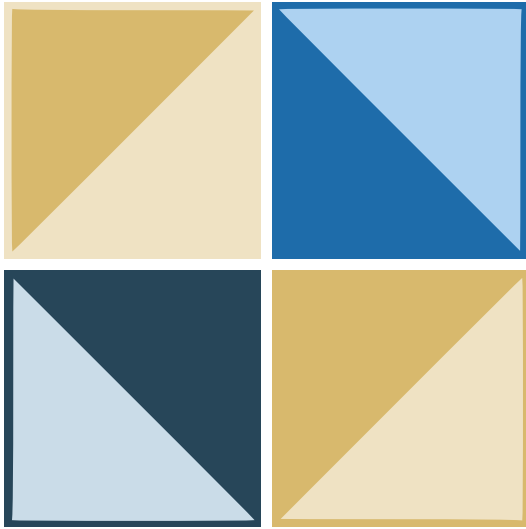
Figure 12. Flowchart for Subroutine GWT2BCT1solve.	81
Figure 13. Flowchart for Subroutine Formulate.	82
Figure 14. Different connected linear network geometries: A, a single linear feature discretized with three Connected Linear Network (CLN) cells; B, a multi-dimensional network of CLN segments.	101
Figure 15. Corrected and Uncorrected Function (f1) for Darcy-Weisbach Equation with $e = 0.01$. Note that the Manning's Equation and Hazen-Williams Equation have a different exponent but similar form.	110
Figure 16. The gradient function f2, for Manning's equation with $e = 0.0001$ – Note that the Hazen-Williams equation has similar form but different exponent.	113
Figure 17. Graph of smooth function used to express the head value used in the flow calculation to the head value of the downstream cell as it transitions from saturated to partially saturated conditions.	119
Figure 18. An example of hydraulic conductivity K_x changing over one stress period of a simulation, showing the use of linear and logarithmic (base 10) interpolation methods.	130
Figure 19. Bootstrapping approach.	158
Figure 20. Scaling approach for bootstrapping	158
Figure 14. Simulation Results in Layer 1 at 3,000 days for Transport through Conduit Example	164
 Figure Ex 1. Simulation Results for Advection in a One-Dimensional, Uniform Flow Field.	160
Figure Ex 2. Simulation Results at 20 Days for Advection, Dispersion, and Decay in a One-Dimensional, Uniform Flow Field.	161
Figure Ex 3. Comparison of Model Results with MT3D Simulation Results for Transport in a 2-D Radial Flow Field at 5,000 days.	162
Figure Ex 4. Comparison of Simulation Results with and without Cross Dispersion for Transport in a 2-D Radial Flow Field at 5,000 days.	163
Figure Ex 5. Simulation Results in Layer 1 at 3,000 days for Transport through Conduit Example.	165
Figure Ex 6. Simulation Results in Layer 1 at 3,000 days for Transport through Conduit Example, Using a Nested Grid.	165
Figure Ex 7. Concentration at Outlet for Dual Domain Transport in a One-Dimensional, Uniform Flow Field.	166
Figure Ex 8. Concentration profiles for Henry Problem Simulation.	167
Figure Ex 9. Temperature Distribution around Borehole.	169
Figure Ex 10. Simulation of matrix diffusion through a single fracture.	172
Figure Ex 11. Comparison of MODFLOW-USG MDT package output with the semi-analytical and analytical models. Note MODFLOW-USG and the semi-analytical curves are identical.	173
Figure Ex 12. Sand Tank Configuration. Based on Muskus and Falta and (2018) Figure 5. Groundwater flow is from left to right (blue arrows). Not to scale.	174
Figure Ex 13. Comparison of MODFLOW-USG MDT model (50 model cells) output with Chapman et al. (2012) MODFLOW/MT3DMS model (8,988 model cells), semi-analytical model (50 model cells), and observed concentrations.	176
Figure Ex 14. Comparison of MODFLOW-USG MDT model output with the semi-analytical model 1 year, 5 years, 25 years, and 50 years after source release.	179
Figure Ex 15. Change in lake stage and volume.	181
Figure Ex 16. Change in lake concentration over time.	181



MODFLOW-USG-Transport

Sorab Panday

Figure Ex 17 . Breakthrough of PFOA for different concentration cases compared to Lyu (2018) experimental results.	183
Figure Ex 18 . Breakthrough of PFOA for different saturation cases compared to Lyu (2018) experimental results.	184



MODFLOW-USG-Transport

Sorab Panday

PREFACE

The computer model described in this report is **MODFLOW-USG-Transport**, released by GSI Environmental. The code and documentation are currently available for downloading from the GSI Website <https://www.gsienv.com/product/modflow-usg/>. Newer versions or releases with revisions and updates will be made available for downloading from the same website. Alternatively, the user can register at the GSI Website for emails that make users aware of code updates.

MODFLOW-USG-Transport (or USG-T), includes advancements to the MODFLOW-USG software first released by the U.S. Geological Survey (USGS) in 2013. The primary functional capability that has been added in USG-T is simulation of three-dimensional chemical species transport. The transport module works seamlessly with the MODFLOW-USG groundwater flow model and is fully compatible with flow-fields generated by the Groundwater Flow (GWF)

and Connected Linear Network (CLN) Processes of MODFLOW-USG.

Other updates to MODFLOW-USG flow routines are also published with USG-T which include, among other processes or convenient features, unsaturated zone flow and transport, turbulent formulations for CLN flow, and density dependent flow transport coupling.

Testing of the new USG-T routines indicated that the code yields reliable, mass conserved results for a wide variety of problems. The user, however, should be aware that the accuracy and efficiency of transport computations can be significantly affected by gridding and time-stepping considerations among other conceptual and numerical factors. A recorded webinar on transport fundamentals, specifically with USG-T, is available commercially from the GSI Environmental website at <https://www.gsienv.com/webinars/webinar-3-simulation-of-solute-and-heat-transport-with-usg-transport-and-other-general-transport-simulation-fundamentals/>.



MODFLOW USG-Transport

Sorab Panday

[This page intentionally left blank]



ACKNOWLEDGMENTS

Development efforts for USG-T were funded in part via a research project from GSI Environmental. Thanks to Christian D. Langevin and Alden M. Provost for their collaboration and continued input to this work and preliminary review of the document. Thanks, are also due to Anthony (Tony) Daus for supporting this effort. Finally, various people have contributed to the development of different modules to USG-T. These contributions are acknowledged in the respective sections.

Disclaimer

The authors of this work and GSI Environmental make no warranties and disclaim liability for **all uses of the software and documentation**.



MODFLOW USG-Transport

Sorab Panday

[This page intentionally left blank]



ABSTRACT

This report presents continued development to the MODFLOW-USG software released by GSI Environmental as USG-Transport. The code primarily includes a transport model simulator for unstructured grids. The transport model is designed to simulate heterogeneous, three-dimensional solute transport of multiple chemical constituents in the subsurface, caused by advection; hydrodynamic dispersion (which includes both mechanical dispersion and molecular diffusion); mixing (or dilution) from fluid sources; simple reactions including first-order and zero-order decay; linear or Freundlich equilibrium adsorption; and non-equilibrium retardation via a dual-porosity representation. Density dependent flow and transport processes are also accommodated as is heat transport, matrix diffusion, and dissolution/precipitation of solutes to/from the solution. Other enhancements include saturated/unsaturated flow, dual porosity flow, and water distribution modules along with several convenient features and enhancements that assist with smooth modeling of flow and transport in the subsurface.

The USG-Transport model is fully compatible with and integrated into MODFLOW-USG. MODFLOW-USG (Panday et al., 2013), released by the U.S. Geological Survey, is a three-dimensional groundwater flow model that uses implicit Control Volume Finite Difference (CVFD) methods to solve for steady-state and/or transient flow on unstructured grids. MODFLOW-USG includes a Groundwater Flow (GWF) Process and a Connected Linear Network (CLN) flow process. The GWF Process solves for flow in a three-dimensional porous medium. The CLN flow process solves for flow through a network of 1-dimensional cells representing wells, rivers, or fracture networks, interacting with the GWF process. Transport is fully compatible with flow in the GWF domain and within the CLN domain and accommodates solute migration between the CLN domain and the GWF domain as part of the solution. The numerical schemes are selected to provide optimal solution speed and accuracy pertinent to most subsurface transport simulation objectives.

This report includes a description of the theoretical basis of the model, the numerical methods, data-input requirements, output options, results of benchmark simulation problems, and an input instructions guide. The code was evaluated for several problems for which exact analytical solutions are available and by benchmarking against other numerical codes for select complex problems for which exact analytical solutions were not readily available. These test results indicate that the model is accurate for a wide range of conditions and yields minimal numerical



dispersion for advection-dominated problems. Mass-balance errors were negligible for the problems that were tested.

INTRODUCTION

This report primarily describes and documents a solute transport model for calculating transient changes in the concentration of solutes in a three-dimensional groundwater flow field. The model was designed to be compatible with the MODFLOW-USG groundwater flow model released by the USGS (Panday et al. 2013) to simulate flow through the subsurface (the GWF Process domain) and through a network of interacting linear segments (the CLN domain) representing discrete preferential flow paths interacting with the subsurface porous medium system. Steady-state or transient flow-fields generated by solution to the flow equation are used by the transport model to evaluate solute transport for one or more chemical components. The solute transport routines are incorporated as a Block-Centered Transport (BCT) Process to the MODFLOW-USG code. Other processes, packages and enhancements made to the software are also discussed in this report. These enhancements form the MODFLOW-USG-Transport, or USG-Transport code.

The transport model is fully integrated into the MODFLOW-USG Groundwater Flow model. MODFLOW-USG solves the groundwater flow equation on unstructured grids, and the reader is referred to the documentation for that model and its subsequent modules for complete details (Panday et al., 2013). In this report it is assumed that the reader is familiar with the MODFLOW family of codes in general, and with MODFLOW-USG in particular. The BCT package of USG-Transport evaluates transport in the subsurface porous medium as well as through the connected linear network domain associated with the CLN Process of MODFLOW-USG. The transport package is therefore built on the same unstructured grid setup used in the flow modules including the porous matrix cells and CLN cells.

The purpose of the transport model is to compute the temperature (for heat transport) or concentration of a chemical species (for solute transport) in the subsurface environment at any specified place and time. Significant processes that affect chemical concentrations in an aquifer include: (1) advective transport, where dissolved chemicals move and mix with steady/unsteady, uniform/non-uniform ground water flow fields; (2) hydrodynamic dispersion (i.e., mechanical dispersion and molecular diffusion) which causes the paths of dissolved chemicals to diverge and spread from the average direction and speed of groundwater flow; (3) fluid sources, where water of one composition is introduced into and mixes with water of a different composition; (4)



reactions, in which some amount of the solute is added to or removed from the ground water because of chemical, biological, and physical processes; (5) equilibrium linear or non-linear adsorption onto the porous medium; (6) precipitation and dissolution of solutes; and (7) non-equilibrium, dual porosity processes. These processes have been implemented into the current version of the BCT package. A Density Driven Flow (DDF) package has further been provided to evaluate cases where solute concentration and associated fluid density impact the flow field.

A primary objective of environmental solute transport model simulations is to assess contaminant mass and migration through space and time within a heterogeneous subsurface, for design of remediation/containment systems or for evaluating long term risk. The numerical schemes of the BCT package are accordingly selected to be applicable for a wide range of situations. A mass conservative, implicit, control volume finite difference (CVFD) scheme is used to solve the divergence form of the advection-dispersion transport equation – solution to the convective form of the transport equation need not conserve mass. The implicit schemes are unconditionally stable and do not impose a restriction on the time-step size as do explicit schemes. Therefore, for transient flow and transient transport simulations, the time-step size for transport is the same as that used for flow (no sub-steps are required or performed for the transport solution). A user-defined implicitness factor has also been implemented, to allow a Crank-Nicolson scheme for higher order time discretization accuracy with the advective and dispersive terms. The implicitness factor can therefore vary from the Crank-Nicolson condition of 0.5 to the fully implicit scheme with a factor of 1.0. This allows the solution to be stable regardless of time-step size. Also, the transport simulation immediately follows solution for flow at a given time-step though options are provided for sequential simulations of transport using flow-fields from a previous simulation of flow.

For transient transport situations in steady-state flow conditions, the time stepping information provided in the OC package is used to guide transport time-stepping. An implicit in time, Total Variation Diminishing (TVD) scheme is used to control numerical dispersion in the advection term. The dispersion terms are formulated implicitly for components along the principal axis. The cross-dispersion terms are optionally formulated in a semi-implicit manner on the right-hand side vector, with iterative updates. The groundwater flow domain of *USG-Transport* further includes the capability to simulate dual porosity flow processes; therefore, the associated transport process includes advection as well as diffusion into the dual porosity matrix blocks. Hence, transport within three scales of fracturing can be simulated by including discrete fractures via the CLN domain within a dual porosity simulation. Finally, for the transport solution, the user is



able to specify a subset or window of active cells within the unstructured grid used to solve the flow equation to enhance overall efficiency of the transport solution by avoiding calculation efforts where they are not required.

The BCT Process is a basic tool that is applicable to a wide range of field problems involving solute and heat transport. However, the user should first become aware of the assumptions and limitations inherent in the model, as described in this report. Also, it is critical to correctly conceptualize and discretize the model to make good transport predictions, and there will be situations in which the model results could be inaccurate or model operation inefficient. The report includes brief guidelines for recognizing and avoiding such issues.

The computer program is written in FORTRAN and has been developed in a modular style similar to the MODFLOW-USG model code. The model is compatible with most modules of the MODFLOW-USG code; however, it does not compute transport with the stream-flow routing processes of the STR7 and SFR packages and does not affect porosity changes caused by deformation simulated with the Subsidence Package. Furthermore, for transient flow conditions, the transport model may be incompatible with the drying and rewetting schemes of MODFLOW-2005 (Harbaugh, 2005) which are implemented into MODFLOW-USG. Specifically, it is not possible to include consideration of adsorbed mass within dry cells that may subsequently re-saturate. These issues do not occur with the NWT scheme for solution to the flow equation. Also it is assumed by the BCT process that flow is not affected by transport; however, there is a density dependent flow (DDF) Package included with USG-Transport which accommodates density effects.

The types of reactions incorporated into the BCT process are restricted to those that can be represented by zero- or first-order reaction rates such as radioactive decay. Reactions with soil grains via instantaneous, reversible, adsorption/desorption onto the soil grains are governed by a linear distribution coefficient or the Freundlich isotherm. An option to include a dual porosity formulation (with mobile and immobile regions occurring at a sub-grid scale) allows for simulating diffusive transport into dead zones or low conductivity materials within a simulation grid-block to include effects of localized heterogeneity. The same storage, adsorption and reaction processes of the mobile domain are also included in the immobile domain.

This report includes a detailed description of the formulation and numerical methods used to solve the solute-transport equation and the additional customized modules developed for USG-



Transport. This is followed by sections discussing the program design, guidance on usage, example problems and input instructions. The data requirements, input format specifications, program options, and output formats of the code are structured in a general manner that is compatible with available data for many field problems. Existing MODFLOW-USG input and output modules, formats and styles are used in developing the BCT process options and features to maximize compatibility and ease of use.

GOVERNING TRANSPORT EQUATIONS AND FORMULATION

The BCT process solves for multi-species component transport in a three-dimensional, non-uniform, steady/unsteady flow field. The flow equation for GWF and CLN domains is first solved to provide the cell-by-cell fluxes required by the BCT process for simulation of chemical transport. For steady-state flow conditions, the flow equations are solved once to achieve the steady-state flow-field, followed by simulation of solute transport for the entire time period of simulation. For a transient flow field, the transport equation solution follows flow equation solution for every time step. The flow equations and solution schemes are detailed in the MODFLOW-USG document (Panday et al, 2013). Details of the chemical transport formulation and solution schemes are documented here. The governing transport equations and associated simulated processes are first discussed for the GWF domain followed by the formulation for the CLN domain. Note that the GWF domain can also be conceptualized by the dual porosity formulation in *USG-Transport*.

Porosity Definitions

It is useful at this stage to provide the various definitions of porosity that will be used in conceptualizing and formulating a general form of the governing equations for transport within a porous medium, as used in this document. **Figure 1** shows the relevant schematic. The total volume within a porous medium is occupied by soil grains and pore space (voids). The volume of voids per total volume of the material is called the total porosity (or simply, the porosity) of the medium, ϕ . For a saturated system, the total pore space is occupied by water that is mobile or is in direct contact with mobile water (collectively termed as the effective water space), and water that is isolated from the active flow and transport system (the ineffective pore space).

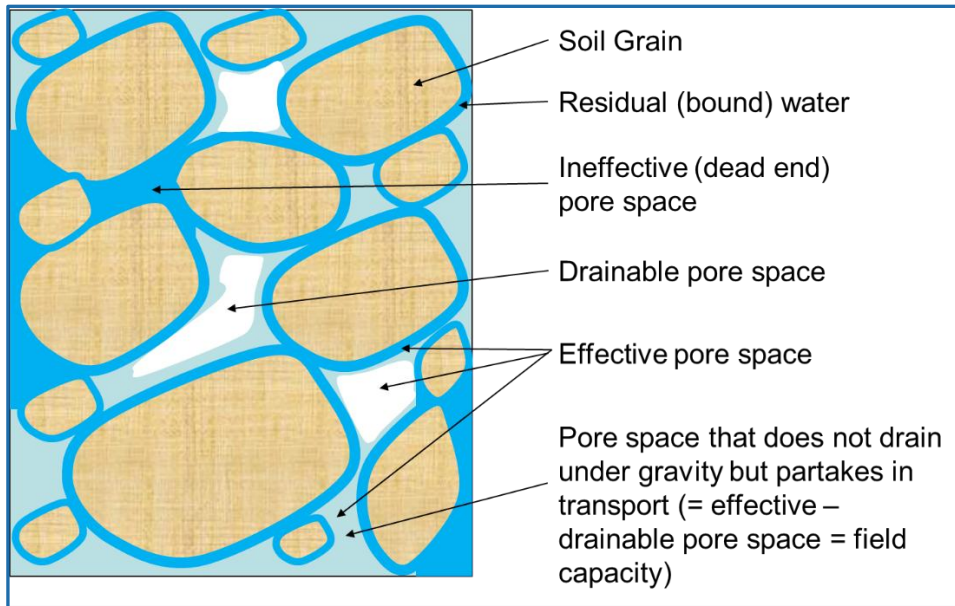


Figure 1. Porosity Definitions

The effective pore space is termed such because it is “effective” in facilitating storage and migration of chemical species. The volume of this pore space that participates in transport per total volume of the medium is called the effective transport porosity, ϕ_e . The remaining pore volume (the dead end pores) does not participate in transport. This volume of the ineffective water space per total volume of the medium is the ineffective porosity, ϕ_{ie} . Finally, if water levels are lowered, a portion of the mobile water space (the drainable pores) can drain under gravity. The volume of drainable pores per total volume of the material is the specific yield, S_y , as used in the unconfined flow simulations of MODFLOW. The field capacity moisture content, ϕ_{fc} , defines the volume of water per total volume of the material after gravity drainage has ceased. Based on this terminology, the following relation can be defined:

$$\phi = \phi_e + \phi_{ie} = S_y + \phi_{fc} + \phi_{ie} \quad (1)$$

It is noted from the definitions above, that $\phi \geq \phi_e \geq S_y$. Also, the effective porosity is equal to the specific yield plus the field capacity. Although some transport solutions may allow users to enter an effective porosity that is less than specific yield, this can lead to conceptual inconsistencies between the flow and transport solution as noted in the above equation.

For fully saturated or confined conditions, the entire effective pore space participates in chemical species transport. By definition, no chemical species enters or leaves the ineffective pore space (the pore space that does not contribute to dissolved species storage or transport) and hence



the total porosity is not a factor in transport computations. Note that water bound to the soil grains or within capillarity but which is accessible to mobile water is considered as part of the effective pore space as species is stored within all of the effective pore space including in residual water and water bound onto soil grains. This is significant as adsorption onto the soil would necessarily be mediated through the bound water. At the pore scale, it is therefore assumed that diffusion processes instantly equilibrate the species mass from the drainable or flowing water to bound water within the effective pore space, and subsequently onto soil grains when adsorption is also considered. Note that this definition is for the classical transport equation in porous medium.

A “dual porosity” or “dual domain” formulation includes small-scale processes within the transport formulation. Specifically, the porous medium system is conceptualized to include a mobile (dynamic) region through which flow occurs, and an immobile (stagnant) region where flow does not occur, but solutes may diffuse in and out. Each region has the same porosity definitions as discussed earlier; however, an additional variable is included that defines how much of the domain is the dynamic region, versus how much is the static region. **Error! Reference source not found.** shows the conceptual representation of a dual porosity model.

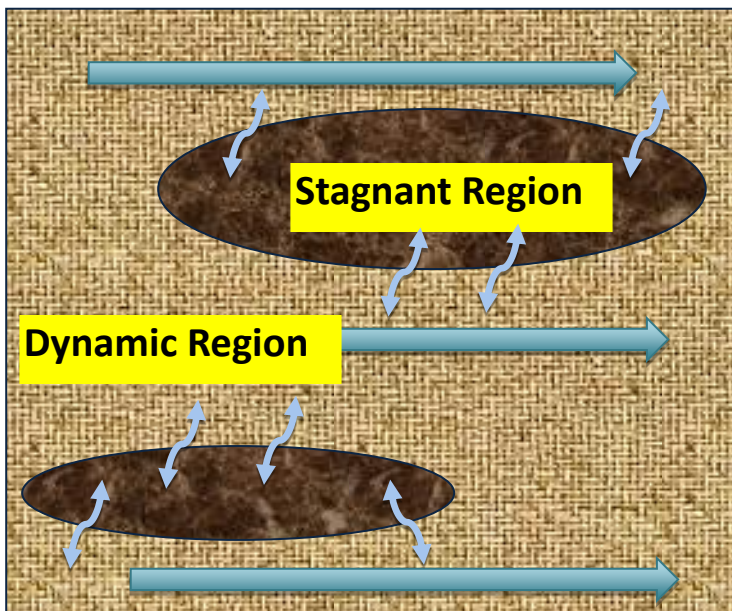


Figure 2. Dual Porosity Definitions.



Transport Formulation for a 3-D Porous Medium

The groundwater transport equation in an unsteady flow field (Zheng and Wang, 1999) is adapted for the BCT process as:

$$\frac{\partial(M)}{\partial t} = \frac{\partial}{\partial x_i} \left[\theta_w D_{ij} \frac{\partial c}{\partial x_j} \right] - \frac{\partial}{\partial x_i} [v_i c] - [\lambda_w \theta_w c + \lambda_s \rho_b c_s] - [\mu_w \theta_w + \mu_s (1 - \phi_e)] + \dot{M} \quad (2)$$

Where

M is the total mass per unit volume, of a component species in water and on soil [M / L^3],

t is the time [T],

x_i (i=1,2,3) are the principal coordinate directions, [L],

D_{ij} is the apparent hydrodynamic dispersion tensor, [L^2/T],

c is the concentration of a component species in water [M_w / L^3], where M_w is the mass of a component species in water,

v_i is the Darcy flux in direction x_i , [L/T],

ϕ_e is the effective (transport) porosity [],

θ_w is the moisture content, []. Its computation is discussed along with the discussion of the storage term, below,

ρ_b is the bulk density of the porous medium [M_s / L^3]; where M_s is the mass of soil solids,

c_s is the adsorbed concentration of component species [M_a / M_s], where M_a is the mass of a component species adsorbed onto the soil,

λ_w is the first-order decay coefficient in water [T^{-1}], = $\ln(2) / t_{1/2w}$ where $t_{1/2w}$ is the half-life of the chemical species in water,

λ_s is the first-order decay coefficient on soil [T^{-1}], = $\ln(2) / t_{1/2s}$ where $t_{1/2s}$ is the half-life of the chemical species adsorbed on the soil,

μ_w is the zero-order decay coefficient in water [$M / (L^3 T)$],



μ_s is the zero-order decay coefficient on soil [$M / (L^3T)$], and

$\dot{M}$ is a source term (negative for sink) for the component species, representing other source and sink processes [$M / (L^3T)$].

A source can be represented as the inflow flux times the dissolved concentration of the chemical species in the inflowing water. Thus, $\dot{M} = Qc'$ where Q is the fluid flux and c' is the concentration of inflow water at the boundary. Thus, all inflow boundary conditions for the flow simulation of *MODFLOW-USG* also include input for species concentrations which are multiplied by the flux to provide the mass input into the modeled system. The species concentration of fluid leaving the aquifer at fluid sinks is commonly assumed to be the same as concentration of the fluid in the aquifer (that is, $c' = c$ for $Q < 0$). Thus, if species concentration is provided in the boundary input file at an outflow boundary, it will be ignored. Note that for evapotranspiration boundaries, solutes are generally assumed to be left behind as water evaporates unless there is also solute uptake by plants or volatilization considerations. Note that for certain boundaries, it is not known a priori whether a cell would be inflow or outflow (for instance, GHB, RIV, STR, LAK). However, other boundary conditions (DRN, EVT) are exclusively outflow boundary types and species concentration input is not required by the code for these packages.

MASS STORAGE TERM

The storage term for component species considers dissolved mass storage in water and adsorbed mass storage on soil. For transient transport in a steady-state flow-field, the storage term is written in the standard form as:

$$\frac{\partial(M)}{\partial t} = \frac{\partial(\phi_e S_w c)}{\partial t} + \frac{\partial(\rho_b c_s)}{\partial t} + \frac{\partial(M_p)}{\partial t} \quad (3)$$

Where S_w is the saturation of water or the fraction of the grid-block height that is below the water table for unconfined groundwater flow (for confined groundwater flow, S_w is unity), [].

The water content is defined as:

$$\theta_w = \phi_e S_w \quad (4)$$

And M_p is the mass of solute that precipitates out of solution above a solubility limit.



For transient transport simulations in a transient flow-field, the storage term for chemical species mass in the water phase needs special considerations within the *MODFLOW* framework. There are two reasons for this. First, for unsaturated or unconfined situations, the effective porosity for transport (pores available to storage and transport of solute) may be different from the specific yield (drainable porosity) used in the flow solution. Secondly, *MODFLOW* handles storage of water differently, for the various transient flow simulation options depending on whether a cell is unconfined, confined, or convertible. The contribution and discussions with Christian D. Langevin regarding the various mass conserved numerical formulations for the transport storage term are acknowledged here.

Unconfined Transient Flow Conditions

When the effective porosity for transport is different from the specific yield used for transient unconfined flow situations for any porous medium cell, the imbalance in net flow causes a change in saturation within the drainable pore space of the cell and not within the effective transport pore space. The resulting chemical species imbalance occurs within this volume of water defined by the specific yield and not the total volume of water within the effective pore space. However, within any cell, the chemical is contained in all the water within the entire effective pore space; the remainder from drainable pore space, defined above within this context as the field capacity, being assumed to be filled completely with water. **Figure 1** shows this situation where only a portion of the effective porosity which partakes in solute transport, is drainable. To maintain mass conservation for such a situation, the effective porosity is split into two portions – a drainable yield portion (S_y) within which saturation changes occur as per the flow equation solution, and the field capacity portion which also contains the chemical species but does not drain. Thus, the storage term is written as:

$$\frac{\partial(M)}{\partial t} = S_y \frac{\partial(S_w c)}{\partial t} + \phi_{fc} \frac{\partial(c)}{\partial t} + \frac{\partial(\rho_b c_s)}{\partial t} + \frac{\partial(M_p)}{\partial t} \quad (5)$$

Where the field capacity and specific yield are assumed to be constant and can be pulled out of the temporal derivatives. The water content in this context is the sum of the drainable water content and the field capacity giving:

$$\theta_w = S_y S_w + \phi_{fc} \quad (6)$$



Note that equations (5) and (6) reduce to equations (3) and (4) respectively, for a temporally constant porosity and when $\phi_e = S_y$. Also note that the mass conserved formulation requires that $\phi_e \geq S_y$ for transport computations in transient flow situations.

Confined Transient Flow Conditions

The specific storage of the aquifer is used by the *MODFLOW* framework for the compressible storage term for confined groundwater flow. The specific storage represents the change in the amount of water in storage per unit change in head and is inherently a combination of porous medium and water compressibility. However, *MODFLOW* does not adjust the specific yield or water density in response to matrix and water compressibility and the specific storage is treated simply as additional storage available due to the compressibility effects. Chemical species mass is associated with this water in additional storage. Thus, the species mass storage term for transport under transient confined flow conditions can be written in a consistent manner as:

$$\frac{\partial(M)}{\partial t} = \phi_e \frac{\partial(c)}{\partial t} + S_s c \frac{\partial(h)}{\partial t} + \frac{\partial(\rho_b c_s)}{\partial t} + \frac{\partial(M_p)}{\partial t} \quad (7)$$

Where h is the hydraulic head in the cell. Since the confined condition is fully saturated (i.e., $S_w = 1$), the effective water content is equal to the effective porosity. Thus, for this situation:

$$\theta_w = \phi_e \quad (8)$$

Note that the head is a reference head and thus the term $S_s h c$ represents an absolute value of chemical species mass held in compressible storage per unit volume at the reference head value. *MODFLOW* computes changes in mass storage and therefore, the absolute starting value does not affect results. For computing the initial mass in compressible storage, the top of the cell is taken as the datum from which its reference head is computed since the system becomes unconfined and equations 5 and 6 are applicable for convertible systems with heads below this value. In other words, any component species held in compressible storage term is considered as zero when a cell becomes unconfined. Also note that water and porous matrix are considered incompressible for transport considerations as the pore volume and liquid density are not changed in a *MODFLOW* simulation and, consistent with *MODFLOW*, the compressible storage component is treated simply as additional storage available due to the compressibility effects that should not otherwise have an impact on the solute concentrations.



Convertible Transient Flow Conditions

For convertible cases (confined and unconfined), both pore storage and compressible storage terms are applied by the flow solution. For this case, the storage term of the transport equation is an extension of that for the flow equation, with equations 5 and 6 used for unconfined conditions and equations 7 and 8 for confined conditions. If a cell converts from confined to unconfined (or vice versa), the portion of the change in head above the top of the cell contributes to confined storage while the portion of the change in head below the top of the cell contributes to unconfined storage. For the upstream weighted formulation, both terms are used in combination and the associated transport mass storage term can be written in a consistent manner as:

$$\frac{\partial(M)}{\partial t} = S_y \frac{\partial(S_w c)}{\partial t} + \phi_{fc} \frac{\partial(c)}{\partial t} + S_w S_s c \frac{\partial(h)}{\partial t} + \frac{\partial(\rho_b c_s)}{\partial t} + \frac{\partial(M_p)}{\partial t} \quad (9)$$

The water content for this case is defined by equation (6) since the effective pore space is segregated into a drainable pore space and the field capacity. Note that the compressible storage term vanishes when a cell is dry (and S_w goes to zero). Equation 9 provides the most general form of the storage term for the transport equation within the *MODFLOW* framework. It reduces to equation 7 for confined transient flow-field conditions (i.e., when $S_w = 1$), to equation 5 for transient unconfined flow-field conditions, and to equation 3 for steady-state flow-field conditions or transient flow-field conditions with effective porosity equal to the specific yield.

Storage Term on Soil

Equilibrium is assumed between adsorbed and dissolved concentration of chemical species. The nonlinear Freundlich isotherm relates the adsorbed concentration of a species to its concentration in water as

$$c_s = k_d c^e \quad (10)$$

where

k_d is the adsorption coefficient [$L^3 M^{-1}$] and

e is the nonlinear Freundlich exponent [].

Note that a linear adsorption isotherm is obtained by setting the exponent e to unity. Note that use of the nonlinear adsorption isotherm with linear decay on soil causes the net decay to also



be nonlinear since the equivalent decay term, $\lambda_s \rho_b c_s = \lambda_s \rho_b k_d c^e$, is nonlinear in terms of the water phase concentration.

The Langmuir isotherm for adsorption is written as:

$$c_s = \frac{k_d}{1 + \eta c_w} c_w \quad (11)$$

Where η is the Langmuir coefficient. Either the linear, Freundlich, or Langmuir coefficients may be used for adsorption on soil.

The soil storage term occurs over the entire volume of the finite volume cell. For unconfined simulations, if the saturated thickness is much smaller than the cell thickness, the adsorbed mass of a solute may be present only over the saturated thickness of the cell (i.e., below the water table). To accommodate this, an option is provided called ONLY_SATADSORB which uses the saturated thickness of the first time step of the first iteration as the volume over which adsorption occurs instead of using the entire cell thickness for the adsorption volume. This option is appropriate for steady-state flow conditions but could be an approximation for transient flow conditions where the water table moves up and down the cell thickness, as the adsorbed mass will be over the saturated cell thickness of that first time-step of the first stress period. Varying the saturated cell thickness to account for the effect of transient flow conditions on the adsorbed volume is not done because that causes mass balance problems. If a transient flow situation has large water table fluctuations then use of multiple layers to discretize the system may be considered to improve accuracy of volumes of soil participating in adsorption.

Solubility and Mass Precipitation

The last term on the right-hand side of equation (3) determines dissolution of solutes from another phase or precipitation of solutes out of the water phase. Dissolution / precipitation of solutes is included in the transport formulation by means of equilibrium partitioning into another phase once the solubility limit is reached. Thus, solutes precipitate out of solution when the concentration exceeds the solubility limit. A linear partitioning coefficient further allows for a distribution of the solute between water and the other phase into which it precipitates. Note here that the word “precipitate” indicates that it moves out of the aqueous phase into a non-aqueous phase. Note also that the presence or absence of this mass as it moves out or into the aqueous phase does not affect the flow field in the flow equation.



The approach to implementing solubility into the solution is to include mass in another phase precipitated out of solution into the calculations. A functional form shown in **Figure 3** correlates the mass of precipitated solutes per unit total volume, with concentration within the water phase.

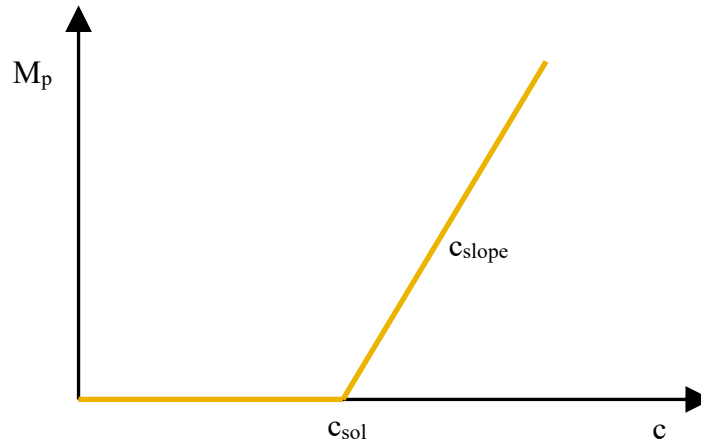


Figure 3. Function correlating mass precipitated out into other phases with solute concentration.

Thus, when solute concentration is below the solubility limit c_{sol} , there is no mass partitioned out into the other (precipitated) phase. At concentrations exceeding c_{sol} , there is a linear mass partitioning of solute between the precipitated phase mass and the water phase concentration. As concentration increases beyond the solubility limit, the precipitated phase mass also increases linearly, with a slope of c_{slope} .

If for a solute, all excess mass precipitates out above c_{sol} such that solute concentration cannot exceed c_{sol} , then the slope of the line above c_{sol} will be infinity (90 degrees); therefore, if the solute concentration tries to increase further, the species precipitates out of solution. Numerically, a large number should be implemented which does not cause truncation errors in tracking the mass. A slope of 100 (89.43 degrees) to 1,000 (89.94 degrees) should be adequate to prevent concentrations from exceeding the solubility limit by significant amounts. However, that slight exceedance of the solubility limit helps to keep track of the mass that was precipitated out of solution on the functional curve.

The solubility / precipitation functionality may also be used to place an initial precipitated mass of a solute within a domain and allow it to dissolve within groundwater. The known mass may be input into the model as an equivalent concentration above the solubility limit on the functional curve and may be computed as $c_i = (M_{pi} / c_{slope}) - c_{sol}$, where c_i is the initial concentration,



M_{pi} is the initial known mass of component in the precipitated phase per volume of cell i , C_{slope} is the slope of the functional curve.

The solubility parameters may be input as a uniform value for a species or as variable in space. A species may dissolve or precipitate out of solution by different amounts in different locations requiring different parameters spatially. Where no precipitation or dissolution occurs, the solubility limit c_{sol} may be put to a very large number, or the slope C_{slope} may be set to zero.

DISPERSION

The hydrodynamic dispersion tensor D_{ij} for a generalized anisotropic porous medium is expressed by Bear (2012) as:

$$D_{ij} = a_{ijkm} \frac{v_k v_m}{v} f(Pe, \delta) \quad (12)$$

where

a_{ijkm} is the geometric dispersivity of the medium, ($i, j, k, m = 1, 2, 3$ for the three principal coordinate directions), v is the average velocity, Pe is the Peclet number, δ is the ratio of length characterizing an individual channel of a porous medium to its hydraulic radius, and $f(Pe, \delta)$ is a function which introduces the effect of transfer by diffusion between adjacent streamlines at a microscopic level (Bear, 2012). There are a total of 81 terms that represent the medium's geometrical dispersivity, where 36 of them are non-zero and 21 non-zero moduli are independent (Bear, 2012). The geometric dispersivity is expressed in a general form as (Bear, 2012):

$$a_{ijkm} = a_{IJ} \delta_{ij} \delta_{km} + \frac{a_I - a_{IJ}}{2} (\delta_{ik} \delta_{jm} + \delta_{im} \delta_{jk}) \quad (13)$$

where

a_I is the dispersivity modulus in the longitudinal direction along I , a_{IJ} is the dispersivity modulus in the transverse direction to plane IJ , ($I, J = 1, 2, 3$ for the three principal coordinate directions) and δ_{ij} is the Kronecker delta which is unity when $i = j$, and zero otherwise. For an isotropic medium, equation 13 reduces to only two dispersivity moduli. For this case, the derivation of Bear (1988) in two dimensions can be extended into the third dimension by inclusion of subscript "3" as:



$$a_{1111} = a_{2222} = a_{3333} = a_L \quad (14)$$

$$a_{1122} = a_{2211} = a_{1133} = a_{3311} = a_{2233} = a_{3322} = a_T \quad (15)$$

$$a_{1212} = a_{1221} = a_{2112} = a_{2121} = a_{1313} = a_{1331} = a_{3113} = a_{3131} = a_{2323} = a_{2332} = a_{3223} = a_{3232} = 0.5(a_L - a_T) \quad (16)$$

Where a_L is the longitudinal dispersivity and a_T is the transverse dispersivity. Furthermore, the components do not change with rotation of the coordinate axis and assuming that $f(Pe, \delta) = 1$ (Bear, 2012), gives the isotropic dispersion relationship of Scheidegger (1961) as:

$$D_{ij} = a_T \bar{v} \delta_{ij} + (a_L - a_T) v_i v_j / \bar{v} \quad (17)$$

This relationship has been used extensively even for anisotropic systems due to the difficulty of determining and parameterizing a fully anisotropic dispersivity tensor.

For horizontally stratified systems with flow along the direction of stratification, it was noted that the isotropic formulation provided plumes with excessive dispersion in the vertical direction which caused unrealistic simulated plume shapes and that a vertical transverse dispersivity two orders of magnitude lower than the horizontal direction value was required to match the Where a_L is the longitudinal dispersivity and a_T is the transverse dispersivity, shape of a plume in California (Burnett and Frind, 1987). The situation was rectified by generalizing the relationship of Scheidegger for a Cartesian system to include a separate transverse vertical dispersivity modulus from the transverse horizontal dispersivity value. Use of a lower value for the vertical dispersivity modulus restricts spreading in the vertical direction thus simulating more realistic plumes.

For general anisotropic conditions in the horizontal and vertical directions, the equation of Scheidegger can be extended in a similar manner to Burnett and Frind (1987), to account for stratification effects in each coordinate direction. Thus, the dispersivity components along the three principal axis in equation 14, or in the transverse plane in equation 15 are not all equal to each other. In Cartesian coordinates the anisotropic dispersion equations can be written as:

$$\theta_w D_{xx} = (a_{Lx} v_x^2 + a_{Txy} v_y^2 + a_{Txz} v_z^2) / |v| + \theta_w D^* \quad (18)$$



$$\theta_w D_{yy} = (a_{Txy} v_x^2 + a_{Ly} v_y^2 + a_{Tyz} v_z^2) / |v| + \theta_w D^* \quad (19)$$

$$\theta_w D_{zz} = (a_{Txz} v_x^2 + a_{Tyz} v_y^2 + a_{Lz} v_z^2) / |v| + \theta_w D^* \quad (20)$$

where

D_{xx} , D_{yy} , and D_{zz} are the principal components of the dispersion tensor, $L^2 T^{-1}$,

a_{Lx} , a_{Ly} , a_{Lz} , a_{Txy} , a_{Tyz} , and a_{Txz} are the longitudinal and transverse dispersivity values for each of the respective coordinate directions, L ,

v_x , v_y , and v_z are the components of the Darcy velocity vector along the x, y, and z axes, LT^{-1} ,

D^* is the effective molecular diffusion coefficient in water, $L^2 T^{-1}$, and

$|v|$ is the net magnitude of the velocity vector computed as

$$|v| = \sqrt{v_x^2 + v_y^2 + v_z^2} \quad (21)$$

The first term of equations 18, 19, and 20 expresses the mechanical dispersion, and the last term of these equations expresses the diffusion component of the total hydrodynamic dispersion. Also, the transverse dispersivity moduli of these equations are symmetric and thus, for example, the modulus for transverse to z in the x-direction is the same as for transverse to x in the z-direction ($a_{Tzx} = a_{Txz}$). The cross-dispersion terms are expressed as

$$\theta_w D_{xy} = \theta_w D_{yx} = (a_{Lxy} - a_{Txy}) v_x v_y / |v| \quad (22)$$

$$\theta_w D_{xz} = \theta_w D_{zx} = (a_{Lxz} - a_{Txz}) v_x v_z / |v| \quad (23)$$

$$\theta_w D_{yz} = \theta_w D_{zy} = (a_{Lyz} - a_{Tyz}) v_y v_z / |v| \quad (24)$$

where

a_{Lxy} , a_{Lxz} and a_{Lyz} are the longitudinal cross direction dispersivity moduli obtained as a harmonic mean of the respective longitudinal component in each direction. The harmonic mean weights the solution towards the lower dispersivity value thus preventing excessive spreading of the cross-term in the direction of low dispersivity values.



The anisotropic dispersion equations 18-24 reduce to the equations of Burnett and Frind (1987) when stratification is only in the vertical direction. The anisotropic dispersion equations reduce to the Scheidegger equation (17) for isotropic conditions.

Since the principal components of the dispersion tensor can be treated implicitly, they are implemented as the default condition when dispersion is simulated. The cross-dispersion terms are treated via the right-hand-side in an iterative manner, as an option. The contribution of Alden M. Provost is acknowledged in developing the formulation for the dispersion terms above.

DUAL POROSITY TRANSPORT FORMULATION

This section discusses the Dual Porosity Transport (DPT) Package included with the BCT Process. The transport equation discussed above represents advective-dispersive transport of chemical species in a porous medium. In some cases, such as transport in fractured media or extremely heterogeneous porous media, it may be more appropriate to use a dual-domain approach wherein the representative volume at the scale of discretization consists of a mobile domain (fractures or zones of high flow within a computational cell) in which advection/dispersion is the predominant means of transport, and an essentially immobile domain (non-fractured matrix or zones of low flow and immobile or relatively stagnant water within the computational cell) with mass transfer between domains via molecular diffusion. A mobile fraction (also referred to as fracture porosity in certain literature) determines how much of the total cell volume is filled by the medium representing the mobile domain, while the remainder (an immobile fraction) constitutes the medium representing the immobile domain. This is known as the dual porosity or dual domain formulation. **Error! Reference source not found.** shows the conceptual representation of a dual porosity model depicting the stagnant and dynamic regions. **Figure 4** shows how the porosities are conceptualized in each domain. Note that the mobile fraction $f_m = V_m/V_t$ as per the figure. Also, the immobile fraction $f_{im} = 1 - f_m$.

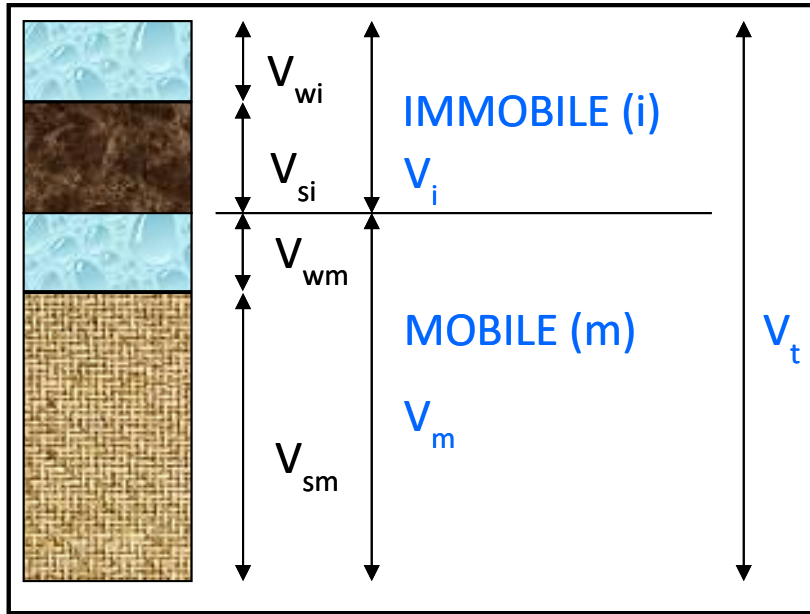


Figure 4. Porosity conceptualization of a dual porosity model.

For the DPT formulation, advective/dispersive transport applies to the mobile fraction of the control volume, while the immobile domain portion provides a storage bin that includes an additional source or sink of component species to the mobile portion. The diffusive exchange between the mobile and immobile domains is typically defined by a lumped kinetic mass transfer term that takes into account the size and nature of the small-scale heterogeneity. For transient flow cases, there is also an exchange of water between mobile and immobile domains which further transfers solute mass between the domains due to advective flux.

Equation 2 can be re-written for the mobile domain of a dual-domain transport formulation as:

$$f_m \frac{\partial(M)}{\partial t} = \frac{\partial}{\partial x_i} \left[f_m \theta_w D_{ij} \frac{\partial c}{\partial x_j} \right] - \frac{\partial}{\partial x_i} [f_m v_i c] - f_m [\lambda_w \theta_w c + \lambda_s \rho_b c_s] - f_m [\mu_w \theta_w + \mu_s (1 - \phi_e)] + \dot{M} - q_{m,im} c_u \quad (25)$$

Where f_m is the fraction of the volume that constitutes the mobile domain [], or the mobile fraction, Γ is the diffusive mass transfer rate from mobile to immobile domain [$ML^{-3}T^{-1}$], $q_{m,im}$ is the flux between mobile and immobile domains, c_u is the concentration of the upstream of the two domains, and where all other properties, parameters and concentrations of equation 25 reference the mobile domain. Note that the last term of equation 25 is non-zero when dual



porosity flow is also simulated, and the flow field is transient. This term accounts for advective movement of solutes between the mobile and immobile domain.

Note that the flow equation that provides the flow-field for transport represents only the mobile domain. A flag is provided in *USG-Transport* to indicate if the hydraulic conductivity then represents that of the entire control volume, or only of the mobile (fracture) domain. For steady-state flow simulations, the immobile domain does not exchange water with the mobile domain and the last term in equation 25 is zero.

The mass balance equation for component species in the immobile domain can be written as:

$$f_{im} \frac{\partial(M_{im})}{\partial t} = -f_{im}[\lambda_{wim}\theta_{wim}c_{im} + \lambda_{sim}\rho_{bim}c_{sim}] - f_{im}[\mu_{wim}\theta_{wim} + \mu_{sim}(1-\phi_{im})] + \Gamma + q_{m,im}c_u \quad (26)$$

Where f_{im} is the fraction of the volume that constitutes the immobile domain ($f_{im} + f_m = 1$),

M_{im} is the total mass per unit volume, of a component species in water and on soil within the immobile domain [M_{im} / L_{im}^3],

c_{im} is the concentration of a component species in water in the immobile domain [M_{im} / L_{im}^3],

θ_{wim} is the moisture content in the immobile domain [],

ϕ_{wim} is the effective porosity of the immobile domain [],

ρ_{bim} is the bulk density of the porous medium within the immobile domain [$M_{s,im} / L_{im}^3$],

c_{sim} is the adsorbed concentration of component species in the immobile domain [$M_{a,im} / M_{s,im}$],

λ_{wim} is the first-order decay coefficient within the immobile domain in water [T^{-1}], $= \ln(2) / t_{1/2wim}$ where $t_{1/2wim}$ is the half-life of the chemical species within the immobile domain in water,

λ_{sim} is the first-order decay coefficient on soil [T^{-1}], $= \ln(2) / t_{1/2sim}$ where $t_{1/2sim}$ is the half-life of the chemical species within the immobile domain adsorbed on the soil,

μ_{wim} is the zero-order decay coefficient within the immobile domain in water [$M_{im} / (L_{im}^3 T)$], and

μ_{sim} is the zero-order decay coefficient within the immobile domain on soil [$M_{a,im} / (L_{im}^3 T)$].



The mass transfer between mobile and immobile domains via the diffusive mechanism is expressed by a first order mass transfer rate as:

$$\Gamma = \zeta(c - c_{im}) \approx [\beta(1 - f)\phi_{im}D_{im}^* / a^2](c - c_{im}) \quad (27)$$

Where ζ is the first-order mass transfer rate coefficient [T^{-1}] which defines the diffusive exchange of species between mobile and immobile domains,

β is a shape factor (3 for rectangular and 15 for spherical aggregates),

D_m^* is an effective diffusion coefficient, and

a is a characteristic half length.

The last term of equation (27) finds an equivalence between the mass transfer rate and matrix diffusion into the stagnant domain.

The advective mass transfer rate between fracture and matrix domains for transient flow situations is expressed by the last term of equation (25). However, for transient flow cases, it often happens that dual porosity transport is considered without simulating dual porosity flow. For such cases an assumption is made that there is no flow between mobile and immobile domains, and thus an immobile domain flow equation is not required since the change of saturation or compressible storage within the immobile domain is zero. The solute mass storage term for the immobile domain can be treated simply for species concentrations in water and on soil, and the special *MODFLOW* storage considerations of equations 5 through 9 are not required for the immobile domain giving:

$$\frac{\partial(M_{im})}{\partial t} = \frac{\partial(\phi_{im}S_{wim}c_{im})}{\partial t} + \frac{\partial(\rho_{bim}c_{sim})}{\partial t} \quad (28)$$

The moisture content of the immobile domain can then be written as:

$$\theta_{wim} = \phi_{im}S_{wim} \quad (29)$$

where

S_{wim} is the water saturation in the immobile domain [].



Various assumptions can be made with regard to immobile domain saturation when the transient flow solution is ignored for the immobile domain: It can be set internally to unity; entered as user input; or set to the same value as the mobile domain saturation at the start of the simulation. It cannot however be changed during the simulation unless flux exchange between the domains is also simulated using the dual porosity flow formulation. Also, if the option ONLY_SATADSORB is used, the volume of the soil in immobile domain that partakes in adsorption will be the saturated thickness when an unconfined condition is applied to the immobile domain of the cell. Note that ONLY_SATADSORB should not be used for cells that simulate unsaturated flow conditions (i.e., use the Richards' Equation) as it may limit the adsorption volume which is not correct.

Note that the dual porosity equations are often expressed in terms of a mobile porosity, θ_m , and an immobile porosity, θ_{im} (for example, Zheng and Wang, 1999). For that case, the mobile and immobile porosities are related to the variables of equations presented above, as:

$$\theta_m = f_m \phi_e \quad (30)$$

And

$$\theta_{im} = f_{im} \phi_{im} \quad (31)$$

The dual domain equations can be used to conceptualize a wide variety of subsurface transport conditions whereby widely differing flow rates exist within a discretized grid-block (i.e., at a sub-grid scale) with mass transfer between these systems. For instance, referring to **Figure 1** and the porosity definitions earlier, if diffusion processes onto bound water (from the flowing and drainable water within the effective pore space) were not considered instantaneous, the immobile domain may be used to represent this water. This space occupied by bound or immobile but connected water, would then not be considered part of the effective porosity but would be conceptualized as part of the immobile fraction instead. For this conceptualization, adsorption would then occur only in the immobile domain (representing the bound water) with which the soil grains are in contact.

Note that the dual porosity formulation includes the ISOLUBILITY keyword to include its own solubility parameters for the process of precipitation and dissolution discussed earlier. The dual porosity formulation also includes the ONLY_SATADSORB option discussed earlier to delineate the adsorption soil volumes for unconfined flow situations. It also includes transport of heat as



described later. The dual porosity formulation also includes the chain decay formulation discussed later.

TRANSPORT FORMULATION FOR CLN CELLS

The equation for transport of solutes through a connected linear network in an unsteady flow field can be written in a manner similar to equation (2), as:

$$\frac{\partial V_s c}{\partial t} = \frac{\partial}{\partial L_{CC}} \left[D_{CC} \frac{\partial c}{\partial L_{CC}} \right] - \frac{\partial (v_{CC} c)}{\partial L_{CC}} + \Gamma_{MC}^* - \lambda_w V_s c - \mu_w V_s \quad (32)$$

where

V_s is the fraction of the total volume of the CLN cell that is saturated during unconfined conditions []. Note that V_s is unity for confined or saturated conditions.

L_{CC} is the length dimension of the CLN cell [L],

D_{CC} is the longitudinal dispersion coefficient along the CLN cell [L^2 / T],

v_{CC} is the velocity of flow along the CLN cell [L / T], and

Γ_{MC}^* is the mass exchange between the GWF cell and the CLN cell [$M / (L^3 T)$].

Note that equations 1 and 15 require this term to also be subtracted from the right-hand-side of a matrix cell interacting with the CLN cell.

In Equation (32), adsorption along the CLN walls is neglected. Dispersion along the CLN domain is expressed as a combination of a velocity-dependent dispersivity and diffusion as:

$$D_{CC} = a_{CC} v_{CC} + V_s D^* \quad (33)$$

where a_{CC} is the longitudinal dispersivity along the CLN cell [L]. Mechanisms for mass exchange between GWF cell and CLN cell, Γ_{MC}^* , include advection as well as dispersion. The dispersion term for GWF-CLN connection may be expressed in a similar manner as equation (33) with a longitudinal dispersion coefficient a_{MC} expressing dispersion that occurs for the GWF-CLN interaction. Thus, the interaction term is written as:



$$\Gamma_{MC}^* = -\frac{\partial(v_{MC}c_{u,MC})}{\partial L_{MC}} + \frac{\partial}{\partial L_{MC}} \left[(a_{MC}v_{MC} + V_s D^*) \frac{\partial c}{\partial L_{MC}} \right] \quad (34)$$

Where

v_{MC} is the flux per unit area from CLN cell to GWF cell [L / T],

$c_{u,MC}$ is the species concentrations of the upstream location between the matrix and CLN cell [M / L^3],

L_{MC} is the length scale for interaction between the CLN and matrix cells [L] taken as the hydraulic radius of the CLN cell plus the effective cell radius of the GWF cell (see equations 14 and 15 of the *MODFLOW-USG* Flow Documentation).

The chain decay formulation discussed later is also included in the CLN transport formulation, as is transport of heat. For use of CLNs in Borehole Heat Exchanger (BHE) simulations, the conduction of heat from GW to borehole is detailed further in the heat formulation section discussed later. The CLN domain also includes the ISOLUBILITY keyword functionality to provide a solubility limit to CLN concentrations. Density dependent flow and transport also occurs through the CLN domain.

IMMOBILE COMPONENT SPECIES AND EQUATION ORDERING

When considering multiple components that may interact with each other, it is useful to consider that all components may not migrate due to advection and dispersion in the groundwater flow-field. Therefore, the USG-Transport code has functionality to simulate mobile components (MCOMP) and immobile components (IMCOMP). The immobile components concentrations and mass are tracked throughout the simulation as they participate in reactions; however, these immobile components are not transported by advection or dispersion. Furthermore, the immobile component concentrations are expressed in terms of their mass per unit total volume (not as a solute concentration) considering that they do not move with water.

The immobile solute species components are ordered after the mobile components. The total number of components in the transport simulation are then $NTCOMP = MCOMP + IMCOMP$. If a thermal simulation is included, MCOMP and NTCOMP are incremented by one. Sequencing of the heat equation immediately follows all mobile component species equations. Thus the



ordering of variables is as follows: Concentrations of mobile components MCOMP, followed by temperature, followed by concentrations of immobile components IMCOMP.

*Note that an **immobile species** is different from an **immobile domain** of a dual porosity medium discussed earlier.* Immobile components can exist in the mobile and immobile domains of a dual porosity system.

CHAIN DECAY

Chain decay is the process whereby a “parent” component generates one or more “daughter” components, some of which may be significant for further analysis in the transport simulation. A typical example of chain decay is the transformation process of PCE → TCE → DCE → VC whereby Tetrachloroethylene transforms ultimately to Vinyl Chloride. Chains may be more complex whereby multiple parents may generate the same daughter product or vice versa. Figure C1 below demonstrates a chain decay process. The mass of any daughter product generated by transformation of a parent species depends on the stoichiometry of the reaction and the molecular weights of the parent and daughter components. The stoichiometric coefficients input for a simulation should therefore consider the units used for concentration in the simulation (mass/volume or moles/volume for instance). A further consideration in chain decay (or Generative Transformation) is that one or more of the species of reaction may not be mobile. This is an especially important consideration for geochemical reactions.

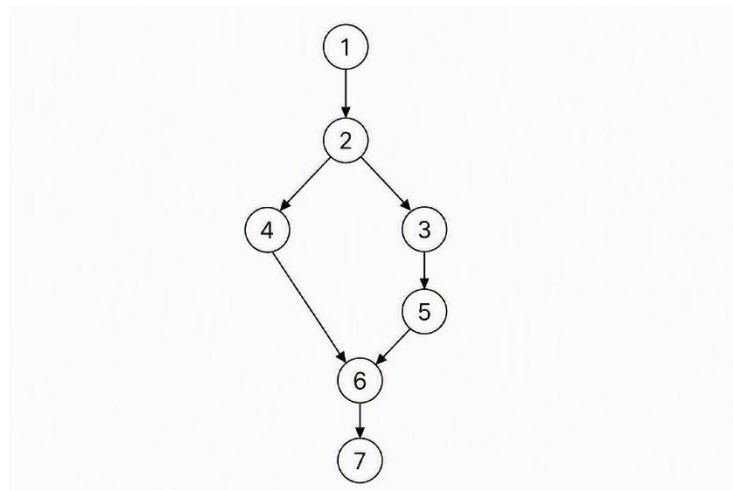


Figure C1. Example of a Branched Chain Decay Process



The decay formulations considered for chain decay are equilibrium first order and zeroth order processes discussed earlier. The generation of daughter products resulting from decay of its parent components in a porous medium can be written as:

$$\dot{M}_k = \sum_{m \in NP} \xi_{mk}^1 [\lambda_w \theta_w c + \lambda_s \rho_b c_s] - \sum_{m \in NP} \xi_{mk}^0 [\mu_w \theta_w + \mu_s (1 - \phi_e)] \quad (C1)$$

Where: $\dot{M}_k$ is the mass generation for species component k , the summation sign is over all m species components that are parents to component k , NP is the number of parents to component k , ξ_{mk}^1 is the ratio of mass of component m consumed, to mass of component k generated for the first order reaction and ξ_{mk}^0 is the ratio of mass of component m consumed, to mass of component k generated for the zeroth order reaction.

Typically, a linear straight or branched chain can be solved in just one transport iteration over the species. This is the case when daughter products have a higher species number than a parent component. However, if a daughter product has a lower species number than its parent, the concentration of the parent would not yet be updated when daughter product generation is computed, thus treating that in an explicit manner. Including an outer iteration to the calculation will redo the calculation with a parent concentration also updated. Also note that the concentrations in equation (C1) are of the parent component m which decays to component k . Therefore, numerical implementation is entirely on the right-hand side vector of the transport solution matrix.

An optional expansion of the formulation enables the stoichiometry to vary by location. Stoichiometry is generally a property of the chemical reaction and not location; however, there could be cases where multiple reactions are occurring on some of the same chemicals in different locations. For instance, abiotic reactions occur in one portion of the domain while biotic reactions in another part of the domain having different chemistry at different locations. The option “SPATIALREACT” turns on location variable stoichiometric reactions for a simulation.

Note: The chain decay formulation is implemented in the CLN domain as well as in the dual porosity transport (DPT) domain.



GENERALIZED MULTI-SPECIES KINETIC REACTIONS

USG-Transport (USGT) incorporates reaction modules through operator splitting, in which solute transport and chemical reactions are solved sequentially within each transport time step. USGT first updates component concentrations for all species with other transport processes within the core BCT routines (storage, adsorption, advection, dispersion, decay, dual domain processes), then passes the resulting concentrations to the reaction module for a separate reaction update. This keeps the code modular, allowing for complex reaction formulations to be added without modifying the core transport solver. User-defined outer iterations over transport and reaction routines allow for iterative implicit computations instead of using a strict operator splitting approach. Solute mass changes induced by the reaction package are also tracked independently for the component mass budgets.

A general reaction between multiple reactants, which then produces multiple products can be written as:

$$x_1 R_1 + x_2 R_2 + \dots + x_i R_i = y_1 P_1 + y_2 P_2 + \dots + y_j P_j \quad (\text{R1})$$

Where: R_i are the “ i ” reactants and P_j are the “ j ” products of the reaction. The reaction module accommodates multiple reaction equations as may be needed in a simulation. These could be independent of each other, or dependent on previous reactions. For instance, the product of one reaction may be a reactant for another.

The reaction equation is written compactly as

$$\sum_{i \in L(r)} s_L(r, i) \cdot X_i \rightarrow \sum_{j \in R(r)} s_R(r, j) \cdot X_j \quad (\text{R2})$$

where X_i denotes the molar concentration of species i [mol m^{-3}]. For each reaction, the user specifies:

- a set of $n_{L,r}$ reactant (left-hand-side) species, identified by component index $i_L(r, 1), \dots, i_L(r, n_{L,r})$, with corresponding positive stoichiometric coefficients $s_L(r, 1), \dots, s_L(r, n_{L,r})$;
- a set of $n_{R,r}$ product (right-hand-side) species, identified by component index $i_R(r, 1), \dots, i_R(r, n_{R,r})$, with corresponding positive stoichiometric coefficients $s_R(r, 1), \dots, s_R(r, n_{R,r})$.

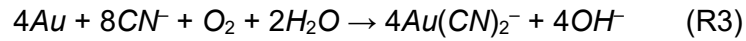
The component indices i_L and i_R range over all N_{TCOMP} mobile and immobile transport species (mobile solutes, optionally heat, and immobile (IMCOMP) species), and may be listed **in any order and combination in the reaction equation**: a species need not be a reactant on the left and a product on the right in only one particular numeric position. For example, a reaction



with component 1 and component 3 as reactants and component 2 as the sole product is fully general input, i.e. $I_L(1,1)=1$, $I_L(1,2)=3$, $I_R(1,1)=2$.

Bilinear Reaction Formulation

The motivation for developing a generalized reaction module stems from simulation of gold extraction during a heap leach process whereby gold ore is placed in a heap and cyanide solution is drizzled over it to extract the gold-cyanide complex. The gold-dissolution rate is a function of cyanide, oxygen, and remaining gold inventory. This is generally expressed by the Elsner cyanidation reaction written as:



With the assumption of constant/excess oxygen availability, the rate law is reduced to keep only the cyanide and remaining-gold dependence in the reaction, as:



A **bilinear lumped rate law** (first order in free cyanide and first order in remaining reactive gold inventory) can be used to approximate the reaction kinetics to reduce the more detailed cyanidation model. This is expressed as:

$$r = k * C_{CN} * M_{Au} \quad (R5)$$

where

- r = Au dissolution rate [$\text{mol m}^{-3} \text{s}^{-1}$]
- k = lumped kinetic constant [$\text{m}^3 \text{mol}^{-1} \text{s}^{-1}$] if using mol m^{-3}

Thus, gold dissolution depends on cyanide concentration and remaining gold inventory, while oxygen is always available. Using equation (R4), the reaction ODEs are

$$\frac{dM_{Au}}{dt} = -r \quad (R6)$$

$$\frac{dC_{CN}}{dt} = -2r \quad (R7)$$

$$\frac{dC_{AuCN}}{dt} = +r \quad (R8)$$

or, explicitly,

$$\boxed{\frac{dM_{Au}}{dt} = -k * C_{CN} * M_{Au}} \quad (R9)$$

$$\boxed{\frac{dC_{CN}}{dt} = -2k * C_{CN} * M_{Au}} \quad (R10)$$

$$\boxed{\frac{dC_{AuCN}}{dt} = +k * C_{CN} * M_{Au}} \quad (R11)$$



Where: C_{CN} is the free cyanide concentration mol m⁻³, M_{Au} is the remaining reactive solid AU inventory mol m⁻³, and C_{AuCN} is the dissolved $Au(CN)_2^-$ concentration mol m⁻³. These equations enforce the tracked-species stoichiometry: 1 mol Au dissolved consumes 2 mol CN and produces 1 mol of $Au(CN)_2^-$.

An analytical solution to the equations produces an exact solution and removes explicit-stability concerns for the local chemistry.

Defining the invariant

$$I = C_{CN}^n - 2 * M_{Au}^n \quad (R12)$$

Then, for $|I| > \varepsilon$,

$$M_{Au}^{n+1} = \frac{I * M_{Au}^n * \exp(-I * k * \Delta t)}{C_{CN}^n - 2 * M_{Au}^n * \exp(-I * k * \Delta t)} \quad (R13)$$

$$C_{CN}^{n+1} = I + 2 * M_{Au}^{n+1} \quad (R14)$$

$$C_{AuCN}^{n+1} = C_{AuCN}^n + (M_{Au}^n - M_{Au}^{n+1}) \quad (R15)$$

And for the special case where $I = 0$ (initial cyanide exactly stoichiometric with reactive Au):

$$M_{Au}^{n+1} = \frac{M_{Au}^n}{1 + 2 * k * M_{Au}^n * \Delta t} \quad (R16)$$

$$C_{CN}^{n+1} = 2 * M_{Au}^{n+1} \quad (R17)$$

$$C_{AuCN}^{n+1} = C_{AuCN}^n + (M_{Au}^n - M_{Au}^{n+1}) \quad (R18)$$

The advantage of a bilinear reaction formulation is that an analytical solution is available for the reactions.

Multilinear Reaction Formulation

The bilinear formulation can be expanded to multiple reactants as in the reaction equation (R2). For this case, the reaction rate law can be expressed as:

$$r_{rxn} = k \cdot \prod_{i \in L(r)} X_i \quad (R19)$$

where $\prod_{i \in L(r)}$ is the product over all reactants (left hand side of equation (R2)) concentrations, for that reaction. Thus, for reaction r , the rate of change of each participating species is obtained from the stoichiometry:

$$dX_i/dt = -s_L(r,i) \cdot r_{rxn} \quad \text{for } i \in L(r) \quad (R20)$$

and

$$dX_j/dt = +s_R(r,j) \cdot r_{rxn} \quad \text{for } j \in R(r) \quad (R21)$$



Species not participating in reaction r are unaffected by it. When $N_{RXN} > 1$, the reactions are applied sequentially (operator split) within the local update, in the order $r = 1, \dots, N_{RXN}$, consistent with the generalized-decay chain treatment already used in USG-Transport.

For any species pair (i, i') both appearing on the left of reaction r , or a reactant/product pair, the linear combination $s_L(r, i') \cdot X_i - s_L(r, i) \cdot X_{i'}$ is conserved during the reaction sub-step, since both derivatives are proportional to the same rate r_{rxn} . For the two-reactant case, this was the invariant used for the exact solution.

No closed-form solution available when a reaction has more than two reactants. Instead, the time step size Δt is subdivided into N_{sub} internal sub-steps of size $h = \Delta t / N_{sub}$, with N_{sub} chosen from a crude stability estimate based on the initial reaction rate. Each sub-step is advanced by explicit (forward Euler) integration of Equations (R20) and (R21). The number of sub-steps is capped at a maximum value to bound computational cost; the explicit sub-stepping with extent clamping keeps all concentrations non-negative regardless of step size, at the cost of being only first-order accurate in time (as opposed to the exact solution for a bilinear reaction).

The cell-by-cell mass change produced by the reaction is accumulated, per node and per component, as the difference between the mass concentration after and before the reaction step, multiplied by an effective reacting volume that depends on whether the component is a mobile species (or heat) or an immobile (IMCOMP) species — consistent with the volumetric convention already used for storage, decay, and total-mass accounting in USG-T. The change in mass per time step due to reactions is given as:

$$\Delta M_i = (C_i^{new} - C_i^{old}) \cdot V_i^{eff} \quad (R22)$$

Where the effective volume is defined as:

$$V_i^{eff} = V \cdot \theta \cdot S_w \quad \text{for mobile solute or heat components } (i \leq MCOMPT) \quad (R23)$$

to express its water phase concentration, and

$$V_i^{eff} = V \quad \text{for immobile solute components } (i > MCOMPT) \quad (R24)$$

to express the total mass of the immobile component within the grid-block volume.

In equations (R23) and (R24), V is the node (or CLN segment) volume, θ is porosity (PRSITY for the mobile domain, PRSITYIM for the immobile domain), and S_w is water saturation.

Mass – Mole Conversion

Transport formulations typically use a mass basis for concentration instead of a molar basis. However, the reaction stoichiometry is developed on a molar basis. Therefore, conversion is required from mass to moles before reaction computations and then reverted to mass after



reaction computations. For each component i with molecular weight W_i [kg mol^{-1}] > 0 , the following conversion is applied before the reaction update on each component:

$$X_i = C_i / W_i \quad (\text{R25})$$

Where W_i is the molecular weight of species i . The inverse conversion $C_i = X_i \cdot W_i$ is applied after the reaction computation.

If $W_i \leq 0$ for a component, the component is treated as already on a molar basis and no conversion is performed — this allows mixed mass- and mole-basis species lists without requiring a molecular weight for every component in the simulation, only for those participating in a reaction. Alternatively, a molecular weight of 1 allows for molar units to be directly used in a transport simulation.

DENSITY DEPENDENT FLOW AND TRANSPORT FORMULATION

Sorab Panday, GSI Environmental, Herndon, VA.

Christian D Langevin, U.S. Geological Survey, Reston, VA.

Alden M Provost, U.S. Geological Survey, Reston, VA.

This section discusses the Density Dependent Flow (DDF) Package included with the BCT Process. The flow and solute transport equations may be coupled for situations whereby solute concentrations affect the density of the fluid, thus affecting the flow-field. This coupling may be necessary for proper analyses of saltwater intrusion in coastal aquifers, analyses of brackish water resources, evaluation of impacts of fluid injection into deep brines, or studies of fingering and instabilities of denser fluids moving through lighter ones. Density dependent flow has typically been solved using the equivalent freshwater head (EFH) formulation (Guo and Langevin, 2002; Langevin et al. 2003). A hydraulic head (HH) formulation is presented here for use with *USG-Transport*. The HH formulation for density driven flow is attractive because it precludes the need for converting back and forth between EFH and HH formulations and avoids complexities that arise due to nonlinearities or boundary conditions. Also, the density terms is compartmentalized such that it can be modularly incorporated into *USG-Transport*. Formulation of the density dependent flow is detailed by Langevin et al. (2020).

The HH formulation for density dependent flow is obtained as follows. Darcy's law provides the fluid flux in terms of the pressure and elevation potentials as:

$$q = -(k / \mu)[\nabla p + \rho g \nabla z] \quad (35)$$



Where:

q is the Darcy flux $[L^3 / T]$, k is the permeability of the medium $[L^2]$, μ is the dynamic viscosity $[M / (LT)]$, p is the pressure $[M / (LT^2)]$, ρ is the density of water at resident concentration $[M / L^3]$, g is the gravity term $[L / T^2]$, and z is the elevation $[L]$. Using chain rule on the right-hand side of equation (35) gives:

$$q = -(k / \mu)[\nabla(p + \rho gz) - gz \nabla \rho] \quad (36)$$

The hydraulic head can be defined as

$$h_\phi = p / (\rho g) + z \quad (37)$$

Substituting equation (37) into equation (36) gives:

$$q = -(k / \mu)[\nabla(\rho g h_\phi) - gz \nabla \rho] \quad (38)$$

Multiply and divide Equation 38 by freshwater density, ρ_o , and freshwater viscosity, μ_o ; bring the constant gravity term out of the gradient operator; and take the freshwater density term in the denominator into the gradient operator gives:

$$q = -(kg \rho_o / \mu_o)(\mu_o / \mu)[\nabla(\rho / \rho_o h_\phi) - z \nabla(\rho / \rho_o)] \quad (39)$$

Which is also written as:

$$q = -K_o(\mu_o / \mu)[\nabla\{(\rho / \rho_o)h_\phi\} - z \nabla(\rho / \rho_o)] \quad (40)$$

Where

K_o is the hydraulic conductivity of freshwater defined as $K_o = kg \rho_o / \mu_o$

The term (μ_o / μ) is taken as unity with the assumption that viscosity changes with salinity are not significant.

Equation (40) reduces to standard Darcy's equation for flow of freshwater when there is only freshwater in a simulation.

Expanding the first gradient operator on right-hand side of equation (40) using chain rule and rearranging gives:



$$q = -K_o \left[(\rho/\rho_0) \nabla h_\phi \right] - K_o (h_\phi - z) \nabla (\rho/\rho_0) \quad (41)$$

Equation (41) indicates that flow occurs as a result of the gradient in HH (as per the standard Darcy's Law equation), with additional terms resulting from manipulating the generalized flow equation. Also, the hydraulic conductivity is scaled by the ratio of solution density to freshwater density. These corrections are applied as separate modules called within *USG-Transport*.

The mass storage term due to density changes in a grid-block can be expanded as Guo and Langevin (2002):

$$\left[S_y S_w + S_s (h_\phi - z) \right] \frac{1}{\rho} \frac{\partial \rho}{\partial t} = \left[S_y S_w + S_s (h_\phi - z) \right] \frac{1}{\rho} \frac{\partial \rho}{\partial c} \frac{\partial c}{\partial t} \quad (42)$$

The derivative of concentration with respect to time is obtained from the transport solution. Assuming a linear relationship between fluid density and concentration, the density-concentration function can be expressed as:

$$\frac{\partial \rho}{\partial c} = \frac{(\rho_{std} - \rho_o)}{c_{std}} \quad (43)$$

Where

ρ_o is the density of freshwater (when concentration is zero); and ρ_{std} is the density of the solution at a concentration of c_{std} . Equations 42 and 43 are also updated in *USG-Transport* in a modularized fashion at the same time when the flow terms are being updated. Additional details are published in Langevin et al. (2020).

Note that the density dependent formulation is applied to the CLN domain as well as to the groundwater flow domain.

A **nonlinear density versus concentration** relationship is also implemented as an option. The keyword **NONLINEAR** on the first input line indicates that a nonlinear relationship is to be used. When the **NONLINEAR** option is active, the normalized density at each cell is computed as $\rho = 1 + S_w (\rho_{tab} / \rho_o - 1)$, where ρ_{tab} is the density interpolated from the table at the cell concentration, and S_w is the cell saturation. The S_w factor ensures that density effects smoothly vanish for dry or partially saturated cells (when $S_w = 0$, $\rho = 1.0$, corresponding to freshwater density, and no density gradient drives flow).



The transient density storage term uses the actual density change from the table, $(\rho_{tab}c_{new} - \rho_{tab}c_{old})$, rather than the linear approximation $\partial\rho / \partial c(c_{new} - c_{old})$.

The contribution of Alden M. Provost and Christian L. Langevin is acknowledged in developing the density dependent flow formulation presented above.

HEAT MODULE OF BTN PACKAGE

Sorab Panday, GSI Environmental, Herndon, VA.

Thomas Conzen, Kühn Geoconsulting GmbH, Bonn, Germany

Introduction

Heat transport is very similar to solute transport and can generally be simulated by a solute transport code. However, it is more convenient and accurate to include input of the heat related parameters and do the temperature computations internally. Therefore, the BTN package has been extended to directly perform computations of energy transport. Practical applications of energy transport simulations include evaluation of Borehole Heat Exchanger (BHE) performance, temperature signatures of mixing waters (groundwater interaction with streams, thermal plume extent from aquifer recharge operations), or geochemical studies. Therefore, the boundary conditions for temperature have also been extended to suit these applications.

The development of the heat transport capability was supported in part by funding from Kühn Geoconsulting GmbH, Bonn, Germany, and research funding from GSI Environmental.

Heat Module Formulation

The energy transport equation in a porous medium can be written as (Hect-Mendes et al, 2010):

$$\phi S_w \rho_w c_w \frac{\partial T}{\partial t} + (1-\phi) \rho_s c_s \frac{\partial T_s}{\partial t} = \frac{\partial}{\partial x_i} \left[(\lambda_m + \phi S_w \rho_w c_w \alpha v_i) \frac{\partial T}{\partial x_j} \right] - \frac{\partial}{\partial x_i} [\phi S_w \rho_w c_w v_i T] + q_b - \Gamma_{MC}^* \quad (H1)$$

Where

ϕ is the porosity [],

S_w is the saturation of water, [].

ρ_w is the density of water [kg / m^3];



c_w is the specific heat capacity of water [$J / kg / K$];

T is the temperature of water [K],

t is the time [s],

ρ_s is the density of soil solids [kg / m^3];

c_s is the specific heat capacity of soil solids [$J / kg / K$];

T_s is the temperature of soil solids [K],

λ_m is the effective thermal conductivity of the medium [$W / m / K$]; It is noted that the effective heat conductivity is assumed to be isotropic for the medium.

v_i is the Darcy flux in direction x_i , [m/s],

α is the dispersivity [m], and

q_b is the boundary heat flux [W/m³],

Γ_{MC}^* is the heat exchange between the matrix GWF cell and the conduit CLN cell [$M / (L^3T)$].

Dividing the energy balance equation by the term $(\rho_w c_w)$, and comparing it with the solute transport equation, the following similarities are noted.

1. Temperature (T) is similar to solute concentration (c) .
2. The storage term for heat in water is similar to the storage term for solutes in water.
3. With the further assumption that temperature in water is the same as that on soil (i.e., $T_s = T$), the storage term for heat in soil is the same as the storage term for solutes on soil with the following equivalence: $(c_s / \rho_w c_w = k_d)$.
4. The dispersion term for heat is similar to the dispersion term for solute.
5. The heat conduction term (thermal diffusivity) in the energy balance equation is similar to the diffusion term in the solute transport equation with the following equivalence: $(\lambda_m / \rho_w c_w = \theta_w D_o)$.
6. The advection term for heat is similar to the advection term for solutes.



7. The decay terms for solute transport do not have an equivalent in heat transport and all decay coefficients are zero.
8. The source/sink terms for heat are similar to the source/sink terms for concentration as noted in the discussion on boundary conditions further below.

The heat conductivity of the medium can be evaluated as a weighted average of the heat conductivity of soil and water, giving:

$$\lambda_m = \phi S_w \lambda_w + (1 - \phi) \lambda_s \quad (\text{H2})$$

Where

λ_w is the heat conductivity of water and λ_s is the heat conductivity of soil. Upon making these substitutions, the solute transport equation can be used to solve for heat transport.

For the CLN domain, the energy conservation equation can be written as:

$$\rho_w c_w \frac{\partial V_s T}{\partial t} = \frac{\partial}{\partial L_{cc}} \left[(\lambda_{CLN} + V_s \rho_w c_w \alpha v_{cc}) \frac{\partial T}{\partial x_j} \right] - \frac{\partial}{\partial x_i} [\rho_w c_w v_{cc} T] + q_b + \Gamma_{MC}^* \quad (\text{H3})$$

Where

V_s is the fraction of the total volume of the CLN cell that is saturated during unconfined conditions []. Note that V_s is unity for confined or saturated conditions within the CLN.

L_{CC} is the length dimension of the CLN cell [L],

λ_{CLN} is the heat conductivity along the length of the CLN domain

v_{CC} is the velocity of flow along the CLN cell [L / T], and

Γ_{MC}^* is the heat exchange between the matrix GWF cell and the conduit CLN cell [M / (L³T)] and it is noted that the sign is opposite of that in equation (H1) because heat flow into the CLN cell is the same as heat flow out of the porous matrix cell for heat conservation.

In equation (H3), it is assumed that the CLN tube itself has zero heat capacity and therefore an adsorption type term is not included. Further, if it is assumed that the heat conductance along



the length of the CLN material is zero, then the net CLN heat conductance can be expressed only by the conductance of the liquid within it as:

$$\lambda_{CLN} = V_s \lambda_w \quad (H4)$$

The interaction term between the CLN cell and the groundwater (GW) cell can be expressed as:

$$\Gamma_{MC}^* = -\frac{\partial(\rho_w c_w v_{MC} T_{u,MC})}{\partial L_{MC}} + \frac{\partial}{\partial L_{MC}} \left[(\rho_w c_w a_{MC} v_{MC} + \lambda_{MC}) \frac{\partial T}{\partial L_{MC}} \right] \quad (H5)$$

Where

λ_{MC} is the heat conductivity between the porous matrix and the CLN cell which is computed as the inverse of the average of the thermal resistance to heat transfer of the GWF and CLN cells, weighted by the length dimension along the direction of heat flow. For the CLN cell, this dimension is the radius of the cell (L_{CLN}). For the GWF cell, this radius is the radius of influence of the GWF cell as determined by Peaceman (1983) formulation (L_{GWF}). Note that this is equivalent to the weighted harmonic mean of the conductivity of porous medium and CLN cell.

The thermal conductivity (inverse of resistance) of the CLN cell is the thermal conductivity of water times water saturation within the CLN cell since only water is within the CLN cell and is defined by equation (H4).

The thermal conductivity of the GWF cell is the average of water and soil, in their respective amounts (saturation times porosity for water and one minus porosity for soil). Thus, $\lambda_{GWF} = \phi \lambda_w S_w + (1 - \phi) \lambda_s$, where λ_s is the thermal conductivity of soil. Thus, thermal conductivity of air in an unsaturated system is neglected.

The first term on the right-hand side of equation (H5) represents convection of heat through transfer of water between the CLN cell and the matrix. The second term on the right-hand side of equation (H5) represent mechanical dispersion plus heat conduction in the fluids between the CLN and matrix.

Recognizing that the heat conductance and dispersion terms are similar to the water flow term in the continuity equation, analogous terms may be developed for the transport equation for the various options available for CLN-GW interaction in MODFLOW-USG (Panday et al, 2013) for



the flow computations. Specifically, two options are provided with heat conductance – one is to use a high thermal conductivity for the CLN material to transfer heat to the CLN fluids, while the other is to use a similar equation to the Thiem equation for water flow, which patches in the small scale effect of the CLN with a larger grid block. Thus, if the coefficient of the last term in equation (H5) is expressed as an effective conductivity λ_{eff} as:

$$\lambda_{eff} = \rho_w c_w a_{MC} v_{MC} + \lambda_{MC} \quad (H6)$$

Then the heat conductance coefficient term of an implicit CVFD solution matrix may be expressed as:

$$C_H = \frac{Area(\lambda_{eff})}{b} = \frac{2\pi r l (\lambda_{eff})}{b} = \frac{2\pi r l (\lambda_{eff})}{r \log(r_o / r)} \quad (H7)$$

Where,

Area is the wetted area of the CLN cell,

r is the radius of the CLN cell,

r_o is the effective radius of influence of the GW cell,

l is the length of the CLN cell,

b is the “separation distance” between CLN cell and the porous medium cell across which flow occurs. This “separation distance” is taken as the radius of the CLN cell. The last equality of equation (H7) comes from a similarity to the Thiem equation for water flow and has been shown for that case, to reduce the grid-discretization impacts on the GW-CLN interaction flux, when the effective radius of influence is computed as per the Peaceman (1983) formulation (see discussion of CLN-GW interaction for fluid flow in Panday et al, 2013). Therefore, the second equality and the last equality of equation (H7) are provided as options to this interaction flux computation (IDISPCLN = 0 or 1). Note that for the second equality (IDISPCLN = 0), the cell radius will be used for the separation distance, while for the third equality (IDISPCLN = 1) the log of the ratio of cell to well radius is used.

It should be noted that the previous USG-Transport versions only provided the first option for the dispersive flux term; however, with this version, since the second option has been incorporated for heat conductivity, it is also available for dispersivity.



An additional generality has been included into USG-Transport, specific to closed-system Borehole Heat Exchangers (BHEs). Essentially, there could be additional resistances to heat transfer between the grout and the BHE fluid including the BHE tube material thermal conductance, λ_{pipe} , and the boundary layer mixing of heat between the BHE wall and the fluid within termed the convective heat transfer, λ_{conv} , which is a function of the Reynolds Number and the Prandtl Number. **Table 1** provides the convective heat coefficient for different diameter pipes and different flow rates. To accommodate these additional resistances, the heat conductivity between matrix and conduit tube λ_{MC} in equation (H6) is a combination of the various impacts. Thus,

$$\lambda_{MC} = \lambda_{net} + \lambda_{conv} \quad (H8)$$

Where:

λ_{net} is the net effective heat conductivity of the porous medium in the GWF cell, the BHE pipe, and the fluid within the BHE pipe. The heat conductivity of the BHE tube material may itself include a combination of resistances that can be evaluated outside the code and provided as input. The convective heat transfer coefficient depends on the thermal conductivity of the BHE fluid and the Prandtl Number and the Reynolds Number of flow within the BHE tube and may be evaluated outside the code and provided as input. The table below provides values for various pipe diameters and flow rates of water as the BHE fluid, using the equation of Choi et al, (2011). [Choi, JC, SR Lee, and DS Lee, 2011. Numerical simulation of vertical ground heat exchangers: Intermittent operation in unsaturated soil conditions, Computers and Geotechnics, 38 (2011) 949-958].

Table 1. Convective Heat Transfer Coefficient for various pipe diameters and flow rates.

Diameter (m)	Flow Rate (cu-m/s)	Convective Heat Coeff. (W/m/K)
0.1	1.00E-05	10.45
0.1	1.00E-04	65.95
0.1	3.30E-04	171.42
0.1	1.00E-03	416.14
0.15	1.00E-05	5.04
0.15	1.00E-04	31.79
0.15	3.30E-04	82.62
0.15	1.00E-03	200.58
0.2	1.00E-05	3.00
0.2	1.00E-04	18.94



0.2	3.30E-04	49.23
0.2	1.00E-03	119.51

The net thermal resistance λ_{net} , is the sum of the resistances offered by the soil, the BHE pipe, and the water in the BHE tube. This is computed as length weighted resistances as:

$$\lambda_{net} = (L_{GWF} + L_{CLN} + L_{pipe}) / (L_{GWF} / \lambda_{GWF} + L_{CLN} / \lambda_{CLN} + L_{pipe} / \lambda_{pipe}) \quad (H9)$$

Where:

L_{CLN} is the radius of the CLN cell; L_{GWF} is the radius of influence of the GWF cell as determined by Peaceman (1983) formulation, and L_{pipe} is the thickness of the BHE pipe material. Note that this is the same as the weighted harmonic mean thermal conductances of the various materials.

Accommodation is also provided for a different BHE fluid than water. The BHE fluid thermal conductivity may be different from that of the pore water and is entered for each CLN segment for this case. It is likely that the BHE fluid has a different density and heat capacity as well. Instead of adding to the input burden for this specific case, the user may enter a scaled thermal conductivity for the fluid to account for its different density and heat capacity, by multiplying its value by fluid density and fluid heat capacity, and dividing by water density and water heat capacity.

For the case where BHE details are different and specific, additional generality for BHE operations is accommodated via an option under OPTION2 called BHEDETAIL in the CLN input file. If this flag is on, then each of the conduit types includes additional input for the heat conductivity of the BHE tube material and its thickness, the BHE fluid thermal conductivity, and for the convective heat transfer coefficient. If the BHE fluid is water, a negative value of fluid thermal conductivity will use the thermal conductivity of the water input in the BCT file for heat parameters. Zero may be entered for the BHE pipe thermal conductance if that is to be neglected for any BHE pipe. Zero may be entered for the convective coefficient if that is to be neglected for any BHE pipe.

Boundary Conditions for Heat Module

The heat transport boundary conditions are applied in a similar manner as for solute transport. A prescribed temperature boundary can be provided at any cell within the model domain and heat flux in or out of that cell will be such as to maintain that prescribed temperature.



Third type boundary conditions for energy transport, where water enters or leave the domain, are provided by supplying the boundary temperature which is internally multiplied by the boundary fluid flux. Thus, the term q_b is given by:

$$q_b = \rho_w c_w q_w T_{u,b} \quad (\text{H8})$$

For outflow, the upstream temperature, $T_{u,b}$, is the temperature of the cell, while for inflow, it is the temperature of incoming fluid. Note that since the entire energy balance equation was divided by $(\rho_w c_w)$, the boundary term is also divided by this term and therefore energy inflow is expressed in the code as

$$q_b / \rho_w c_w = q_w T_{u,b} \quad (\text{H9})$$

Also, since MODFLOW-USG does not directly have a heat inflow or solute mass inflow boundary input, the heat inflow can be provided to a model by providing a small flux ($10^{-6} \text{ m}^3/\text{s}$) with an associated boundary temperature T_b , such that the product is the energy input to the model,

$$q_w T_{u,b}.$$

Operation of BHE systems requires that a simulation provide a constant heat flux boundary condition to a BHE whereby the heat inflow minus heat outflow is a constant. Considering Equation (H8), this implies that the temperature at the inlet of a BHE minus the temperature at the outlet is a constant. This condition can be applied within USG-Transport using the modified DRT Package.

The DRT package allows for flow of water out of a domain if water levels are above a drain elevation, with all or portion of the flow returned to the domain. The DRT package has been modified to include solutes and heat whereby the solute (or heat) removed from a location can be reapplied (or a portion of it) at the return flow location. When applied to the CLN cells at the inlet and outlet of a borehole, this allows for the inflowing water to have a temperature that is different from the outflowing temperature by a user defined fixed amount. The DRT package documentation herein includes some discussion on applying the constant heat rate boundary condition for a BHE tube.

Input Specific to Heat Module

The USG-Transport code can be used to solve for heat transport by including a flag on the first line of the BCT (Block Centered Transport) input data file, IHEAT = 1. When this flag IHEAT is



on, the code adds temperature as the last of the mobile transport species and requests input for the various energy balance equation parameters. If solutes are solved for in the simulation as well, then the heat equation follows solution for all mobile solutes and the total solute components solved for transport (MCOMPT) include the mobile solutes (MCOMP) plus one (temperature). Input instructions for the heat transport capability are provided at the end of the section.

When IHEAT = 1, the third type boundary conditions for the energy equation ($W / m^3 / s$) are supplied for each boundary condition via an auxiliary variable in that boundary condition's input file. The energy flux is computed as a fluid flux times the temperature as per the boundary condition discussion. This is similar to supplying boundary conditions for solute species. Note that the auxiliary variables are named C01, C02, etc. up to the total number of mobile species. [For heat transport, the auxiliary variable is named TMPR.](#)

If IHEAT = 1, the prescribed temperature boundary condition is provided via the PCB file which is also used for prescribed concentration of solute transport runs. The heat transport equation is solved after all solute transport equations are solved, so the species number in the PCB package for prescribing temperature of a heat equation will be NCOMP+1, where NCOMP are the components that are solved for solute species.

Output Specific to Heat Module

Output for the energy transport equation is similar to that for solute transport. The Listing file reflects input parameters, transport iteration behavior and output of temperatures and energy balances along with other data of the simulation. The temperature is optionally written to the binary CON file as one more species of solution (the CON file is the binary file that outputs concentrations of the solute species). The boundary exchange of energy and energy storage terms are also reported with the mass balance output.



MATRIX DIFFUSION TRANSPORT (MDT) PACKAGE

Sorab Panday and Shahla Farhat, GSI Environmental Inc., Houston, Texas.
Ronald Falta and Kien Pham, Clemson University, Clemson, South Carolina.
Alan Lemon, Aquaveo LLC., Salt Lake City, Utah.

Acknowledgement

Development efforts for the Matrix Diffusion Transport package were funded by the Department of Defense's Environmental Security Technology Certification Program (ESTCP-ER19-5028).

Disclaimer

The authors of this work, the United States government, GSI Environmental, Clemson University, and Aquaveo LLC. make no warranties and disclaim liability for all uses of the software and documentation.

Formulation

The MDT package is based on a semi-analytical method implemented in the screening level model REMChlor-MD (Falta et al., 2018; Farhat et al., 2018; Muskus and Falta, 2018). The method is based on an approach originally developed for simulating heat conduction in low permeability cap rocks (Vinsome and Westerveld (1980). It was adapted to the analogous problem of dissolved solute diffusion in aquitards by Falta and Wang (2017), and extended to the case of embedded heterogeneities by Muskus and Falta (2018).

Matrix diffusion in low permeability materials can be approximated as a one-dimensional process that depends on the distance from the interface with the high permeability material, z_l . The governing partial differential equations for transient matrix diffusion assuming first order decay of a component $m-1$ that produces a daughter compound, m are:

$$R_{lm} \frac{\partial C_{lm}}{\partial t} = \tau_l D \frac{\partial^2 C_{lm}}{\partial z_l^2} + -\lambda_{lm} C_{lm} \quad \text{for } m = 1 \quad (\text{MD1})$$

$$R_{lm} \frac{\partial C_{lm}}{\partial t} = \tau_l D \frac{\partial^2 C_{lm}}{\partial z_l^2} + y_{lm-1} \lambda_{lm-1} C_{lm-1} - \lambda_{lm} C_{lm} \quad \text{for } m > 1 \quad (\text{MD2})$$

Here, the subscript l refers to the low permeability material. C_{lm} is the aqueous mass concentration of component m in the low permeability material, R_{lm} is the retardation factor of component m , τ_l is the tortuosity, D is the molecular diffusion coefficient, λ_{lm} is the first order decay rate of component m , and y_{m-1} is the mass yield of daughter component m from parent



compound $m-1$. Equation (2) can be repeated for subsequent daughter products to form a complete decay chain. For example, at a site where tetrachloroethylene (PCE) undergoes decay to form trichloroethylene (TCE) and the TCE decays to form cis-1,2-dichloroethylene (DCE) which decays to form vinyl chloride (VC), PCE would be the parent compound ($m=1$), governed by Equation (1), while TCE, DCE, and VC would be daughter compounds governed by Equation (2) with $m=2, 3$, and 4 , respectively.

The semi-analytical method is based on the thermal conduction approximation developed by Vinsome and Westerveld (1980). Bear *et al.* (1994) and Falta and Wang (2017) adapted the Vinsome and Westerveld model to the problem of matrix diffusion in a semi-infinite aquitard using the mathematical analogy between transient heat conduction and transient matrix diffusion. The approach is based on the use of a fitting function to describe the concentration profile as a function of distance from the interface in the low permeability matrix:

$$C_{lm}(z_l, t) = (C_m^{t+\Delta t} + p_m z_l + q_m z_l^2) e^{-z_l/d_m} \quad (\text{MD3})$$

where $C_m^{t+\Delta t}$ is the current concentration of component m at the interface between high permeability and low permeability zones and p_m and q_m are fitting parameters. The concentration $C_m^{t+\Delta t}$ is the concentration that is solved for in each gridblock of a normal numerical simulation, and it is assumed to represent the average concentration in the high permeability part of the gridblock. The concentration penetration depth, d_m , is defined by:

$$d_m = \frac{\sqrt{\kappa_{lm} t}}{2}; \quad \kappa_{lm} = \frac{\tau_l D}{R_{lm}} \quad (\text{MD4})$$

The zero level of the concentrations in Equation (16) correspond to the initial (uniform) concentrations in the low permeability zone, which is usually zero. The fitting parameters p_m and q_m are determined by two conservation of mass laws. The first constraint requires the fitting function to satisfy the governing equation at the high permeability/low permeability interface:

$$R_{lm} \frac{\partial C_{lm}}{\partial t} = \tau_l D \frac{\partial^2 C_{lm}}{\partial z_l^2} \Big|_{z_l=0} + y_{lm-1} \lambda_{lm-1} C_{lm-1} \Big|_{z_l=0} - \lambda_l C_{lm} \Big|_{z_l=0} \quad (\text{MD5})$$

The component production term in Equation (5) is only used for daughter products ($m>1$).



In order to discretize the equation, a first-order finite difference approximation is applied to the time derivative. C_{lm} and C_{lm-1} are replaced with the trial functions on the right-hand side, which results in:

$$R_{lm} \left(\frac{C_m^{t+\Delta t} - C_m^t}{\Delta t} \right) = \tau_l D \left(\frac{C_m^{t+\Delta t}}{d_m^2} - \frac{2p_m}{d_m} + 2q_m \right) + y_{lm-1} \lambda_{lm-1} C_{m-1}^{t+\Delta t} - \lambda_{lm} C_m^{t+\Delta t} \quad (\text{MD6})$$

The second constraint is the mass conservation of the components in the low permeability material. This requires that the rate of change of total mass in the matrix is to equal the mass flux across the interface minus the rate of decay in the matrix plus the rate of production in the matrix. For an ideal semi-infinite aquitard case, the integral of distance into low permeability areas is defined from zero to infinity (Falta and Wang, 2017). This study deals with finite embedded heterogeneities, such as low permeability lenses and layers or fractured porous media with parallel fractures. Thus, the mass conservation constraint must account for a finite diffusion distance. The characteristic average diffusion length, L corresponds to the average maximum depth or distance of diffusion into the low permeability material. The low permeability matrix mass balance constraint becomes:

$$R_{lm} \frac{\partial}{\partial t} \int_0^L C_{lm} dz_l = -\tau_l D \left. \frac{\partial C_{lm}}{\partial z_l} \right|_{z_l=0} + y_{lm-1} \lambda_{lm-1} \int_0^L C_{lm-1} dz_l - \lambda_{lm} \int_0^L C_{lm} dz_l \quad (\text{MD7})$$

The concentration integrals in Equation (7) are directly related to the mass of the components in the low permeability material. The solution of the concentration integrals in Equation (7) using the trial functions have the form:

$$I_m(t) = \int_0^L C_{lm} dz_l = C_m(t) d_m + p_m d_m^2 + 2q_m d_m^3 - \left(C_m(t) d_m + p_m d_m L + p_m d_m^2 + q_m d_m L^2 + 2q_m d_m^2 L + 2q_m d_m^3 \right) e^{-L/d_m} \quad (\text{MD8})$$

Following Pruess and Wu (1988, 1993), this integral can be written as a weighted function of $C_m(t)$, p_m , and q_m :

$$I_m(t) = \delta_m C_m(t) + \gamma_m p_m + \beta_m q_m \quad (\text{MD9})$$

where

$$\delta_m = d_m - d_m e^{-L/d_m} \quad (\text{MD10})$$



$$\gamma_m = d_m^2 - (d_m L + d_m^2) e^{-L/d_m} \quad (\text{MD11})$$

$$\beta_m = 2d_m^3 - (L^2 d_m + 2d_m^2 L + 2d_m^3) e^{-L/d_m} \quad (\text{MD12})$$

For the special case of an ideal infinite aquitard, where $L \rightarrow \infty$, $\delta_m = d_m$; $\gamma_m = d_m^2$; $\beta_m = 2d_m^3$, corresponding to the definition of $I(t)$ in Falta and Wang (2017). Replacing the derivative in Equation (7) with a finite difference approximation of the concentration integral, and substituting C_{lm} with the fitting functions in the space derivative and decay and production terms gives:

$$R_{lm} \left(\frac{\delta_m C_m^{t+\Delta t} + \gamma_m p_m + \beta_m q_m - I_m^t}{\Delta t} \right) = \tau_l D \left(\frac{C_m^{t+\Delta t}}{d_m} - p_m \right) + y_{lm-1} \lambda_{lm-1} I_{m-1}^{t+\Delta t} - \lambda_{lm} (\delta_m C_m^{t+\Delta t} + \gamma_m p_m + \beta_m q_m) \quad (\text{MD13})$$

Solving the linear equations (6) and (13) by substitution provides the expressions for the fitting parameters p_m and q_m :

$$p_m = \frac{I_m^t + \frac{y_{lm-1} \lambda_{lm-1} I_{m-1}^{t+\Delta t} \Delta t}{R_{lm}} - E C_m^{t+\Delta t} - A_m \left(\frac{C_m^{t+\Delta t} - C_m^t}{2\kappa_{lm} \Delta t} - \frac{C_m^{t+\Delta t}}{2d_m^2} - \frac{y_{lm-1} \lambda_{lm-1} C_{m-1}^{t+\Delta t}}{2R_{lm} \kappa_{lm}} + \frac{\lambda_{lm} C_m^{t+\Delta t}}{2R_{lm} \kappa_{lm}} \right)}{\frac{A_m}{d_m} + B_m} \quad (\text{MD14})$$

$$q_m = \frac{\frac{(C_m^{t+\Delta t} - C_m^t) d_m^2}{\kappa_{lm} \Delta t} - C_m^{t+\Delta t} + 2d_m p_m - \frac{y_{lm-1} \lambda_{lm-1} C_{m-1}^{t+\Delta t} d_m^2}{R_{lm} \kappa_{lm}} + \frac{\lambda_{lm} C_m^{t+\Delta t} d_m^2}{R_{lm} \kappa_{lm}}}{2d_m^2} \quad (\text{MD15})$$

where

$$A_m = \beta_m + \frac{\lambda_{lm} \Delta t}{R_{lm}} \beta_m \quad (\text{MD16})$$

$$B_m = \gamma_m + \kappa_{lm} \Delta t + \frac{\lambda_{lm} \Delta t}{R_{lm}} \gamma_m \quad (\text{MD17})$$

$$E_m = \delta_m - \frac{\kappa_{lm} \Delta t}{d_m} + \frac{\lambda_{lm} \Delta t}{R_{lm}} \delta_m \quad (\text{MD18})$$



Unique values of p_m and q_m are calculated in the aquifer gridblocks containing low permeability zones, and they are updated at each time step to represent the changing concentration profile in the low permeability zones. The concentration integral $I_m^{t+\Delta t}$ is recalculated at every time step in every gridblock using Equation (9) and it is stored for use in the next time step (I_m^t). For the special case of diffusion in a semi-infinite aquitard with no reactions, the formulation becomes equivalent to the original Vinsome and Westerveld (1980) method for heat conduction and to the *Bear et al. (1994)* method for diffusion.

Calculation of Matrix Diffusion Mass Flow

The matrix diffusion mass flow entering (+) or leaving (-) the high permeability zone is described by Fick's first law of diffusion. Substituting the low permeability material concentration by the trial function:

$$\dot{m}_m = A_{md} \phi_l \tau_l D \left. \frac{\partial C_{lm}}{\partial z_l} \right|_{z_l=0} = A_{md} \phi_l \tau_l D \left(-\frac{C_m^{t+\Delta t}}{d_m} + p_m \right) \quad (\text{MD19})$$

where ϕ_l is the porosity of the low permeability material and A_{md} is the matrix diffusion area, defined as the interfacial area between the high permeability and low permeability zones. The equation for p_m can be rewritten as:

$$p_m = a_m C_m^{t+\Delta t} + b_m \quad (\text{MD20})$$

with

$$a_m = \frac{-E_m - \frac{A_m}{2\kappa_{lm}\Delta t} + \frac{A_m}{2d_m^2} - \frac{A_m \lambda_{lm}}{2R_{lm}\kappa_{lm}}}{\frac{A_m}{d_m} + B_m} \quad (\text{MD21})$$

$$b_m^{t+\Delta t} = \frac{I_m^t + \frac{A_m C_m^t}{2\kappa_{lm}\Delta t} + \frac{A_m y_{lm-1} \lambda_{lm-1} C_{m-1}^{t+\Delta t}}{2R_{lm}\kappa_{lm}} + \frac{y_{lm-1} \lambda_{lm-1} I_{m-1}^{t+\Delta t} \Delta t}{R_{lm}}}{\frac{A_m}{d_m} + B_m} \quad (\text{MD22})$$

Then the expression for the matrix diffusion mass flow rate, is:



$$\dot{m} = A_{md} \phi_l \tau_l D \left(\left(a_m - \frac{1}{d_m} \right) C_m^{t+\Delta t} + b_m^{t+\Delta t} \right) \quad (\text{MD23})$$

The matrix diffusion mass flow becomes a linearly concentration dependent source/sink term added in the numerical transport model gridblocks that contain (or are adjacent to) low permeability materials. The $b_m^{t+\Delta t}$ term (Equation (22)) that appears in the gradient expression for component m does not depend on the current concentration of m , but it does depend on the previous concentration and integral of m , as well as on the current concentration and integral of the parent compound, $m-1$.

The semi-analytical approximation for matrix diffusion with a finite diffusion length does not force the concentration gradient at the symmetry boundary $z_l = L$ to be zero. This might seem to be a significant limitation of the formulation, but we have not found that to be true in practice (Muskus and Falta, 2018). It appears that it is sufficient to accurately represent the concentration gradient at the interface, and the mass balance in the low permeability material. This observation is similar to that made by Pruess and Wu (1993) in their study of heat conduction in cubic matrix blocks using a similar semi-analytic method. They tried adding a cubic term to the fitting function so that the additional free parameter could be used to force the concentration gradient to be zero at $z_l = L$, but found that this formulation gave less accurate results than the two parameter fitting function. They observed that forcing the zero gradient condition at $z_l = L$ resulted in a less accurate representation of the gradient at the interface.



AIR-WATER INTERFACE ADSORPTION PACKAGE FOR PFAS IN THE UNSATURATED ZONE

Introduction to PFAS Behavior

The ITRC document on Environmental Fate and Transport for Per- and Polyfluoroalkyl Substances defines PFAS as a large group of compounds that vary in molecular weight and can have multiple structures and functional groups, and that have environmental and health effects. Therefore, understanding the fate and transport of a PFAS chemical in the environment is fundamental to its investigation and remediation at a contaminated site.

The four major sources of PFAS to groundwater include fire training/fire response sites, industrial sites, landfills, and wastewater treatment plants/biosolids. Other sources may be locally significant.

Transport mechanisms for PFAS compounds in the subsurface include advection and dispersion. Retardation of PFAS in the subsurface is affected by multiple partitioning mechanisms including hydrophobic and lipophobic effects, electrostatic interactions, and interfacial behaviors. The hydrophobic and lipophobic effects drive the association with organic carbon in soils. Electrostatic interactions are a function of the charge of the polar functional group at the head of the molecule. Therefore, at environmental conditions, PFCAs and PFSAAs (present as organic anions) are relatively mobile in groundwater but tend to associate with organic carbon fraction in soil. Thus, partitioning potential can be evaluated by the normalized distribution coefficient (K_{oc}), though other factors (pH and other geochemical conditions) may also affect PFAS sorption. In general, the longer chain PFAS compounds have a higher K_{oc} value compared to shorter chain PFAS compounds.

PFAS compounds exhibit surfactant-like behavior due to the hydrophobic and hydrophilic portions and form films at the air-water interface. This preference for the air-water interface may influence vadose zone transport, where unsaturated conditions provide significant air-water interfacial area.

PFAAs likely do not degrade or otherwise transform under ambient environmental conditions. Therefore, decay of PFAAs is not of significant concern in simulation analyses.

The ITRC PFAS web page (<https://pfas-1.itrcweb.org/>) provides several resources to understand PFAS issues in the environment.



Adding PFAS simulation capabilities to USG-Transport

USG-Transport has the capability to simulate saturated-unsaturated flow and transport of solutes in the subsurface using the Richards Equation, with van Genuchten moisture retention curves and Brooks Corey or van Genuchten relative permeability functions in addition to generalized tabular input of these functions for solving the flow equation. Processes of advection, dispersion and adsorption to soil are also simulated for multiple solute species in a three dimensional transient or steady-state flow-field. Thus, additional processes required for simulating transport of PFAS compounds in the subsurface include the complexities of retardation specific to PFAS compounds; specifically, adsorption of PFAS to the air-water interface.

PFAS adsorption to the air-water interface may also affect the flow properties of water. Specifically, the surface tension may be affected thus impacting the air-water capillary retention curve for flow in the vadose zone (termed surface tension driven flow), and the viscosity may be affected, impacting the hydraulic conductivity of the soil to the solution for saturated or unsaturated flow. These flow effects are generally significant only at high concentrations of PFAS.

The transport equation for a PFAS compound in saturated-unsaturated soil includes adsorption on the air-water interface as an additional storage term in the solute transport Equation (2). Thus, the mass storage term of Equation (3) is written as:

$$\frac{\partial(M)}{\partial t} = \frac{\partial(\phi_e S_w c)}{\partial t} + \frac{\partial(\rho_b c_s)}{\partial t} + \frac{\partial(M_p)}{\partial t} + \frac{\partial(c_{aw})}{\partial t} \quad (P1)$$

Where, c_{aw} is the net concentration of PFAS solute adsorbed onto the air-water interface $[ML^{-3}]$, and the other terms have been defined earlier. The concentration of PFAS adsorbed onto the air water interface may be expressed as Guo et al., (2020)

$$c_{aw} = A_{aw} K_{aw} c_w \quad (P2)$$

Where, A_{aw} is the specific air-water interfacial area $[L^2/L^3]$, and K_{aw} is the air-water interfacial adsorption coefficient $[L^3/L^2]$. It is noted that Silva et al., (2021) write the equation slightly differently for the solute concentrations in the air-water interface. Instead of a net concentration at the interface, they define a unit concentration at the interface Γ , $[ML^{-2}]$, where $c_{aw} = \Gamma A_{aw}$.



The air-water specific interfacial area, A_{aw} , is a function of water saturation empirically expressed as (Guo et al, 2020):

$$A_{aw} = x_2 S_w^2 + x_1 S_w + x_0 \quad (P3)$$

Where x_2 , x_1 , and x_0 are fitting parameters [1/L]. A simplified correlation of interfacial area to mean grain diameter was also presented by Brusseau et al., (2019) where A_{aw} is computed as:

$$A_{aw} = A_{max} (1 - S_w) \quad (P4)$$

The value of A_{max} is correlated to d the median grain diameter (in cm) as: $A_{max} = 3.9d^{-1.2}$ (Lyu et al., 2018).

Silva et al., (2021) present A_{aw} as a function of the moisture content as:

$$A_{aw} = \frac{\rho_w g}{\sigma_o} \int_{\theta_w}^{\theta_s} h \quad d\theta_w \quad (P5)$$

Where ρ_w is the density of water [ML⁻³], g is gravitational acceleration [LT⁻²], σ_o is the surface tension of water [MT⁻²], θ_s is the saturated water content (or porosity), and θ_w is the water content which is the porosity times saturation. Assuming constant porosity in equation (P5) gives:

$$A_{aw} = \frac{\rho_w g \phi}{\sigma_o} \int_{S_w}^1 h \quad dS_w \quad (P6)$$

Numerical integration can be performed on equation (P6) to get the interface area (not yet included in the code).

The partitioning coefficient for PFAS onto the air-water interface K_{aw} [L³/L²] is expressed as (Brusseau et al., 2019):

$$K_{aw} = \frac{1}{RT} \frac{\sigma_o b_{aw}}{a_{aw} + c_w} \quad (P7)$$

Where R is the universal gas constant (J/K/mol) = 8.314, T is temperature [K], and a_{aw} [mol/L³] and b_{aw} [-] are fitting parameters, and σ_o is the surface tension of water [dyn/cm or mN/m or 0.001kg/s²]. *Note that with this definition of K_{aw} the concentrations are defined in terms of moles of the solute and not mass of the solute. Multiplying equation (P7) and the coefficient a_{aw} by molecular weight of the solute will provide equation (P7) in terms of mass of solute.*



Substituting (P7) into (P2) gives

$$c_{aw} = \frac{1}{RT} \frac{\sigma_0 b_{aw}}{a_{aw} + c_w} A_{aw} c_w \quad (P8)$$

Silva et al., (2021) present a Langmuir type isotherm for partitioning at the air-water interface as:

$$c_{aw} = \frac{c_{\max,aw} K_{L,aw}}{1 + K_{L,aw} c_w} A_{aw} c_w \quad (P9)$$

Where $K_{L,aw}$ is the Langmuir coefficient [$L^3 M^{-1}$] and $c_{\max,aw}$ is the maximum unit concentration on the air-water interface [$M L^2$].

Essentially, equations (P8) and (P9) are Langmuir equations general form:

$$c_{aw} = \frac{A}{1 + B c_w} A_{aw} c_w \quad (P10)$$

Where A and B are the generalized Langmuir coefficients with equivalences to coefficients of equation (P8) or (P9).

As an alternate to the above equations to compute the air-water specific interfacial area as a function of water saturation, or use the Langmuir isotherm for the air-water interface partition coefficient, the most accurate approach would be to measure A_{aw} for the specific soil at the relevant water saturations and K_{aw} for the specific PFAS and its concentration in solution. There may be other functional forms that fit better and thus a tabular input of these functions provides flexibility in that they can be pre-calculated and entered into the model.

Guo et al., (2020) also present kinetic sorption domain models for sorption to soil as well as to the air-water interface where a fraction of sorbent adsorbed is instantaneous while the remaining is adsorbed in a kinetic manner which approaches the equilibrium condition and both processes are reversible. Brusseau (2021) details simulation of rate-limited multi-process retention of PFAS whereby kinetic processes govern transfer of PFAS into a non-advective domain from an advective domain with adsorption on soil and air-water interfaces in both domains.

Silva et al., (2021) also consider the impact of a PFAS solution on groundwater flow. Specifically, surface-tension of the solution is different than for water and the retention function is applied to a capillary head scaled by the ratio of surface tensions as:



$$h = h_o \left(\frac{\sigma}{\sigma_o} \right) = h_o \alpha_h \quad (\text{P11})$$

Where α_h is the pressure head scaling factor, and the surface tension of the solution is obtained as:

$$\frac{\sigma}{\sigma_o} = 1 - b_\sigma \ln \left(\frac{c_w}{a_\sigma} + 1 \right) \quad (\text{P12})$$

where a_σ [ML^{-3}] and b_σ [-] are constants determined by fitting (P12) to measured surface tension isotherms for the compound of interest. For PFAS of current environmental significance, solution concentrations more than 10 mg/L are generally needed before the surface tension effects contribute to unsaturated flow. AFFF solution concentrations of 50 mg/L or greater are needed for surface tension flow effects to contribute to infiltration of AFFF solutions (Silva et al., 2020).

In addition, Silva et al (2021) include solution viscosity effects as a function of solute concentration. The viscosity of AFFF solutions can be as much as 2 to 15 times greater than that of water, which will reduce the net hydraulic conductivities of the solution. Silva (2019) used a similar expression to the surface-tension function to model viscosity changes as a function of solute concentration as:

$$\frac{\mu}{\mu_o} = 1 - b_\mu \ln \left(\frac{c_w}{a_\mu} + 1 \right) \quad (\text{P13})$$

Where a_μ [ML^{-3}] and b_μ [-] are constants determined by fitting (P15) to measured viscosity isotherms for the compound of interest.

The impact of surface tension and viscosity effects on flow of PFAS solutions was awaiting validation (Silva et al., 2021).

Implementation into MODFLOW USG-Transport

Two processes need to be included into USG-Transport to enable simulation of PFAS transport in the vadose zone. First, the adsorption onto the air-water interface needs to be included in the transport equation. Secondly, the effect of PFAS solution on capillarity and viscosity need to be included in the flow equation. Note that the required modifications for transport are made to the BCT package as well as to the DPT package to accommodate mobile and immobile domains when simulating dual domain transport. Modifications required to the flow equation for surface tension driven flow may be accommodated in the BCF/LPF package as well as the DPF package when Richards' equation is simulated. The impact of



PFAS concentrations on viscosity of water may be turned on in the density driven flow (DDF) package as the viscosity impact on conductance is accommodated there. *The flow modifications (viscosity impact and surface tension driven flow) are not currently accommodated in USG-Transport and are only significant at high PFAS concentrations.*

Modification to Transport Equation

If a transport simulation includes air-water interface adsorption for any solute, the optional keyword A-W_ADSORB should be used in the first line of the BCT package (see input instructions of the BCT package for more details). Similarly, if the simulation includes a dual domain conditions, the DPT package may include the keyword A-W_ADSORBIM to indicate that air-water interface adsorption should also be included in the matrix of the dual domain simulation (see input instructions for the DPT package for details). The last term on the RHS of equation (P1) needs to be included into the transport equation of MODFLOW USG-Transport to include the effect of adsorption of a solute onto the air-water interface, with equation (P3) providing the equilibrium partitioning relation between the concentration in the interface and the water phase concentration. The partitioning onto the interface is a function of the interfacial area A_{aw} , which is a function of saturation either measured, or as per equations (P3), (P4), or (P6); and of the air-water partition coefficient K_{aw} , which is a function of the solute concentration either measured, as per equations (P7), (P9), or (P10). Integers that follow the keyword A-W_ADSORB indicate the option used for computing the air-water interface area (iarea_fn), and for computing the air-water partition coefficient (ikawi_fn)

Options for Air-Water Interface Area Computations

Many options are provided for calculating the air-water interface specific area, A_{aw} . The first option (iarea_fn = 1) is to use equation (P4) with a user defined A_{max} . It is repeated here as:

$$A_{aw} = A_{max}(1 - S_w) \quad (P14)$$

Where A_{max} is input by the user for each grid block. A second option (iarea_fn = 2) is to compute A_{max} using its correlation to d the median grain diameter (in cm) as given by Lyu et al., (2018). Thus,

$$A_{max} = 3.9d^{-1.2} \quad (P15)$$



A third option (iarea_fn = 3) for calculating the air-water interface specific area is to use equation (P14) with $A_{\max}$ calculated from equation (P6) and repeated here as:

$$A_{\max} = \frac{\rho_w g \phi}{\sigma_o} \quad (\text{P16})$$

For this case, the factor $\rho_w g / \sigma_o$ is input by the user and the code internally multiplies it by the porosity of each cell to use as the value of $A_{\max}$ in equation (P14).

A fourth option (iarea_fn = 4) for calculating the air-water interface specific area is to use the equation (P3) provided by Guo et al, (2020), repeated here as:

$$A_{aw} = x_2 S_w^2 + x_1 S_w + x_0 \quad (\text{P17})$$

Where x_2 , x_1 , and x_0 are fitting parameters [1/L].

A fifth option (iarea_fn = 5) for calculating the air-water interface specific area is to use tabular input of A_{aw} is a function of water saturation.

Options for Air-Water Interface Adsorption Computations

For evaluating the air-water interface adsorption coefficient, K_{aw} , the generalized Langmuir equation (P10) is used in USG-Transport repeated here as:

$$c_{aw} = \frac{A}{1 + B c_w} A_{aw} c_w \quad (\text{P18})$$

Where A and B are input by the user for each grid block. This is the first option (ikawi_fn = 1). For a linear partitioning case, this option may be used with B equal to zero and A equal to the linear air-water partition coefficient (L).

As a second option (ikawi_fn = 2), the coefficients A and B may be evaluated from Silva et al., (2021) as $A = c_{\max,aw} K_{L,aw}$, and $B = K_{L,aw}$, considering equation (P9) where $c_{\max,aw}$ and $K_{L,aw}$ are input by the user and A and B computed internally by the code. Note that here, B is the Langmuir coefficient $K_{L,aw}$.

As a third option (ikawi_fn = 3), the coefficients A and B may be obtained as per Brusseau et al., (2019) from equation (P7) as $A = \sigma_0 b_{aw} / RT a_{aw}$, and $B = 1 / a_{aw}$ where σ_0 / RT , a_{aw} and b_{aw} are input by the user and A and B computed internally by the code.



As a fourth option ($ikawi_fn = 4$), the partition coefficient K_{aw} of equation P2 is directly input in a tabular form as a function of water phase concentration of the solute species. This allows for any form of input for the partitioning function.

To accommodate site specific functions for A_{aw} and K_{aw} if available, as well as other generalized equations that relate interfacial area to saturation, and air-water partition coefficient for a solute to its concentration, USG-Transport provides the tabular input option for these relationships. The flag $iarea_fn = 5$ provides for tabular input of specific area versus saturation, and the flag $ikawi_fn = 4$ provides for tabular input of the air-water interface partition function versus solute concentration. This provides further flexibility for the various units of concentrations (moles, mg/L, ppm, etc) that may be used in simulations, and conversions may be accommodated as a preprocessing step by the user to provide the tabular input in appropriate consistent set of units for these relationships. as the concentration units of simulation.

Decay and Daughter Component Generation on Air-Water Interface

The solutes adsorbed on the air-water interface can decay in a similar manner to decay in water and on soil. Thus, decay on the air-water interface can occur as a zero-order process or as a first-order process (or both). Also, decay of a component can generate daughter components and the chain decay processes that occur to solutes in water and on soil also occur to the solutes on the air-water interface. Separate decay coefficients can be provided for decay in each of these domains (i.e., decay coefficients can be different in water, on soil and on the air-water interface).

PFAS in the Immobile Domain of a Dual Porosity Simulation

PFAS may also be simulated in the matrix of a dual domain (or dual porosity) simulation. The PFAS can migrate between the mobile and immobile domains, can adsorb on soil in the immobile domain, adsorb on the air-water interface in the immobile domain if it is simulated as saturated / unsaturated using Richards' Equation, and decay with generation of daughter components in water, soil and on the air-water interface in the immobile domain. The coefficients for adsorption and decay in the immobile domain may be variable in space and can be different from those in the mobile domain.

Modification to Flow Equation

The flow equation can be modified due to PFAS presence causing a change in the viscosity and surface tension of the solution. The flow and transport coupling is performed by USG-Transport through the density dependent flow package which may be used for concentration dependent viscosity coupling also, as it affects the hydraulic conductivity of the fluid. This coupling is significant only at high PFAS concentrations (not significant in most environmental conditions related to PFAS contamination) and is not currently available in USG-Transport.

Surface tension scaling may be provided for unsaturated zone flow, to the moisture retention curves as a function of concentration of PFAS. Equation (P14) can be used to provide the scaling factor and the coefficients a_σ and b_σ may be input for the PFAS compound. This coupling is also significant only at high PFAS concentrations (not significant in most environmental conditions related to PFAS contamination) and is not currently available in USG-Transport.

References

- Brusseu, M.L., 2021. Simulating PFAS Transport Influenced by Rate-limited Multi-process Retention, *Water Research* 168: 115179, Published online 2019 Oct 15. doi: [10.1016/j.watres.2019.115179](https://doi.org/10.1016/j.watres.2019.115179)
- Guo, B., J. Zeng, and M.L. Brusseau, 2020. A Mathematical Model for the Release, Transport, and Retention of Per- and Polyfluoroalkyl Substances (PFAS) in the Vadose Zone, *Water Resources Research* 56(2): [e2019WR026667](https://doi.org/10.1029/2019WR026667). Published online 2020 Jan 10. doi: [10.1029/2019wr026667](https://doi.org/10.1029/2019wr026667).
- Brusseu, M.L., N. Yan, S. Van Glubt, Y. Wang, W. Chen, Y. Lyu, B. Dungan, K.C. Carroll, F. O. Holguin, 2019. Comprehensive retention model for PFAS transport in subsurface systems, *Water Research* 148 (2019) 41-50.
- Peng, S., Brusseau, M.L., 2005. Impact of soil texture on air-water interfacial areas in unsaturated sandy porous media. *Water Resour. Res.* 41, W03021. <https://doi.org/10.1029/2004WR003233>.
- Lyu, Y., Brusseau, M.L., Chen, W., Yan, N., Fu, X., Lin, X., 2018. Adsorption of PFOA at the air-water interface during transport in unsaturated porous media. *Environ. Sci. Technol.* 52, 7745e7753.



Silva, J.A.K., 2019. Baseline Data Acquisition and Numerical Modeling to Evaluate the Fate and Transport of PFAS within the Vadose Zone, Final Report, SERDP Project ER18-1389.

Silva, J.A.K., J. Simunek, and J.E. McCray, 2021. A Modified HYDRUS Model for Simulating PFAS Transport in the Vadose Zone, *Water* 2020, 12, 2758; doi:10.3390/w12102758.

Silva, J.A.K.; McCray, J.E.; Martin, W.A. Final Report: Baseline Data Acquisition and Numerical Modeling to Evaluate the Fate and Transport of PFAS within the Vadose Zone. SERDP Project ER18-1389. Available online: <https://www.serdp-estcp.org/Program-Areas/Environmental-Restoration/Contaminated-Groundwater/Emerging-Issues/ER18-1389> (accessed on 10 September 2020).

NUMERICAL TREATMENT FOR TRANSPORT EQUATIONS

The unstructured grid discretization schemes have been discussed in the Groundwater Flow document of *MODFLOW-USG* (Panday et al, 2013). The BCT process solves for solute and/or heat transport using the same grid that was used for the flow solution. Gridding flexibility and details are provided by Panday et al, 2013).

The governing transport equations are discretized using a Control Volume Finite Difference (CVFD) approach similar to the flow formulation of *MODFLOW-USG*. A fully implicit solution scheme is used to provide solution stability for all Courant number conditions. The implicitness factor can be varied up to the Crank-Nicolson condition to provide a higher order temporal accuracy and yet keep the solution unconditionally stable. A Total Variation Diminishing (TVD) scheme is used for expansion of the advective term to minimize numerical dispersion without incurring unphysical numerical oscillations. The full dispersion tensor is expanded into its component directions with the principal dispersion components handled fully implicitly and the cross-dispersion terms included optionally on the right-hand-side of the matrix equation with iterative updates. The equations for the GWF and the CLN domains are assembled into one matrix and solved simultaneously at each time step in accordance with the flow modules of *MODFLOW-USG*. Numerical treatment of the BCT package is discussed below.

Numerical Expansion of the Transport Equation for an Unstructured Grid

Equation (2), with equation (9) providing the most general form of the mass storage term, can be expanded for a general finite volume discretization with cell n connected to cells m as (Forsyth et al., 1998):



$$\begin{aligned}
 & \frac{S_y \nabla_n}{\Delta t} [S_w c_n^{t+\Delta t} - S_w c_n^t] + \frac{\phi_{fc} \nabla_n}{\Delta t} [c_n^{t+\Delta t} - c_n^t] + \frac{S_w S_s \nabla_n}{\Delta t} [h c_n^{t+\Delta t} - h c_n^t] + \frac{\nabla_n}{\Delta t} \rho_b [c_{sn}^{t+\Delta t} - c_{sn}^t] = \\
 & \sum_{m \in \eta_n} \tilde{\theta}_w D_{nm} (c_m - c_n) - \sum_{m \in \eta_n} Q_{nm} c_{nm} - \nabla_n [\lambda_w \theta_w c_n + \lambda_s \rho_b c_{sn}] - \nabla_n [\mu_w \theta_w f(c_n) + \mu_s (1 - \phi_e) f(c_n)] + \\
 & \theta_w D_n^{cross} + \dot{M}
 \end{aligned} \tag{44}$$

Where

∇_n is the volume of grid block n ,

Δt is the time step size,

$\tilde{D}_{nm}$ is the inter-cell dispersion conductance term between cells n and m resulting from the principal components of the dispersion tensor,

c_n is the concentration of component species in water for cell n ,

c_{sn} is the concentration of component species in soil for cell n ,

$Q_{nm} c_{nm}$ is the inter-cell advection term between cells n and m ,

Q_{nm} is the net flux between cells n and m ,

D_n^{cross} is the cross-dispersion flux at cell n , and

θ_w is obtained from equation (6) for this case. Note that the storage term and water content are appropriately selected for the *MODFLOW* simulation case, for transient flow conditions.

The superscript on the left-hand side of equation (44) represents the time step value. The summation in the first and second terms on the right-hand side of the equation is over all m cells connected to cell n and represents the net dispersive and advective flux entering cell n , from all of the cell faces. The term $f(c_n)$ is a smooth function that reduces the zero-order decay rate smoothly to zero when the concentration approaches zero, to prevent the zero-order decay rate from removing more mass than is available. Currently, it is hardwired in the code to reduce the zero-order decay to zero, when concentration reduces below 0.01. A Newton expansion of this nonlinearity further facilitates higher order convergence properties and provides converged solutions that are mass conserved.



All terms on the right-hand side of equation (44) are expressed at the new time level for a fully implicit formulation. The Crank-Nicolson scheme expresses those terms half-way between the old time-step value and the new time-step value. In general, the right-hand side of equation (44), RHS_{44} can be expressed as:

$$RHS_{44} = \theta(RHS_{44})^{t+\Delta t} + (1 - \theta)(RHS_{44})^t \quad (44b)$$

Where θ is the time implicitness factor which is 1.0 for a fully implicit scheme; 0.5 for a Crank-Nicolson scheme; or can be any time factor in between. Note that a factor less than 0.5 may cause solutions that are not unconditionally stable.

The implicitness factor of a transport simulation can be optionally changed in USG-Transport using the TIMEWEIGHT option. An implicit factor of 0.5 gives the Crank-Nicolson Scheme. This semi-implicit scheme is applied only to the advective and dispersive terms and helps to reduce numerical dispersion in the solution, the closer it is to the Crank-Nicolson condition (TIMEWEIGHT = 0.5) than to the fully implicit condition (TIMEWEIGHT = 1.0). Boundary terms and decay terms are kept fully implicit therefore change of solute mass due to sources or sinks is not altered. In addition, the velocity of a transient flow simulation is also treated fully implicitly. In this manner, different terms use different degrees of implicitness.

Equation (11) is substituted into equation (44) to provide the adsorbed concentration in terms of the water phase concentration. For a non-linear adsorption isotherm, equation (11) is expanded using the Newton formulation to provide mass conserved solutions as noted by Huang et al, (1998). With coefficients for a cell and its connection being defined appropriately, equation (44) presents the CVFD expansion to the governing transport mass balance equation for a component species in a porous medium.

Expansion of the Advection Term

The term c_{nm} is the inter-cell concentration between cells n and m . A TVD expansion of this term is provided as (Forsyth et al., 1998):

$$c_{nm} = c_{nm}^{ups} + \frac{\sigma(r_{nm})}{2} [c_{nm}^{down} - c_{nm}^{ups}] \quad (45)$$

Where

c_{nm}^{ups} is the concentration of the upstream grid-block between cells n and m ,



c_{nm}^{down} is the concentration of the downstream grid-block between cells n and m , and

$\sigma(r_{nm})$ is the flux limiter, which depends on the smoothness sensor, r_{nm} .

A suitable flux limiter is given by van Leer as (Forsyth et al., 1998):

$$\sigma(r_{nm}) = \begin{cases} 0 & \text{if } r_{nm} \leq 0 \\ \frac{2r_{nm}}{1 + r_{nm}} & \text{if } r_{nm} > 0 \end{cases} \quad (46)$$

Where the smoothness sensor r_{nm} is defined as (Forsyth et al., 1998):

$$r_{nm} = \frac{c_{nm}^{ups} - c_{nm}^{2up}}{[L_{nm}^{ups} + L_{nm}^{2up}]} \cdot \frac{[L_n + L_m]}{c_{nm}^{down} - c_{nm}^{ups}} \quad (47)$$

Where

$2up$ is the second point upstream cell (the upstream cell to grid-block ups),

c_{nm}^{2up} is the concentration of grid-block $2up$,

$[L_{nm} + L_{mn}]$ is the perpendicular distance between cells n and m , and

$[L_{nm}^{ups} + L_{nm}^{2up}]$ is the perpendicular distance between cells ups and $2up$.

The upstream grid-block to node n in three-dimensional flow is selected as the connecting node with the highest inflow to node n . Similarly, the second point upstream node (upstream grid-block to node ups) in three-dimensional flow is selected as the connecting node with the highest inflow to node ups .

Note that equation 45 reduces to the upstream weighted implicit formulation when the last term is zero. In effect, therefore, the TVD scheme can be viewed as applying an upstream weighted formulation with a correction term that sharpens the numerical dispersion to the point that oscillations do not occur in the solution, as a result of the TVD property being maintained.

Expansion of the Dispersion Term

The inter-cell dispersion conductance term $\tilde{D}_{nm}$ is defined as the inter-block dispersion coefficient times the cross-sectional area across which the dispersion flux is evaluated, divided by the distance over which the concentration gradient is calculated (Zheng and Wang, 1999). Thus,



$$\tilde{D}_{nm} = \frac{A_{nm} D_{nm}}{[L_{nm} + L_{mn}]} \quad (48)$$

Where

A_{nm} is the perpendicular flow area between cells n and m ,

D_{nm} is the inter-block dispersion term between cells n and m , and

L_{nm} and L_{mn} are the perpendicular distances between the respective cell centers and the n - m interface.

See **Figure 5** for the definition of the geometric terms of equation (48) for various grid types.

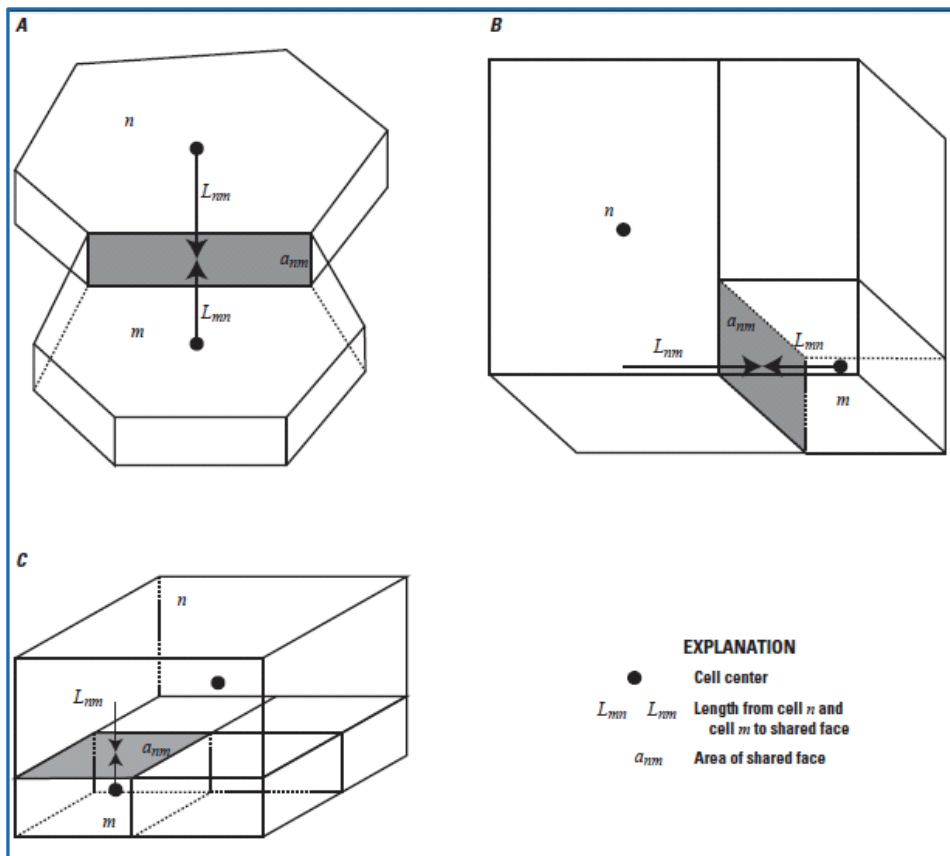


Figure 5. Definition of Geometrical Properties for Various Grid Types.

Since the dispersion term is resolved along its principal coordinate directions and associated cross components [equations 18 through 24], the orientation of the faces of unstructured grid-blocks is required as input to the simulation. For an unstructured grid-block n , **Figure 6** shows



the angle α_{nm} between the normal to face nm and the x-coordinate direction for any connecting grid block m . Note that $\alpha_{mn} = \pi + \alpha_{nm}$ because the normal outward direction is reversed. Therefore only the upper triangular portion of the matrix containing the α_{nm} values is stored. Also note that α_{nm} is internally computed (to be 0 or $\pi/2$ as appropriate) for a rectangular grid that is aligned with the principal coordinate directions.

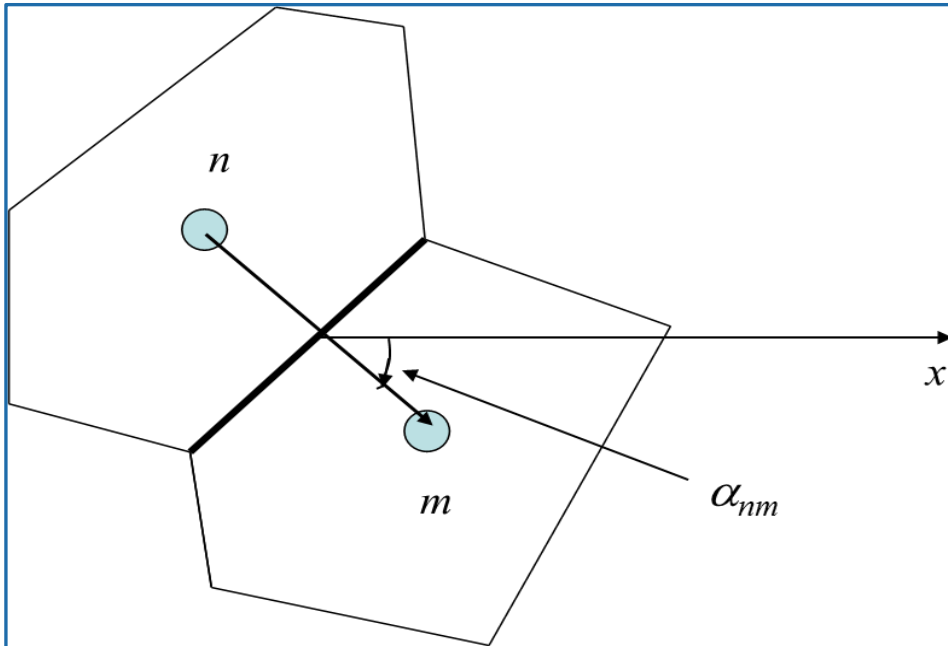


Figure 6. Angle between x-coordinate direction and the perpendicular to face nm .

The horizontal velocity components in each principal direction for a cell (v_x and v_y) are computed as a weighted average of the respective velocity component through all the faces of the cell with horizontal normals. The weighting in the x-direction is performed as per the square of the cosine of the angle between the perpendicular to the face and the x-coordinate direction (Figure 7).

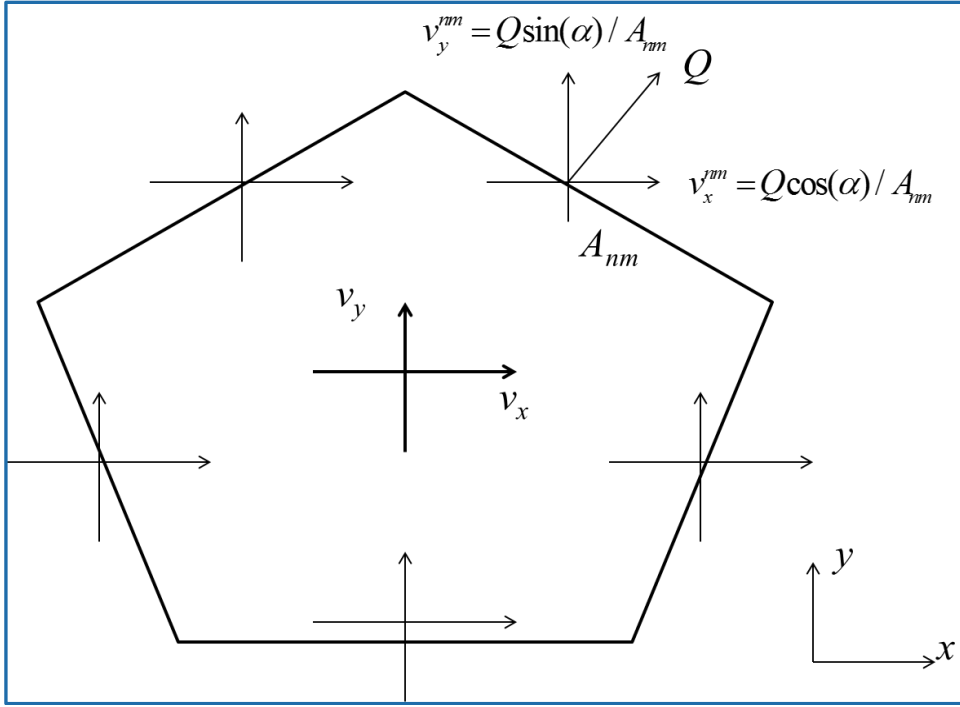


Figure 7. Schematic for computing the average x and y- direction velocities of a cell.

The weighting in the y-direction is performed as per the square of the sine of the angle α_{nm} . Thus, the velocity components of a cell along the horizontal principal coordinate directions are expressed at the center of the cell as:

$$v_x = \sum_{m \in \eta_{nh}} \left\{ v_x^m [\cos(\alpha_{mn})]^2 \right\} / \sum_{m \in \eta_{nh}} \left\{ [\cos(\alpha_{mn})]^2 \right\} \quad (49)$$

Where η_{nh} is the set of all connecting faces to cell m that have a horizontal normal direction, and

$$v_y = \sum_{m \in \eta_{nh}} \left\{ v_y^m [\sin(\alpha_{mn})]^2 \right\} / \sum_{m \in \eta_{nh}} \left\{ [\sin(\alpha_{mn})]^2 \right\} \quad (50)$$

It is noted that if a face is parallel to a velocity component (for example, the bottom face of the cell in **Figure 4** is parallel to the x-direction), then that component of the velocity across the face is zero because the two directions are parallel. Therefore, the weighting scheme applied above ensures that this face is not counted during averaging. It is also noted for rectangular grids, that the x-direction weighting for a face is maximum (one) and the y-direction weighting is zero when the angle α_{nm} of the face is zero. When the angle α_{nm} for a face is $\pi/2$, the x-direction weighting for the face is zero and the y-direction weighting is maximum (one). Therefore, the weighting scheme also ensures that the average computed velocity of a cell reduces appropriately for



rectangular grids oriented with the principal coordinate axis. The x- and y-direction components of velocity at each face are obtained from the flux across the face (determined during solution to the flow equation of the *MODFLOW-USG* flow modules) and the face angle as (see **Figure 4**):

$$v_x^{mn} = Q_{nm} / \cos(\alpha_{mn}) / A_{nm} \quad (51)$$

And

$$v_y^{mn} = Q_{nm} / \sin(\alpha_{mn}) / A_{nm} \quad (52)$$

Where

A_{nm} is the area of face nm . Substituting equation 51 into 49 and equation 52 into 50 gives the average velocity components in the x- and y-directions for a cell n , in terms of the flow across the cell faces, the face areas and the face angles.

The vertical direction velocity term v_z is treated in the *MODFLOW* manner – i.e., it is assumed that the layering is vertical and vertical distortion effects are ignored. Thus, v_z is computed as an average of the velocities across all faces identified as being normal to the horizontal direction. This is given as

$$v_z = \sum_{m \in \eta_{nz}} v_z^{mn} / \sum_{m \in \eta_{nz}} 1 \quad (53)$$

Where

η_{nz} is the set of all connected faces to cell n that are normal to the horizontal direction. Note that the z-direction velocity is computed from the inter-face flux as

$$v_z^{mn} = Q_{nm} / A_{nm} \quad (54)$$

Thus, substituting equation (54) into (53) gives the average z-direction flux component for cell n in terms of the vertical face fluxes and face areas. The average cell velocities obtained from equations 49, 50 and 53 are used to compute the principal components of the dispersion coefficient for each cell using equations 18, 19 and 20. The dispersion coefficients in the horizontal principal directions thus obtained, are then used to compute the dispersion coefficient across a horizontal connection nm . This can be done from tensoral flux considerations or from angle considerations using a weighted arithmetic mean rotation (see **Figure 8**), as:



$$D_{nm} = D_{xx} \cos(\alpha_{mn}) ** 2 + D_{yy} \sin(\alpha_{mn}) ** 2 \quad (55)$$

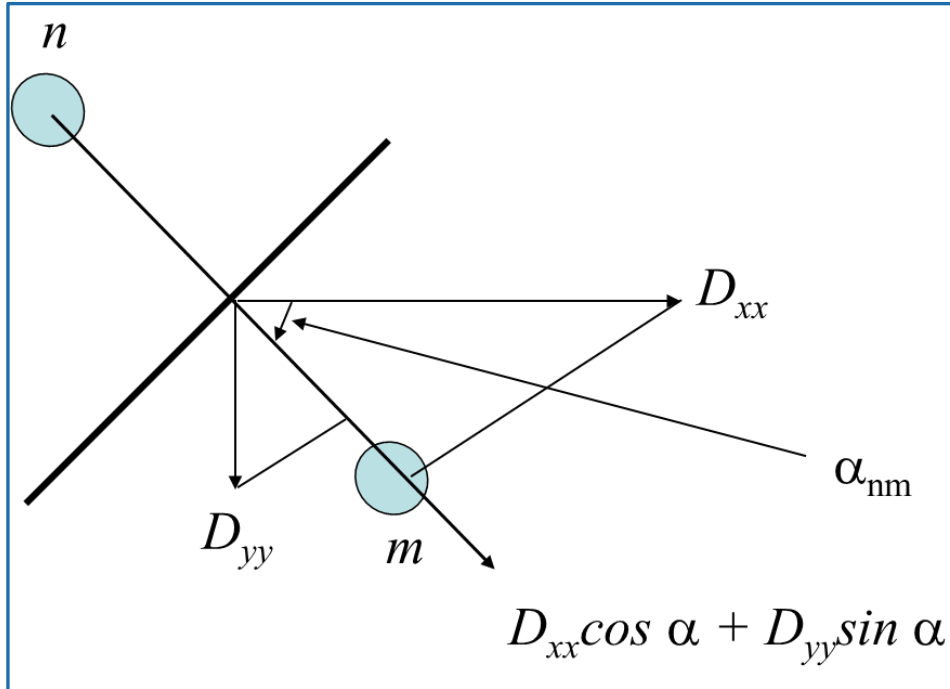


Figure 8. Principal dispersion components of cell m across face nm .

The tensoral derivation is similar to the anisotropic hydraulic conductivity implementation in MODFLOW-6 (Langevin et al, 2017) or SUTRA (Voss and Provost, 2010).

The vertical dispersion coefficient in the principal direction is treated in the *MODFLOW* manner – i.e., it is assumed that the layering is vertical and thus vertical distortion effects are ignored. Therefore, for a vertical connection, the principal component of the dispersion coefficient is expressed as

$$D_{nm} = D_{zz} \quad (56)$$

The net dispersion coefficient for the connection nm is the average of the dispersion coefficients for each of the connected cells obtained from equations 46 and 47. Note that the principal-component dispersion terms are expressed fully implicitly and therefore, the coefficients of equations 55 and 56 are applied to the left-hand side matrix. Also note that equation 55 reduces to the principal direction dispersion coefficients for a rectangular grid oriented along the principal coordinate directions.



The cross-dispersion terms are included with the transport equation as an option. They are included on the right-hand side vector and expressed as:

$$D_n^{cross} = \frac{\partial}{\partial x} \left[D_{xy} \frac{\partial c}{\partial y} \right] + \frac{\partial}{\partial y} \left[D_{yx} \frac{\partial c}{\partial x} \right] + \frac{\partial}{\partial x} \left[D_{xz} \frac{\partial c}{\partial z} \right] + \frac{\partial}{\partial y} \left[D_{yz} \frac{\partial c}{\partial z} \right] + \frac{\partial}{\partial z} \left[D_{zx} \frac{\partial c}{\partial x} \right] + \frac{\partial}{\partial z} \left[D_{zy} \frac{\partial c}{\partial y} \right] \quad (57)$$

The cross dispersion terms may be expanded in several ways. The methodology used here is to express all terms for a connection between two cells as components in the respective principal directions. Hence, equation (57) can be written in a CVFD form for any connection between cells n and m as:

$$D_n^{cross} = \sum_{m \in \eta_n} \left[\begin{aligned} & \left\{ A_{nm} \left[D_{xy} \frac{\partial c}{\partial y} \right]_{nm} \right\} \cos(\alpha_{mn}) + \left\{ A_{nm} \left[D_{yx} \frac{\partial c}{\partial x} \right]_{nm} \right\} \sin(\alpha_{mn}) \\ & + \left\{ A_{nm} \left[D_{xz} \frac{\partial c}{\partial z} \right]_{nm} \right\} \cos(\alpha_{mn}) + \left\{ A_{nm} \left[D_{yz} \frac{\partial c}{\partial z} \right]_{nm} \right\} \sin(\alpha_{mn}) \\ & + \left\{ A_{nm} \left[D_{zx} \frac{\partial c}{\partial x} \right]_{nm} \right\} + \left\{ A_{nm} \left[D_{zy} \frac{\partial c}{\partial y} \right]_{nm} \right\} \end{aligned} \right] \quad (58)$$

Equation (58) is obtained by taking the component along the connection between cells n and m , of the dispersion terms, $D_{yx} \frac{\partial c}{\partial x}$, $D_{xz} \frac{\partial c}{\partial z}$, $D_{yz} \frac{\partial c}{\partial z}$, $D_{zx} \frac{\partial c}{\partial x}$, and $D_{zy} \frac{\partial c}{\partial y}$ (see **Figure 9**). The value of each term on the right-hand-side of equation 58, at the face between the cells n and m , is computed as a distance weighted average of the value at each of the cells. Therefore, equation 58 expands the connectivity of a cell, to include all the connections of each of its connections and therefore, rather than expand the matrix, this cross term is implemented on the right-hand-side vector as stated earlier.

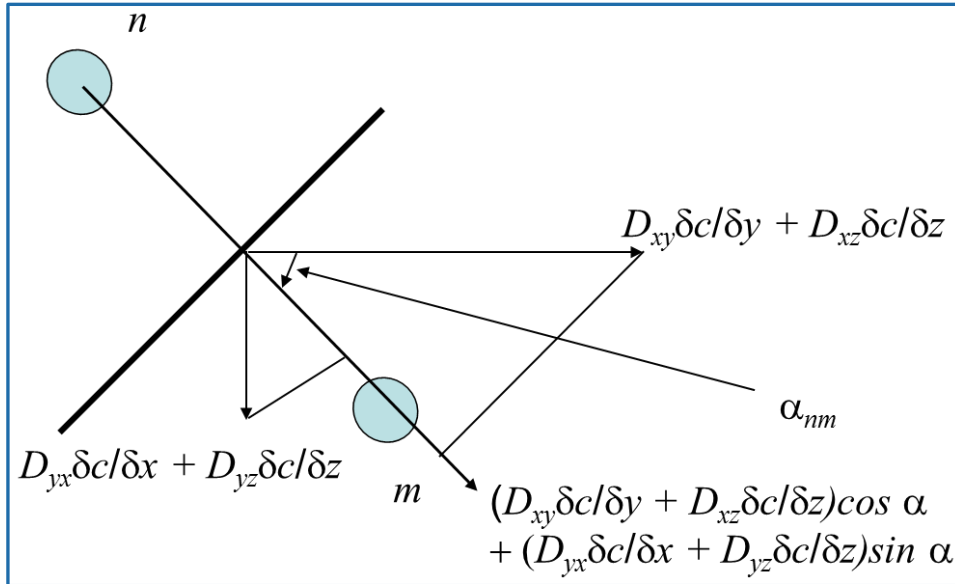


Figure 9. Cross dispersion components across face nm .

The terms $\partial c / \partial x$, $\partial c / \partial y$, and $\partial c / \partial z$ are computed for each cell as follows (see **Figure 10**). For each m^{th} connection of cell n , the concentration gradient between the cells $\left[\frac{\partial c}{\partial L} \right]_{nm}$ is computed as

$$\left[\frac{\partial c}{\partial L} \right]_{nm} = (c_n - c_m) / (L_{nm} + L_{mn}) \quad (59)$$

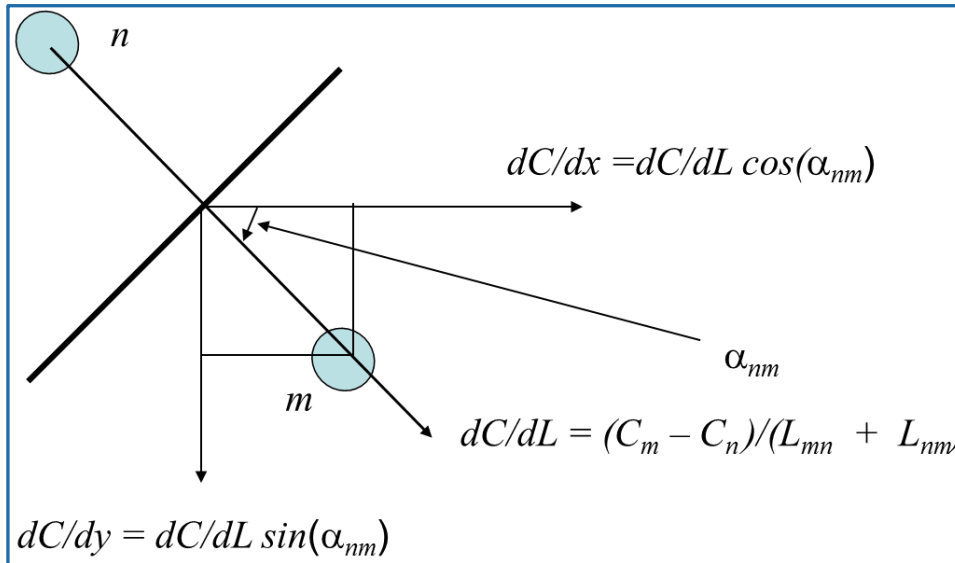


Figure 10. dc/dx and dc/dy for each face.

The x- and y-components of this gradient are resolved as per **Figure 10**, to give:



$$\left[\frac{\partial c}{\partial x} \right]_{nm} = \left[\frac{\partial c}{\partial L} \right]_{nm} \cos \alpha_{nm} \quad (60)$$

and:

$$\left[\frac{\partial c}{\partial y} \right]_{nm} = \left[\frac{\partial c}{\partial L} \right]_{nm} \sin \alpha_{nm} \quad (61)$$

The derivative terms for each cell are then obtained as a weighted average of the terms across each of the cell's face, in a similar manner to how the velocity components v_x , and v_y were computed for a cell in equations 49 and 50. Hence,

$$\frac{\partial c}{\partial x} = \sum_{m \in \eta_{nh}} \left\{ \left[\frac{\partial c}{\partial x} \right]_{nm} \text{abs}[\cos(\alpha_{mn})] \right\} / \sum_{m \in \eta_{nh}} \{ \text{abs}[\cos(\alpha_{mn})] \} \quad (62)$$

and

$$\frac{\partial c}{\partial y} = \sum_{m \in \eta_{nh}} \left\{ \left[\frac{\partial c}{\partial y} \right]_{nm} \text{abs}[\sin(\alpha_{mn})] \right\} / \sum_{m \in \eta_{nh}} \{ \text{abs}[\sin(\alpha_{mn})] \} \quad (63)$$

The z-direction gradient $\partial c / \partial z$ is treated in the *MODFLOW* manner – i.e., it is assumed that the layering is vertical and vertical distortion effects are ignored. Thus, $\partial c / \partial z$ is computed as an average of the vertical gradients across all horizontal faces (i.e., with a vertical normal). This is given as

$$\frac{\partial c}{\partial z} = \sum_{m \in \eta_{nz}} \left[\frac{\partial c}{\partial z} \right]_{nm} / \sum_{m \in \eta_{nz}} 1 \quad (64)$$

Note that these terms reduce to the regular formulation for a finite difference grid aligned with the principal x- and y-directions since $\sin \alpha_{nm} = 0$ and $\cos \alpha_{nm} = 1$ for the rectangular grid faces.

The contribution of Alden M. Provost is acknowledged in developing the numerical treatment for dispersion presented above. Alden has developed a sophisticated cross-dispersion scheme implemented in XT3D (Provost, Personal Communication), and used in MODFLOW 6 (Langevin, et al. 2017) which may be implemented in a later version.



Numerical Treatment of Dual Porosity Transport (DPT) Package

For dual domain transport evaluations, the flow field is only representative of the mobile domain. Therefore, the mobile domain fraction, f_m , is multiplied into the storage terms of the flow equation terms appropriately, for analysis of transport in a transient flow field. In addition, an option exists to interpret the input hydraulic conductivity as that representing the fracture (mobile) domain only. In that case, the hydraulic conductivity per unit total volume is obtained by multiplying the fracture hydraulic conductivity by the mobile domain fraction, during solution to the flow equation.

The dual domain transport formulation of equation (25) is similar to the formulation for transport in a porous medium of equation (2). The differences include multiplication of the storage, advection, dispersion and decay terms by the mobile domain fraction, f_m , and the additional source/sink terms resulting from mass transfer with the immobile domain, Γ . The former is incorporated into the numerical treatment for porous medium transport, by performing the multiplication by f_m , in the read and prepare stages of execution. The latter is implemented via equation 29, which is a function of the concentration of the mobile and the immobile domains.

Numerical Treatment of Density Dependent Flow (DDF) Package

For density dependent flow and transport evaluations, the only additional input required are the density of freshwater, the reference density of the solution, and the reference concentration for which the reference density is provided. During read and prepare, the density of the solution is computed for initial concentration conditions. Then, after filling the matrix and right-hand side vector for the flow equation, the density term updates are made using equation 41 for the flow term. The flow equation is in terms of HH and therefore, the modification only requires that the matrix be multiplied by $\left[\left(\rho/\rho_0\right)\right]$, and the term $-K_o\left(h_\phi - z\right)\nabla\left(\rho/\rho_0\right)$ is added to the right-hand side vector; this term can also be further expanded to treat the coefficient of h_ϕ in an implicit manner. At the same time, the term that computes mass storage due to density changes in and equations 42, and 43 are also added to the right-hand side vector. The density of water column between two cells in equation (41) is the average density of water in each of the cells, the head at the interface of two cells (the additional term in equation 41) is the average head between the cells, and the average interface elevation is the average of the z-elevation of the cells. This average may be obtained as an arithmetic average, or optionally as a vertical thickness weighted average. The vertical thickness weighted average would generally be more appropriate to use.



The density term is updated in a time-lagged manner. The density coupling is generally taken to be weak, and this time-lagged update was noted to be sufficient (Guo and Langevin, 2002).

The contribution of Christian D. Langevin is acknowledged in developing the numerical treatment for the DDF Package presented above.

Numerical Treatment for CLN Transport

The equations governing transport through the CLN domain and mass transfer with the GWF domain are expressed by equations 32 and 34. These equations are solved simultaneously with those of the porous medium domain by assembling all equations into one matrix in a similar manner to solution of the flow equations by *MODFLOW-USG*.

Equation 32 for the CLN cells may be numerically expanded for a general finite volume discretization in a manner similar to equation 44 for matrix nodes. The advective term for transport through the fracture or for transport between matrix and fracture is expanded using the implicit TVD formulation in a similar fashion to equations 45, 46 and 47. Note that the upstream and second-point upstream locations as required by the TVD scheme may be in either GWF or CLN domain, depending on consideration of flux through all the connected cells. The dispersion term for transport through the fracture or for transport between matrix and fracture is expanded as in equation 48. Only a longitudinal dispersion component is associated with CLN-CLN interactions and with CLN-matrix interactions. Therefore, the inter-block dispersion term is simply treated as in equation 56, with equation 18 providing the dispersion coefficient for only a non-zero longitudinal dispersivity. Note that for CLN-CLN interactions, the terms L_{nm} or L_{mn} in equations 47 and 48 are each computed as half the length of the CLN cell. For CLN-matrix interactions, the terms L_{nm} or L_{mn} are twice the hydraulic radius of the CLN geometry (equal to radius of a cylindrical geometry) from the CLN side, and the effective cell radius computed using the extended Peaceman formulation provided in the *MODFLOW-USG* Flow Documentation, from the matrix side.

Numerical Treatment of Matrix Diffusion Transport Package

The MDT Process for *MODFLOW-USG* is implemented directly into the code as subroutines and modules that are called from within the main program or within the BCT solute transport program. The program design for *MODFLOW-USG* is expanded to include appropriate calls from the main program for allocating the required arrays, reading and preparing the information,



and formulating and solving the governing transport equations. **Figure 11** is flow chart that illustrates how the MDT subroutines are implemented in the *MODFLOW-USG* code.

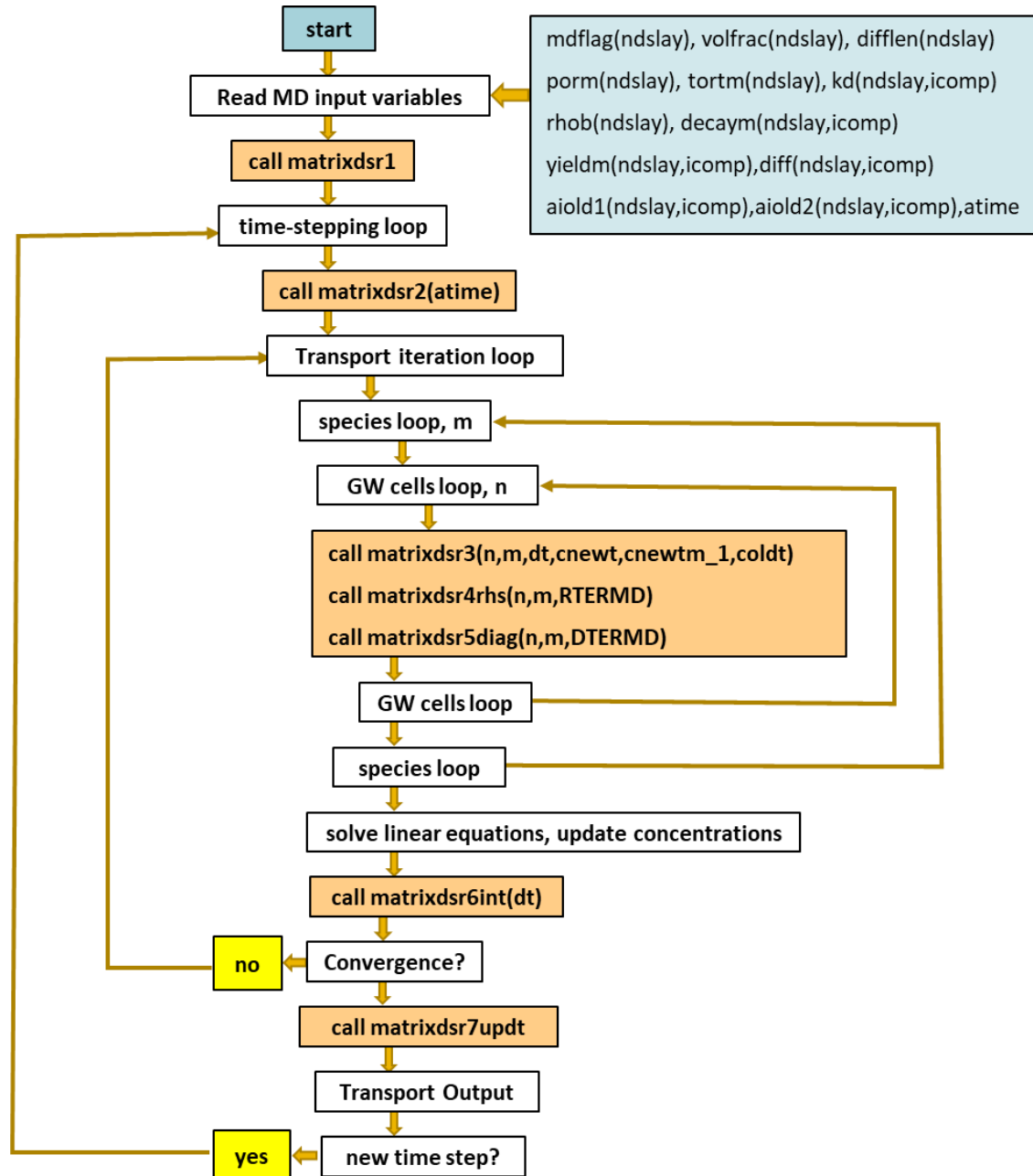


Figure 11. Flow chart showing the MDT subroutine implementation in *MODFLOW-USG*.

The code structure for the BCT process is discussed under transport implementation and program design further below.



Output Files from MDT Package

Concentration related data output for the MDT package is included in the MODFLOW-USG transport concentration output files. The binary output files for the MDT package include the coefficients AIOLD1MD and AIOLD2MD, which are used to compute the polynomial coefficient curves for the matrix. These coefficients are useful for restarting a simulation. These arrays are written to the binary file if the flag and unit number IMDTCF is greater than zero. The header for output is 'COEFF AI1MD' or 'COEFF AI2MD' for a single species simulation. For multiple species, the species number is appended at the end of the text (for example ('COEFF AI1MD 01', 'COEFF AI1MD 02', 'COEFF AI1MD 03', etc.).

If the flag and unit number IMDTCB is greater than zero, then the rate of mass diffusion between fracture and matrix domain are written to a binary file under the title ' MD MASS STORAGE'. For a multi-species simulation, the rate of mass transfer between matrix and fracture domain is output sequentially for each species.

Starting an MDT Run with Non-zero Matrix Concentrations

In many cases, the transport simulation may start under current site conditions and not include the historical conditions that would load solutes into the matrix domain. Since the analytical solution of matrix diffusion starts only with zero initial solute concentration in the matrix domain, it must be loaded up with solute concentration prior to running the MDT transport simulation. The analytical solution of the MDT routines uses interpolation coefficients computed from the loading mass and time of loading (AIOLD1MD and AIOLD2MD) to define the concentration profile within the matrix domain and there is no direct concentration input to initialize the matrix domain with solutes. Therefore, these coefficients need to be generated for non-zero conditions so they can be subsequently used in the transport simulation. Reasonable assumptions can be made regarding the solutes in the matrix, considering the history and solutes within the mobile domain. Specifically, the matrix domain can be appropriately loaded with solutes, given solute concentrations within the fracture domain, and a residence time over which the solutes may have bled into the matrix domain. The following steps may be used to achieve this:

Step 1: Set up initial concentrations in the mobile domain to match solute concentrations at each cell, which would load up the matrix domain, and set up a transport simulation for the historical duration required. Use the flag ITRNSP = 5 (set in the BCT package input) and run the model for the number of years required to initialize the matrix. Under



this flag, the flow computations are skipped entirely, the initial concentrations of the mobile domain are set to prescribed concentrations, and transport simulation is performed for a saturated domain with zero flow, to allow only diffusion from the mobile domain into the matrix domain to occur. The outputs from the MDT routines are AIOLD1MD and AIOLD2MD instead of concentrations within the matrix domain. These coefficients are used by the MDT routines to establish solutes within the matrix domain and may be used in a subsequent simulation of transport.

Step 2: The AIOLD1MD and AIOLD2MD arrays generated for each solute in Step 1 are extracted and inserted into the MDT input file. Note that these coefficients are zero and therefore not read when starting a simulation with zero solutes in the matrix domain. The time of loading from the previous run in Step 1 is also required to define the solute profile of the matrix domain. The ITRNSP number is changed back to the original number and the forward MDT model is run as originally designed with the matrix now being loaded up.

Numerical Treatment of Localized Flow Errors in Transport Solution

The mass conservative transport equation is based on the premise that the flow model is balanced throughout the simulation domain. The equation assumes that the net flow of water into a cell is balanced by net outflow (plus storage for transient flow-fields), and does not even consider that numerically it may be otherwise. However, in complex simulations, there can be significant localized flux balance errors that can sometimes have severe consequences to the transport solution. Ideally, it is advisable to reduce flow errors in such cases. However, that may not be possible or practicable throughout the domain of large simulations with hundreds of thousands of cells. A methodology is developed here which is optionally implemented into the code, to include these flow errors in the computation of advective solute transport such that they do not cause errors (often unnoticed) in solute concentration computations. The scheme reflects flow conservation errors as transport mass conservation errors which are then summarized in the mass balance table and output for every grid-cell in the domain (using the same binary formats as for the HDS, DDN, or CON files) for further review and evaluation.

For an implicit upstream weighted scheme, the advective flux of solute from a cell to an adjacent cell is computed by inserting the flux value of a cell computed from the flow solution into the discretized coefficient matrix at the upstream location. Thus, all outflow from a cell is summed into the diagonal term (the cell itself is upstream for outflow faces) while all inflows into the cell



are applied to the appropriate off-diagonal location of the row representing the cell. Ignoring sources and sinks as well as transient flow storage effects for the moment, the advective flux matrix off-diagonals balance the diagonal term if there is no balance error in the flow computations for that cell. However, for cells where head convergence was not sufficient there may be flow balance errors which are quantified as a residual.

The suggested methodology is to quantify the impact of that residual on transport mass balance by adding the residual flux to the diagonal term of the coefficient matrix. Effectively, a positive flux error is dissipated as a sink and a negative flux error is balanced by smaller outflow amounts thus solving the solute transport equations with a physically balanced representation of the fluxes. Furthermore, transport solution errors resulting from flow-field errors are back-calculated after solution to the transport equation as resulting concentration times the flux error at each cell. The error is also added up and presented in the transport mass balance summary in the LST file. Details are published by Panday et al. (2017).

The contribution of Christian D. Langevin and Vivek Bedekar is acknowledged in developing the numerical treatment of localized flow errors as presented above.

TRANSPORT IMPLEMENTATION AND PROGRAM DESIGN

The BCT Process for *MODFLOW-USG* is implemented directly into the code as subroutines and modules that are called from within the main program only. The program design for *MODFLOW-USG* is expanded to include appropriate calls from the main program for allocating the required arrays, reading and preparing the transport related information, and formulating and solving the governing transport equations. Solution for transport immediately follows solution for the required flow-field so the code does not need to read a cell-by-cell flow file to generate the required transport velocities – options are however available to do so using steady or transient flow-fields. This section describes how the BCT Process is implemented into the *MODFLOW-USG* code. The contribution and discussions with Christian D. Langevin are acknowledged in designing the program.

Program Structure

The program structure for the BCT Process modules is similar to that of *MODFLOW-USG*. Array allocation and read-and-prepare subroutines are called from within the main code as per the modular *MODFLOW-USG* structure. The code passes control to the transport solution modules via subroutine GWT2BCT1SOLVE immediately after the flow solution. For a steady-state



flow-field, the flow equations are by passed after the first time-step of every stress period. Thus, a mixture of steady and transient flow-fields can be accommodated for different stress periods in accordance with the associated flow solution option. Options are also available to read in the flow-fields from a previously executed flow model to perform the transport calculations.

The subroutine GWT2BCT1SOLVE accommodates solution to the transport equation. This subroutine controls formulation, solution, and mass balance computations for every component species at any time step of a simulation. The flowchart for subroutine GWT2BCT1SOLVE is shown **Figure 12** and **Figure 13**. As noted on **Figure 12**, the required arrays are initialized and terms that are constant for the current time step are first computed before entry into a component species loop. For each component species, the transport equations are then formulated and solved, till convergence of the nonlinear transport terms (nonlinear adsorption and the TVD scheme contribute to this nonlinearity), or till the maximum number of iterations is reached. A separate species loop is then set up to compute mass balance and output the transport results for all components. A separate loop is maintained for the mass balance computations to accommodate future enhancements that may include an outer nonlinear loop to compute nonlinear interactions among transport species. The shell for this outer loop is already developed in the code; therefore, any reaction module can be readily implemented using an operator splitting approach.

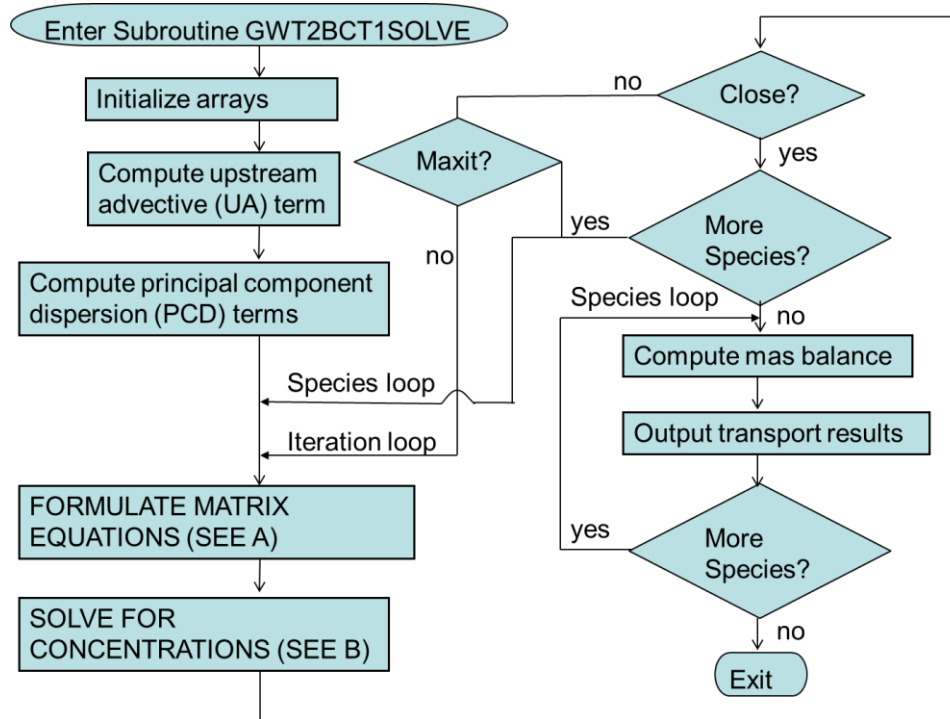


Figure 12. Flowchart for Subroutine GWT2BCT1solve.

As shown on **Figure 12** the terms that are constant for the current time step are first assembled into the coefficient matrix. These include the upstream advective term, and the principal components of the dispersion term. It is noted that these terms are constant throughout a simulation, if the flow-field is steady-state and therefore, their repeated computation may be skipped for such conditions. After filling the upstream weighted terms, the matrix is further formulated as noted on **Figure 13**. The cross-dispersion terms are optionally included in the right-hand-side vector, followed by assembly of the TVD terms. The storage and decay terms are then assembled into the matrix equations followed by assembly of the dual domain equation for such situations. The formulation is finalized by entering the boundary conditions into the matrix equations. The equations are solved by first condensing the immobile domain terms into the mobile domain using a partial Gauss elimination and then doing a back-substitution for the immobile domain concentrations after matrix solution. The linear matrix is solved using the same solvers used for the flow equations of *MODFLOW-USG*. Note that the transport equation is asymmetric and therefore, if the symmetric PCG schemes from the solver were used for the flow solution, the code automatically selects an asymmetric scheme for the transport solution. Aside from this and the closure tolerance for concentrations, solver parameters for transport are the same as those used for the flow simulation, as set in the SMS input file.

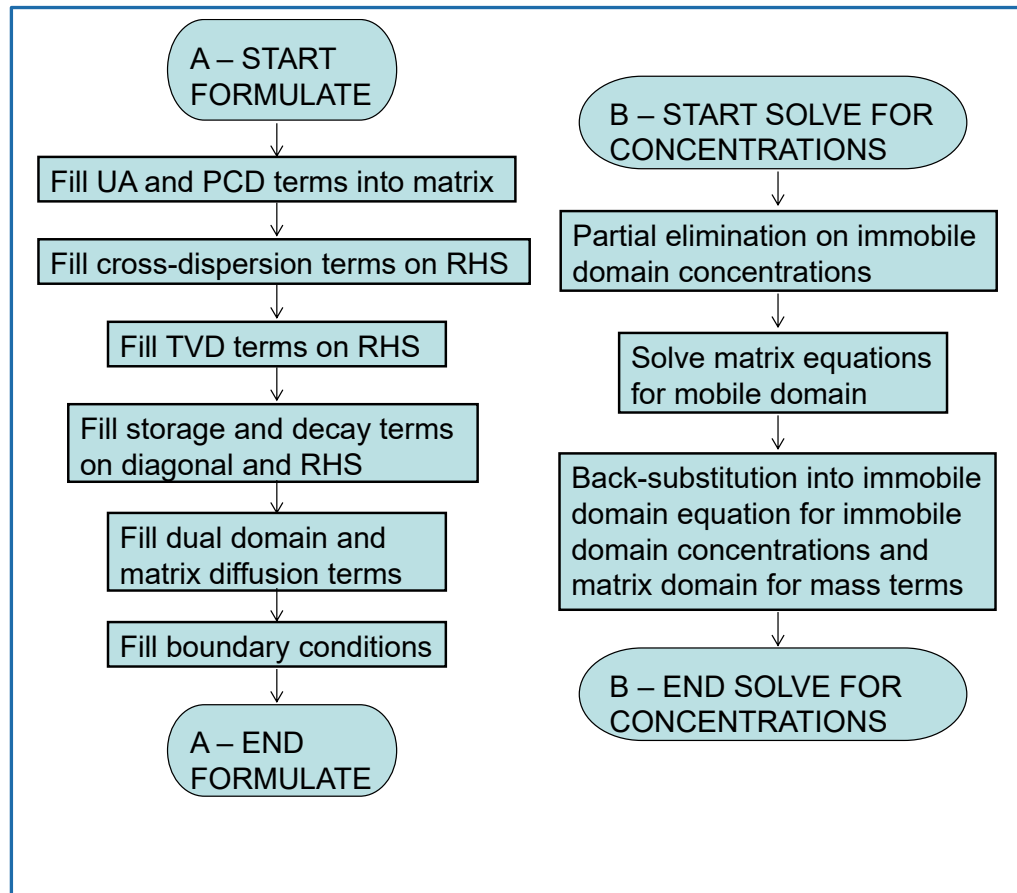


Figure 13. Flowchart for Subroutine Formulate.

Internal Array Storage and Precision of Variables

The array storage methodology for the BCT process is similar to that of *MODFLOW-USG*. All material property arrays are one-dimensional with one element for each node of the simulation. Therefore, all material properties can be heterogeneous and can vary on a node-by-node basis. The compressed sparse row (CSR) format is used for the coefficient matrix and for information required for each connection. With the CSR format, all nonzero coefficients in a row of the matrix are stored sequentially, beginning with the first one in the row and ending with the last. Pointers IA and JA index the location within the array, where the information for rows and columns of the matrix are stored. This is identical to the array structure used for the flow equation of *MODFLOW-USG*, and therefore, the pointer arrays are not duplicated.

Mixed precision is used for the BCT process. Variables are declared as **Real** or **Integer** depending on their purpose. In addition, **Double Precision** is used where needed. Parameter

values are stored in single precision to save on memory allocation. The coefficient matrix, right-hand-side vector, and arrays for concentrations of chemical species at old and new time steps are stored in double precision formats to provide accuracy required for matrix inversion.

Model Input and Output

The input structure for the BCT process follows that used by the flow modules of *MODFLOW-USG*. A transport simulation requires all the flow simulation files in addition to files required specifically for transport. For the GWF domain, a transport simulation requires a BCT input file for porous matrix transport, a DPT input file for dual porosity transport, a MDT input file for matrix diffusion transport, and a file named PCN to supply prescribed concentration input. Additional input required for transport within the CLN domain is supplied directly within the CLN package input file (a separate file is not required). The *NAME* file of *USG-Transport* is used to define all input files. The BCT, DPT, MDT and PCN conditions are simulated if the associated keywords are present in the *NAME* file. Since the transport simulation uses the same unstructured grid used for flow, a separate discretization file is not needed. Also, the boundary condition files used for flow are extended to provide the associated transport mass flux boundary condition, therefore, separate boundary condition files are not required, except as noted above, for providing prescribed concentrations to the simulation.

Input of Transport Boundary Conditions

The transport boundary conditions are input as auxiliary variables within each of the respective boundary input files (see the *MODFLOW-2005* documentation by Harbaugh, 2005, for details on auxiliary variables and their use). Multiple auxiliary variables may be used for multiple chemical species or for other identification or grouping purposes. If an auxiliary variables name is "Cxx_" (where xx is a two digit number and _ is a blank-space), that auxiliary variable is identified as the concentration of a species (species identification number being the two digits following the "C") to be used with the transport module. For example, the name C01 identifies the auxiliary variable as the concentration of species 1; C02 identifies it as the concentration of species 2 and so on. Note that the fourth space in the auxiliary variable name indicated above is a blank that acts as a separator distinguishing it from a name that has "Cxx" as part of a longer auxiliary variable name. Also, if an auxiliary variable is not provided to define inflow concentration for any solute species, then the concentration is defaulted to zero.



The concentrations supplied in the input file are multiplied by the fluid flux at inflow nodes, to provide the respective solute mass influx. Note that outflow cells do not require a boundary concentration, and if one is provided in the input file, it is ignored. Therefore, transport boundary conditions are not provided at purely outflow boundaries such as the EVT and DRN packages. Finally, prescribed head nodes for a flow simulation should be provided using the CHD input package, so that the inflow concentrations can be provided via auxiliary variables therein. If prescribed heads are provided to the flow simulation via negative values of the IBOUND array in the BAS package, it is not possible to provide concentrations at inflow nodes with the current BCT packages. In that case, the code assumes that the concentration of species in the inflowing water is zero.

Transport Output Files

The output structures for the BCT routines are similar to those used for flow with *MODFLOW-USG*. The main listing file reflects the simulation data input for transport along with input details for the flow simulation. Before transport simulation begins, a printout of the initial mass in the domain is also provided for each component species. A printout of the concentrations and transport mass balance is provided for each time-step, if requested, after printout of the appropriate flow simulation output. Convergence behavior for the transport solution is also output appropriately in the listing file.

Binary output files are also generated for output concentrations and the cell-by-cell mass flux terms. Separate files are generated for the immobile domain, if dual domain transport is simulated, with output options provided in the DPT input file. Similarly, separate files are generated for output from the MDT routines, as directed in the MDT input file.

By default, all species concentration outputs are placed into one binary output file if multiple species exist in a simulation. A "MULTIFILE" option is provided that further separates concentration output by species, for easier file handling. The binary output files use the same utility programs as used by the flow modules and hence output formats are identical. The concentration output file header contains the word "CONC" if the simulation is for only one species. If multiple species are simulated, the binary concentration file contains the header "CONC01", "CONC02", etc., for the various species simulated. The concentration output can be in double precision, if the DPOUT option is used.



Cell-by-cell binary files for mass budgets of the various component species are also output from transport simulations. All component budgets are output only in one binary file. The mass budget file includes storage, decay, generation, and boundary in/out terms. The cell-by-cell transfer of mass through advection and dispersion are not output with the mass budgets. Component species mass balance output for the various supported flow boundary types are in single precision and may be provided in either matrix format or compressed format as per *MODFLOW* conventions.

GUIDANCE FOR USING THE BCT PROCESS

The BCT process and its packages extend the capabilities of *MODFLOW-USG* to simulate fate and transport of multiple component species using unstructured grids. A thorough understanding of the formulation, governing equations and numerical approximations is crucial for obtaining accurate solutions and understanding code limitations. Also, since the BCT process follows the structure and formats of the flow modules, experience with flow simulations using *MODFLOW-USG* is essential for understanding gridding concepts, creating the datasets and running the model. Flow module details are available in Panday et al. (2013). This section provides guidance for applying *MODFLOW-USG* with the BCT process to solve for fate and transport of component species in the subsurface.

General Considerations for Grid Design and Temporal Discretization

Obtaining an accurate numerical solution to the equations governing transport in the subsurface is not a simple matter. Several schemes are available to discretize and solve the governing transport equations including implicit and explicit temporal discretization schemes, upstream, central, or multi-point weighted spatial discretization methods, particle tracking schemes, and hybrid Eulerian-Lagrangian techniques. Some methods are unconditionally stable while others may impose restrictions on temporal or spatial discretization. Some methods preserve sharp fronts but may not conserve mass. Others may be mass conservative but may introduce excessive numerical dispersion or unphysical oscillations. Yet other techniques may be extremely accurate for homogeneous systems and uniform flow conditions but may introduce oscillations and mass balance errors for heterogeneous systems.

The schemes used by the BCT package that are discussed in this document, are considered optimal for solution to transport in subsurface systems for a large number of situations. These analyses are often performed to delineate and address contamination within the subsurface and



evaluate how species mass evolves through space and time due to advective, dispersive, reactive, and equilibrium/non-equilibrium retardation processes under various hydrologic and imposed stress conditions. In this context, a scheme that conserves mass is important to the evaluation – an analysis would be severely hindered if contaminant mass were to horrendously appear or conveniently disappear simply due to the numerical solution scheme. Use of the original (divergence) form of the transport equation with a CVFD discretization by the BCT package provides mass conserved solutions even for heterogeneous, non-uniform flow-fields as would be present in the subsurface.

Unphysical oscillations or excessive numerical dispersion also compromise a transport solution. Use of the TVD scheme by the BCT package to discretize the advective term eliminates unphysical oscillations while at the same time minimizing numerical dispersion – note that numerical dispersion is not completely eliminated and appropriate discretization and grid refinement should be used in critical regions of interest, for advection dominated problems. No definite restrictions exist for general CVFD schemes on grid sizes and shapes in three dimensions, however, numerical dispersion errors are minimal when grid Peclet numbers are small. Users are therefore encouraged to estimate the grid Peclet number at significant locations and times during the simulation to estimate the significance of possible numerical dispersion in a subjective though quantitative manner.

A subsurface flow-field can be complex with contrasting velocities within the GWF domain itself, as well as between the GWF and CLN domains. Explicit schemes have time-step size restrictions for solution stability. Therefore, explicit schemes may require extremely small time-step size values with prohibitive associated simulation times to accommodate the largest velocities with a stable grid Courant number. Implicit schemes do not have this stability restriction and are therefore appropriate for subsurface transport evaluations that can span timescales from months to years, decades to centuries. For this reason, the implicit time-stepping solution scheme was selected for the BCT package. Implicit schemes may however incur numerical dispersion associated with time averaging over large time-step sizes and therefore time-step size selection is important for accuracy considerations. The implicitness factor can optionally be varied down to 0.5 for the Crank-Nicolson scheme that provides higher order temporal accuracy than the fully implicit formulation with a factor of 1. To evaluate the magnitude of the time-discretization error in a subjective though quantitative manner, users should evaluate the cell Courant numbers at critical times and in critical regions of the model domain to select appropriate maximum time-step sizes.



Accuracy of the transport simulation should be an important consideration in grid design. Often, vertical resolution is needed within an aquifer in the contaminated region to allow for vertical resolution of plume concentrations and migration within the aquifer. The sub-layering capability of *MODFLOW-USG* can be a useful feature in this regard; however, the flow-field also needs to be generated on this sub-refined grid as there is no capability of refining the grid for only the transport simulation. For grid refinement in the vertical or horizontal direction, the user should also be aware that Peclet number effects may be alleviated with smaller grids, however, that also increases the Courant number; possibly requiring smaller time-step sizes for the same degree of temporal accuracy.

Accurate transport simulations also require appropriate conceptualization of the flow and transport system. Often, the flow system is treated as steady-state or varying at a seasonal or larger timescale. However, field conditions may fluctuate at smaller time scales, with velocity variations that are not captured by the simulated flow-field, but which would still affect transport behavior. The dispersivity terms can help account for this condition as well. In addition, the effective porosity is difficult to quantify unless tracer tests have been performed at the site and use of other porosity definitions to approximate the effective porosity may not properly represent subsurface plume migration behavior. Evaluating the sub-grid scale heterogeneity with respect to fast versus slow flow paths and appropriate implementation of the dual domain concept or matrix diffusion processes, is also critical in obtaining accurate transport behavior for complex field situations.

Transport Solution Options

The BCT package input files provide several solution-option flags on the first line of data input. Some of these flags indicate physical processes that are activated for the simulation and help with associated memory allocation. Others are related to solution speed and accuracy. If dual domain transport is simulated, the DPT input file provides flags indicating the physical processes that are active in the immobile domain and help with associated memory allocation (and similarly for simulating matrix diffusion using the MDT process). Finally, if density driven flow is simulated, the DDF input file provides the necessary input.

The physical process flags indicate the number of transport component species (mobile and immobile); whether adsorption is simulated and whether it is linear or nonlinear; whether dispersion is simulated and if so, whether the homogeneous or the heterogeneous formulation is being used; if molecular diffusion is included; whether first order or zeroth order decay are



simulated and if so, whether that occurs in water, soil, or both; if chain decay is simulated; whether heat transport is simulated; and whether precipitation/dissolution are simulated. There is also a flag that direct whether transport is sequential to flow in the same simulation or whether transport simulation uses a previously determined flow field.

Flags associated with simulation speed and accuracy include options to set the active transport domain; optionally turn on cross-dispersion; a TVD flag and iteration counter; a concentration tolerance limit for the iterative solver; and a flag to trap impact of local flow balance errors. These items are discussed further below.

The grid for a transport simulation can be a subset of the flow simulation grid. This can significantly speed up the transport simulations if the flow domain is considerably larger than the region of interest for transport. A flag ICBNDFLG is available with the BCT, DPT and MDT packages to inactive unwanted portions of the respective transport domains. If this flag value is one, the active domain for transport is the same as for flow.

The cross-dispersion flag indicates whether cross-dispersion components are included in the simulation. On rectangular grids, the cross-dispersion term tends to smoothen the diamond-shaped patterns that may occur around extraction or injection wells if they are neglected. Experience has shown that its impact is not too significant when simulating transport in complex flow-fields. More experience is needed with this term for other cell shapes.

The TVD flag indicates whether the upstream-weighted scheme or the TVD formulation will be used and also indicates the number of iterations that will be taken for solution to each component, to resolve the nonlinearities arising out of the TVD scheme with right-hand-side updates of the TVD term. The upstream weighted scheme incurs more numerical dispersion than the TVD scheme; however, it does not require more than one iteration if other nonlinearities are absent and can therefore be quicker.

The Freundlich adsorption isotherm, Langmuir adsorption isotherm, the TVD scheme, and the cross-dispersion term all induce a nonlinearity to the solution of the transport equation for any given species. This nonlinearity is resolved iteratively till a convergence tolerance is achieved or up to the maximum number of iterations set by the ITVD flag. Therefore, the solution accuracy is improved with tighter tolerance and a larger number of iterations at the expense of computation speed. Experience suggests that a predictor-corrector approach with just two iterations is sufficient to significantly improve accuracy for nonlinearities associated with the TVD



expansion of the advective term. Nonlinearities associated with the Freundlich or Langmuir isotherm may require a greater number of iterations for sufficient accuracy. Use of smaller time-step sizes helps with achieving convergence as well as improving accuracy associated with time-step size averaging. The output listing file details the nonlinear iterations, the worst concentration error, and the associated cell location for each species in a similar manner to how convergence is reported for the nonlinear flow solution.

Local imbalances in the flow field can also cause errors in transport simulations. An optional flag IFMBC indicates if these flow field errors should be trapped and reflected in mass transport errors. If this flag is off (=0) then the errors are not trapped, and transport equation solution is done in the traditional manner. If the flag is on (=1), then additional input is provided to include Fortran Unit numbers for binary output of the flow residuals at each cell and the associated transport mass balance error for the GWF and CLN domains. These unit numbers and associated file names are also required in the NAME file to appropriately open these files. Output to these files is in a similar format to that of the flow simulation HDS and DDN files or the CON file of a transport simulation. The mass balance output table in the Output Listing (LST) file also summarizes the transport errors that occurs as a result of local imbalances in the flow solution.

The option "WRITE_GWMASS imasswr" indicates that the mass of each control volume cell is written to an output file in binary format (same format as that of the CONC file) when concentration output is written. The file on which this output is written is attached to Fortran unit number "imasswr". The mass output is the total mass of a species in a cell including in water phase, on soil solids (if adsorption is on) and on the air-water interface (if air-water interface is on).

Either the XMD or the PCGU matrix solver may be used to solve the implicit set of matrix equations. If asymmetric solution options were selected with these solvers for flow, the same solvers will be used for the transport simulations. If symmetric solution options were selected with these solvers for flow, the transport solution will automatically select the BCGS option of the PCGU solver or the ORTHOMIN option of the XMD solver for transport solutions. Linear solver options such as levels of fill, number of orthogonalizations, etc. are currently the same for both flow and transport solutions.



Additional Considerations for Transport of Multiple Components

Additional considerations have been included with this version of the code to include component species that are not transported (immobile component species that are still accounted for in a simulation but not transported, denoted as IMCOMP). This is in preparation for attaching more complex geochemical or reaction packages to the code. Thus, the total number of components NTCOMP is equal to the total number of mobile components plus temperature if heat is solved (MCOMPT). The variables are also ordered in the same manner – i.e., the mobile components are first, followed by the temperature, followed by the immobile components. Input concentrations are provided for these additional species which may then be changed by a reaction or geochemical package. Output of the reacted concentrations is then also provided by USG so that the reaction species are also tracked and reported by USG.

Also in preparation for several species that may be present in a geochemical simulation, the concentration output file has been further optionally split into multiple files. Specifically, the default approach was to write all species concentrations to a single .CON file. An option is now provided whereby each species is written to a separate .CON file. With this option, MULTIFILE, the output of each species goes to a separate file. The root name of the file is defined with the MULTIFILE option, and each species number (CONC01, CONC02, etc) gets concatenated to the root filename with the extension “.CON”. For temperature output, the MULTIFILE option appends the word “TMPR” to the root filename followed by the extension “.CON”.

OTHER ENHANCEMENTS IN USG-TRANSPORT

Several other enhancements have been made to the USG-Transport code to increase functionality and promote useability for model design, calibration, and application. These enhancements are discussed here to document these capabilities and demonstrate their usage. The formulation as well as their implementation are discussed here. Associated example problems are presented with the Example Problems section, and associated input instructions are supplied under the section on Description of Input Files.

Further customization of the flow routines accommodate additional capabilities and flexibility. A brief introduction to these customized solutions is presented here. The additional packages (or capability) that have been incorporated into the current version includes:

- The Transient IBOUND (TIB) Package discussed below.
- Recharge (RCH) Package modified to include non-zero inflow concentrations



- The Segmented Evapotranspiration (ETS) Package modified to include transport input if needed
- The Drain with Return Flow (DRT) Package
- The Sink with Return Flow (QRT) Package
- The Sparse Matrix Solver (SMS) Package with “SOLVEACTIVE” enhancement
- The River (RIV) Package with option to compute river-bed leakance independently from input leakance and groundwater cell leakance.
- Unsaturated zone flow option in BCF or LPF packages
- Dual Porosity Flow (DPF) Package
- Specified Gradient Boundary (SGB) Package
- CLN Package Modifications
- Solute Transport in LAK Package

Richards' Equation for Unsaturated Flow

Formulation

The general form of the CVFD balance equation for flow is expressed by equation (1) of the MODFLOW-USG document (Panday et al, 2013) as:

$$\sum_{m \in \eta_n} C_{nm}(h_m - h_n) + Storage_n(h_n) = W_n, \quad (U1)$$

where C_{nm} is the inter-cell conductance between cells n and m , h_n and h_m are the hydraulic head at cells n and m , $Storage_n$ is the storage term for cell n , which is a function of the head h_n , and W_n is the source or sink on cell n .

This equation was applied to solution of confined and unconfined flow in MODFLOW-USG by using the upstream weighted formulation of MODFLOW-NWT (Niswonger, et al, 2011). The same equation may be applied to solve for unsaturated zone flow, by considering storage to consist of saturation of water in pore storage, and the inter-cell conductance to be controlled by a saturated hydraulic conductivity term times a relative permeability term. The relative permeability may be expressed by the Brooks Corey function [Brooks, R.H. and A.T. Corey, 1966. Properties of porous media affecting fluid flow. *ASCE J. Irrig. Drain. Div.*, 92 (IR2): 61-88.], written as:

$$k_r = S_e^n \quad (U2)$$



Where k_r the relative permeability of the soil, n is the Brooks-Corey exponent, and S_e is the effective soil saturation expressed as:

$$S_e = \frac{S_w - S_r}{1 - S_r} \quad (\text{U3})$$

Where S_w is the soil water saturation, and S_r is the residual water saturation. Note that the relative permeability is zero if the water saturation falls below the residual water saturation, and flow of water from one cell to the next becomes zero as a result. The relative permeability curve can alternately be represented by the van Genuchten function as:

$$k_r = S_e^{1/2} \left[1 - (1 - S_e^{1/\gamma})^\gamma \right]^2 \quad (\text{U4})$$

For unsaturated zone flow, the water saturation is a function of the capillary head, which in turn, is a function of the hydraulic head h . Assuming that air is at atmospheric pressure, the capillary head is expressed as:

$$h_c = z - h \quad (\text{U5})$$

Where z is the elevation of the cell. The water saturation may be expressed as a function of the capillary head via the van Genuchten equation [van Genuchten (1980). A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, *Soil Science of American Journal* 44, 892-898.]. This equation is modified in the current formulation to extend its capability, and is expressed as:

$$S_e = \begin{cases} [1 + \{\alpha(h_c - h_b)\}^\beta]^{-\gamma} & \text{for } h_c < 0 \\ 1 & \text{for } h_c > 0 \end{cases} \quad (\text{U6})$$

Where α , β and $\gamma (= 1 - 1/\beta)$ are the van Genuchten parameters, and h_b is the bubble point or air entry head. The standard form of the van Genuchten parameter has a zero air entry head. The bubbling point pressure also allows the retention function to represent gas saturations that would be generated below the bubbling point pressure in shale gas formations. This extension is invoked if the **“BUBBLEPT” option** is used.

A further optional extension of the van Genuchten equation is provided here, whereby equation (U6) expresses the actual water saturation S_w , instead of the effective water saturation S_e . Thus, the water saturation in a medium can range from completely dry to fully saturated, though flow



of water from one cell to the next only occurs above residual water saturation as per equation (4). This is a useful extension to accommodate fully dry conditions which may occur due to evaporation. This capability is invoked with the **“FULLYDRY” option**. Note that the unsaturated zone flow equation is also known as the Richards equation.

Additional options for retention curves are provided by a **tabular input capability**. Thus, any functional form of the retention curve and relative permeability curve may be provided as input. Furthermore, an option is provided along with tabular input, for integrating the retention curves over the grid-block height to provide more accurate functions for thick grid-blocks than the point value at the cell center. An Excel spreadsheet is available to assist users with converting the retention curves to integrated tabular curves for input to MODFLOW-USG. Tabular input is set using groups of soil types that are then scattered around the model cells using a zonal array.

Implementation

The unsaturated zone flow equation solution is invoked in USG-Transport by using the option “RICHARDS” on the first line in the BAS datafile. This additional option determines that the Richards equation will be solved instead of the unconfined flow solution, and additional input will be requested as needed, in the BCF or LPF packages, for the unsaturated zone parameters, α , β , S_r , and n , which follows the other material property input for every layer, if the unsaturated zone equation is solved. Furthermore, if the option BUBBLEPT is invoked (in BCF/LPF file) in using Richards Equation, the bubble point head is also required as input. Finally, if the option FULLYDRY is invoked (in BCF/LPF file), then the moisture retention curve expresses the total water saturation and not the effective water saturation.

The unsaturated zone flow equation can also be invoked for any one or more layers in an unconfined flow simulation. **For this case, the option “RICHARDS” should not be invoked.** Instead, the LAYCON value for that layer in the BCF input (or the LAYTYP value for that layer in the LPF input) should be set to 5 to indicate that the layer is treated via the unsaturated flow equation formulation, with additional associated input of the retention function parameters. For the unsaturated flow equation formulation with tabular input, the LAYCON value for that layer in the BCF input (or the LAYTYP value for that layer in the LPF input) should be set to 6. Finally, it should be noted that if Richards equation is used in the BCF or LPF packages, then if dual domain flow is invoked using the DDF package, the same form of the equation will be used in the dual domain flow package as well.



Evapotranspiration Formulation for Solving Richards' Equation

With assistance from Martin Vonk^{1,2}

1. Department of Water Management, Faculty of Civil Engineering and Geosciences, Delft University of Technology. Delft, South Holland, The Netherlands.

2. Artesia B.V., Schoonhoven, South Holland, The Netherlands

Evapotranspiration from the unsaturated zone can be simulated when solving Richards' Equation using the EVT or ETS Package. The associated inputs are translated / equivalenced as discussed below.

Evapotranspiration in the unsaturated zone depends on the water content and its relation can be approximated in different ways. The most common is to assume that the evapotranspiration rate for a crop is maximum (CROPET) at and above the limiting point (θ_{lp}). The actual evapotranspiration rate drops to zero as water content reduces to the wilting point water content (θ_{wp}). The limiting point is in between the wilting point and the field capacity (θ_{fc}). The relation between the ET rate and the water content is depicted in Figure RE1 below.

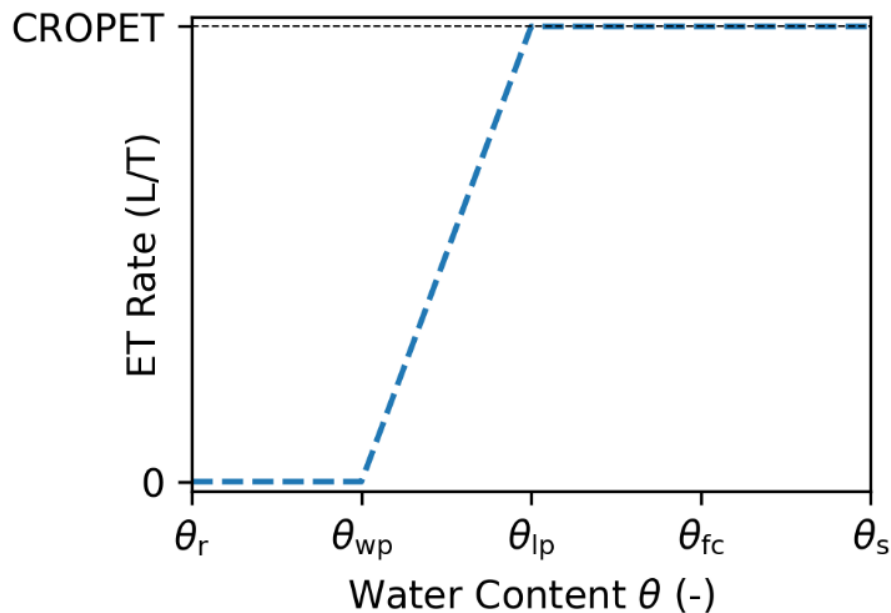


Figure RE1 – Actual Evapotranspiration as a Function of Water Content

Figure RE2 below shows a soil water retention curve using the van Genuchten equation. The figure depicts the water content on the x-axis and the pressure head (h_p) on the y-axis (note that the pressure head in the unsaturated zone is the negative of the capillary head).

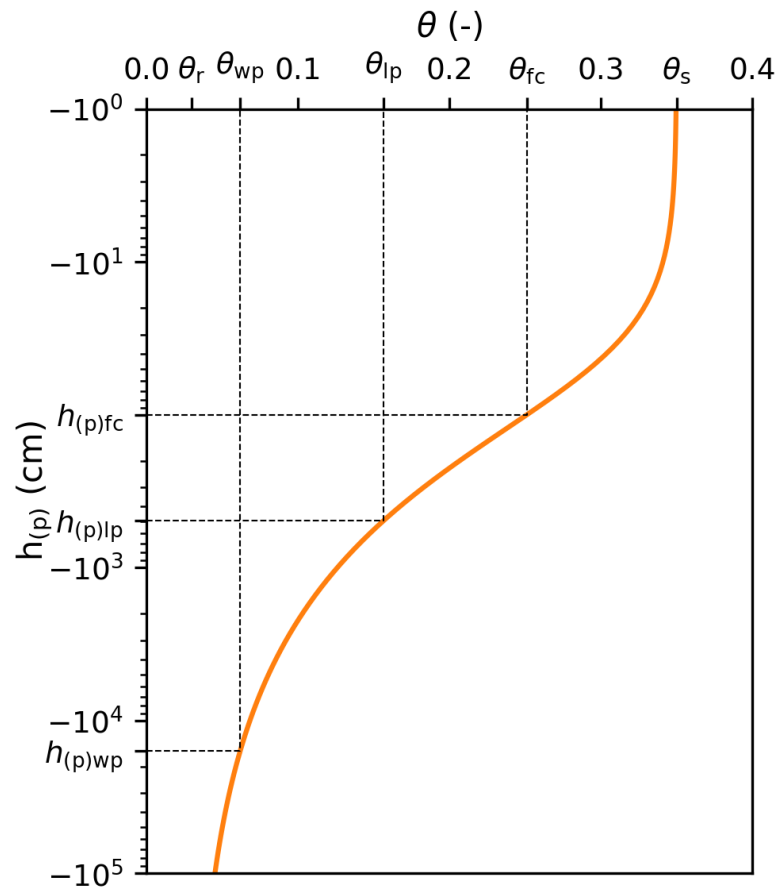


Figure RE2 – van Genuchten Soil Water Retention Curve highlighting the Relationship between the Various Moisture Contents and Pressure Heads

Thus, from knowing the capillary curve of the soil and the saturated water content (θ) which is also the soil porosity, the wilting point and field capacity pressure heads can be found. The wilting point and field capacity hydraulic heads can then be obtained by adding the elevation to the respective pressure heads since hydraulic head ($h(h)$) equals pressure head ($h(p)$) plus elevation (z). The wilting point and field capacity are soil dependent, but common definitions are at $h(p) = -10^{4.2}$ and -10^2 cm respectively. The limiting point water content is dependent on the soil type, plant type and root depth, but commonly chosen halfway in between (depletion factor=0.5, FAO Irrigation and Drainage Paper. Vol. 56, Ch. 8, 1998) the field capacity and the wilting point water contents.



MODFLOW EVT implementation

MODFLOW EVT input is for a potential evapotranspiration rate (PET), an ET surface (SURF) above which the PET is completely satisfied, and an extinction depth (EXTDP) at which the ET rate drops to zero in a linear fashion from the maximum ET rate.

The maximum evapotranspiration rate for a crop (CROPET) in a Richards' equation simulation translates to the potential evapotranspiration rate (PET) in MODFLOW. Thus, **PET = CROPET**.

The field capacity water content (θ_{fc}) in a Richards' equation simulation can be converted to the field capacity pressure head ($h_{(p)fc}$) through the van Genuchten equation, and further to the field capacity hydraulic head ($h_{(h)fc}$) by adding the elevation to the pressure head term. The field capacity hydraulic head ($h_{(h)fc}$) thus calculated translates to the ET surface (ETSURF) in MODFLOW. Thus, **ETSURF = $h_{(h)fc}$**

Similarly, the wilting point water content (θ_{wp}) in a Richards' equation simulation can be converted to the wilting point pressure head ($h_{(p)wp}$) through the van Genuchten equation, and further to the wilting point hydraulic head ($h_{(h)wp}$) by adding the elevation to the pressure head term. The wilting point hydraulic head ($h_{(h)wp}$) thus calculated is the elevation at which the ET rate drops to zero, which is equal to the ET surface, minus the extinction depth in MODFLOW. Thus, $ETSURF - EXTDP = h_{(h)wp}$ which can be rewritten as $EXTDP = ETSURF - h_{(h)wp}$. As above, $ETSURF = h_{(h)fc}$, therefore, **$EXTDP = h_{(h)fc} - h_{(h)wp}$** . Table RE1 below summarizes the inputs.

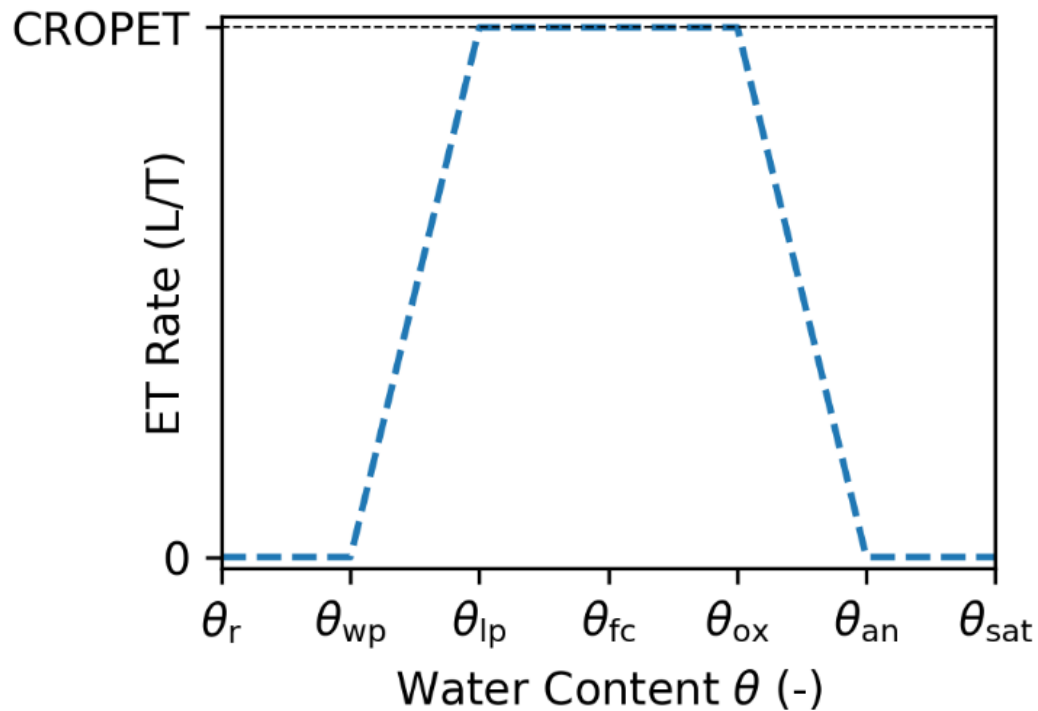
Table RE1 – Suggestion for Input for EVT Package

MODFLOW EVT	Unsaturated Zone Parameters with θ_{lp}	Unsaturated Zone Parameters assuming $\theta_{lp} = \theta_{fc}$
PET	CROPET	CROPET
ETSURF	$h_{(h)lp}$	$h_{(h)fc}$
EXTDP	$h_{(h)lp} - h_{(h)wp}$	$h_{(h)fc} - h_{(h)wp}$



MODFLOW ETS implementation

A more complicated evapotranspiration function for the unsaturated zone considers further, an oxic water content (θ_{ox}), above which point the evapotranspiration rate starts again to decrease



down to zero at the anoxic water content (θ_{fc}), when the roots are waterlogged and do not transpire further. This is depicted in Figure RE3.

Figure RE3 – Alternate Relationship of Evapotranspiration as a Function of Water Content

This function can be simulated using the ETS Package of MODFLOW-USG. The parameters ETSURF and EXTDP are defined in the same manner as for the EVT package above. However, the ETS package should include two segments to determine PET. The first segment that defines PET will be the CROPET from zero to the oxic water content fraction (θ_{ox}/θ). The second segment for PET will be zero from the oxic water content fraction up to the anoxic moisture content fraction (θ_{an}/θ). Example of the segment table is below.



Table RE2 – Input for ETS Package Segments

Fraction of EXTDP	PET value at that fraction
0	CROPET
θ_{ox}/θ	CROPET
θ_{an}/θ	0

Therefore, the function works similarly to the EVT package, only, the segments are required to then reduce the ET amounts beyond the oxic water content to zero at the anoxic water content. Without segmentation at the oxic water fraction, the ET would continue at CROPET values.

Note that the wilting point water content is larger than the residual water content as it sits on the retention curve. The van Genuchten retention curve flattens out at residual saturation and saturation cannot below that amount. USG provides an option to have the residual water content only apply to the relative permeability, with a zero residual saturation value in the soil water retention curve. This option allows water saturation to go to zero as a result of evapotranspiration or other such processes, while flow ceases at residual saturation as it is applied to the relative permeability function.

CLN Package Modifications – Turbulent Flow; Well Efficiency

The CLN Process is documented in the MODFLOW-USG report (Panday et al. 2013). The capability of flow within the CLN domain has been extended here to include a variety of turbulent flow formulations. Also, the CLN-GW connection can be governed by a “well efficiency” instead of a skin resistance factor. These capabilities are documented here. For completeness, the entire CLN document is included here including the laminar flow formulation and the formulations for CLN-GW connection.

CLN Flow with Turbulent Flow Option

The CLN Process was developed for MODFLOW-USG to provide the framework for incorporating one-dimensional connected features into a structured or unstructured three-dimensional GWF Process grid. A one-dimensional CLN feature is any hydrogeologic or hydrologic water conveyance feature that has a cross-sectional dimension which is much smaller than the longitudinal flow dimension and the size of the encompassing GWF cell. Flow is computed in the longitudinal direction of the network of connected one-dimensional features



using specified cross-sectional properties; flow between CLN cells and GWF cells is computed across the upstream wetted perimeter of the one-dimensional CLN feature. The CLN Process thus provides a mechanism for including features with small cross-sectional areas, relative to GWF cell sizes, without having to build this level of detail into the grid used for the GWF domain. An example problem is included in this report to demonstrate use of the CLN Process. The MODFLOW-USG document (Panday et al, 2013) provides details on the CLN flow process for laminar flow. This process has been further extended here to include a variety of turbulent flow formulations. The entire formulation including the laminar flow implementation of MODFLOW-USG is repeated here for completeness.

The CLN Flow Process is solved simultaneously with the GWF Process. This means that the total number of cells in a MODFLOW-USG simulation is equal to the combined total of GWF cells and CLN cells. The CLN flow process solves for flow of water within a network of linear features as well as for the interaction of the features with the porous medium. There are then, two types of flow calculations that occur with the addition of the CLN domain—flow within the CLN domain, and flow between CLN and GWF cells. The CLN Process does not inactivate dry cells as is done in MODFLOW-2005 for GWF cells. Instead, a head value is always calculated for active CLN cells as in the UPW formulation. As the saturated thickness approaches zero, the conductance values with connected cells also approach zero and the cell responds as if it were dry.

For flow within the CLN domain, the CLN Process implements solution of one nonlinear equation per CLN cell. The CLN cells may represent wells, pipes, fractures, canals, rivers, streams or other linear 1-D features within a simulation domain that need to be represented by flow connections that are separate from those of the aquifer. The formulation is general enough whereby different types of features can flow into each other. For instance, pipes can flow from or to open-channel features. Extension of the code to include other features or geometries is a straightforward exercise in implementing functional forms for cross-sectional areas, and volumes as functions of flow depths at the CLN cells within the appropriate subroutines.

The current version of MODFLOW-USG only incorporates cylindrical and rectangular conduit geometry types. Other geometry types and flow formulations can be added in the future, as needed. The current implementation could also be extended to support other pipe or section geometries, tabular input of volume and cross-sectional flow area as a function of head, and turbulent flow conditions. These geometry types and flow conditions can be included by



providing functional forms of the longitudinal flow conductance terms within the appropriate subroutines.

Flow within each CLN segment can be laminar or turbulent since the formulations are derived in a general manner. Laminar flow is computed using the Hagen- Poiseuille equation. Three optional formulations are provided to simulate turbulent flow – the Darcy-Weisbach equation, the Hazen-Williams equation, and the Manning's equation. The turbulent flow formulations neglect the inertial term in the St Venant equations and therefore solve the diffusion wave equation. The dynamic wave equation for flow in open channels requires solution to two equations per channel segment (one for continuity and a separate one for momentum) and therefore is not readily accommodated by the present form of the CLN Process.

For flow between CLN cells and connected GWF cells, the CLN Process implements solution of one nonlinear flow equation per CLN-to-GWF connection. Various options are provided to compute the effective leakance between them, including input of the effective leakance value, computation from skin conductance and thickness, or use of a Thiem solution to provide the radial influences and efficiency considerations for flow between CLN and GWF cells. The formulation is general enough whereby extension of the code to include other connection geometries or special considerations is straightforward as long as functional forms for the respective wetted surface areas with flow depths for the CLN-to-GWF connection can be provided. Furthermore, the CLN-to-GWF transfer equations are generalized to accommodate unstructured grid cells.

The CLN Process uses the concept of a CLN segment, which is one or more CLN cells connected end to end. A CLN segment may be used to represent a well, for example. CLN segments can also be connected to one another to form a network, as shown in **Figure 14**. A CLN network may be useful for representing tile drains, radial collector wells, or other connected linear features. A CLN cell can be vertical, horizontal, or tilted. Cell lengths may be dimensioned such that several CLN cells (from the same or different CLN segments) may be connected to one GWF cell, or one CLN cell may be connected to several GWF cells depending on the scale of conceptualization of flow within the CLN and GWF domains. In addition, there is no need for specific upstream to downstream sequential ordering of cells as may be necessary for other packages or codes.

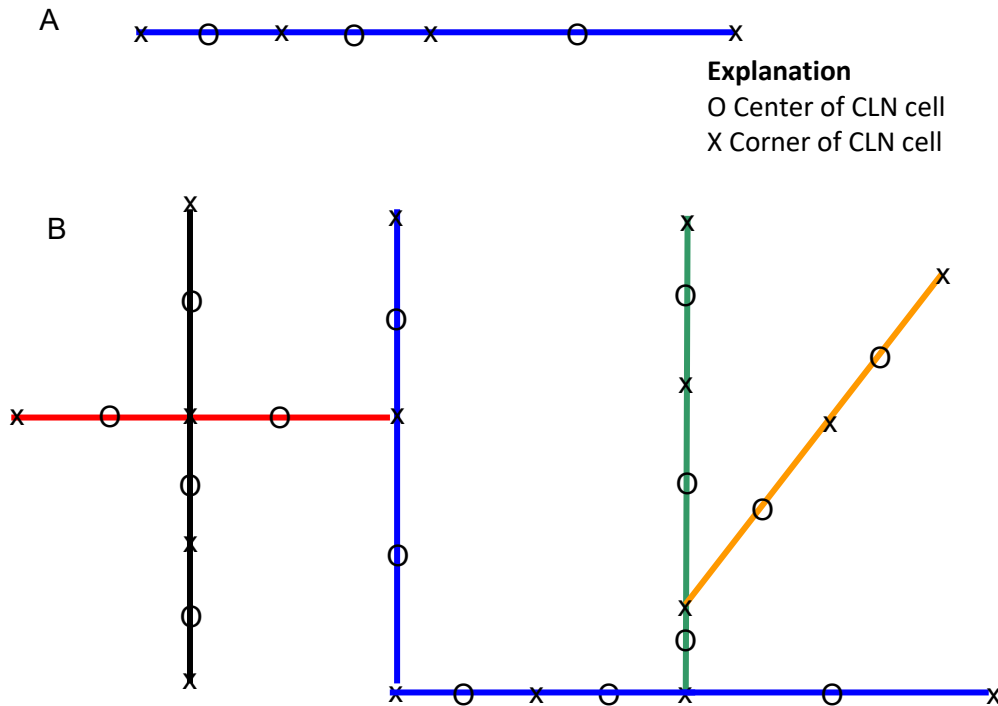


Figure 14. Different connected linear network geometries: A, a single linear feature discretized with three Connected Linear Network (CLN) cells; B, a multi-dimensional network of CLN segments.

Process with the cylindrical conduit geometry type can be used to simulate flow through a linear network of karst conduits similar to what can be done with the Conduit Flow Process (CFP) (Shoemaker and others, 2007) of MODFLOW-2005. Areal flow functionality of CFP (CFP modes 2 and 3) is presently not available in MODFLOW-USG. An advantage of the MODFLOW-USG implementation is that the conduit network is solved simultaneously with the groundwater flow equation, with a Newton-Raphson linearization.

The CLN Process, when applied with cylindrical conduits and the Thiem solution for CLN-to-GWF connection, provides some of the functionalities of the Multi-Node Well (MNW and MNW2) (Halford and Hanson, 2002; Konikow and others, 2009) Packages of MODFLOW-2005. A single cylindrical CLN cell connected to multiple GWF cells may be pumped to simulate multi-node well conditions, in which the CLN cell extracts water from the GWF cells connected to it as part of the solution to the coupled CLN and GWF flow equations. A CLN cell can be pumped by use of the Well (WEL) Package by assigning a source or sink to that CLN cell. If flow through a narrow conduit is important to the production of a multi-layer well (for instance, the well-bore resistance governs how much water is produced from each of multiple aquifers) the well may also be



represented by a segment consisting of multiple CLN cells. In this case, a CLN cell can be added to each GWF cell or layer, and flow would be simulated within the conduit. This may be especially important when the conduit radius is small and resistance in the well bore affects flow through the conduit. This approach would also be useful for simulating borehole flow within wells that are screened to multiple aquifers.

A CLN network may be useful for representing tile drains, radial collector wells, or other connected linear features. A CLN cell can be vertical, horizontal, or tilted. Cell lengths may be dimensioned such that several CLN cells (from the same or different CLN segments) may be connected to one GWF cell, or one CLN cell may be connected to several GWF cells depending on the scale of conceptualization of flow within the CLN and GWF domains. In addition, there is no need for specific upstream to downstream sequential ordering of cells as may be necessary for other packages or codes.

The CLN Process with the cylindrical conduit geometry type can be used to simulate flow through a linear network of karst conduits similar to what can be done with the Conduit Flow Process (CFP) (Shoemaker and others, 2007) of MODFLOW-2005. Areal flow functionality of CFP (CFP modes 2 and 3) is presently not available in MODFLOW-USG. An advantage of the MODFLOW-USG implementation is that the conduit network is solved simultaneously with the groundwater flow equation, with a Newton-Raphson linearization.

The CLN Process, when applied with cylindrical conduits and the Thiem solution for CLN-to-GWF connection, provides some of the functionalities of the Multi-Node Well (MNW and MNW2) (Halford and Hanson, 2002; Konikow and others, 2009) Packages of MODFLOW-2005. A single cylindrical CLN cell connected to multiple GWF cells may be pumped to simulate multi-node well conditions, in which the CLN cell extracts water from the GWF cells connected to it as part of the solution to the coupled CLN and GWF flow equations. A CLN cell can be pumped by use of the Well (WEL) Package by assigning a source or sink to that CLN cell. If flow through a narrow conduit is important to the production of a multi-layer well (for instance, the well-bore resistance governs how much water is produced from each of multiple aquifers) the well may also be represented by a segment consisting of multiple CLN cells. In this case, a CLN cell can be added to each GWF cell or layer, and flow would be simulated within the conduit. This may be especially important when the conduit radius is small and resistance in the well bore affects flow through the conduit. This approach would also be useful for simulating borehole flow within wells that are screened to multiple aquifers.



Formulation of Flow within the CLN Domain

The continuity equation for flow through a CLN cell is a function of flows from connected CLN cells and flows from connected GWF cells and may be written in difference form as

$$\frac{V_n f_{vn}}{\Delta t} = \sum_{m \in \eta_n} Q_{nm} + \sum_{p \in \eta_n} \Gamma_{cpn} \quad (1)$$

Where

V_n is the volume of CLN cell n ,

f_{vn} is the fraction of the volume of CLN cell n that is saturated (or the variable part of the volume term which is a function of the head of cell n),

Q_{nm} is the volumetric flow between connected CLN cells n and m , and

Γ_{cpn} is the volumetric flow from a connected GWF cells p to CLN cell n .

The left-hand side of equation (1) is the storage term of the CLN cell, the first term on the right-hand side is the flow between cell n and connected CLN cells, m , and the second term on the right-hand side is the interaction flow between cell n and all connected GWF cells, p . The summation in the first flow term on the right-hand side is over all m CLN cells connected to CLN cell n , whereas the summation in the second term on the right-hand side (the interaction term), is over all p GWF cells connected to CLN cell n .

The fraction of the total volume of a CLN cell that is saturated, f_v , can be computed depending on whether the 1-dimensional cell is vertical or horizontal. For a vertical CLN cell, the saturated fraction, f_v , is obtained as

$$\begin{aligned} f_v &= 0 & \text{for } (h - BOT) < 0 \\ f_v &= \frac{(h - BOT)}{l} & \text{for } 0 < (h - BOT) < l \\ f_v &= 1 & \text{for } (h - BOT) > l \end{aligned} \quad (2)$$

where

BOT is the bottom elevation of the CLN cell, and



l is the length of the CLN cell in its direction of flow.

If the cell is angled, the assumption is made that $l \gg r$ where r is the hydraulic radius of the CLN cell (that is, the length of the CLN segment is much larger than its cross-sectional dimension), and therefore, the term l in equation (2) is replaced by $l \sin(\theta)$, where θ is the angle that the conduit makes with the horizontal. The second part of equation (2) contains a singularity when the angle θ is zero; therefore, if $l \sin(\theta)$ is less than twice the hydraulic radius of the CLN cell ($2r$), a horizontal conduit is assumed, governed by a different set of equations as shown next. In practice, equation (2) is smoothened to determine f_{vn} in order to remove slope discontinuities as required by the Newton-Raphson method.

For a horizontal CLN segment, the saturated fraction, f_v , is obtained for any cell n as

$$f_v = \frac{A_w}{A}, \quad (3)$$

where A_w is the wetted cross-sectional area of CLN cell n , and A is the saturated cross-sectional area of CLN cell n . Note that subscript " n " is omitted for convenience. The wetted cross-sectional area can be obtained from geometric considerations of the cross-section as a function of the flow depth.

Laminar Flow within the CLN domain

Flow within the CLN domain may be computed by various alternative equations. For laminar flow between any two CLN nodes n and m , the volumetric flux may be expressed by the Hagen-Poiseuille equation as:

$$Q_{nm} = \frac{AgD_h^2}{32\nu L} \cdot f_{unm} \cdot (h_n - h_m) \quad (4)$$

Where

g is the gravitational acceleration constant [L/T^2],

D_h is the hydraulic diameter ($= 4A/P$ where P is the perimeter), [L]. Note that the hydraulic diameter required here is for a fully flowing CLN cell

ν is the kinematic viscosity of water [L^2/T],



L is the separation distance between nodes n and m [L], and

f_{unm} is the fraction of the upstream CLN cell volume that is saturated ($= f_{vm}$, if n is upstream and $= f_{vm}$ if m is upstream) [-]. This term extends the Hagen-Poiseuille equation to partially wetted conditions and reduces the flow from a dry node to zero in a physical manner, with the upstream formulation.

Note that equation (4) contains a constant term (the property-geometry factor), a term that is dependent on the upstream head value, and a linear driving force term expressed by the head difference between the CLN cells.

Turbulent Flow with the Darcy-Weisbach Equation

For turbulent flow through the CLN domain, the volumetric flux may be expressed by the Darcy-Weisbach equation as:

$$Q_{nm} = A_w \sqrt{\frac{2gD_h^2(h_n - h_m)}{fL}} \quad (5)$$

Where

f is the Darcy-Weisbach friction factor [-].

The Colebrook and White formula provides an equation to compute the friction factor as:

$$\frac{1}{\sqrt{f}} = -2 \log \left[\frac{k_c}{3.71D_h} + \frac{2.51}{R_e \sqrt{f}} \right] \quad (6)$$

Where

k_c is the mean roughness height of the CLN domain walls [L], and

R_e is the Reynolds number of flow [-].

The Reynolds number is defined as:

$$R_e = \frac{QD_h}{Av} \quad (7)$$



Manipulation of equations (3), (5), (6) and (7), gives a closed form formula for the volumetric flux using Darcy-Weisbach equation as:

$$Q_{nm} = A \sqrt{\frac{32g}{L}} \cdot f_v \left[\frac{A_w}{P_w} \right]^{0.5} \cdot (|h_m - h_n|)^{-0.5} \cdot \log \left[\frac{k_c}{3.71D_h} + \frac{2.51\nu}{\sqrt{\frac{2gD_h^3 |h_m - h_n|}{L}}} \right] \cdot (h_m - h_n) \quad (8)$$

Where,

P_w is the wetted perimeter [L].

Note that the first term of equation (8) is a constant (the property-geometry factor). The second term is dependent on the upstream head value. This term physically reduces the flow to zero when the upstream cell becomes dry. The third term is the nonlinear turbulent flow coefficient which is a function of the head difference between the connected nodes, and the last term is the linearized portion of the nonlinear driving force for turbulent flow.

Turbulent Flow with the Hazen-Williams Equation

For turbulent flow through the CLN domain, the volumetric flux may be expressed by the Hazen-Williams equation as:

$$Q_{nm} = \frac{A(0.849C_{HW})}{L^{0.54}} \cdot f_v \left[\frac{A_w}{P_w} \right]^{0.63} \cdot (|h_m - h_n|)^{-0.46} \cdot (h_m - h_n) \quad (9)$$

Where

C_{HW} is the Hazen-Williams coefficient [$L^{0.37}/T$], typically provided for various material types in SI units.

Note that the first term of equation (9) is a constant (the property-geometry factor). The second term is dependent on the upstream head value. This term physically reduces the flow to zero when the upstream cell becomes dry. The third term is the nonlinear turbulent flow coefficient which is a function of the head difference between the connected nodes, and the last term is the linearized portion of the nonlinear driving force for turbulent flow.

**Turbulent Flow within the CLN Domain using the Manning's Equation**

For turbulent flow through the CLN domain, the volumetric flux may be expressed by the Manning's equation as:

$$Q_{nm} = \frac{A}{nL^{0.5}} \cdot f_v \left[\frac{A_w}{P_w} \right]^{(2/3)} \cdot (|h_m - h_n|)^{-0.5} \cdot (h_m - h_n) \quad (10)$$

Where

N is the Manning's coefficient [$T/L^{(1/3)}$], typically provided for various materials in SI units.

Note that the first term of equation (10) is a constant (the property-geometry factor). The second term is dependent on the upstream head value. This term physically reduces the flow to zero when the upstream cell becomes dry. The third term is the nonlinear turbulent flow coefficient which is a function of the head difference between the connected nodes, and the last term is the linearized portion of the nonlinear driving force for turbulent flow.

Numerical formulation of the laminar and turbulent CLN-CLN flux

As noted from the above equations, the flux along the CLN domain is expressed in a general form as:

$$Q_{nm} = A_{cnm} C_{onm} f_{lunm} f_{gnm} (h_n - h_m) \quad (11)$$

Where

$A_{cnm} C_{onm}$ is the saturated linear conductance of connected CLN cells n and m ,

A_{cnm} is the cross-sectional area of the connection between the CLN cells $= \min(A_{cn}, A_{cm})$,

A_{cn} is the cross-sectional area of CLN cell n ,

A_{cm} is the cross-sectional area of CLN cell m , and

f_{lunm} is the upstream CLN cell's nonlinear conductance term ($= f_{1n}$, if n is upstream and $= f_{1m}$ if m is upstream),

f_{1n} is the nonlinear conductance term of cell n



f_{lm} is the nonlinear conductance term of cell m , and

f_{gnm} is the turbulent flow coefficient (the “gradient term”) which is a function of the head difference between connected CLN cells n and m .

Treatment of Constant Term

The constant term of the equation is the saturated linear conductance between two CLN cells, $A_{cnm} C_{onm}$. The term C_{onm} , is expressed differently depending on which flow equation is used. For the laminar flow equation, C_{onm} is obtained from equation (4) and the definition for hydraulic diameter as:

$$C_{onm} = \frac{g}{2\nu(0.5[l_n + l_m])} \left[\frac{A}{P} \right]^2 \quad (13)$$

Where

l_n and l_m are the full length of CLN cells n and m . Note that the connection distance between two cells is half of the sum of the lengths of the respective cells.

For turbulent flow, C_{onm} can be expressed using the Darcy-Weisbach equation (8) as:

$$C_{onm} = \sqrt{\frac{32g}{0.5[l_n + l_m]}} \quad (14)$$

Using the Hazen-Williams equation (9) as:

$$C_{onm} = \frac{0.849 C_{HWnm}}{(0.5[l_n + l_m])^{0.54}} \quad (15)$$

And using the Manning’s equation (10) as:

$$C_{onm} = \frac{1}{(0.5[l_n + l_m])^{0.5} n_{nm}} \quad (16)$$

The Hazen Williams coefficient in equation (15) for flow between segments n and m , C_{HWnm} , and the Manning’s coefficient in equation (16) for flow between segments n and m , n_{nm} , are computed as a harmonic mean of the respective coefficients for each of the cells.



Treatment of Upstream Term

The upstream term, f_{lunn} , has different values depending on which flow equation is used. For laminar confined flow, this term is unity. For laminar unconfined flow, this term decreases with saturated volume fraction of the upstream cell such that it goes to zero when the upstream saturation is zero thus preventing any flow from a dry cell. Thus, for laminar flow, we have: saturated fraction, f_{vn} , is obtained as

$$f_{lunn} = f_v \quad (17)$$

Where the term on the right-hand side is the upstream of cells n and m ($= f_{vn}$, if n is upstream; $= f_{vm}$ if m is upstream).

For turbulent flow, f_{lunn} for any cell n can be expressed using the Darcy-Weisbach equation (8) as:

$$f_{lunn} = f_v \left[\frac{A_w}{P_w} \right]^{0.5} \quad (18)$$

Where the term on the right-hand side is the upstream of cells n and m .

For turbulent flow, f_{lunn} for any cell n can be expressed using the Hazen-Williams equation (9) as:

$$f_{lunn} = f_v \left[\frac{A_w}{P_w} \right]^{0.63} \quad (19)$$

Where the term on the right-hand side is the upstream of cells n and m .

For turbulent flow, f_{lunn} for any cell n can be expressed using the Manning's equation (10) as:

$$f_{lunn} = f_v \left[\frac{A_w}{P_w} \right]^{(2/3)} \quad (20)$$

Where the term on the right-hand side is the upstream of cells n and m .

For the turbulent flow equations, the slope of the upstream term approaches infinity as the function value approaches zero (which is when the depth of flow approaches zero). To avoid



problems with the Newton-Raphson slope term for this condition, and maintain continuity of the function and its slope, the upstream function, f_{unm} , is modified by a patch function that is applied when the depth of flow is between 0 and ε , (ε = a small number = 10^{-2}), given as:

$$f_{unm} = \begin{cases} f_{patch} f_{unm} & \left[\frac{A}{P} \right] < \varepsilon \\ f_{unm} & \left[\frac{A}{P} \right] \geq \varepsilon \end{cases} \quad (21)$$

Where

$$f_{patch} = -\frac{2}{\varepsilon^3} \left[\frac{A}{P} \right]^3 + \frac{3}{\varepsilon^2} \left[\frac{A}{P} \right]^2 \quad (22)$$

Which has the property that $f_{patch} = 0$ at $[A/P] = 0$; $df_{patch} = 0$ at $[A/P] = 0$; $f_{patch} = 1$ at $[A/P] = \varepsilon$ and $df_{patch} = 0$ at $[A/P] = \varepsilon$. Therefore, the corrected function has the properties that $f_{unm} = 0$ at $[A/P] = 0$; $df_{unm} = 0$ at $[A/P] = 0$; $f_{unm} = f_{unm}^*$ at $[A/P] = \varepsilon$ and $df_{unm} = df_{unm}^*$ at $[A/P] = \varepsilon$, where f_{unm}^* is the value of the original function. **Figure 15** shows the corrected and uncorrected functions as a function of $[A/P]$.

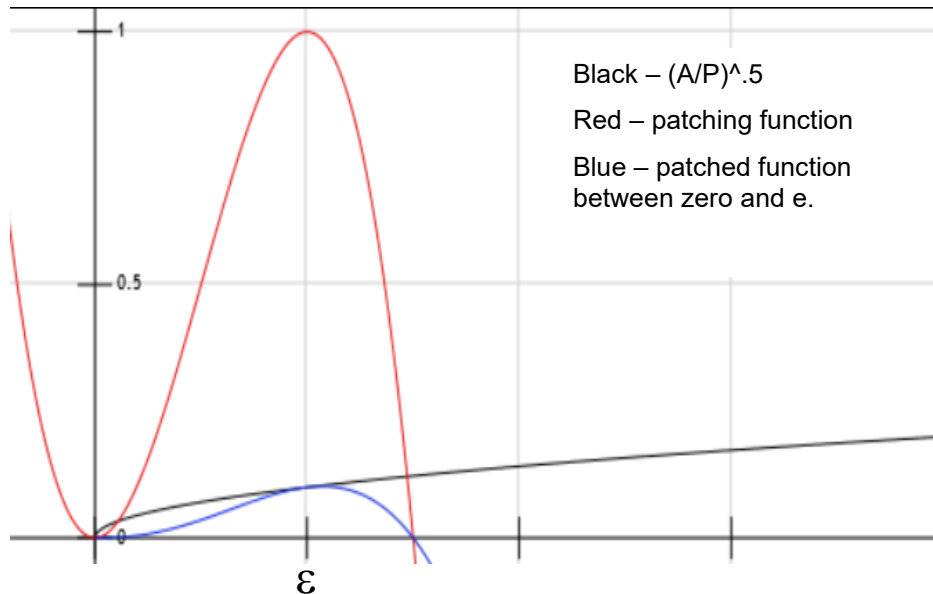


Figure 15. Corrected and Uncorrected Function (f_1) for Darcy-Weisbach Equation with $e = 0.01$. Note that the Manning's Equation and Hazen-Williams Equation have a different exponent but similar form.



Treatment of the Turbulent Flow Coefficient

The turbulent flow coefficient, f_{gnm} , has different values depending on which flow equation is used. For laminar flow, this term is unity.

The turbulent flow coefficient, f_{gnm} for any connection nm can be expressed using the Darcy-Weisbach equation (8) as:

$$f_{gnm} = (|h_m - h_n|)^{-0.5} \log \left[\frac{k_c}{3.71D_h} + \frac{2.51\nu}{\sqrt{\frac{2gD_h^3 |h_m - h_n|}{L}}} \right] \quad (23)$$

Substituting for D_h and rearranging gives:

$$f_{gnm} = (|h_m - h_n|)^{-0.5} \log \left[\frac{k_c P}{14.84A} + \frac{0.222\nu}{\sqrt{\frac{g |h_m - h_n|}{L} \left(\frac{A}{P}\right)^3}} \right] \quad (24)$$

Which can be written in a more simplified form as:

$$f_{gnm} = h_d^{-0.5} \log \left[\alpha + \frac{\beta}{\sqrt{h_d}} \right] \quad (25)$$

Where

$$h_d = |h_m - h_n| \quad (26),$$

$$\alpha = \frac{k_c P}{14.84A}$$

And

$$\beta = \frac{0.222\nu}{\sqrt{\frac{g}{L} \left(\frac{A}{P}\right)^3}} \quad (27)$$



The function of equation (25) tends to infinity when the head difference between the nodes, h_d tends to zero. A clipping function is therefore required to limit the turbulent flow coefficient to a reasonably high value. Furthermore, the clipping function needs to have continuity and continuous derivatives to satisfy the Newton-Raphson requirement. The following function satisfies the required conditions:

$$f_{g\varepsilon} = C_1 h_d^2 + C_2 \quad (28)$$

Where

$f_{g\varepsilon}$ is the clipped function for when h_d is less than e . Equation (25) is used when h_d is greater than e . The values of C_1 and C_2 are as follows:

$$C_1 = \frac{-(\alpha\sqrt{\varepsilon} + \beta)\ln(\alpha + \beta/\sqrt{\varepsilon}) - \beta}{9.21034\varepsilon^{(5/2)}(\alpha\sqrt{\varepsilon} + \beta)} \quad (29)$$

And

$$C_2 = \frac{\log(\alpha + \beta/\sqrt{\varepsilon})}{\sqrt{\varepsilon}} - C_1 \varepsilon^2 \quad (30)$$

The function of equation (28), with constant values obtained from in equations (29) and (30) has a derivative of zero at $h_d = 0$; $f_{g\varepsilon} = f_{gnm}$ at $h_d = e$; and $df_{g\varepsilon} = df_{gnm}$ at $h_d = e$.

Summarizing, for the Darcy-Weisbach equation, the gradient term is expressed by Equation (25) when h_d is greater than e , and by equation (28) when h_d is less than e . **Figure 16** shows the functional form of the equations.

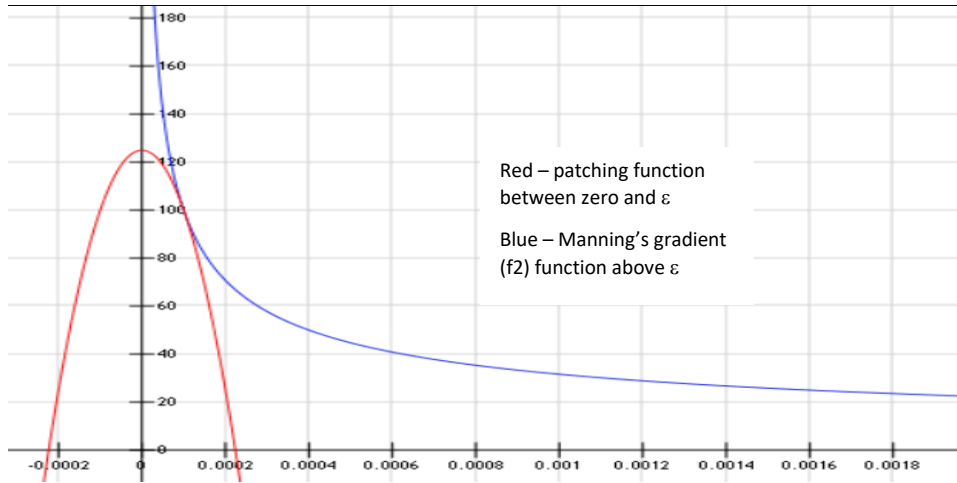


Figure 16. The gradient function f_2 , for Manning's equation with $e = 0.0001$ – Note that the Hazen-Williams equation has similar form but different exponent.

The turbulent flow coefficient, f_{gnm} for any connection nm can be expressed using the Hazen-Williams equation (9) as:

$$f_{gnm} = (|h_m - h_n|)^{-0.46} = |h_d|^{-0.46} \quad (31)$$

The turbulent flow coefficient, f_{gnm} for any connection nm can be expressed using the Manning's equation (10) as:

$$f_{gnm} = (|h_m - h_n|)^{-0.5} = |h_d|^{-0.5} \quad (32)$$

The function of equations (31) and (32) may be written in the general form as:

$$f_{gnm} = |h_d|^{(\eta-1)} \quad (33)$$

Where

$\eta = 0.54$ for the Hazen-Williams equation, and $\eta = 0.5$ for the Manning's equation. Equation (33) tends to infinity when the head difference between the nodes, h_d tends to zero. A clipping function is therefore required to limit the turbulent flow coefficient to a reasonable value. Furthermore, the clipping function needs to have continuity and continuous derivatives to satisfy the Newton-Raphson requirement. The following function satisfies the required conditions:

$$f_{g\epsilon} = \frac{(\eta-1)}{2} \epsilon^{(\eta-3)} h_d^2 + \epsilon^{(\eta-1)} \frac{(3-\eta)}{2} \quad (34)$$



Where

f_{ge} is the clipped function for when h_d is less than e . Equation (33) is used when h_d is greater than e .

Formulation of Flow between the CLN and GWF Domains

The interaction flow between CLN and GWF cells, Γ_{cpn} , in equation (1) may be expressed as

$$\Gamma_{cpn} = \alpha_{cpn} f_{upn} (h_p - h_n), \quad (35)$$

Where

α_{cpn} is the saturated conductance between the CLN cell n and GWF cell p ,

h_p is the head in the GWF cell, and

f_{upn} is the wetted fraction of the upstream perimeter, which is a function of the flow depth.

The term f_{upn} is the same as the saturated fraction, f_{vn} , for vertical and angled CLN cells.

Hence, for vertical and angled CLN cells,

$$f_{upn} = f_{vn}, \quad (36)$$

Where

f_{vn} is computed from equation (2) using the upstream head of h_p and h_n . For horizontal CLN cells, the wetted fraction is the ratio of wetted perimeters of the CLN cells, obtained as

$$f_{upn} = P_{wu} / P_t, \quad (37)$$

Where

P_{wu} is the wetted perimeter computed using the upstream head of h_p and h_n , and

P_t is the total perimeter of the CLN cell.

Several options are provided for computing the conductance term, α_{cpn} , between the CLN cell and the GWF cell. The first option uses a CFP formulation option for water exchange between the GWF and CLN cells in which the conductance values, α_{cpn} , are input to the model. The



second option also uses a CFP formulation option in which skin thickness and skin hydraulic conductivity are input to the model and the conductance is computed using CLN geometry as

$$\alpha_{cpn} = \frac{K_s P_l l}{b_s} \quad (38)$$

Where

K_s is the hydraulic conductivity of the skin surrounding the CLN-to-GWF interface, and

b_s is the thickness of the skin surrounding the CLN-to-GWF interface.

The third option for computing the conductance term α_{cpn} follows Bennett and others (1982) and provides for head loss between the GWF cell and the CLN cell by use of the Thiem equation to compute the flow, as adapted in the MNW Package (Halford and Hanson, 2002; Konikow and others, 2009). For this case, the conductance is approximated for a vertical CLN cell connection with a GWF cell as

$$\alpha_{cpn} = \left[\frac{\ln(r_{oz} / r_n) + S_f}{2\pi l K_{xx} \sqrt{1 / \mathfrak{R}_{rz}}} \right]^{-1} + \left[C \Gamma_{cpn}^{P-1} \right]^{-1} \quad (39)$$

Where

r_{oz} (representing the radius of influence in the Thiem equation) is the effective external radius of a GWF cell for a connected vertical CLN cell,

S_f is the skin resistance factor,

$\mathfrak{R}_{rz}$ is the x:y anisotropy ratio (K_{xx} / K_{yy}),

C is the nonlinear well-loss coefficient; and

P is the exponent for the nonlinear well loss term.

For a horizontal CLN cell connection, the conductance is approximated as

$$\alpha_{cpn} = \left[\frac{\ln(r_{oh} / r_n) + S_f}{2\pi l K_{xx} \sqrt{1 / \mathfrak{R}_{rh}}} \right]^{-1} + \left[C \Gamma_{cpn}^{P-1} \right]^{-1} \quad (40)$$



Where

r_{oh} is the effective external radius of a GWF cell for a connected horizontal CLN cell, and

$\mathfrak{R}_{rh}$ is the horizontal to vertical anisotropy ratio (K_{xx} / K_{zz}).

The last term in equations (39) and (40) is nonlinear and may cause numerical problems; moreover, this term is sometimes unnecessary. As a result, the term has been eliminated from the current CLN Process (that is, $C = 0$ in equations (39) and (40)). The vertical conduit assumption is used for a slanted conduit, thereby incorporating the horizontal radius of influence and anisotropy approximations of equation (39).

The effective external radius of the GWF cell with a vertical CLN cell connection is computed as

$$r_{oz} = 0.14 \left[\frac{\sqrt{2 \sum_{m \in \eta_n} (L_{nm})^2}}{N_m} \right], \quad (41)$$

Where

N_m is the number of adjacent connections to GWF cell, n .

Equation (41) reduces to the isotropic equation for the effective external radius for a rectangular coordinate system of a finite-difference connectivity, as defined by Peaceman (1983). The horizontal isotropic approximation of equation (41) is considered adequate for unstructured grids, as opposed to evaluating components along the horizontal principal coordinate directions for horizontally anisotropic conditions.

The effective external radius of a GWF cell with a horizontal CLN cell connection is computed as

$$r_{oh} = 0.28 \left[\frac{\sqrt{\Delta H^2 \sqrt{1 / \mathfrak{R}_{rh}} + \Delta Z^2 \sqrt{\mathfrak{R}_{rh}}}}{\sqrt[4]{1 / \mathfrak{R}_{rh}} + \sqrt[4]{\mathfrak{R}_{rh}}} \right], \quad (42)$$

Where

ΔZ is the thickness of the GWF cell, and



ΔH is the horizontal cell dimension normal to the line of the CLN cell, approximated as

$$\Delta H = \left[\frac{2 \sum_{m \in \eta_n} (L_{nm})}{N_m} \right] \quad (43)$$

Equations (42) and (43) reduce to the anisotropic equation for the effective external radius, for a rectangular coordinate system of a finite-difference connectivity, as defined by Peaceman (1983). The approximations of equations (42) and (43) are deemed adequate, as opposed to evaluating components along the horizontal coordinate directions normal to the line of the conduit cell, because the basic equations of Peaceman (1983) are themselves approximations for horizontal conduits. Because suitable equations for estimating cell-to-well conductance of horizontal wells are not well defined, it is not warranted to make the connection to horizontal conduits more complex. Furthermore, because of incomplete knowledge of conduit geometry or of the skin factor or well efficiency, and regardless of the option selected to assemble α_{cpn} , it is likely that an estimated value of this parameter will be modified during model calibration. Equations (39) through (43) are generalizations of the equations for rectangular finite-difference cells presented by Halford and Hanson (2002) and Konikow and others (2009).

Confined Option for CLN-CLN Flow

An option for confined flow between CLN cells is also provided and is similar to the confined option available for each layer of the GWF Process grid. Each CLN cell can be designated as confined or convertible. With this option, the term f_{unm} in equation (4) is held constant at a value of 1, allowing for flow from an upstream cell to occur under saturated conditions, regardless of the saturation condition of the CLN cell. The option can be useful for different cases of flow in horizontal or vertical conduits. This option applies only to CLN-CLN flow and not to flow between CLN and GWF domains.

Flow to Convertible Cells for CLN-CLN Flow

Special considerations are required for flow to convertible cells that can dry and rewet. When flow is between two cells that are wet, the head difference between them creates the driving force for flow. When the downstream cell is dry, however, a reference elevation can be used instead of the downstream head to express flow between the nodes. This “flow-to-dry-cell”



option is the default condition for the BCF and LPF Packages of MODFLOW-2005 for downward flow to a partially saturated cell.

The “flow-to-dry-cell” option is also available for flow between two CLN cells or for flow between a CLN cell and a GWF cell if the cells are convertible. For this case, the respective flow can be expressed as

$$\begin{aligned} Q_{nm} &= C_{nm}^0 f(h_{ups})(h_m - h_n) & \text{for } h_n > e_n \\ Q_{nm} &= C_{nm}^0 f(h_{ups})(h_m - e_n) & \text{for } h_n < e_n \end{aligned} \quad (44)$$

Where

C_m^0 is the constant portion (fully saturated) of the conductance term between cells n and m ,

$f(h_{ups})$ is the upstream-weighted nonlinear term, and

e_n is a reference elevation for the downstream node, n .

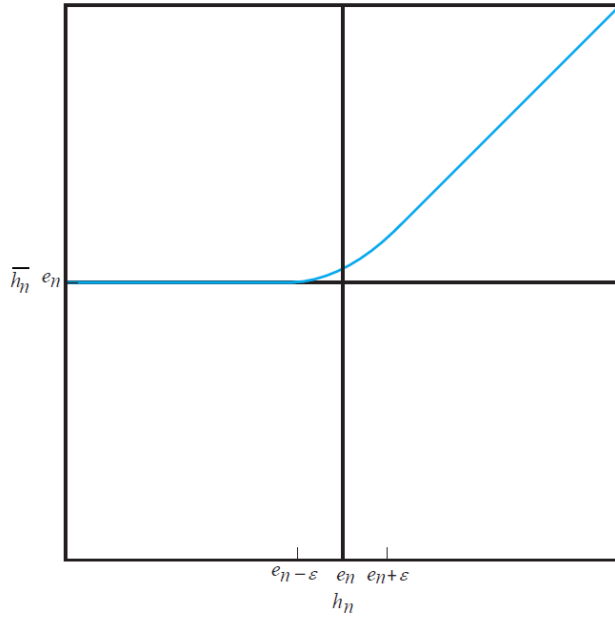
For the remainder of this section, the upstream node is assumed to be m and the downstream node is assumed to be n . In the GWF Process, when downward flow is from cell m to a partially saturated cell n , the top elevation of cell n is the reference elevation, e_n . Equation (44) has a slope discontinuity at $h_n = e_n$. For a smooth transition, which is required by the Newton-Raphson method, equation (44) can be recast as:

$$Q_{nm} = C_{nm}^0 f[h_{ups}] f_gh_m - \bar{h}_n, \quad (45)$$

Where

$\bar{h}_n$ is a function that transitions between head at the downstream node, h_n , and the reference elevation, e_n in equation (45), in a smooth manner as shown on **Figure 17**. The function is given as:

$$\bar{h}_n = \begin{cases} h_n & \text{for } (h_n - e_n) > \varepsilon \\ \frac{(h_n - e_n)^2}{4\varepsilon} + \frac{(h_n - e_n)}{2} + \frac{\varepsilon}{4} + e_n & \text{for } \varepsilon > (h_n - e_n) > -\varepsilon \\ e_n & \text{for } (h_n - e_n) < -\varepsilon \end{cases} \quad (46)$$



EXPLANATION

—	Smooth function	$\bar{h}_n$	Head value used in flow calculation
e_n	Reference elevation	h_n	Head value in cell n
ε	Short smoothing distance		

Figure 17. Graph of smooth function used to express the head value used in the flow calculation to the head value of the downstream cell as it transitions from saturated to partially saturated conditions.

Note that ε is a small number (10^{-4} in MODFLOW-USG).

Where

ε is the distance over which the function is smoothed. Equation (46) satisfies the continuity of the function and of the derivatives at $(h_n - e_n) = \pm \varepsilon$. Thus,

$$\bar{h}_n = e_n \quad \text{for} \quad (h_n - e_n) = -\varepsilon \quad (47)$$

$$\frac{d\bar{h}_n}{dh} = 0 \quad \text{for} \quad (h_n - e_n) = -\varepsilon \quad (48)$$

and

$$\frac{d\bar{h}_n}{dh} = 1 \quad \text{for} \quad (h_n - e_n) = \varepsilon \quad (49)$$



Just as was done for vertical flow correction in the GWF Process of MODFLOW-2005, flow to a convertible cell is expressed using Picard linearization in MODFLOW-USG as a correction on the right-hand-side vector of the matrix equations, with assembly of the conductivity term into the left-hand-side coefficient matrix in a symmetric manner, as is done for the traditional Darcy flow equation. Thus, equation (45) can be expanded as

$$Q_{nm} = C_{nm}^0 f[h_{ups}] f_g[h_m - \bar{h}_n](h_m - h_n) + C_{nm}^0 f[h_{ups}] f_g[h_m - \bar{h}_n](h_n - \bar{h}_n), \quad (50)$$

where the first term on the right-hand-side of equation (50) is assembled into the coefficient matrix and the second term is assembled into the right-hand-side vector.

For flow between two CLN cells, the reference elevation, e_n , is taken as the bottom elevation of the upstream CLN cell; for flow between a CLN cell and a GWF cell, the reference elevation, e_n , is taken as the higher of the two bottom elevations:

$$e_n = \begin{cases} BOT_{CLN} & \text{for } BOT_{CLN} > BOT_{GWF} \\ BOT_{GWF} & \text{for } BOT_{GWF} > BOT_{CLN} \end{cases} \quad (51)$$

Where

BOT_{CLN} is the bottom elevation of the CLN cell, and

BOT_{GWF} is the bottom elevation of the GWF cell.

Cylindrical Conduit Geometry Type

The CLN Process is designed in a modular fashion to support different geometry types. In the present MODFLOW-USG version, the CLN Process supports laminar flow in cylindrical conduits. Other one-dimensional features could also be represented by the process, provided functional forms can be provided for the following:

- The cross-sectional area, perimeter, hydraulic radius, and saturated hydraulic conductivity, as a function of its input geometric characteristics, and
- The saturated cross-sectional area and wetted perimeter as a function of the potentiometric head. This latter condition is only required for nonlinear (unconfined) conditions and is not required for confined (linear) conditions.



For a cylindrical conduit represented as a CLN cell, the input required is the radius of the cylindrical conduit, R_{cond} . The cross-sectional area, a_c , is then computed for a cylindrical conduit as

$$a_c = \pi R_{cond}^2 \quad (52)$$

The total perimeter, P_t , for a cylindrical conduit is the circumference, given as

$$P_t = 2\pi R_{cond} \quad (53)$$

The hydraulic radius, r_n , of a cylindrical conduit is the same as its radius and, therefore,

$$r_n = R_{cond} \quad (54)$$

The saturated hydraulic conductivity for flow in a cylindrical conduit may be obtained by assuming laminar flow. Following Sudicky and others (1995), equation (4) can be expressed in a similar manner to Darcy's Law by rearranging the terms to give an effective hydraulic conductivity as

$$K_c = \frac{\rho g}{8\mu} R_{cond}^2 \quad (55)$$

Where

ρ is the density of water,

g is the gravitational constant, and

μ is the dynamic viscosity of water.

In addition, the wetted cross-sectional area of a cylindrical conduit is required for computing the saturated fraction of equation (34) for a horizontal unconfined CLN cell. This area is computed for a given potentiometric head from geometric considerations as:



$$\begin{aligned}
 a_w &= 0 & \text{for } d < 0 \\
 a_w &= R_{cond}^2 \cos^{-1} \left(\frac{R_{cond} - d}{R_{cond}} \right) - (R_{cond} - d) \sqrt{R_{cond}^2 - (R_{cond} - d)^2} & \text{for } 0 < d < R_{cond} \\
 a_w &= R_{cond}^2 \left[\pi - \cos^{-1} \left(\frac{d - R_{cond}}{R_{cond}} \right) \right] + (d - R_{cond}) \sqrt{R_{cond}^2 - (R_{cond} - d)^2} & \text{for } R_{cond} < d < 2R_{cond} \\
 a_w &= \pi R_{cond}^2 & \text{for } d > 2R_{cond}
 \end{aligned} \tag{56}$$

Where

d is the depth of flow in the horizontal conduit, obtained as

$$d = h - BOT \tag{57}$$

Finally, the wetted perimeter is required to compute the surface area for interaction between the CLN and GWF domains, as noted in equation (37). The wetted perimeter for a circular conduit section is obtained as

$$\begin{aligned}
 P_w &= 0 & \text{for } d < 0 \\
 P_w &= 2R_{cond} \cos^{-1} \left(\frac{R_{cond} - d}{R_{cond}} \right) & \text{for } 0 < d < R_{cond} \\
 P_w &= 2R_{cond} \left[\pi - \cos^{-1} \left(\frac{d - R_{cond}}{R_{cond}} \right) \right] & \text{for } R_{cond} < d < 2R_{cond} \\
 P_w &= 2\pi R_{cond} & \text{for } d > 2R_{cond}
 \end{aligned} \tag{58}$$

Similar equations can also be written for different section types.

For fully saturated flow in a CLN cell, the saturated fraction $f_v = 1$. For this case, equation (31) expresses linear steady-state flow through the CLN domain and equations (32), (34), (55), and (56) are not required. Furthermore, for the saturated case, equation (37) is unity and equation (57) is not required.

Newton Raphson Linearization of CLN flow terms

The equations generated by the turbulent flow formulation are an extension of those generated by the upstream weighted formulation for unconfined laminar flow. Specifically, the flow terms of the turbulent CLN equations includes a nonlinear function of the head difference between nodes which also benefit from a Newton Raphson expansion. The derivation below, generalizes



the finite-difference expansion to include this turbulent flow coefficient term, whose contribution is noted in blue font, to identify additions to the generalized matrix developed for laminar flow.

For any node n , the flow term Q_{nm} between nodes n and neighbor m may be expressed in general form by rewriting equation (11) as:

$$Q_{nm} = C_{nm}^0 f[h_{ups}] f_g[h_m - h_n] (h_m - h_n) \quad (59)$$

Where

C_{nm}^0 is the constant portion of the conductance term between nodes n and m , $f_g[h_m - h_n]$ is the extra non-linear turbulent flow coefficient portion of the conductivity matrix, and $f(h_{ups})$ is the upstream weighted nonlinear term. Applying the Newton formulation to Q_{nm} for row n of the matrix gives:

$$\frac{\partial Q}{\partial h_n} \Delta h_n + \frac{\partial Q}{\partial h_m} \Delta h_m = -Q \quad (60)$$

Where

Δh is the change in the head value from the previous nonlinear iteration to the current one expressed as

$$\Delta h = h^k - h^{k-1} \quad (61)$$

Where

h^k is the unknown head value being estimated for the current time step and h^{k-1} is the estimate at the previous Newton iteration. Equation (60) can be expressed in terms of the MODFLOW-2005 variable of solution (the unknown h^k), by substituting (61) into (60) and rearranging as

$$\frac{\partial Q}{\partial h_n} h_n^k + \frac{\partial Q}{\partial h_m} h_m^k = -Q + \frac{\partial Q}{\partial h_n} h_n^{k-1} + \frac{\partial Q}{\partial h_m} h_m^{k-1} \quad (62)$$

The derivative with respect to h_n and h_m are obtained from equation (59) as



$$\frac{\partial Q}{\partial h_n} = \left\{ -C_{nm}^0 f[h_{ups}] + C_{nm}^0 \frac{\partial f[h_{ups}]}{\partial h_n} (h_m - h_n) \right\} f_g[h_m - h_n] - C_{nm}^0 f[h_{ups}] (h_m - h_n) \frac{\partial f_g[h_m - h_n]}{\partial [h_m - h_n]} \quad (63)$$

And

$$\frac{\partial Q}{\partial h_m} = \left\{ C_{nm}^0 f[h_{ups}] + C_{nm}^0 \frac{\partial f[h_{ups}]}{\partial h_m} (h_m - h_n) \right\} f_g[h_m - h_n] + C_{nm}^0 f[h_{ups}] (h_m - h_n) \frac{\partial f_g[h_m - h_n]}{\partial [h_m - h_n]} \quad (64)$$

Note that the derivative of $f_g[h_m - h_n]$ with respect to $[h_m - h_n]$ can be obtained numerically for any of the turbulent flow formulations: the Darcy-Weisbach, Hazen-Williams or Manning's equation appropriately clipped before the function grows to a very large number and smoothed for Newton Raphson considerations as was shown earlier.

The right-hand side of equation (62) is expressed in terms of known variables estimated from the previous nonlinear iteration. This term can be expanded further by substituting Equations (59), (63) and (64) for the flux and its derivatives to give

$$RHS = \left\{ \left[C_{nm}^0 \frac{\partial f[h_{ups}]}{\partial h_n} (h_m - h_n) \right] h_n + \left[C_{nm}^0 \frac{\partial f[h_{ups}]}{\partial h_m} (h_m - h_n) \right] h_m \right\} f_g[h_m - h_n] + C_{nm}^0 f[h_{ups}] (h_m - h_n)^2 \frac{\partial f_g[h_m - h_n]}{\partial [h_m - h_n]} \quad (65)$$

Thus, if the flow term is of the form of equation (59), the Newton Raphson procedure can be implemented within the MODFLOW-2005 solution framework by adding the last terms of equations (63) and (64) to the left-hand side coefficient matrix generated by Picard discretization of the flow equation – the first term on the right-hand side of equations (63) and (64) being the flow conductance term that is already assembled by the MODFLOW-2005 formulate routines. Also, equation (65) is added to the right-hand side vector to complete the Newton procedure of linearizing a flow term expressed in the form of equation (59).

Newton Raphson with Turbulent flow and Flow to Dry Cell Option (conversion between diffusion wave and kinematic wave)



The flow term with flow to dry cell option can be written for node m being the upstream node as:

$$Q_{nm} = C_{nm}^0 f[h_m] f_g[h_m - \bar{h}_n] (h_m - \bar{h}_n) \quad (66)$$

Which is re-written for Picard considerations as:

$$Q_{nm} = C_{nm}^0 f[h_m] f_g[h_m - \bar{h}_n] (h_m - h_n) + C_{nm}^0 f[h_m] f_g[h_m - \bar{h}_n] (h_n - \bar{h}_n) \quad (66a)$$

Where the second term is put on the RHS vector and the first term goes into the LHS matrix for Picard considerations. The Newton-Raphson equations (60), (61) and (62) are now applied to the flow term of equation (66). The derivative terms for this case are obtained as

$$\frac{\partial Q_{nm}}{\partial h_n} = -C_{nm}^0 f[h_m] f_g[h_m - \bar{h}_n] \frac{\partial \bar{h}_n}{\partial h_n} - C_{nm}^0 f[h_m] \frac{\partial f_g[h_m - \bar{h}_n]}{\partial [h_m - \bar{h}_n]} \cdot \frac{\partial \bar{h}_n}{\partial h_n} (h_m - \bar{h}_n) \quad (67)$$

and

$$\begin{aligned} \frac{\partial Q_{nm}}{\partial h_m} = & C_{nm}^0 f[h_m] f_g[h_m - \bar{h}_n] + \\ & C_{nm}^0 \frac{\partial f[h_m]}{\partial h_m} f_g[h_m - \bar{h}_n] (h_m - \bar{h}_n) + C_{nm}^0 f[h_m] \frac{\partial f_g[h_m - \bar{h}_n]}{\partial [h_m - \bar{h}_n]} (h_m - \bar{h}_n) \end{aligned} \quad (68)$$

Where n is the downstream node and $\bar{h}_n$ is a function that transitions between head at the downstream node, h_n , and the reference elevation, e_n when flow to dry cell option is used. Similar equations can be obtained for m being a downstream node. Because the $C_{nm}^0 f[h_{ups}] f_g[h_m - \bar{h}_n]$ term is already appropriately entered into the solution matrix for Picard linearization of the equations (negative on the diagonals), the Newton-Raphson equation (62) further requires that the flow term on the diagonal of row n and off-diagonal term of row m be multiplied by $\partial \bar{h}_n / \partial h_n$. The pivot term of row n contains $-C_{nm}^0 f[h_m] f_g[h_m - \bar{h}_n]$ as well as other terms. Therefore, the adjustment required for the pivot is to add the term $C_{nm}^0 f[h_m] f_g[h_m - \bar{h}_n]$ and subtract the term $C_{nm}^0 f[h_m] f_g[h_m - \bar{h}_n] \partial \bar{h}_n / \partial h_n$. Also, the second and third terms of equation (68) are added to the pivot term of row m and off-diagonal term of row n to form the Jacobian matrix when m is the upstream node. Similar computations can be done for n being the upstream node.

As per equation (62), the right-hand-side vector is expanded as



$$\begin{aligned}
 RHS = & -C_{nm}^0 f(h_m) f_g[h_m - \bar{h}_n] \left[\frac{\partial \bar{h}_n}{\partial h_n} h_n - \bar{h}_n \right] + C_{nm}^0 \frac{\partial f(h_m)}{\partial h_m} f_g[h_m - \bar{h}_n] (h_m - \bar{h}_n) h_m \\
 & - C_{nm}^0 f(h_m) \frac{\partial f_g[h_m - \bar{h}_n]}{\partial [h_m - \bar{h}_n]} (h_m - \bar{h}_n) \left[\frac{\partial \bar{h}_n}{\partial h_n} h_n - h_m \right]
 \end{aligned} \tag{69}$$

Considering that the negative of the second term of equation (66a) is already entered into the right-hand-side vector during Picard considerations, application of the Newton-Raphson update necessitates removing it and adding equation (62) to give the update to the right-hand-side vector for this “flow-to-dry-cell” option as

$$\begin{aligned}
 RHS_{update} = & -C_{nm}^0 f(h_m) f_g[h_m - \bar{h}_n] \left[\frac{\partial \bar{h}_n}{\partial h_n} h_n - h_n \right] + C_{nm}^0 \frac{\partial f(h_m)}{\partial h_m} f_g[h_m - \bar{h}_n] (h_m - \bar{h}_n) h_m \\
 & - C_{nm}^0 f(h_m) \frac{\partial f_g[h_m - \bar{h}_n]}{\partial [h_m - \bar{h}_n]} (h_m - \bar{h}_n) \left[\frac{\partial \bar{h}_n}{\partial h_n} h_n - h_m \right]
 \end{aligned} \tag{70}$$

In tests, it was noted that the Newton expansion of the gradient term created convergence difficulties when the water level gradient was close to zero. Therefore, Newton expansion of the Newton term has been disconnected from the code and has proven to provide strong convergence behavior.

Relation between Skin Friction and Well Efficiency

The CLN package includes an option to read in the well efficiency, and have the code compute the skin resistance factor (S_f of Equation 39) internally. This is done, through the definition of well efficiency, as:

$$E = Q_a / Q_t \tag{71}$$

Where

E is the well efficiency; Q_a is the actual pumping; and Q_t is the theoretical pumping rate for the same unit drawdown. If we use the Thiem equation with a skin resistance to estimate the actual pumping, and the Thiem equation without skin resistance to estimate the theoretical pumping, then we get the efficiency in terms of skin factor as

$$E = \frac{\ln(r_e / r_w)}{\ln(r_e / r_w) + S_f} \tag{72}$$



Rearranging, gives the skin factor in terms of efficiency as:

$$S_f = \frac{(1-E)}{E} \ln(r_e / r_w) \quad (73)$$

Thus, if the index IFCON=4, then efficiency is read and skin factor is computed internally by the code as per equation (73) above.

Dual Porosity Flow (DPF) Package

A dual porosity storage term is also included with the transient flow formulation of USG-Transport. The dual porosity flow formulation was developed originally for petroleum reservoirs [Warren, J. E., & Root, P. J., 1963, September 1. The Behavior of Naturally Fractured Reservoirs. Society of Petroleum Engineers. doi:10.2118/426-PA]. Dual porosity flow is significant when the subsurface is comprised of contrasting flow systems, such as a network of fractures (fast flow) within a porous matrix (slow flow) subsurface system. Mobile and immobile spaces, or inclusion of fine clay lenses within a transmissive sandy aquifer system are also examples of a dual porosity (or dual domain) system.

The dual porosity flow formulation accommodates sub-grid scale physics in fractured reservoirs, where flow is dominant through the high permeability connected fracture network, while storage is significant in the otherwise lower permeability porous matrix blocks. The dual porosity formulation therefore conceptualizes out the two domains at the grid-block scale, attributing a fraction of the grid-block to the fracture (mobile) domain, while the remainder of the grid-block is occupied by the matrix (immobile) domain. Water flows from one grid-block to the next within the fracture domain as governed by the flow equation (1). The equation is scaled by the “fracture porosity”, which is defined as the fraction of the grid-block that is occupied by the fracture domain. Properties of the flow equation that are read in the BCF or LPF file represent the total medium by default. Thus, the hydraulic conductivity terms are effective properties of the bulk system. However, the FRAHK option may be supplied to allow input hydraulic conductivity to represent that solely of the fracture domain.

Within a grid-block, movement of water can occur between the fracture and matrix domain. A mass transfer coefficient represents this movement and is generally governed by the hydraulic conductivity of the matrix. The mass transfer rate is proportional to $n \cdot K \cdot A / L$ of each representative matrix block, where K is the hydraulic conductivity of the matrix domain, A is the



face area between fracture and matrix domains, L is the length from the face to the center of the immobile domain matrix block, and n is the number of immobile domain matrix blocks per unit volume within the numerical cell.

The flow equation of USG-Transport includes a variety of formulations for the storage term including confined, unconfined, convertible and unsaturated formulations. For the dual porosity flow formulation, the various storage options apply for the fracture domain, as well as for the matrix domain. Furthermore, when solving the unsaturated flow equation, if the options BUBBLEPT and FULLYDRY are used in the BCF or LPF file, then these options apply also to the dual porosity flow matrix domain. Specifically, when the Richards equation is solved, the BUBBLEPT option includes a bubble point (air entry) pressure that will also be input for the immobile domain of dual porosity flow. Also, the FULLYDRY option allows the soil saturation to go to zero, below the prescribed residual saturation; however, flow ceases at or below residual saturation levels. In this case, the van Genuchten moisture retention function is applied with no residual saturation so the saturation can vary from 0 to 1. However, the relative permeability function uses an effective saturation that makes the relative permeability zero when saturation is at or below residual saturation causing flow to also be zero. Evapotranspiration or other local sink mechanisms can still remove water to be below the residual saturation value. Details on this option are further provided in the Unsaturated Flow Solution section.

It is important to note that if there is transient flow, the dual porosity flow option may be important if the simulation further considers dual porosity transport. This is because water containing solutes would move in and out of storage in both fracture and matrix domains – in many cases, storage of water is more dominant in the matrix domain and thus it is essential to represent this advective movement of solutes as water flows in and out of storage within the matrix domain. The dual porosity transport formulation of USG-Transport includes this advective transfer term between fracture and matrix along with the diffusive mass transfer rate term. Also, it is significant to note that the porosity within each of the domains should be equal to or larger than the associated specific yield term to have physical consistency as discussed with the porosity definitions at the beginning of the document.

Note that when flow is dual porosity, the immobile domain material properties are read here. Specifically, these include the mass transfer rate, specific storage, specific yield, and, if the simulation is for unsaturated flow, the van Genuchten parameters for the matrix (immobile) domain. Fracture porosity is also read here. Also, the storage properties that were read from the



BCF or LPF input file represent the fracture (mobile) domain. Thus, these are multiplied by the fracture porosity (equal to fracture volume over total volume) to solve the problem in terms of a total volume. Also note that the dual porosity flow package can be used only if the upstream weighting formulation is used for the unconfined flow equation. This is because upstream weighting is also performed on the transfer term between domains to keep the results physical.

Time-Variant Materials Package (TVM v2) for MODFLOW-USG

Damian Merrick, HydroAlgorithmics, Canberra, Australia.

Vivek Bedekar, S. S. Papadopoulos and Associates, Maryland, USA.

Introduction and Formulation

The Time-Variant Materials (TVM) Package was designed and developed by Damian Merrick to vary parameters of a flow model and has been incorporated into USG-Transport. [Vivek Bedekar at S. S. Papadopoulos and Associates, Inc](#) has further assisted with incorporating routines for the TVM package to work with solute transport.

The TVM Package allows hydraulic conductivity and storage values specified in the LPF package to be changed as a step function between stress periods or in a continuous manner throughout a transient simulation. [Information in this document specific to the USG-Beta and USG-Transport versions of MODFLOW-USG is highlighted by use of a blue font for convenience.](#)

New hydraulic conductivity and storage values (K_x , K_z , S_s and S_y) may be given at stress period boundaries, for some or all cells in a model. The TVM package updates K_x , K_z , S_s and S_y values at each time step of the simulation by interpolating between the value at the start of the period and the value specified for the end of the period. Interpolation may be linear or logarithmic to a given base. The step function option provides instantaneous changes to the property values at each stress period. Note that this may cause large disequilibrium conditions between the state of the system and the new property value which may require smaller initial time step sizes to recover.

If linear interpolation is used, for starting and ending property values $VALUEA$ and $VALUEB$ in stress period N , the resulting property value will be calculated as:

$$VALUE = VALUEA + \frac{PERTIM(VALUEB - VALUEA)}{PERLEN(N)}$$



where PERTIM is the current simulation time within stress period N, and PERLEN(N) is the length of stress period N.

If logarithmic interpolation is used, the resulting property value will be calculated as:

$$\text{VALUE} = B^{\log_B(\text{VALUEA}) + \frac{\text{PERTIM}(\log_B(\text{VALUEB}) - \log_B(\text{VALUEA}))}{\text{PERLEN}(N)}}$$

where B is the user-specified logarithm base.

Figure 18 below shows a comparative example of these interpolation methods with hydraulic conductivity Kx starting at a value of 0.1 at the start of a simulation, remaining unchanged for stress period 1, and then changing to a value of 1.0 at the end of stress period 2.

Values are assumed to change continuously over time, such that a property value for a given cell at the end of any stress period is the same as its value at the start of the next stress period.

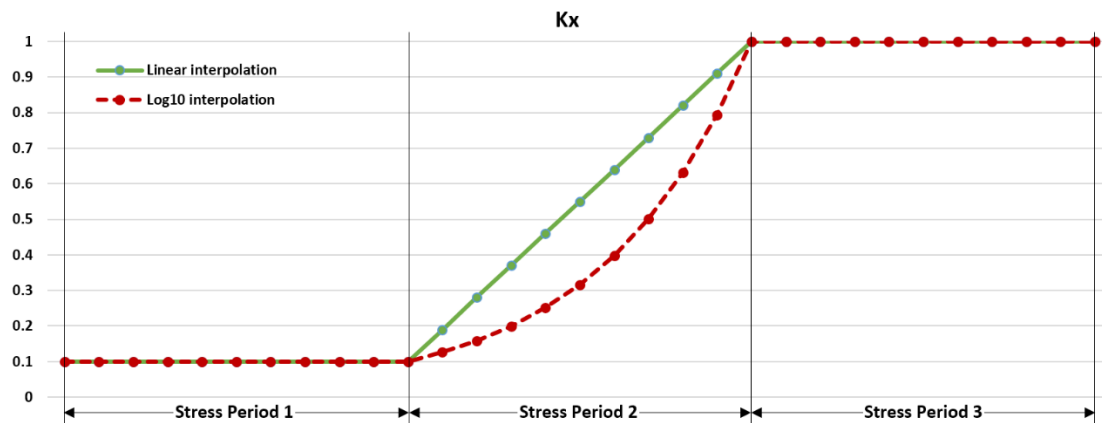


Figure 18. An example of hydraulic conductivity Kx changing over one stress period of a simulation, showing the use of linear and logarithmic (base 10) interpolation methods.

At the beginning the TVM package first reads a set of values for the start of the simulation; at the beginning of the first stress period, these override the equivalent values given in the LPF package. Following this, a set of values is read for every stress period; these specify the values that properties should take at the end of each stress period. In total, TVM reads (NPER + 1) sets of changed values, where NPER is the number of stress periods in the simulation.

All value changes are optional, meaning that new values need only be entered for those model cells and properties that are to change, and only at the stress periods at which they should



change. Any cell that does not have a new property value specified at a given stress period retains the same value for that property throughout that stress period.

The TVM package does not alter the time-stepping behavior of MODFLOW-USG, and only acts once per time step. Consideration should be given in specifying time steps for the simulation such that values do not change so rapidly as to cause instability. The Adaptive Time Stepping package may be useful for this purpose.

Specific storage and specific yield values are converted by TVM into storage capacities, SC1 and SC2, as used internally by MODFLOW-USG. It is important to note that values are printed by TVM in the listing file for the storage capacities SC1 and SC2, not the specific storage or specific yield values entered in the TVM package input file. SC1 values are calculated as the original S_s values multiplied by cell volume and SC2 values are calculated as the S_y values multiplied by cell area. If the STORAGEEFFICIENT option in the LPF package is used, original S_s values actually represent storage coefficients instead of specific storage, and SC1 will be calculated as the original value multiplied by cell area.

Compatibility of TVM2 with Other Packages

Note that the TVM package is not compatible with the unsaturated flow options of the SFR2 package. Changed hydraulic conductivity and storage values will not take effect in SFR package calculations. TVM is also incompatible with the BCF package; LPF must be used instead when TVM is active.

Extension of TVM2 for Dual Domain Flow

If dual domain flow is also simulated the TVM package will allow for the dual domain rate transfer term to also change at any stress period for any cell as per the methodologies used above for the porous medium material properties. Options of logarithmic, linear or step function may also be used for the rate transfer term. Input instructions pertaining to the dual domain or immobile domain properties are noted in blue. Also note that data items 1 and 2 require an entry for the dual domain term even if dual domain is not simulated. Data item 7 is not entered if dual domain is not simulated.



Extension of TVM2 for Solute Transport Simulations

TVM package will allow for transient porosity values for solute transport simulations. Changes to porosity are incorporated in the solute transport equation as a mass conserved formulation, as detailed below. Like transient flow storage properties (S_s and S_y), step function changes for porosity are not permitted.

TVM2 Adjustments to Storage Formulation

As of version 2 of TVM, changes to storage properties S_s and S_y are incorporated into the groundwater flow equations such that corresponding changes to head occur to correctly conserve water volume. No changes to the TVM input file are required if coming from version 1 of TVM. However, these changes must be integrated over at least one time step, and step function changes to S_s and S_y are no longer permitted in TVM version 2 (step function changes to K_x and K_z are still allowed).

The mathematical formulation for storage changes in a cell depends on the LAYTYP flag specified in the LPF package for the cell's layer, and each of these cases is detailed below. It is recommended that LAYTYP 4 be used for convertible layers where storage changes are made, as LAYTYP 4 provides a continuous storage function as required by the Newton solver. Using the traditional convertible layer type $\text{LAYTYP} > 0$ ($\neq 4$) results in a discontinuity at the boundary of full saturation, which may result in instability when a storage change forces a cell's head to change from confined to unconfined conditions. [Note that the Richards formulation \(LAYTYP 5\), if enabled, uses the same storage adjustment as LAYTYP 4.](#)

CONFINED (LAYTYP = 0)

Without storage changes, MODFLOW-USG implements the confined time-dependent storage term as

$$\Delta S = \frac{S_s V}{\Delta t} (h^{t+\Delta t} - h^t)$$

Where

S_s is specific storage, V is cell volume, Δt is time step length, h^t is head at the beginning of the time step, and $h^{t+\Delta t}$ is head at the end of the time step.

When specific storage is changed during a simulation, TVM adjusts this to



$$\Delta S = \frac{V}{\Delta t} (S_s^{t+\Delta t} h^{t+\Delta t} - S_s^t h^t)$$

Where

S_s^t is specific storage at the beginning of the time step and $S_s^{t+\Delta t}$ is specific storage at the end of the time step.

CONVERTIBLE (LAYTYP > 0, LAYTYP ≠ 4, **LAYTYP ≠ 5**)

Without storage changes, MODFLOW-USG implements the convertible time-dependent storage term as

$$\Delta S = \frac{1}{\Delta t} \begin{cases} S_s V (h^{t+\Delta t} - h^t), h^t > TOP \wedge h^{t+\Delta t} > TOP \\ S_y V (S_w^{t+\Delta t} - S_w^t), h^t \leq TOP \wedge h^{t+\Delta t} \leq TOP \\ [S_y V (S_w^{t+\Delta t} - 1) + S_s V (TOP - h^t)], h^t > TOP \wedge h^{t+\Delta t} \leq TOP \\ [S_s V (h^{t+\Delta t} - TOP) + S_y V (1 - S_w^t)], h^t \leq TOP \wedge h^{t+\Delta t} > TOP \end{cases}$$

Where

S_w^t is the saturated fraction of the cell's thickness at the beginning of the time step, $S_w^{t+\Delta t}$ is the saturated thickness fraction at the end of the time step, S_s is specific storage, S_y is specific yield, V is cell volume, TOP is cell top elevation, Δt is time step length, h^t is head at the beginning of the time step, and $h^{t+\Delta t}$ is head at the end of the time step.

When either or both of specific storage and specific yield are changed during a simulation, TVM adjusts the formulation to

$$\Delta S = \frac{1}{\Delta t} \begin{cases} \begin{bmatrix} S_s^{t+\Delta t} V (h^{t+\Delta t} - TOP) \\ -S_s^t V (h^t - TOP) + (S_y^{t+\Delta t} - S_y^t) V \end{bmatrix}, h^t > TOP \wedge h^{t+\Delta t} > TOP \\ \begin{bmatrix} S_y^{t+\Delta t} V S_w^{t+\Delta t} \\ -S_y^t V S_w^t \end{bmatrix}, h^t \leq TOP \wedge h^{t+\Delta t} \leq TOP \\ \begin{bmatrix} S_y^{t+\Delta t} V S_w^{t+\Delta t} \\ -S_s^t V (h^t - TOP) - S_y^t V \end{bmatrix}, h^t > TOP \wedge h^{t+\Delta t} \leq TOP \\ \begin{bmatrix} S_s^{t+\Delta t} V (h^{t+\Delta t} - TOP) + S_y^{t+\Delta t} V \\ -S_y^t V S_w^t \end{bmatrix}, h^t \leq TOP \wedge h^{t+\Delta t} > TOP \end{cases}$$



Where

S_s^t is specific storage at the beginning of the time step, $S_s^{t+\Delta t}$ is specific storage at the end of the time step, S_y^t is specific yield at the beginning of the time step and $S_y^{t+\Delta t}$ is specific yield at the end of the time step.

In the adjusted formulation, the initial stored volume of water is subtracted, with the old specific yield S_y^t controlling the portion up to top elevation (i.e. saturated thickness fraction between 0 and 1), and the old specific storage S_s^t controlling any volume above top elevation. The new volume is added in, similarly with the new specific yield $S_y^{t+\Delta t}$ controlling the portion up to top elevation and the new specific storage $S_s^{t+\Delta t}$ controlling any volume above top elevation.

CONVERTIBLE WITH UPSTREAM WEIGHTING (LAYTYP = 4, [LAYTYP = 5](#))

Without storage changes, MODFLOW-USG implements the continuous time-dependent storage term as

$$\Delta S = \frac{V}{\Delta t} [S_s Y_n^{t+\Delta t} (h^{t+\Delta t} - h^t) + S_y (Y_n^{t+\Delta t} - Y_n^t)]$$

Where

S_s is specific storage, S_y is specific yield, V is cell volume, Δt is time step length, Y_n^t is smoothed saturated thickness fraction (as per the smoothing function Y_n detailed in the MODFLOW-NWT documentation) at the beginning of the time step, $Y_n^{t+\Delta t}$ is smoothed saturated thickness fraction at the end of the time step, h^t is head at the beginning of the time step, and $h^{t+\Delta t}$ is head at the end of the time step.

When either or both of specific storage and specific yield are changed during a simulation, TVM adjusts the formulation to

$$\Delta S = \frac{V}{\Delta t} (S_s^{t+\Delta t} Y_n^{t+\Delta t} h^{t+\Delta t} - S_s^t Y_n^t h^t + S_y^{t+\Delta t} Y_n^{t+\Delta t} - S_y^t Y_n^t)$$

Where

S_s^t is specific storage at the beginning of the time step, $S_s^{t+\Delta t}$ is specific storage at the end of the time step, S_y^t is specific yield at the beginning of the time step and $S_y^{t+\Delta t}$ is specific yield at the end of the time step.



Note that the implementation of the LAYTYP 4 storage formulation in MODFLOW-USG involves a Newton expansion of the terms in the above storage expressions; this is not detailed here, but the derivatives terms used in the Newton expansion are also modified appropriately for the adjusted storage formulation. Refer to the MODFLOW-NWT documentation for details on the Newton linearization implementation.

TVM2 Adjustments to Transport Storage Formulation

A mass conserved approach is applied to incorporate changes in porosity into the solute transport equation. These changes must be integrated over at least one time-step, and step function changes to porosity are not permitted. Transient porosity values are supplied in the TVM package as item 8, as detailed in the input instructions below.

The mathematical formulation for storage changes in a cell depends on the LAYTYP flag specified in the LPF package for the cell's layer, and each of these cases is detailed below and broadly classified as confined storage, unconfined storage, and convertible (with both confined and unconfined) storage.

CONFINED

MODFLOW-USG implements the confined transport storage term as

$$\frac{\Delta M}{\Delta t} = \frac{\phi_e V}{\Delta t} (c^{t+\Delta t} - c^t) + \frac{S_s V}{\Delta t} (c^{t+\Delta t} h^{t+\Delta t} - c^t h^t)$$

Where

ΔM is the change in mass, ϕ_e is effective transport porosity, c^t is concentration at the beginning of the time step, and $c^{t+\Delta t}$ is concentration at the end of the time step. Note that the effective porosity and specific storage in the above equation are assumed constant.

For time-varying porosity and specific storage, TVM adjusts the formulation to

$$\frac{\Delta M}{\Delta t} = \frac{V}{\Delta t} (\phi_e^{t+\Delta t} c^{t+\Delta t} - \phi_e^t c^t) + \frac{V}{\Delta t} (S_s^{t+\Delta t} c^{t+\Delta t} h^{t+\Delta t} - S_s^t c^t h^t)$$

Where

ϕ_e^t and $\phi_e^{t+\Delta t}$ are porosity values at the beginning and the end of time step, respectively.

**UNCONFINED**

MODFLOW-USG implements the unconfined transport storage term as

$$\frac{\Delta M}{\Delta t} = \frac{V(\phi_e - S_y)}{\Delta t} (c^{t+\Delta t} - c^t) + \frac{VS_y}{\Delta t} (S_w^{t+\Delta t} c^{t+\Delta t} - S_w^t c^t)$$

Where

ΔM is the change in mass, ϕ_e is effective transport porosity, S_w^t and $S_w^{t+\Delta t}$ are the saturation at the beginning and at the end of the time step, respectively, and c^t and $c^{t+\Delta t}$ are concentration at the beginning and at the end of the time step, respectively. Note that the effective porosity and specific yield in the above equation are assumed constant.

For time-varying porosity and specific yield, TVM adjusts the formulation to

$$\frac{\Delta M}{\Delta t} = \frac{V}{\Delta t} ((\phi_e^{t+\Delta t} - S_y^{t+\Delta t})c^{t+\Delta t} - (\phi_e^t - S_y^t)c^t) + \frac{V}{\Delta t} (S_y^{t+\Delta t} S_w^{t+\Delta t} c^{t+\Delta t} - S_y^t S_w^t c^t)$$

Where

ϕ_e^t and $\phi_e^{t+\Delta t}$ are porosity values at the beginning and the end of time step, respectively, and S_y^t and $S_y^{t+\Delta t}$ are specific yield values at the beginning and the end of time step, respectively.

CONVERTIBLE (BOTH CONFINED AND UNCONFINED STORAGE)

MODFLOW-USG implements the convertible layer with both the confined and unconfined transport storage terms as

$$\frac{\Delta M}{\Delta t} = \frac{V(\phi_e - S_y)}{\Delta t} (c^{t+\Delta t} - c^t) + \frac{VS_y}{\Delta t} (S_w^{t+\Delta t} c^{t+\Delta t} - S_w^t c^t) + \frac{VS_s S_w^{t+\Delta t}}{\Delta t} (c^{t+\Delta t} h^{t+\Delta t} - c^t h^t)$$

Where

ΔM is the change in mass, ϕ_e is effective transport porosity, and c^t and $c^{t+\Delta t}$ are concentration at the beginning and at the end of the time step, respectively. Note that effective porosity, specific yield, and specific storage in the above equation are assumed constant.

For time-varying porosity, specific yield, and specific storage, TVM adjusts the formulation to



$$\frac{\Delta M}{\Delta t} = \frac{V}{\Delta t} \left((\phi_e^{t+\Delta t} - S_y^{t+\Delta t}) c^{t+\Delta t} - (\phi_e^t - S_y^t) c^t \right) + \frac{V}{\Delta t} (S_y^{t+\Delta t} S_w^{t+\Delta t} c^{t+\Delta t} - S_y^t S_w^t c^t) + \frac{V S_w^{t+\Delta t}}{\Delta t} (S_s^{t+\Delta t} c^{t+\Delta t} h^{t+\Delta t} - S_s^t c^t h^t)$$

Where

ϕ_e^t and $\phi_e^{t+\Delta t}$ are porosity values at the beginning and the end of time step, respectively, S_y^t and $S_y^{t+\Delta t}$ are specific yield values at the beginning and the end of time step, respectively, S_s^t and $S_s^{t+\Delta t}$ are specific yield values at the beginning and the end of time step, respectively.

Transient IBOUND Package to Simulate Excavation/Reclamation, or Well Drilling/Plugging

The Transient IBOUND (TIB) Package provides the flexibility to change the IBOUND value of any groundwater or CLN cell at any stress period of the simulation. This is advantageous to activate wellbores only when they are drilled, or inactivate them after they are plugged, or to inactivate parts of domains after excavation activities or re-activate them after reclamation. It also provides a mechanism for turning on and off prescribed head cells. It is noted that the MODFLOW FHB and CHD packages allow for turning on a prescribed head cell at any time but once a cell was given a prescribed head condition it could never change back to a regular active cell.

The TIB package is implemented immediately upon entering a stress period. Therefore, changes made within it may be superseded by prescribed head implementations of the FHB and CHD packages. The TIB package sets the IBOUND value to unity for an activated cell, zero for an inactivated cell, and negative one for a prescribed head cell. *For the first stress period, TIB package overrides IBOUND and ICBUND settings in the BAS and BCT packages, respectively.*

Solute Transport in LAK Package

Vivek Bedekar, S. S. Papadopoulos and Associates, Inc.

Introduction

Flow between groundwater and surface water features such as lakes is an important consideration for integrated flow systems. The lake (LAK) package of MODFLOW integrates flow in and out of lakes with the groundwater system. Solute transport capability was added to the LAK package of MODFLOW-USG to enable solute transport integrated with the groundwater system. The development for solute transport capability in the LAK package was supported by



Arcadis Chile SpA. and Kinross Gold Chile and the capability was implemented into USG-Transport by S. S. Papadopoulos and Associates, Inc.

Formulation

Solute transport in the LAK package of MODFLOW-USG is available in two modes: (1) lakes as boundary condition; and (2) solute transport in lakes. In the first mode when lakes are simulated as a boundary condition to the groundwater system lake concentration is a user-specified value to compute mass entering groundwater from lakes. In the second mode, mass balance for lakes is considered and lake concentrations are calculated.

The formulation and implementation of the second mode, i.e. lake solute transport calculations are based on the MOC3D program (Merritt and Konikow, 2000) and MT3D-USGS (Bedekar et al, 2016). A mass conserved mixing model is utilized to account for mass entering and leaving lakes via lake boundaries, like lake precipitation, runoff, lake withdrawal, and from interaction with the aquifer. The formulation of solute transport in lakes in MODFLOW-USG follows the same assumptions listed in Merritt and Konikow (2000). The primary assumption includes complete and instantaneous mixing of solute within lakes. This assumption may be acceptable for smaller lakes but solute transport across larger lakes will cause mass to be transported instantaneously over a larger distance, albeit at low concentrations. Details of the lake transport formulation are detailed in Merritt and Konikow (2000) and Bedekar et al (2016).

Input for LAK package for solute transport

Input instructions for the solute transport capability with the LAKE package are provided later.

Output from LAK package for solute transport

Mass exchanged between lakes and groundwater is reported as a separate term in the groundwater mass balance summary in the output file. When lake transport is considered, a separate lake-specific mass balance is also written to the output file. Concentration for a lake can be printed using the gage package and specifying a lake as an observation point in the gage package.

Recharge (RCH) Package Modifications

The Recharge Package for MODFLOW-USG was developed by modifying the MODFLOW-2005 RCH Package, Version 7, to be compatible with, and include the flexibility of unstructured grids. Recharge can be limited by a user-defined seepage elevation (SEEPELEV) and can reduce or



turn to discharge if water levels try to go above the seepage elevation. Temporally varying recharge input as a time series is also accommodated outside of the stress-period structure of MODFLOW using the RTS option. In addition, the package and its input have been modified to include solute/heat transport with solutes or heat entering the domain as a third type boundary condition with the inflowing recharge water – this is facilitated via an input option (CONC or CONCENTRATION) that indicates such input is required. If concentration of the recharge water is not input for any solute species, it is assumed to be zero (also for temperature). Input instructions for the RCH package provide further details on how these options may be used.

Evapotranspiration (EVT) Package Modifications

The Evapotranspiration Package for MODFLOW-USG was developed by modifying the MODFLOW-2005 EVT Package, Version 7, to be compatible with, and include the flexibility of unstructured grids. Temporally varying evapotranspiration input as a time series is also accommodated outside of the stress-period structure of MODFLOW using the INIZEVT flag. In addition, the package and its input have been modified to include solute/heat transport. By default, solutes or heat are left behind and only water evaporates. However, the IETFACTOR flag can allow components to leave with the water as a fraction of the water leaving, which can be input for each species individually. No concentration input is required since the EVT package does not allow inflow of water. Input instructions for the EVT package provide further details on how these options may be used.

Segmented Evapotranspiration (ETS) Package Modifications

The Segmented Evapotranspiration Package for MODFLOW-USG was developed by modifying the MODFLOW-2005 ETS Package, Version 7, to be compatible with, and include the flexibility of unstructured grids. Details of the package formulation are provided in Banta (2000). [Banta, E. R., 2000. MODFLOW-2000, The U.S. Geological Survey Modular Ground-Water Model—Documentation of Packages for Simulating Evapotranspiration with a Segmented Function (ETS1) and Drains with Return Flow (DRT1), USGS Open File Report 00-466.].

Temporally varying evapotranspiration input as a time series is also accommodated outside of the stress-period structure of MODFLOW using the INIZETS flag. In addition, the package and its input have been modified to include solute/heat transport. By default, solutes or heat are left behind and only water evaporates. However, the IETSFACTOR flag can allow components to leave with the water as a fraction of the water leaving, which can be input for each species



individually. No concentration input is required since the EVT package does not allow inflow of water. Input instructions for the ETS package provide further details on how these options may be used.

DRT Package with Multiple Return Locations, GHB-RT and Transport

The Drain with Return Flow (DRT) Package for MODFLOW-USG was developed by modifying the MODFLOW-2005 DRT Package, Version 7, to be compatible with, and include the flexibility of unstructured grids. Details of the package formulation are provided in Banta (2000). [Banta, E. R., 2000. MODFLOW-2000, The U.S. Geological Survey Modular Ground-Water Model—Documentation of Packages for Simulating Evapotranspiration with a Segmented Function (ETS1) and Drains with Return Flow (DRT1), USGS Open File Report 00-466.].

A drain cell can supply water to multiple nodes by having many drains on a single cell (each drain could be at a different drain elevation too within the same cell), with return flow occurring to different cells. The water extracted is thus distributed to each of the surface-water cells depending on how much water is extracted from the drain. However, it may be desirable to reapply water from a drain to multiple cells at a uniform recharge rate, for example when the drained water applied to say a spreading ground or an agricultural field that includes several nodes (of possibly different sizes). To accommodate this, a further modification of the DRT package was made to allow for multiple return flow nodes from the same drain.

For solute (or heat) transport simulations, the DRT package makes a connection between the solute concentration (temperature for heat transport) that is re-applied, with the extracted solute concentrations. The DRT package further accommodates some manipulation of the solute concentration (or temperature) before reapplication. Current manipulations include: IDCHNGTYP = 0 for no changes; IDCHNGTYP = 1 for addition (or subtraction) of concentration (or temperature) before reapplication; and IDCHNGTYP = 2 for addition (or removal) of solute mass (or heat) to the reappplied water; IDCHNGTYP = 3 for changing the extracted concentration to a fixed value of concentration (or temperature); IDCHNGTYP = 4 for changing the extracted mass to a fixed value of mass (or heat) before reapplication; IDCHNGTYP = 5 for where concentration is **reduced** to a prescribed value if above that value, IDCHNGTYP = 6 for where concentration is **reduced** to a fraction of the incoming concentration value, and IDCHNGTYP = 7 for where mass is **reduced** to a fraction of the incoming mass value at the DRT drain. The auxiliary variables C01, C02 etc. are used to provide the magnitude of change for each solute



component (01, 02, etc.). The auxiliary variable TMPR is used to identify the heat equation boundary condition.

The DRT package has been further extended to provide the ability to simulate conditions specific to closed-loop borehole heat exchanger (BHE) U-Tubes. Specifically, the U-tube of a BHE is required to simulate steady flow of water in and out of a closed-circuit U-tube system. The CLN package is well suited to represent the BHE U-tube. Unfortunately, boundary conditions of flow in and flow out at the ends of a closed-circuit U-tube cannot be applied since a reference head is required for a steady-state solution, at least at one end of the U-Tube. To accommodate this, an option is provided which allows for extraction of water under a GHB condition, with a specific quantity (the flow rate within the U-tube) reapplied at the return location. This case is called the GHB-RT condition and is identified by providing a negative conductance factor in the DRT input file. If the conductance factor “Condfact” is negative it is interpreted as the conductance of the GHB boundary and the input of “elevation” is the GHB head value at the extraction location. On the return flow side for GHB-RT, the term “Rfprop” defines the volumetric flow rate of the return flow water for this case. Thus, a BHE U-tube can be represented with a prescribed head condition at its outflow end (using a high value of “Condfact”) and a prescribed flux at the inflow end. Furthermore, the temperature of the inflow water is related to temperature of outflow from the BHE with manipulations that can add or remove heat, or adjust the temperature before inflow, if needed.

Sink with Return Flow (QRT) Package

The Sink with Return Flow (QRT) Package for MODFLOW-USG was developed by combining the WEL Package with the return flow concept of the DRT Package. Thus, the QRT Package allows for return of some or all of the pumped water to other portions of the domain. The QRT package allows for return to multiple cells and not just one cell. For that case, the return flow volumes are distributed as per the horizontal grid-block areas of the cells to which flow is returned (i.e., it is areally uniformly distributed to simulate possible land irrigation conditions). **It is noted that the extracted Q value is positive in the QRT dataset as it relates strictly to extraction of water and reapplying of the extracted water, while the opposite may not be realistic.**

The significance of QRT is more pronounced when transport simulations are conducted. Specifically, the solute concentration (or temperature for heat transport) that is re-applied depends on the extracted solute concentrations. The QRT package provides that transport link



and further accommodates some manipulation of the solute concentration (or temperature) before reapplication. Current manipulations include: $IQCHNGTYP = 0$ for no changes; $IQCHNGTYP = 1$ for addition of concentration (or temperature) before reapplication; and $IQCHNGTYP = 2$ for addition of solute mass (or heat); $IQCHNGTYP = 3$ for changing the extracted concentration to a fixed value of concentration (or temperature); $IQCHNGTYP = 4$ for changing the extracted mass to a fixed value of mass (or heat) before reapplication; and $IQCHNGTYP = 5$ for where concentration is **reduced** to a prescribed value if above that value. The auxiliary variables C01, C02 etc are used to provide the magnitude of change for each solute component (01, 02, etc). The auxiliary variable TMPR (code for temperature) is used to identify the heat equation boundary condition.

The QRT package may be optionally extended to use a hydrograph type input (similar to the FHB package) for the extraction rate, instead of using the stress periods to vary the flow. This is done by use of the **TRANSIENTQ** optional keyword. This is followed by NBDQTIM, the number of time interpolation values to be used in the Q hydrograph. A negative value for NBDQTIM allows for the interpretation to be stepped instead of interpolated. Note that the Q values entered during stress periods is ignored when TRANSIENTQ is used. Also, sorting and seeking routines have not been coded and therefore, the maximum number of QRT sinks (MXAQRT) and the number of sinks listed within each stress period of QRT extraction cells should be the same as the number used in each stress period, and also the list of transient hydrographs should be in the same order as the input provided for each QRT sink cell (however, the repeat index on ITMP may be used at each stress period so all cells need not be listed). Note that if a sink does not exist at any stress period, it should still be entered in the stress-period list with a Q value of zero so that the order of sink cells in the transient hydrograph list and that in the stress period list is the same. The cell should be included in every stress period list even if it is inactive for that stress period, if the TRANSIENTQ option is used. Finally note that the the QRT package does an integration of the flux under the curve for each time-step (similar to the FHB package). Therefore, the flux is not the value of the interpolated FHB flux at the total time value of the simulation, but an integration of the flux during a time step.

Flow and Head Boundary (FHB) Package Modification

The original FHB package of Leak and Lilly (1997) has been modified to work with unstructured grids. The FHB package has been further extended to accommodate a staircase representation of the time versus flux (or head) hydrographs.



Note that the FHB package does an integration of the flux under the curve for each time-step. Therefore, the flux is not the value of the interpolated FHB flux at the total time value of the simulation, but an integration of the flux for each time step. An optional keyword STAIRCASE activates this condition. If adaptive time stepping is used with the staircase option, then the time stepping will adjust to exactly accommodate the time values in the table. That way, the changes are always consistent, and fluxes are exactly integrated for the simulation.

Discretization file extension

The discretization dataset includes a flag SS or TR to indicate if a stress period is for steady-state or transient flow conditions. This flag has been extended to a flag TRTOSS, to include capability of solving the steady-state simulation by running it in a transient mode for a long time period. This may be needed for a model that may otherwise have difficulty converging for a one-step steady-state solution.

Any solution can be made to reach steady-state conditions by running the model for a very long time. However, once the storage term becomes small, it has essentially reached steady-state conditions and running it for longer only wastes computational effort. The flag TRTOSS optimizes on this computational effort by finishing the stress period when the rate of change of the storage term is below a user defined fraction of the total flow rates in the system. This fraction immediately follows the keyword TRTOSS in the input dataset for each stress period that is treated as transient to steady-state.

A TRTOSS stress period should be set up for a long duration (1000s of years) with optimal time stepping using adaptive time-stepping procedures discussed in the Output Control (OC) Package routines and input instructions. That way, the simulation progresses as rapidly as possible with a long enough duration to ensure it reaches steady state conditions. At every time step of the simulation, MODFLOW-USG computes the ratio of the net rate of storage terms to the net total water budget rate (storage plus boundary flow rates) terms. When this ratio is smaller than the user defined fraction, the simulation takes one more time step so that appropriate output can be performed for heads, drawdowns, and fluxes and ends the stress period.

When TRTOSS is used for a stress period, heads and fluxes will be output for the interim time steps to reach steady-state, unless the variables in the OC file are appropriately set to only write at the end of a stress period. The default of printing or saving output (heads, drawdowns,



fluxes) at the end of a stress period only, is set by using NSTEPS = 0 in the OC file (NSTEPS is otherwise the number of time-steps after which output is written).

Sparse Matrix Solver (SMS) Package

Rodrigo Herrera, SGM Co., Chile.

The original SMS Package of MODFLOW-USG has been considerably expanded in USG-T. Thanks to Rodrigo Herrera for implementing and testing several of these modifications. The modifications are noted below.

The xMD solver returned NaNs under certain conditions. Rodrigo Herrera has implemented guardrails to abort a simulation if the matrix inversion causes the NaN condition.

The linear solver from MODFLOW 6 (IMS solver) had built-in guardrails against NaN errors. Rodrigo Herrera has included it in the SMS Package in the current version of USG-T as an additional optional linear solver, activated with solver parameter LINMETH = 5. Details for the linear solver scheme are available in the MODFLOW 6 documentation (Hughes and others, 2017). Depending on the complexity of the problem, the PCGU, xMD, or IMS solver may perform better.

The SMS Package has further been modified to include only active nodes in the linear solution routines using the SOLVEACTIVE option. This enhancement speeds up computations when there are many inactive nodes (with IBOUND=0) within a model domain.

The SMS Package has also been modified to include the bottom dampening option available in MODFLOW-NWT. This feature was helpful in obtaining convergence for several highly nonlinear problems and was implemented into Version 1.4 of MODFLOW-USG as a default setting. The current default in USG-T is to not do this dampening. This feature is activated here using the DAMPBOT option.

Another possible solver enhancement may be obtained by shifting the solution vector via the option SHIFT. This option translates the right-hand side vector by a factor times the previous solution value (factor is currently set at 0.01) for calculation by the linear solver. The translation is applied in reverse to the solution vector after solution. This has helped sometimes with convergence, especially when the absolute values of the solution (head or concentration) are large.



A truncated Newton option has also been included with the SMS package with all of the linear solvers (PCGU, xMD, and IMS). The formulation was developed by Dr. Damian Merrick of HydroAlgorithmics () who also implemented this feature in the code. The option is set by using the option TRUNCATEDNEWTON on the first line of inputs. This keyword uses a default cutoff limit residual fraction of 0.9. Alternatively, the keyword TRUNCATEDNEWTONCUTOFF allows the cutoff residual fraction to be input right after the keyword. This is the upper limit residual fraction that is used for the residual cutoff (so the default of 0.9 means that the residual is limited to 90% of the previous residual). Values as low as 0.00001 have been noted to help reduce inner iterations without affecting the overall solution.

An additional enhancement to the SMS package is to include flexibility to change solver settings for the nonlinear schemes and the linear solver closure tolerance on a stress-by-stress basis. This can be significant in long simulations where certain stress periods always behave differently than the rest, and can be specifically tuned accordingly. Previous USGT versions only allowed certain solver parameters to be reset at other stress periods after the simulation starts, in the OC file with adaptive time stepping routines. This has been retained for backward compatibility, however, it is suggested to use the stress-period-varying solver parameters from the SMS package to keep these contained.

An option has also been included to output the nonlinear solver details to a separate output file. This is invoked by using the option SOLVEROUTPUTFILE detailed in the I/O document.

The Truncated Newton Method

Dr. Damian Merrick, <http://www.hydroalgorithmics.com>.

The Truncated Newton method is an optional adjustment to the numerical solution process that may promote faster convergence. It can be enabled by using the option TRUNCATEDNEWTON on the first line of the SMS file.

The solution of the linearized system, when allowed to converge completely, often progresses too far and must be pulled back via backtracking to find the nonlinear solution. Early in the nonlinear solution, it is often better to take smaller steps and then re-linearize to avoid overshooting the solution of the nonlinear system. In practice, a sufficiently large step towards the solution can often be achieved with relatively few iterations of the linear solver - even with a single iteration of an appropriately preconditioned Krylov subspace method.



As such, it can be beneficial to limit the number of linear solver iterations at each outer (nonlinear) iteration to avoid wasted effort. A simple way to do this is to manually set a maximum number of inner iterations to a small number, e.g. 2 or 3. This can have the desired effect, but can result in more matrix formulations than necessary in near-linear systems, and when the solver is close to the solution of more nonlinear systems.

A more comprehensive approach can be found in the truncated Newton method. This works similarly, but instead of setting a fixed maximum number of linear iterations, an automated residual-based control mechanism is used. Each nonlinear iteration after the linear solver runs, the system is re-linearized and an updated (nonlinear) residual is calculated. While the linear solution produces a result with a high nonlinear residual, the linear solver is kept from progressing too far by stopping at a relatively high linear residual cutoff – this is typical in early iterations when solving a highly nonlinear system. When the solution is close enough that the linear solver is effectively progressing towards the nonlinear solution (where the nonlinear residual is decreased), the linear solver is allowed to progress further with a lower linear residual cutoff.

To implement this, we adopt equation 2.6 (“Choice 2”) from Eisenstat and Walker (1994) to determine an appropriate residual cutoff fraction η_k for the linear solver at each nonlinear iteration k :

$$\eta_k = \gamma \left(\frac{\|F(x_k)\|}{\|F(x_{k-1})\|} \right)^\alpha$$

with $\gamma = 0.9$ and $\alpha = (1 + \sqrt{5})/2$, where $\|F(x_k)\|$ is the nonlinear residual (sum of squares) at iteration k . Eisenstat and Walker’s “Choice 2” safeguard is applied to ensure that η_k does not become too small: modify η_k by $\eta_k \leftarrow \max\{\eta_k, \gamma\eta_{k-1}^\alpha\}$ whenever $\gamma\eta_{k-1}^\alpha > 0.1$. As an upper bound, η_k is capped to a maximum value of 0.9.

Once the cutoff fraction η_k is computed, it is multiplied by the residual to obtain a sum-of-squares residual limit for the linear solver, $r_{cutoff} = \eta_k \|F(x_k)\|$. At the first iteration, $k = 1$, the linear solver is allowed to solve completely as per the normal convergence criteria, without a residual cutoff.

With this cutoff in place, the nonlinear solution is allowed to run until the normal head closure criterion is met. Then, as a final step to ensure convergence to the same tolerance, the cutoff is disabled, and the solver continues until fully converged as per the normal solution process.



Eisenstat, S.C. and Walker, H.F., 1994, Choosing the forcing terms in an inexact Newton method. Technical report CRPC-TR94463, Center for Research on Parallel Computation, Rice University, Houston, TX.

River (RIV) Package Modifications

The River Package for MODFLOW-USG was developed by modifying the MODFLOW-2005 RIV Package, Version 7, to be compatible with, and include the flexibility of unstructured grids. The RIV package has further been modified to include solute and heat transport capability.

An option is also provided to compute the net conductance of a riverbed and the cell on which it is connected. The default operation of the RIV package in MODFLOW is such that the riverbed conductance, which is input, is used to transmit flow between the river boundary and the groundwater cell. However, if the riverbed elevation is considerably above the elevation of the groundwater cell center, the groundwater cell vertical conductance can also have an impact in transmitting or restricting flow between the boundary and the cell. This resistance to flow is neglected in the traditional MODFLOW river boundary package, or is assumed to be incorporated into the river bed conductance value input here. Thus, if a RIV bed was located on a silty soil versus a sandy soil, the resistance to flow would be the same in either case, if the input river-bed conductance were the same. The option “MERGE_BED_K1” allows for internal computation of this leakance by the code, in combination with a river-bed leakance that may only be a function of the river bed type (e.g., grassy, concrete lined, gravelly, or sandy channel-beds). A harmonic mean of the leakance terms is used to merge the input value of the bed leakance with the vertical leakance of the associated groundwater cell to give the combined leakance to be used for computations, when the river bed elevation is higher than the cell center elevation.

The RIV package was also modified to accept binary input for the river data which can be used for any USG-T simulation. An OPTION called “BINARY ib” on the first line of input will allow for the riverbed elevation, conductance, and stage to be read at each stress period in binary formats, at each river node, from a binary file that is attached to Fortran unit “ib” as opened in the NAMEFILE of the simulation. *This binary data option was provided specifically to speed-up data transfer during IHM simulations where the surface-water model of IHM provides updated stage data to USG-T at every integration time step (equal the MODFLOW stress period). See **IHM integration with USG-T section and input instructions for details.***



Specified Gradient Boundary (SGB) Package

The specified gradient boundary (SGB) package allows for a boundary flux to be a function of the prescribed gradient at the boundary. This boundary type is useful for prescribing the boundary gradient and letting the code compute the flux, considering the hydraulic conductivity of the medium. It can therefore be applied at lateral boundaries with known (or interpolated) water level gradients at the boundary. This can be useful to provide a flux condition at a boundary when the hydraulic conductivity value near the boundary is not known and is in the process of being calibrated. A unit gradient boundary condition is also appropriate to apply at the bottom of the domain within the unsaturated zone when solving Richards Equation (McCord, 1991. "Application of Second-Type Boundaries in Unsaturated Flow Modeling, Water Resources Research, Vol. 27, No 12, 3257-3260, December 1991.).

The specified gradient condition at a boundary is written as

$$Q = A_n q = A_n K_n k_r \frac{\partial h}{\partial n} = A_n K_n k_r G = A_n \mathfrak{R}_n K_x k_r G \quad (1)$$

Where Q is the total boundary flow, A_n is the area of the boundary face through which flow occurs, q is the unit boundary flux, K_n is the saturated conductivity in the direction normal to the boundary face n , k_r is the relative permeability term which is a function of the head h , n is the direction normal to the boundary plane along which the gradient is computed, G is the gradient that is prescribed at the boundary ($G = \partial h / \partial n$), $\mathfrak{R}_n$ is the anisotropy ratio between the normal direction n and the x-direction ($\mathfrak{R}_n = K_n / K_x$), and K_x is the saturated hydraulic conductivity in the x-direction. Note that a positive value of the gradient represents inflow while a negative value for the gradient represents outflow from the boundary cell. For a confined condition, the relative permeability term is unity and therefore the flux is a constant ($Q = A_n K_n G$) which is applied to the right-hand side vector after the flow terms are assembled. For unconfined conditions or solution to Richards Equation, the flux is nonlinear and is implemented using Newton Raphson linearization. Thus,

$$\frac{\partial Q}{\partial h} \Delta h = -Q \quad (2)$$

The specified gradient condition is implemented in a generic manner without taking into consideration the direction of the normal to the boundary. Therefore, MODFLOW-USG expects input of the flow area times the gradient times the anisotropy ($\mathfrak{R}_n A_n G$) for the normal direction



to flow not being along the x-direction, and for anisotropy between the normal direction and the x-direction. The flux is then internally computed by multiplying this input by $K_x k_r$.

WEL Package Modification

The WEL package of MODFLOW-USG includes an “AUTOFLOWREDUCE” option which reduces the extraction of a pumping well to zero, if the water level in the well drops to below the bottom of the cell. This keeps the solution in the physical range so a well cannot pump from a dry cell (or only pump as much as is flowing to that cell under the bottom-hole elevation). The AUTOFLOWREDUCE function has further been enhanced by allowing a user-defined bottom-hole condition, instead of just using the bottom of the Groundwater Flow (GWF) or Connected Linear Network (CLN) cell. This flexibility allows the pumping elevation to be at any location within the cell and not just at its bottom. Also, this location can vary with stress periods, as does pumping at any well.

The modification for flexible bottom-hole elevation is invoked by using an additional keyword “WELLBOT” on the first data line. If this option is set, the well bottom-hole elevation for each well is input right after the well pumping rate (and before any auxiliary variables). If this keyword is not provided, then the bottom of the GWF or CLN cell on which the WEL is located, is used as default. Finally, an internal check is also performed on the user-input WBOT values to ensure that they lie within the cell. If not, the cell bottom is set as the WBOT elevation and a warning is sent to the output listing file. The pumping elevation can be above the top of a cell. In that case, pumping is limited when water levels fall to that pump elevation and will fall below that elevation only after pumping becomes zero if other conditions are also reducing the water levels in a model.



MODFLOW USG-Transport

Sorab Panday

IHM Integration with USG-T

Sorab Panday, GSI Environmental, Herndon, VA.

Jeff S Geurink, and Jeff L. Shelby, Tampa Bay Water, Clearwater, FL.

Introduction

IHM is the Integrated Hydrologic Model (Ross and Geurink, 2018, Geurink et al., 2018) that dynamically combines HSPF (Bicknell et al., 2005) to simulate surface water hydrology (uplands with different land use / land cover features and soils and water bodies including lakes, reservoirs, wetlands, swamps, wet prairies, bogs, fens, etc.), with MODFLOW to simulate groundwater flow. USG-T includes this IHM integration capability and can therefore be used as the groundwater model for IHM.

Development of the IHM integration capability in USG-T was supported in part by funding from Tampa Bay Water, Clearwater, Florida.

Capabilities and Functionality

IHM is capable of simulating all significant hydrologic processes including precipitation, potential evapotranspiration (PET), abstraction storages (interception and depression), evapotranspiration (ET) from impervious uplands, pervious uplands (from abstractions, the vadose zone, and ground-water) and water bodies, infiltration, overland flow, surface runoff, percolation, infiltration redistribution, interflow, recharge, irrigation flux applied to uplands, vadose zone storage, uplands flow routing to water bodies, flow routing between water bodies, flow exchange between water bodies and the ground water system, surface-water diversions to/from water bodies, lateral groundwater flow, inter-aquifer leakage, ground-water storage, spring discharge, and well pumping.

IHM couples the HSPF and MODFLOW models in a sequentially coupled manner with user-defined time steps independently for the HSPF and MODFLOW portions of the code, and for when the two models pass information between each other. IHM passes information between the surface and subsurface domains at each integration time-step, which is a stress period of MODFLOW where the stresses (boundary heads or fluxes) may change in the simulation.

HSPF simulations are spatially discretized using a semi-distributed framework of HRUs (Hydrologic Response Units), while MODFLOW simulations use a gridded format for spatial discretization (though with USG-T one can conceivably create grids shaped to match HRUs). Thus, for passing information back and forth between the surface-water and groundwater processes, IHM maps the heads, parameters, or fluxes between the two models spatially using IHM land and waterbody fragments, as well as accumulates water volumes to pass between surface and subsurface for each integration time-step. IHM fragments represent the intersection of HSPF land segments (upland simulation units) with MODFLOW grid cells and HSPF reaches (waterbody simulation units) with MODFLOW river cells.

Information that is passed from the surface-water model to the groundwater model at each integration time step includes the recharge, remaining potential evapotranspiration (maximum ET rate), and river stages. Thus, information in the RCH, EVT, and WEL packages is updated based on HSPF simulation results and passed from IHM to the MODFLOW (USG-T) model at every stress period. All well pumping time series are stored in an IHM input database which IHM uses to dynamically update the WEL package at every stress period.



MODFLOW USG-Transport

Sorab Panday

Vadose zone processes and capillarity can also influence the pore storage in unconfined portions of the aquifer, and therefore, this information is also computed by IHM depending on HSPF-simulated soil moisture, MODFLOW-simulated depth to water, and capillary effects, and passed to USG-T at each stress period. Since this process affects the pore storage term, the specific yield (SF2 in MODFLOW terminology) also changes at every stress period. The IHM integration incorporated into USG-T therefore allows for the SF2 parameter also to be modified at each stress period. The Theory Manual (Section 3.8.2) of IHM provides more details on temporal changes to SF2 and a water balance correction term related to temporally-variable SF2.

Input requirements for IHM simulations include time series inputs (climatic, anthropogenic stresses, and ground-water boundary conditions), physical characteristics of topography, hydrologic characteristics of land use (pervious and impervious uplands, water bodies) soils, flow routing schematic (land to reach, reach to reach), flow rating functions and discharge invert conditions for reaches, open-water storage parameters for water bodies, functions which capture temporally-varying phenological conditions of vegetation for interception storage and vegetative ET, aquifer hydraulic conductivity and storage properties, bed leakance representing the hydraulic connection between water bodies and the ground-water system, spring conductance, and initial conditions. Time series inputs include precipitation, potential ET, irrigation flux, surface-water diversions, well pumping, and MODFLOW GHB heads (if required as boundary conditions). The IHM User Guide (Geurink et al., 2018) provides details on input requirements for IHM.

Output from an IHM simulation includes detailed water balance information, time series and summaries for all ET components, vadose soil moisture, streamflow, water depths in reaches, flux exchange between pervious uplands and the ground-water system, aquifer water levels, inter-aquifer water level differences, spring discharge, flow exchange between water bodies and the ground-water system, lateral ground-water flow, inter-aquifer leakage, occurrence of simulated ground-water above land, dry grid cells, boundary flows and storage of the ground-water system. The water balance also provides insights about lateral and vertical components of the simulated system. Each water balance includes all applicable flow, flux, and storage terms of hydrologic processes of the surface-water and ground-water systems and of terms IHM tracks related to flow, flux and storage transfers between the component models during each simulation. The IHM User Guide (Geurink et al., 2018) provides details on IHM simulation outputs.

When an IHM simulation is started, IHM reads input from several tables that exist in the IHM input database and launches HSPF and MODFLOW which read their respective standard input files (but do not execute any simulation) to provide IHM the information required for surface and groundwater conditions. Using initial aquifer heads to define initial depth-to-water table (DWT) and database tables to define initial soil and waterbody storage, and initial baseflow to waterbodies for HSPF, IHM prepares updates to several HSPF parameters that are a function of DWT. Then IHM signals HSPF to begin simulation for one integration step. HSPF pauses at the end of each integration timestep and IHM reads HSPF-simulated outputs to update inputs to several MODFLOW (USG-T) packages including recharge (RCH), maximum groundwater ET rate (EVT), stage for each river cell (RIV), and specific yield (BCF/LPF). Also, IHM updates well pumping rates (WEL) using tables in the IHM input database. Note that HSPF includes an algorithm to dynamically define irrigation demand, however, this has not been implemented into the IHM framework. There are several different algorithms to estimate irrigation demand suitable for different climatic conditions and users



may apply their model of choice to populate irrigation water source tables in the IHM input database for well pumping and/or diversions.

After IHM has completed updates for the groundwater package files, IHM signals MODFLOW to begin simulation for one integration timestep. MODFLOW pauses at the end of each integration timestep and IHM reads the related groundwater-simulated outputs to update parameter inputs to HSPF for soil storage, vadose ET, infiltration capacity rate, waterbody ET, and baseflow rate (loss/gain) to waterbodies. Since HSPF and MODFLOW run on an independent time-step basis, the IHM integrates the related surface-water and groundwater outputs over the IHM integration time step to provide volume-conserving mass transfers and temporally-weighted parameter updates for both HSPF and MODFLOW.

Thus, one IHM integration step includes the following tasks:

- Parameter updates for HSPF based on initial conditions or MODFLOW-simulated output,
- HSPF simulation for uplands and water bodies,
- Parameter updates for MODFLOW based on HSPF-simulated output, and
- MODFLOW simulation for the groundwater system.

IHM may be used in various climatic and hydrogeologic conditions. However, its application is most beneficial for regions with shallow water tables where surface and subsurface flow processes are closely related and influence each other. Within the water table feedback zone (i.e., groundwater ET extinction depth), the strength of IHM dynamic feedback increases between surface and subsurface models as distance between the water table and land surface decreases. Thus, dynamic feedback may not be important for deep water table conditions where a loosely coupled model or individual surface or subsurface models may suffice. Due to the full integration of surface and subsurface domains, IHM simulations can test impacts of individual stresses or parameters on the entire hydrologic system. For instance, changes to flows and heads for a specific stress type or land use change can be isolated by IHM within a scenario that includes changes to only that stress type. Management and mitigation alternatives can be explored using an understanding of the contribution of each stress type to the total hydrologic behavior. For an IHM-calibrated model, scenario changes to input stresses (e.g., well pumping) used for calibration induce dynamic feedback responses throughout the simulated system, causing changes to infiltration, surface runoff, baseflow, streamflow, total ET, recharge to saturated groundwater, ET from saturated groundwater, heads, etc. The resulting changes to flows and heads occur as a solution to the flow dynamics.

IHM produces IHM-specific outputs and standard time series outputs from HSPF and MODFLOW, which are formatted for IHM postprocessing (Ross and Geurink, 2018, Geurink et al., 2018). IHM-specific outputs include time series for: (a) dynamic parameter changes for the 11 IHM integration paths including temporally-variable specific yield, (b) water budget terms which are required for balance closure within the integrated model control volume, and (c) flow exchange between water bodies and groundwater for HSPF reaches representing more details than are available from HSPF.

Four IHM postprocessing utilities are available to summarize IHM simulated results in the following ways (Geurink et al., 2018): (a) SUMMARY produces output at observation locations for six hydrologic target groups including evapotranspiration, streamflow, springflow, aquifer



MODFLOW USG-Transport

Sorab Panday

heads, depth-to-water table, and surface water stage; (b) BINARYREADER produces output for groundwater heads and flow, IHM integration flow linked to MODFLOW, and IHM temporally-variable specific yield; (c) BALANCE produces water balance output as fluxes and volumes for multiple control volumes, allowing balance results to be reviewed for the entire integrated model domain or for isolated parts such as only HSPF or MODFLOW domains. Balance results are also available for spatial subregions of the entire simulated domain. (d) HOTSTART produces a revised set of model input files containing initial conditions representing simulated results at a specific time within a previously completed simulation. A hotstart model is used to restart a model at a different start date than the original start date which produces results that are not discernible from the model using the original start date. IHM postprocessing utilities produce csv files, and graphic plots if a specific third-party software license is available to the user.

Testing and History

A calibrated application of the IHM has been used by Tampa Bay Water and the Southwest Florida Water Management District (SWFWMD) for 20 years as a decision-support tool for water resource management, planning, operations, and regulation. As a regional water-supply utility, Tampa Bay Water has used the model for spatio-temporal optimization of groundwater pumping to maintain wetland health, for special studies to demonstrate compliance with water pumping permits and Minimum Flow and Level (MFL) requirements, to evaluate sustainability of potential future water supply sources, and to investigate the impact of landuse change and climate change on sustainability of existing water supply sources. As a regional water resources regulatory and planning agency, the SWFWMD has used the model to support development of and evaluate compliance with MFLs for rivers, springs, and lakes and to evaluate proposed groundwater pumping for compliance with permitting criteria. The IHM code development has been thoroughly documented and the code has passed several certification tests and an external peer review which ensure that IHM works as intended. IHM developments in USG-T also passed these certification tests (GSI, 2023a). A Design and Implementation Plan was also developed to ensure that the IHM modules were developed and coded to function as intended. (GSI, 2023b).

Integrated hydrologic modeling using an HSPF-MODFLOW coupled approach has been ongoing and improving for 35 years, since 1988. The IHM is the third generation of HSPF-MODFLOW coupling. The FIPR Hydrologic Model (FHM) represented the first coupling of HSPF and MODFLOW which was developed by the University of South Florida and two consulting firms, BCI and SDI. The Integrated Surface Ground Water (ISGW) model was the second generation that coupled HSPF and MODFLOW which was adapted from the FHM and modified by SDI for larger scale model domains. Starting in the mid-1990s, the wholesale, regional public water supply utility Tampa Bay Water began using the ISGW model for hydrologic investigations of wellfield pumping and surface-water withdrawals. In the late 1990s to early 2000s, external peer reviews of the ISGW model resulted in recommendations for significant modifications to the ISGW model to meet emerging objectives of Tampa Bay Water and the Southwest Florida Water Management District (SWFWMD), the regional water resources regulator in west-central Florida. Instead of modifying the ISGW, Tampa Bay Water and the SWFWMD partnered and collaborated in the early 2000s to create the IHM to meet the emerging objectives of each agency. The original development team for the IHM included two consulting firms, INTERA and Aqua Terra (now Respec), and the University of South Florida.



MODFLOW USG-Transport

Sorab Panday

Since the original IHM development, the agencies have continued to partner and collaborate to improve the IHM and to develop a calibrated model using the IHM which covers 4,000 square miles of west-central Florida. The calibrated model (INTB) was externally peer reviewed in 2013. The IHM code was peer reviewed in 2017. Replacement of MODFLOW-96 with a later version of MODFLOW was a recommendation of the IHM code peer review. As a standard, USG-T will replace MODFLOW-96 but MF-USG has also been modified to be compatible with the IHM.

The IHM is a generalized platform to integrate the surface-water and groundwater models HSPF and MODFLOW and can be used in different settings where integrated modeling is required to manage the water resources. Completion of successful peer reviews and the expectation that others will find the IHM useful for decision support of water resources issues have prompted the agencies to offer the IHM for public use. More details about the history of the IHM are provided in Section 1 of the IHM Theory Manual.

IHM Availability, Details, Training

The USG-T document here only provides input instructions for the USG-T portion of an IHM simulation as pertaining to the groundwater model of an IHM simulation. Please refer to Ross and Geurink, 2018 and Geurink et al., 2018 for understanding details of IHM and its input and output details. Also, regular IHM training sessions are conducted by Tampa Bay Water. Details on obtaining IHM and associated training are available at <https://www.integratedhydrologicmodel.org/>.

References

- Bicknell, B.R., Imhoff, J.C., Kittle, Jr., J.L., Jobes, T.H., and Donigan, Jr., A.S. (2005), *Hydrological simulation program – FORTRAN: HSPF user's manual for Release 12.2*. U.S. Environmental Protection Agency, Athens, GA.
- GSI Environmental, 2023a, Certification of USG-Transport with IHM Integration Module. Prepared for Tampa Bay Water.
- GSI Environmental, 2023b, Design and Implementation Plan for USG-Transport with IHM Integration Module. Prepared for Tampa Bay Water.
- Geurink, J.S., Shelby, J.L., Murch, R., and Tara, P., 2018, Integrated Hydrologic Model Version 3 User Guide. Prepared for Tampa Bay Water and the Southwest Florida Water Management District, Brooksville, FL.
- Ross, M.A. and Geurink J.S., 2018, Integrated Hydrologic Model Version 3 Theory and Implementation. Prepared for Tampa Bay Water, Clearwater, FL and the Southwest Florida Water Management District, Brooksville, FL.



Output Control Package for Simulating Flow and Transport: Input Instructions

Introduction to Output Control Options

MODFLOW-USG provides adaptive time stepping, output control algorithms, and convenient features that enhance robustness and efficiency of highly nonlinear groundwater flow simulations. These options are discussed below.

Adaptive time stepping

Adaptive time stepping is based on the concept that solution to the problem should be easy when the time-step size is small, and vice versa. This is because smaller time-step sizes result in a dominant storage term due to dividing by the smaller Δt term. Hence, even for a very rapidly evolving system, the state of flow within the domain should not be much different from the previous state, for a sufficiently small time-step size and that would assist with convergence. The algorithm used in MODFLOW-USG has been adapted from the Hydrus code of Simunek et al, (2013). [Simunek J., M. Sejna, H. Saito, M. Sakai, and M. TH. van Genuchten, 2013. *The HYDRUS-1D Software Package for Simulating the One-Dimensional Movement of Water, Heat, and Multiple Solutes in Variably-Saturated Media Version 4.16*, March, 2013.]

The adaptive time-stepping algorithm of MODFLOW-USG is tied to the number of nonlinear iterations that a time step takes to converge to the solution. If the problem was solved within one third the total number of nonlinear iterations, then the solution is assumed to be easy and the time-step size is increased by a user selected adjustment factor, for the next time. If the problem was solved within two third (but greater than one third) the total number of nonlinear iterations, then the solution is considered optimal, and the time-step size is kept the same for the next time. If the problem required more than two third the total number of nonlinear iterations, then the solution is considered hard and the time-step size is decreased by the user selected adjustment factor, for the next time, to try and prevent non-convergence. If however, the system fails to converge for the maximum prescribed number of nonlinear iterations, the time-step size is reduced by a time-step cutting factor and solution is reattempted for this reduced time-step size.

The adaptive time-stepping algorithm further adjusts time-step sizes to evaluate the solution at various target time values. Target times may result from user defined output times in the vector TIMOT (when the number of print time steps NPTIMES is greater than zero) or at time values when there is a step change in stresses (for example using the step function in the FHB or QRT package or using the time series inputs for RCH or EVT packages) or at a change in the stress



periods. Note that if the initial time-step of any stress period is larger than the next print time value in TIMOT, the code will print a warning and stop. The user should then use a DELTAT (initial time-step value) for that stress period, smaller than or equal to the next time value in TIMOT or use a larger value for the first output time in TIMOT, than DELTAT.

The adaptive time-stepping algorithm is further subject to constraints on the time step size. A user defined maximum time-step size value is not exceeded. Furthermore, if the adaptive time step size is computed to be smaller than a user defined minimum time-step size then the simulation is aborted. The minimum time-step size is further used in adapting the time-step size of a simulation to achieve target times that may not be smaller than this minimum time-step size.

Output control flexibility

Output control flexibility is included with adaptive time stepping. Output can be requested at every nth time step, or at user defined time values. The former option is useful when beginning a project and evaluating how the simulation is behaving. The latter option is useful for production simulations of a project when output values are desired at particular times.

With MODFLOW's time stepping procedures, the print flags are set at every time step to determine what is to be output (heads and drawdowns in the ASCII listing and binary output files, cell-by-cell flow terms and mass balance components). These print flags are set at every stress period when adaptive time stepping is used, with the relevant output provided at every nth time step, or at the user defined time values. If solute transport is simulated, the output control also applies in a similar manner to the concentration listing and binary output as well as the cell-by-cell mass flux.

Adaptive solver settings through stress periods

Use of adaptive time stepping further allows adaptation of some of the solver parameters between stress periods. Specifically, the closure tolerance for nonlinear convergence (HCLOSE), the maximum outer number of iterations (MXITER) and the backtracking tolerance factor (BTOL) may be allowed to vary between iterations for better numerical control. For instance, it may be more beneficial for steady-state stress periods to use larger number of iterations (MXITER=200) with a tight backtracking tolerance factor (BTOL=1.1) to force the solution to behave. On the other hand, a smaller number of iterations (MXITER=15) and large backtracking factor (BTOL=1.0E6) may be more suitable for transient stress periods with small changes from previous conditions. ***If MXITER is varied in the OC package, it should be used***



with caution. Specifically, the value set in the SMS package should be larger than any of the values set here in the OC package. This is because the iteration error arrays are dimensioned using MXITER in the SMS package and that dimension should not be exceeded. MXITER can be set at the first iteration in the OC package to the user's requirements, if different from the value in the SMS package. Note that these solver parameters can be adjusted only if the alphabetic input is used for the OC Package and cannot be controlled if the numeric input format is used.

Fast-forwarding files

A user may want to start a simulation from somewhere in the middle of a previous simulation (or if the previous simulation was not completed) instead of restarting the simulation from the start. For this case, the FASTFOWRARD and FASTFORWARD options allow the simulation to use results from previous binary files, at any time-step and stress period, to restart a simulation for flow and/or transport respectively.

Bootstrapping

For transient simulations, robustness and efficiency may be enhanced by providing a good initial guess to the solution at every time step, instead of using the head at the old time value (HOLD) as the value for HNEW at the first iteration of every new time step. This estimate can be obtained from the solution of a previous run for instance during parameter estimation, to jumpstart the current simulation.

A simple bootstrapping methodology is provided that appropriately reads the transient binary file of a previous simulation for this purpose. The algorithm does not require that the file be written/read at every step or that both simulations have the same time stepping sequences – the algorithm interpolates for the change in values obtained from the bootstrapping file and adds the change to HOLD to give the estimate for HNEW of the new time step for the current simulation.

Thus, as the simulation parameters evolve through calibration for example, the head value itself may be different between simulations, however, the change in head may not be that different therefore providing a reasonably good initial estimate to jumpstart a new time-step, which may be closer to the actual solution than using HOLD as the initial estimate.

Figure 19 below shows the bootstrapping approach used.

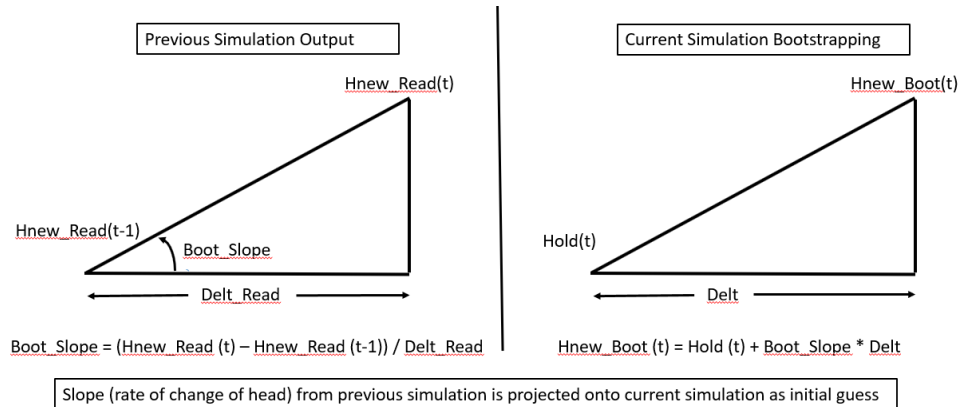


Figure 19, Bootstrapping approach.

In addition to the bootstrapping methodology, a scaling algorithm is also optionally incorporated to scale the bootstrapped head slope by the actual head slope of a previous time-step. The scaling is performed to account for differences between the current simulation and the previous one from which the bootstrapping heads are read. The scaling algorithm is shown on **Figure 20**.

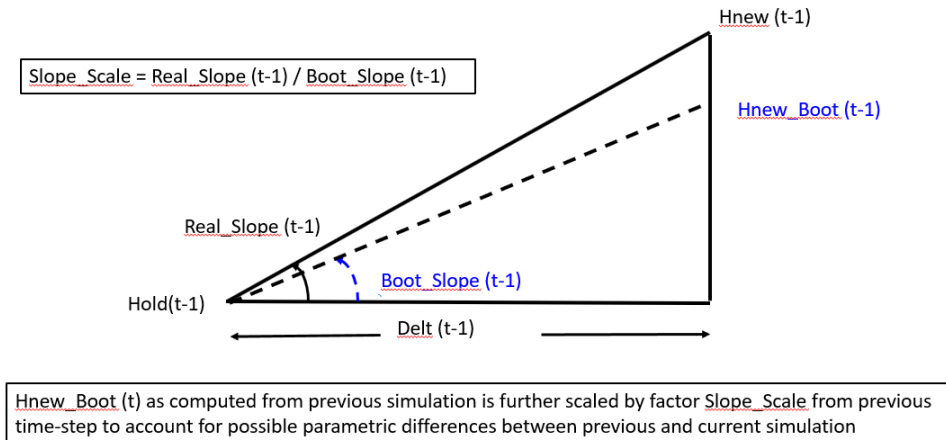


Figure 20. Scaling approach for bootstrapping

Note that bootstrapping does not need the previous run to have every time step recorded to the file, or that the time-stepping is the same between the previous and current simulation. However, in theory it would be most efficient if the head output of the previous simulation were made at every time step and if the time-step sizes were the same between runs.

This capability of jumpstarting every time-step of a transient simulation is turned on by use of a BOOTSTRAPPING option which is available when adaptive time stepping is turned on. When



this option is used, it is followed by the unit number of the HDS file for the groundwater domain, and optionally, the unit numbers for the CLN and DPF domains (if they exist). The bootstrapping can further be turned on or off at any stress period as is the case for the solver parameters discussed above by optional keywords BOOTSTRAP or NOBOOTSTRAP. By default, the last of BOOTSTRAP or NOBOOTSTRAP is maintained for all stress periods till altered at a subsequent stress period. Scaling of the bootstrapped head can be turned on or off by the optional keywords BOOTSTRAPSCALE or NOBOOTSTRAPSCALE.

EXAMPLE PROBLEMS

Several benchmark and verification simulations have been conducted with the BCT Process modules to test accuracy and performance. The code has been tested in 1-, 2-, and 3-dimensions, against analytical solutions as well as against other numerical codes; specifically, MT3D (Zheng and Wang, 1999). The following example problems are provided to demonstrate application of the BCT Process. It is recommended that users familiarize themselves with the different simulation options, code accuracy under various conditions, and input/output structures of the BCT Process via these test problems.

Advection, Dispersion and Decay in a One-Dimensional Uniform Flow Field

This test problem discusses one dimensional advective dispersive transport with first-order decay of a chemical species, from a prescribed concentration source in a uniform, steady-state flow field. A 1000-foot-long domain is discretized into 2 layers, 2 rows, and 101 columns using $\Delta x = \Delta y = 10$ feet, and $\Delta z = 5$ feet. Two layers and rows were selected for convenience. The flow-field is setup using a hydraulic conductivity of 10 ft/day and constant head boundaries of 1,100 feet and 100 feet at either end of the domain. The seepage velocity is thus $v = 50$ feet/day, for an effective porosity of 0.2.

Various transport simulations were conducted with this setup to test the transport components of the code individually and in combination. Case 1 conducts the simulation with zero dispersion. Case 2 includes a retardation of 2 by using a bulk density value of 1 kg/L and an adsorption coefficient (k_d) of 0.2 L/kg. Case 3 further includes a longitudinal dispersivity value of 10 feet (grid Peclet number of 1) while Case 4 includes a longitudinal dispersivity value of 1 foot (grid Peclet Number of 10). Finally, Case 5 also includes first order decay with a half-life of 10 days (first order decay rate of 6.9315×10^{-2} /day) on the simulation with the high grid Peclet Number. Note that the grid Peclet number is $Pe = \Delta x / \alpha_l$ when diffusion is zero.



Transport simulations were performed for 20 days using 1000 time-steps of fixed size $\Delta t = 0.2$ days. Initial concentration of water within the domain was zero, and a prescribed concentration of 1 mg/L was set at the upstream end of the domain for the duration of the simulation. Simulation results were compared with the analytical solution for the respective cases. **Figure Ex 1.** Simulation Results for Advection in a One-Dimensional, Uniform Flow Field. shows the simulation results after 10 days of simulation, when the advective front of Case 1 moves halfway into the one-dimensional domain. Results are presented for an upstream weighted solution and for a solution using the TVD scheme with 4 TVD iterations. The purely advective analytical solution is also shown on the figure for comparison. It is noted that the TVD scheme greatly reduces numerical dispersion associated with the upstream weighted scheme; the sharp front is resolved over a span of 11 grid-blocks with the TVD scheme, as compared to about 32 grid-blocks with the upstream weighted scheme. However, this too may be considered too numerically dispersed for an advective solution involving non-linear reactive transport (where the reactions dominate only in the dilute fringes of the plume) and therefore a finer discretization should be provided in regions where a sharp advective front may be encountered and where such accuracy is important. **Figure Ex 1** also shows the simulation results after 20 days of simulation for Case 2 using 4 TVD iterations. With a retardation of 2, this front is noted to move the same amount in 20 days as for Case 1 in 10 days.

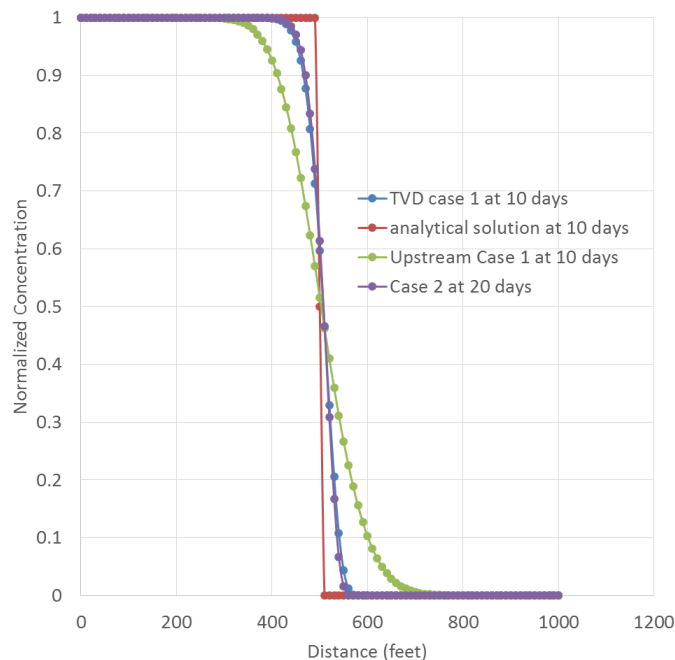


Figure Ex 1. Simulation Results for Advection in a One-Dimensional, Uniform Flow Field.



Results for the simulation cases 3, 4 and 5 are shown on **Figure Ex 2**. Simulation Results at 20 Days for Advection, Dispersion, and Decay in a One-Dimensional, Uniform Flow Field, along with analytical solution results for the respective cases at 20 days. The Domenico spreadsheet analytical solution was used for comparison ([www.elibrary.dep.state.pa.us/dsweb/Get/Version-49262/ Quick Domenico.xls](http://www.elibrary.dep.state.pa.us/dsweb/Get/Version-49262/Quick_Domenico.xls)). The simulation results for all three cases are almost the same as the respective analytical solution results. The largest errors occurred for Case 4 with a high Peclet number of 10, however, inclusion of decay diminished that error as noted for Case 5. Thus, it is noted that solution accuracy of advective transport improves substantially if a reasonable amount of dispersion or solute decay is present. Numerical experiments with different numbers of TVD iterations (including use of just two iterations as in a predictor/corrector approach) did not noticeably change the results for any of the cases discussed above.

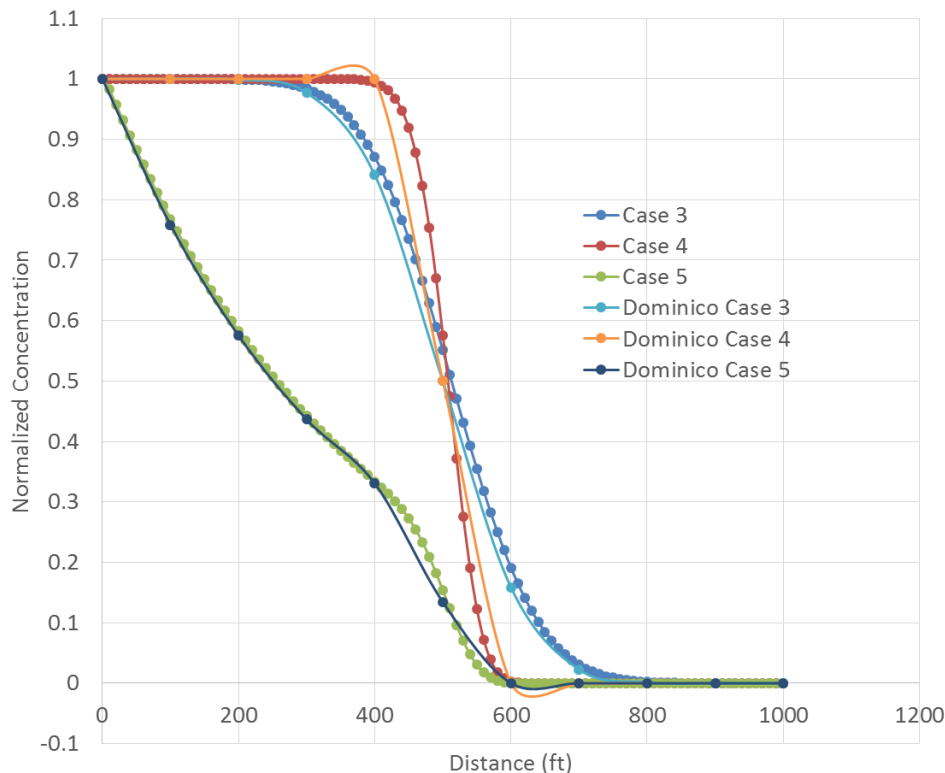


Figure Ex 2. Simulation Results at 20 Days for Advection, Dispersion, and Decay in a One-Dimensional, Uniform Flow Field.

Advection and Dispersion in a Two-Dimensional Confined Radial Flow Field

This test problem discusses advective dispersive transport of a chemical species in a radial flow field resulting from injection of a dissolved chemical species at the center of a 10,000 feet by



10,000 feet square simulation domain. The domain is discretized into 1 layer, 100 rows, and 100 columns with $\Delta x = \Delta y = 100$ feet, and $\Delta z = 15$ feet. A confined flow-field is setup using a hydraulic conductivity of 100 ft/day, a constant head boundary condition of 20 feet around the perimeter, and a well at row = 50 and column = 50, that injects fluid at a rate of 10,000 ft³/day. The concentration of water in the domain is zero at the start of the simulation. The species concentration in injected water is 1mg/L. The dispersivity values used were $\alpha_l = 500$ feet and $\alpha_t = 50$ feet for the longitudinal and transverse directions respectively, and the effective porosity value used was 0.2.

The transport simulation was conducted for 5,000 days with 50 time steps using a fixed time-step size of 100 days. Also, the cross-dispersion option was activated. Simulation results for this test case are compared with results from an MT3D (Zheng and Wang, 1999) simulation with an identical setup. **Figure Ex 3** shows a comparison of the results from the BCT package and MT3D simulation at 5,000 days. A good comparison is noted between the simulations.

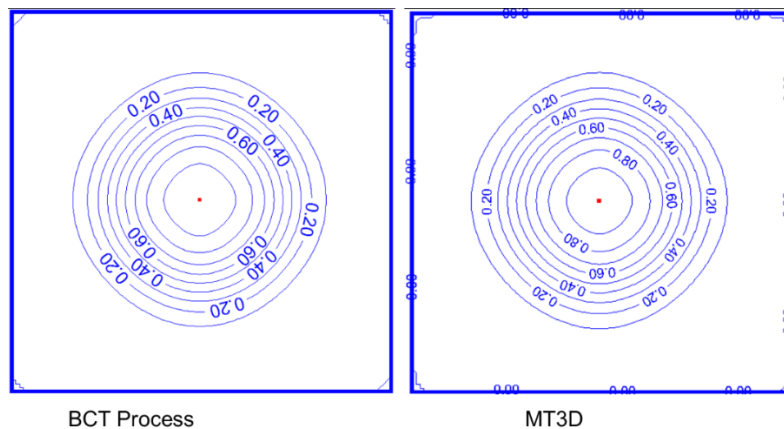


Figure Ex 3. Comparison of Model Results with MT3D Simulation Results for Transport in a 2-D Radial Flow Field at 5,000 days.

The same situation was also simulated without the optional cross-dispersion term activated. **Figure Ex 4** shows a comparison of the results at 5,000 days from the BCT package with and without cross-dispersion. It is noted that the cross-dispersion term rounds off the contours which are otherwise diamond-shaped when the cross-dispersion term is ignored.

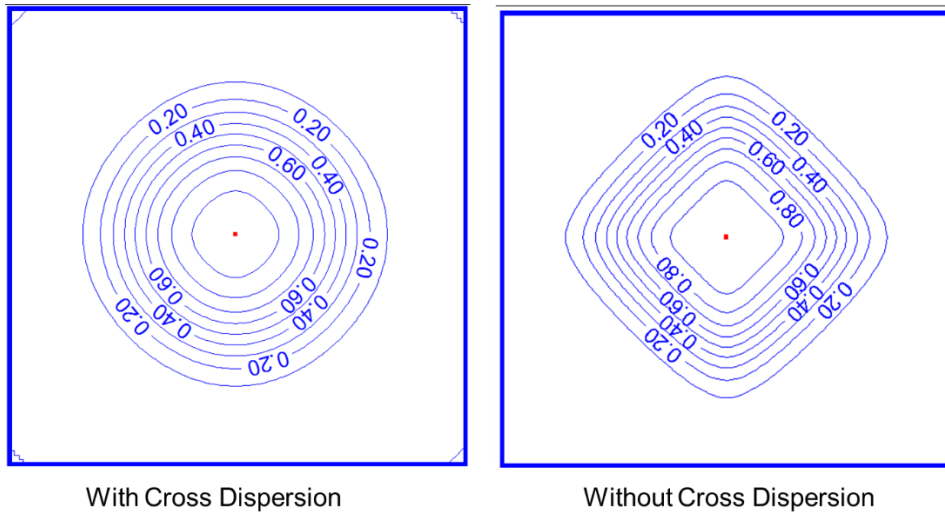


Figure Ex 4. Comparison of Simulation Results with and without Cross Dispersion for Transport in a 2-D Radial Flow Field at 5,000 days.

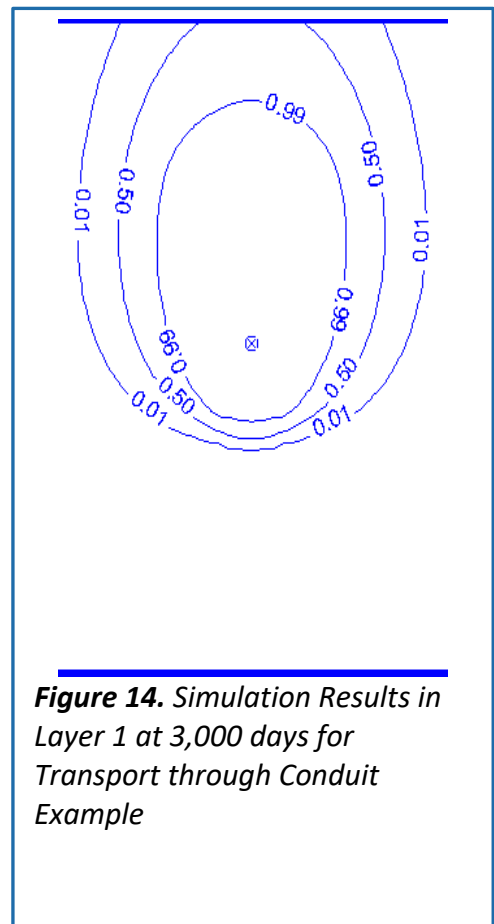
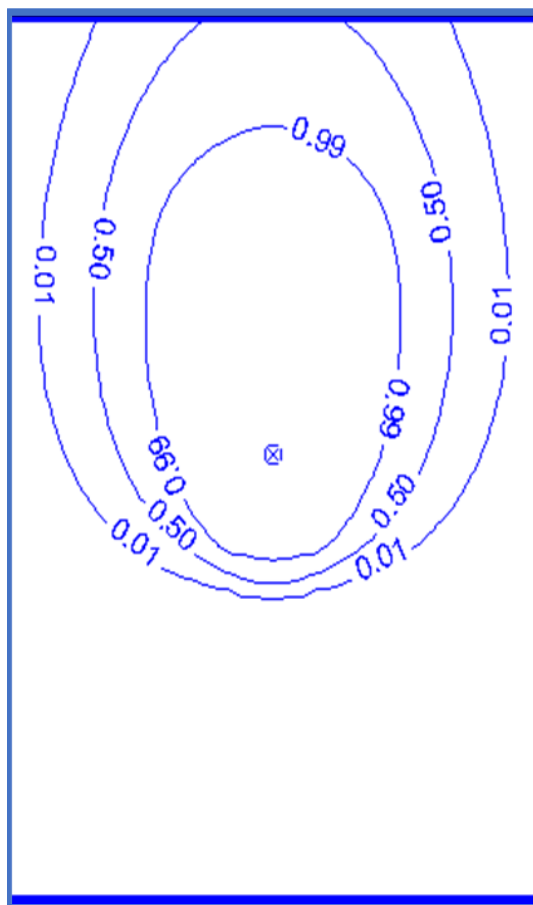
Transport of Solute through a Conduit within a Multi-Aquifer System

This test problem discusses transport of a dissolved chemical species through a conduit connecting two confined aquifers. The simulation domain of 47,000 feet by 47,000 feet square is discretized by 2 layers, 100 rows, and 100 columns of dimensions $\Delta x = \Delta y = 100$ feet, and $\Delta z = 10$ feet. The two layers are separated by a confining aquitard with zero leakance and a thickness of 10 feet. The overlying aquifer has an ambient gradient from south to north with a constant head boundary value of 30 feet along the south boundary and a constant head boundary value of 10 feet along the north boundary. The lower aquifer has a constant head of 60 around the entire perimeter. A conduit of 1 foot diameter connects the two aquifers resulting in flow up through the conduit into layer 1 from the bottom aquifer as a result of the head differential, with a subsequent radial flow component due to the mound and a northward flow component due to the ambient gradient. The steady-state flow-field thus generated was used for the transport simulation.



The transport simulation with the BCT package was conducted for a period of 3,000 days with a fixed time-step size of 30 days. The concentration of water in the upper aquifer is zero at the start of the simulation, while the concentration of water in the lower aquifer is one at the start of the simulation. The upstream weighting scheme was used for the simulation with dispersivities in the longitudinal and transverse directions set to zero. **Figure Ex 5** shows the results of the simulation at 3,000 days, in layer 1 indicating that solutes from layer 2 migrated into layer 1 through the conduit due to head gradients between the aquifers causing the resulting plume in layer 1.

A simulation was also conducted for this case with use of a nested grid. The region around the conduit was nested with each cell being further subdivided in two along the row and column directions. Note that the nesting is not ideal for this problem, as the plume crosses the nested region in the lateral and longitudinal directions. However, such a setup depicts the code accuracy in evaluating transport across nested regions. **Figure Ex 6** shows the nested grid used





for this simulation and the simulation results in layer 1 at 3,000 days. The results are very similar to those of **Figure Ex 5**.

Figure Ex 5. *Simulation Results in Layer 1 at 3,000 days for Transport through Conduit Example.*

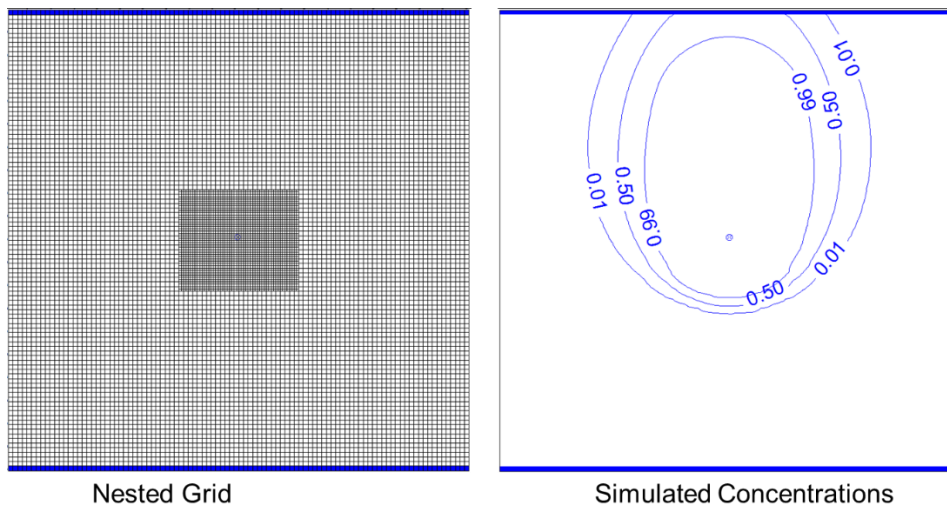


Figure Ex 6. *Simulation Results in Layer 1 at 3,000 days for Transport through Conduit Example, Using a Nested Grid.*

Dual Domain Transport in a One-Dimensional, Uniform Flow Field

This test problem discusses one dimensional dual domain transport in a uniform steady-state flow-field. A 150 foot long horizontal soil column is discretized into 1 layer, 1 row, and 300 columns using $\Delta x = 0.5$ feet, $\Delta y = 1$ foot, and $\Delta z = 1$ foot. The flow-field is setup using a hydraulic conductivity of 1000 ft/day and constant head boundaries of 10 feet and 9 feet at either end of the domain. The simulation considers a dual porosity system with a mobile domain fraction of 0.4. Transport related parameters for the mobile domain include a longitudinal dispersivity of 0.5 feet, zero molecular diffusion, a porosity value of 0.35, a soil bulk density value of 1.6 kg/L, and an adsorption coefficient (k_d) value of 0.1 L/kg. Transport parameters for the immobile domain include a porosity value of 0.2, a soil bulk density value of 1.6 kg/L, an adsorption coefficient (k_d) value of 0.1 L/kg, and a mass transfer rate of 0.1 day^{-1} . The concentration of water in both mobile and immobile domains is zero at the start of the simulation.



A transport simulation was performed for this setup with a prescribed species concentration of 1mg/L at the upstream end of the soil column within the mobile domain, for a period of 20 days. Subsequently, the concentration of inflow water was made to zero for a period of 30 days to evaluate flushing of the system. Each stress period contains 100 time steps of uniform size – 0.04 day step size for the first stress period when the component species front is advancing, and a 0.08 day step size for the second stress period when the soil column is being flushed. Simulation results were compared with results from a MT3D simulation of the same setup, using the TVD solution scheme. Note that the mobile porosity in MT3D is equal to the porosity of the mobile domain (0.35) times the mobile domain fraction (0.4), and that the immobile porosity in MT3D is equal to the porosity of the immobile domain (0.2) times the immobile domain fraction (which is one minus the mobile domain fraction = 0.6). **Figure Ex 7** shows the concentration versus time plot in the mobile domain, at the outlet of the domain. The MT3D and BCT Process simulation results are almost the same.

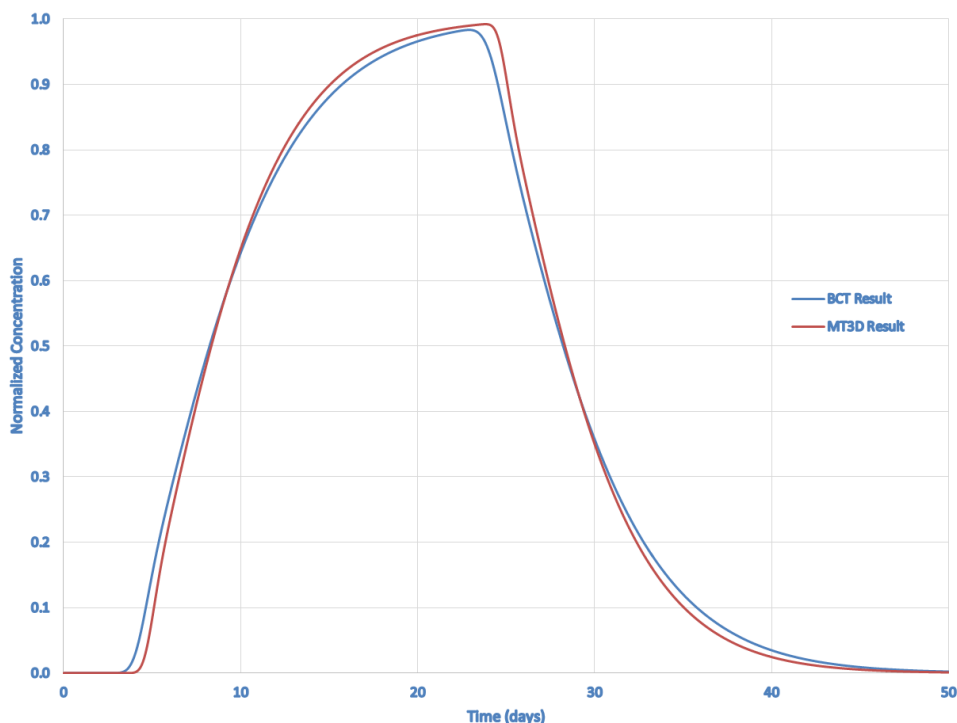


Figure Ex 7. Concentration at Outlet for Dual Domain Transport in a One-Dimensional, Uniform Flow Field.

Henry Problem for Density-Dependent Flow and Transport

The Henry Problem depicted by Guo and Langevin (2002) is replicated here to evaluate the ability of the BCT Process with density dependent flow capabilities coded into *USG-Transport*.



A cross-sectional domain 2-m long, by 1-m high, and by 1-m wide is provided a constant flux of fresh ground water along the left boundary at a rate (Q_{in}) of $5.702 \text{ m}^3/\text{d}$ per meter with a concentration (C_{in}) equal to zero. A zero constant head boundary is applied to the right side of the cross-section to represent seawater hydrostatic conditions. The upper and lower model boundaries are no flow. The finite-difference model grid used to discretize the problem domain consists of 1 row with 21 columns and 10 layers. Each cell, with the exception of the cells in column 21, is 0.1 by 0.1 m in size. Cells in column 21 are 0.01-m horizontal by 0.1-m vertical. The narrow column of cells in column 21 was used to better locate the seawater hydrostatic boundary at a distance of 2 m. The WEL package was used to assign injection wells, with constant inflow rates of $0.5702 \text{ m}^3/\text{d}$ to each cell of column 1. Constant freshwater heads were assigned to the cells in column 21 using a head value of 1.0 m and a concentration of 35 kg/m^3 . The concentration for inflow from these constant head cells was specified at 35 kg/m^3 . An identical problem setup in SEAWAT was also simulated for comparison.

Figure Ex 8 shows the results after 1 day, from the SEAWAT and the USG-Transport simulations. The results are virtually identical.

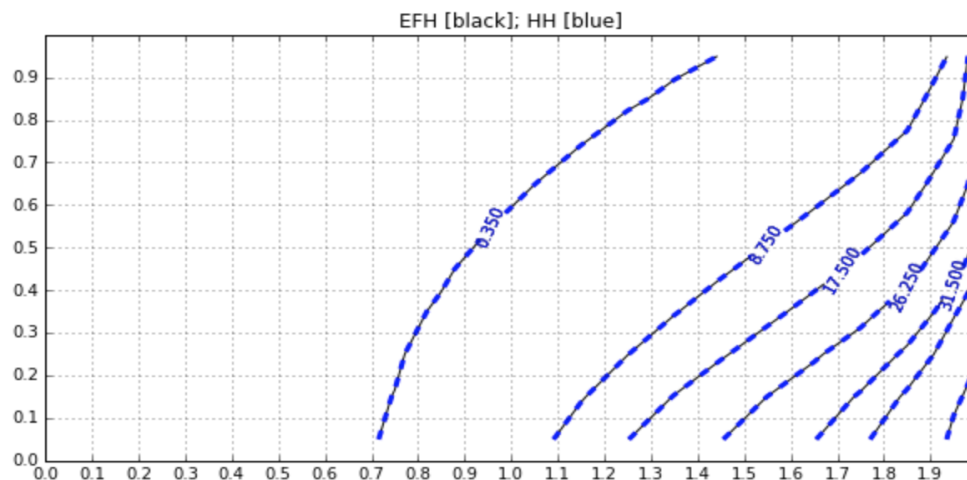


Figure Ex 8. Concentration profiles for Henry Problem Simulation.

The contribution of Christian D. Langevin is acknowledged in providing this example simulation problem.

Verification Example 1 for Heat Module

This verification example compares the results of a heat transport simulation with that of an equivalent solute transport simulation. A domain 400* m long by 120 m wide by 100 m high is



discretized using a variable grid spacing of 128 rows and 217 columns and 1 layer. The smallest grid spacing is 0.25 m and the largest is 4.5 m. Flow is from left to right with boundary head of 101 m to the left and 100.6 m to the right. Initial temperature within the domain is 283.15 K with a borehole heat source within the domain. The source input is 50 W/m of borehole giving 5,000 W of energy to the 100 m long vertical borehole. This was provided to the model as a small flux $q = 10^{-6} \text{ m}^3/\text{s}$ with a temperature of $T = 1200 \text{ K}$. Note that the energy equation is divided by ($\rho_w c_w$) and therefore the energy input at the borehole is also divided by the same amount to provide the boundary condition to the model. Parameters for heat and equivalent solute transport simulations are shown on **Table 2** below.

Table 2. Parameter values for thermal or equivalent solute simulations.

Symbol	Variable	Value	Unit
n	Porosity	0.2	(-)
λ_w	Effective thermal conductivity of water	0.65	(W/m/K)
λ_s	Effective thermal conductivity of soil	2.84	(W/m/K)
λ_m	Effective thermal conductivity of porous medium	2.402	(W/m/K)
ρ_w	density of water	1000	(kg/m ³)
c_w	Specific heat capacity of water	4.18E+03	(J/kg/K)
ρ_s	Density of solid material	2650	(kg/m ³)
c_s	Specific heat capacity of solid	750	(J/kg/K)
ρ_b	Dry bulk density	2120	(kg/m ³)
c_{sv}	volumetric heat capacity of solid	1.99E+06	(J/m ³ /K)
Kd	Partition coefficient	1.79E-04	(m ³ /kg)
R	Retardation factor	2.90	
α_l	Longitudinal dispersivity	0.5	(m)
α_{th}	Transverse horizontal dispersivity	0.05	(m)
α_{tv}	Transverse vertical dispersivity	0.05	(m)
Do	Thermal diffusivity	5.75E-07	(m ² /s)
Tu	Undisturbed temperature of ground	283.15	(K)
key	required for heat equation solution		
key	required for equivalent solute equation solution		
key	Intermediate calculation		
key	required for both solutions		

Simulated results after 1 year of simulation were same for the temperature and solute distribution of the two respective models. The temperature distribution around the borehole was as shown in **Figure Ex 9**.

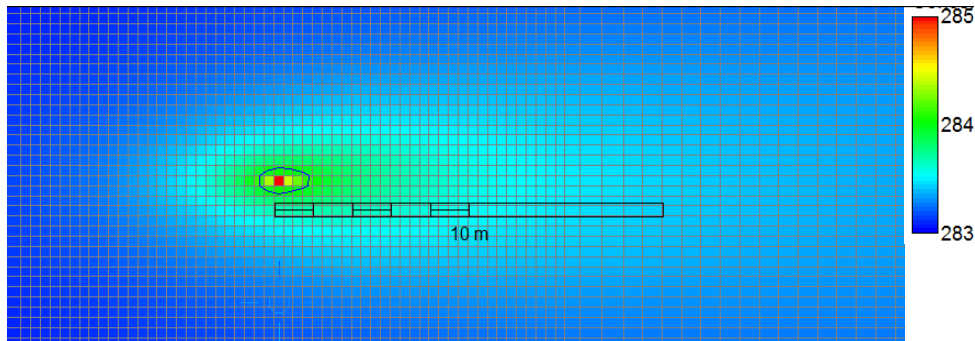


Figure Ex 9. Temperature Distribution around Borehole.

Verification Example 2 for Heat Module

This verification example was given by David Krcmar at Univerzita Komenského in Slovakia. The problem compares the MODFLOW-USG solution with the Stallman (1965) analytical solution for transient heat flow in the subsurface in response to a sinusoidally varying temperature boundary at land surface. The problem consists of a saturated vertical soil column 60 m long having a hydraulic conductivity of 1.0×10^{-4} m/sec (28.35 ft/d). Downward groundwater flow with a Darcy velocity is of 5×10^{-7} m/sec is generated by a constant head of 60 m at the top and 59.7043 m at the bottom. With a porosity of 0.35, the pore velocity is 12.3 cm/d = 1.42×10^{-6} m/sec. The column is discretized into 200 cells with a uniform thickness of 0.5 m each.

The transport model is simulated with both heat and equivalent solute conditions. The initial temperature of the subsurface is 10°C , and the temperature at the surface varies as $T_{\text{top}} = 10 + 5\sin(2\pi t/T)$ where t is the time and T is the wave length of 1 year. The heat transport parameters and equivalent solute parameters are shown on Table H1 below.

Table H1: Heat and equivalent solute transport parameters for variation example of heat module

Symbol	Variable	Value	Unit
n	Porosity	0.35	(–)
λ_w	Effective thermal conductivity of water	0.58	(W/m/K)
λ_s	Effective thermal conductivity of soil	2	(W/m/K)
λ_m	Effective thermal conductivity of porous medium	1.503	(W/m/K)
ρ_w	density of water	1000	(kg/m ³)
c_w	Specific heat capacity of water	4.17E+03	(J/kg/K)
ρ_s	Density of solid material	2650	(kg/m ³)
c_s	Specific heat capacity of solid	800	(J/kg/K)
ρ_b	Dry bulk density	1709.5	(kg/m ³)
c_{sv}	volumetric heat capacity of solid	2.12E+06	(J/m ³ /K)
K_d	Partition coefficient	1.92E-04	(m ³ /kg)
R	Retardation factor	1.94	
α_l	Longitudinal dispersivity	0	(m)



α_t	Transverse dispersivity	0	(m)
Do	Thermal diffusivity (diffusion coefficient)	1.03E-06	(m ² /s)
key	required for heat equation solution		
key	required for equivalent solute equation solution		
key	Intermediate calculation		
key	required for both solutions		

Results of the simulation at 10 years are shown on Figure H1. The Stallman analytical solution and MODFLOW-USG heat and equivalent solute transport simulations compare very well.

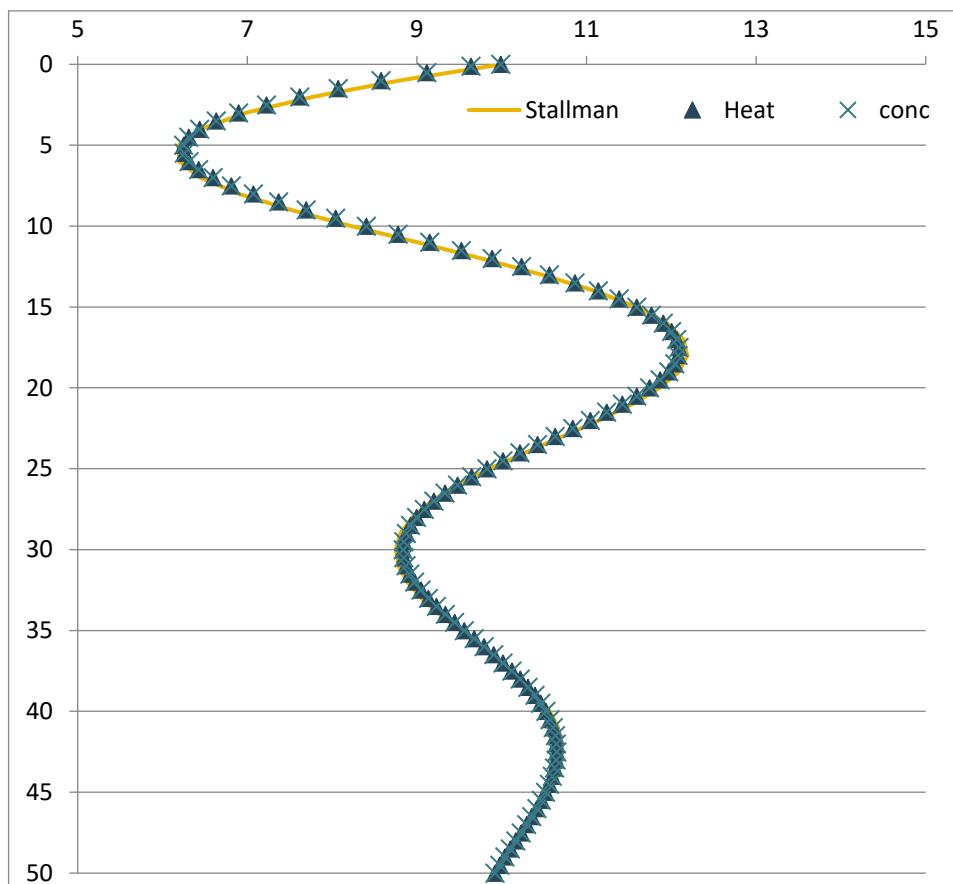


Figure H1: Comparison of temperature profile simulated with MODFLOW-USG as heat or equivalent solute transport against the Stallman (1965) analytical solution at 10 years.

Example Problems for Matrix Diffusion Transport Package

Several benchmark and verification simulations have been conducted with the MDT Package modules to test accuracy and performance. The code has been tested in 1-, 2-, and 3-dimensions, against analytical solutions as well as against other numerical codes; specifically, MT3D (Zheng and Wang, 1999). The following example problems are provided to demonstrate

application of the MDT Process. It is recommended that users familiarize themselves with the different simulation options, code accuracy under various conditions, and input/output structures of the MDT Process via these test problems.

Example MD1: Comparison with Semi-Analytical Solution

Matrix diffusion from a single fracture (**Figure Ex 10**) is used to demonstrate the MODFLOW-USG MDT package's ability to duplicate the semi-analytical solution method detailed in Falta and Wang (2017). This combination of an extremely small fracture in a much larger unfractured surface area for diffusion represents an extreme case for matrix diffusion. In this example, tritium flows through a 100 μm fracture with a Darcy velocity of 0.1 m/day at a constant concentration for 30 years (Falta and Wang, 2017; Sudicky and Frind, 1982).

The system was modeled as a one-dimension (1-D) grid along the fracture in the direction of groundwater flow with unit thickness (perpendicular to groundwater flow) representing the thickness of the fracture. To replicate Falta and Wang (2017), 61 1-m wide cells were used parallel to the groundwater flow with source concentrations assigned to the first upgradient cell. Constant head boundaries were assigned to the first and last grid cells. Model input parameters are provided in **Table 3**. A fracture-matrix interfacial area (A_{md}) of 2 m^2 was used in each model cell to account for matrix diffusion from both the top and bottom of the fracture. The model was run for 50 years using a time-step of 0.1 years. Similar to the Falta and Wang (2017) setup, longitudinal dispersion was not included in the model.

Table 3. Model input parameters for the simulation of matrix diffusion through a single fracture.

Parameter	Value	
	Fracture	Unfractured Media
Fracture thickness (μm)	100	
Darcy velocity (m/d)	0.1	0
Tritium source concentration (mg/L)	1.0	
Tritium loading period (years)	30	
Porosity (unitless)	1	0.01
Retardation factor (unitless)	1	1
Tortuosity (unitless)	1	0.1
Diffusion coefficient (m^2/s)	1.60×10^{-9}	1.60×10^{-9}
Tritium decay rate (1/yr)	5.61×10^{-2}	5.61×10^{-2}

A comparison of the MODFLOW-USG MDT package with the semi-analytical and analytical solutions of Falta and Wang (2017) is shown in **Figure Ex 11**. While the source is on, the plume



continues to expand with concentrations decreasing with increasing distance from the source (year 5 and year 25 curves in **Figure Ex 11**). However, once the source is removed in year 30, back diffusion of tritium from the unfractured media into the fracture is what feeds the plumes observed for years 31, 33, and 50. As stated by Falta and Wang (2017):

The semi-analytical method seems to perform quite well for this demanding case, almost perfectly matching the concentrations during the loading period. At 31 years, one year after the source is removed, the semi-analytical method is able to capture the main features of the concentration reversal near the source, and it continues to perform well as the back diffusion proceeds past 50 years.

As shown in **Figure Ex 11**, the MODFLOW-USG MDT package in duplicated the semi-analytical solution outputs.

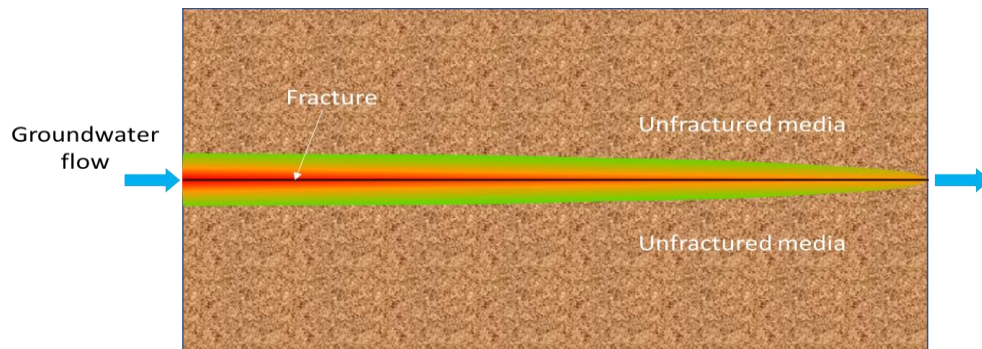


Figure Ex 10. Simulation of matrix diffusion through a single fracture.

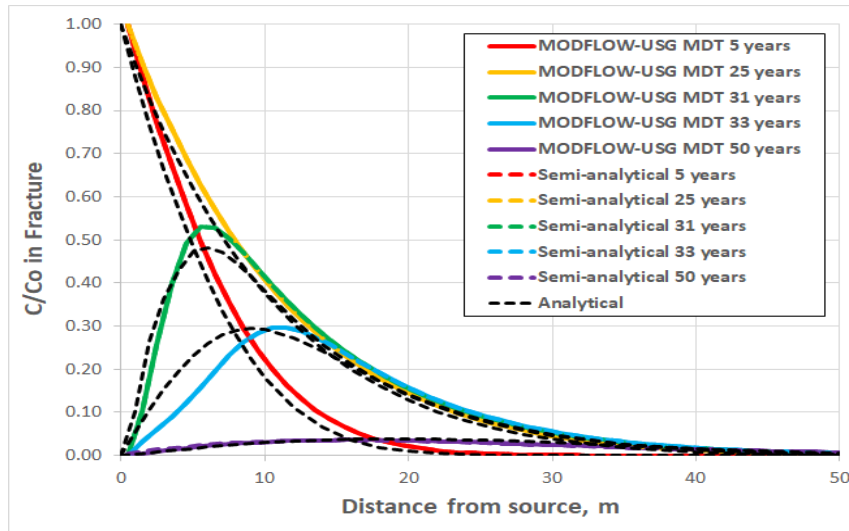


Figure Ex 11. Comparison of MODFLOW-USG MDT package output with the semi-analytical and analytical models. Note MODFLOW-USG and the semi-analytical curves are identical.

Example MD2: Comparison with Embedded Low Permeability Zones

Flow of fluorescein through a sand tank embedded with several low permeability zones is used to demonstrate the MODFLOW-USG MDT package's ability to model the effects of diffusion into and from low-k zones. The sand tank study is described in detail in Chapman et al. (2012) and a comparison with the semi-analytical solution is described in Muskus and Falta (2018). In this laboratory experiment, a tracer solution containing fluorescein and bromide was flushed through a $1.07 \times 0.03 \times 0.84$ m sand tank embedded with four clay lenses (**Figure Ex 12**). Twenty-two days of tracer flushing through the system were followed by 100 days of flushing with clean water. A flow rate of 0.9 mL/min was used for the first 10 days after which a constant rate of 1.5 mL/min was employed.

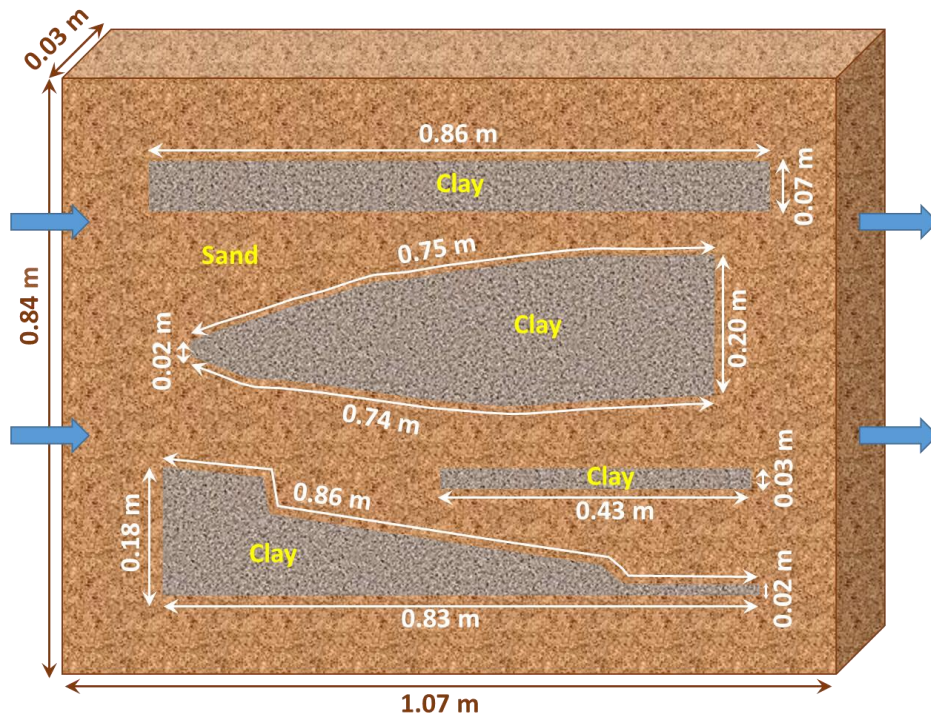


Figure Ex 12. Sand Tank Configuration. Based on Muskus and Falta and (2018) Figure 5. Groundwater flow is from left to right (blue arrows). Not to scale.

Chapman et al. (2012) used various numerical models to simulate the tank experiment. Although their models closely matched the experimental data, they required high resolution grids with ~9,000 to 24,000 cells. Comparatively, Muskus and Falta (2018) simulated the sand tank as a 1-D model with 50 cells in the direction of groundwater flow (x direction) with the clay lenses represented as an embedded matrix diffusion area.

Developed as a 1-D model, the MODFLOW-USG MDT model contains a total of 50 cells (1 row, 50 columns, and 1 layer). Input parameters are shown in **Table 4**. Matrix diffusion was modeled as being from the clay lenses embedded in the transmissive zone. A total matrix diffusion area of 0.193 m^2 was obtained as the sum of the surface area (perimeter times the thickness of the sand tank) of the four clay lenses; resulting in an interfacial surface area (A_{md}) per cell of $3.85 \times 10^{-3} \text{ m}^2$ (Muskus and Falta, 2018). A volume fraction of 0.711, a characteristic diffusion length of $4.05 \times 10^{-2} \text{ m}$, and a Darcy velocity of 31.29 m/yr were also obtained from Muskus and Falta (2018). Note that while the sand tank experiment employed variable volumetric flowrates (0.9 mL/min for the first 10 days and 1.5 mL/min thereafter), Muskus and Falta (2018) selected the most prevalent flow rate (1.5 mL/min) for the entire time period for their comparison. This

same methodology was applied to the MODFLOW-USG-MD model. Fluorescein source concentrations were maintained at 400 mg/L for the first 22 days, after which the tracer concentrations were set to zero.

Table 4. Model input parameters for the simulation of the sand tank.

Parameter	Value	
	Sand	Clay
Darcy velocity (m/yr)	31.29	
Porosity (unitless)	0.45	0.6
Source concentration (mg/L)	400	
Loading period (days)	22	
Retardation factor (unitless)	1.39	1
Tortuosity (unitless)	1.0	0.3
Global Parameters		
Matrix diffusion location	embedded	
Characteristic diffusion length (m)	0.0405	
Transmissive zone volume fraction (%)	71.1	
Matrix diffusion area (m ²)	0.00385	
Diffusion coefficient (m ² /yr)	0.0173	
Length along groundwater flow direction (x) (m)	1.07	
Width perpendicular to groundwater flow direction (y) (m)	0.03	
Depth (z) (m)	0.84	
Δx (m)	0.0214	
Δy (m)	0.03	
Δz (m)	0.84	
Δt (d)	0.5	

As shown on **Figure Ex 13**, observed fluorescein concentrations increase over the first 22 days while the tracer is introduced into the sand tank. Once the tracer is removed on day 22, concentrations start to decrease and tail off by day 120.

Comparison of the MODFLOW-USG MDT model effluent concentrations with the experimental results is shown on **Figure Ex 13**. Also shown on the figure are the simulated concentration curves reported by Muskus and Falta (2018) and Chapman et al. (2012). The MODFLOW-USG MDT package was able to reproduce the observed effluent concentrations reasonably well. The slight left shift of the leading edge of both the MODFLOW-USG MDT and semi-analytical curves compared to the observed concentrations is expected due to the difference in volumetric flowrates over the first 10 days between the simulated and experimental results.

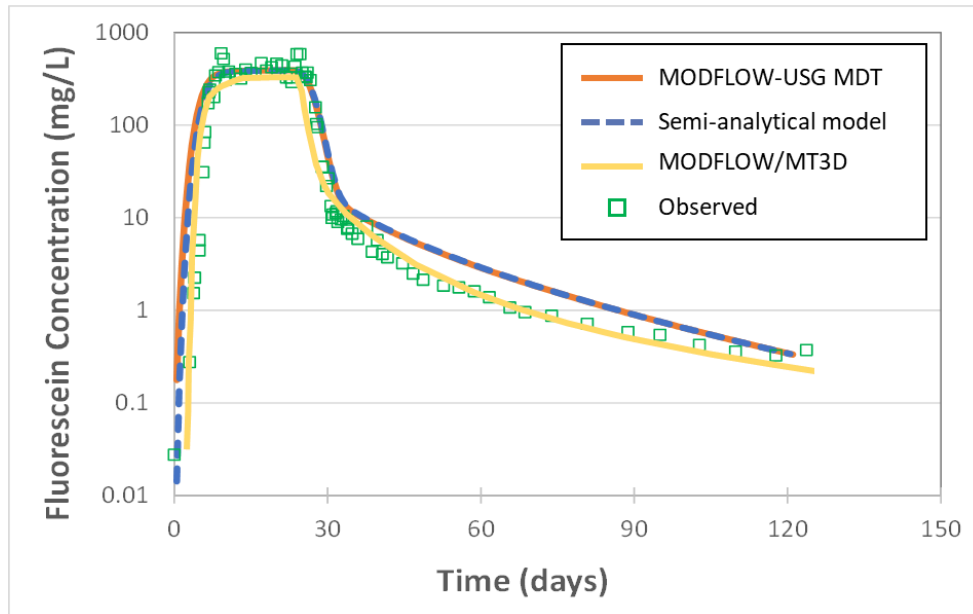


Figure Ex 13. Comparison of MODFLOW-USG MDT model (50 model cells) output with Chapman et al. (2012) MODFLOW/MT3DMS model (8,988 model cells), semi-analytical model (50 model cells), and observed concentrations.

Example MD3: Demonstration of PCE Decay

A hypothetical site with a tetrachloroethene (PCE) release is used to demonstrate MODFLOW-USG MDT package's ability to model the effects of sequential decay in addition to diffusion into and from low-k zones (**Table 5**). The transmissive zone is a sandy aquifer that is interbedded with clay lenses. Hydraulic conductivity of the transmissive zone is 12,500 m/yr and the hydraulic gradient across the site is 0.002 m/m. Porosities for the sand and clay are 0.33 and 0.4, respectively. A tortuosity of 0.7 is assumed for both sand and clay. PCE is continuously released into groundwater at a concentration of 100 mg/L. The source area is 10 m perpendicular to groundwater flow and 3 m deep.

Approximately 40% of the transmissive zone comprises of clay lenses with a characteristic diffusion length of 0.5 m. Therefore, matrix diffusion was modeled as being from the clay lenses embedded in the transmissive zone. PCE is assumed to undergo reductive dechlorination to trichloroethene (TCE), then to cis-1,2-dichloroethene (cis-DCE), and finally to vinyl chloride (VC) with decay rates of 0.4 yr⁻¹ (PCE), 0.15 yr⁻¹ (TCE), 0.1 yr⁻¹ (cis-DCE), and 0.2 yr⁻¹ (VC) (Wiedemeier et al., 1999; Aziz et al., 2002). Retardation factors assigned to each of the constituents are shown in **Table 5**.



MODFLOW USG-Transport

Sorab Panday

The MODFLOW-USG MDT model contains 5 layers, 20 rows, and 100 columns. Input parameters are shown in **Table 5**.

Comparison of the MODFLOW-USG MDT model concentrations with the semi-analytical model results over various years is shown on **Figure Ex 14**. As shown in the figure, the MODFLOW-USG MDT package was able to reproduce the constituent concentrations reasonably well for all constituents.



Table 5. Model input parameters for the simulation of sequential decay.

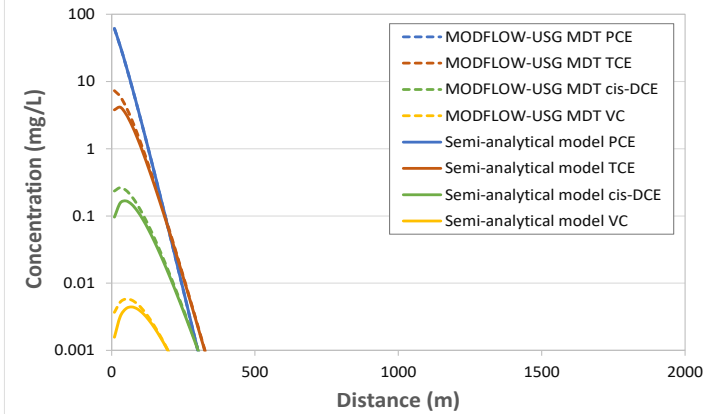
Parameter	Value	
	Sand	Clay
Hydraulic Conductivity (m/yr)	12,500	
Hydraulic gradient (m/m)	0.002	
Porosity (unitless)	0.33	0.4
Source concentration (mg/L)	100	
Retardation factor (unitless)		
PCE	1.94	1.78
TCE	1.57	1.47
Cis-DCE	1.18	1.15
VC	1.07	1.05
First order degradation rates (per year)		
PCE	0.40	0.40
TCE	0.15	0.15
Cis-DCE	0.10	0.10
VC	0.20	0.20
Distribution coefficient (mL/g)		
PCE	0.194	0.194
TCE	0.117	0.117
Cis-DCE	0.0363	0.0363
VC	0.0137	0.0137
Tortuosity (unitless)	0.7	0.7
Bulk density (g/mL)	1.6	1.6
Global Parameters		
Matrix diffusion location	Embedded	
Characteristic diffusion length (m)	0.50	
Transmissive zone volume fraction (%)	40	
Matrix diffusion area (m ²)	360	
Diffusion coefficient (m ² /yr)	0.0316	
Source width perpendicular to groundwater flow direction (m)	10	
Source depth (m)	3	
Model length along groundwater flow direction (x) (m)	2000	
Model width perpendicular to groundwater flow direction (y) (m)	100	
Model depth (z) (m)	15	
Yield of Component 2 from Component 1	0.795	
Yield of Component 3 from Component 2	0.737	
Yield of Component 4 from Component 3	0.64	
Δx (m)	20	
Δy (m)	5	
Δz (m)	3	
Δt (d)	1	



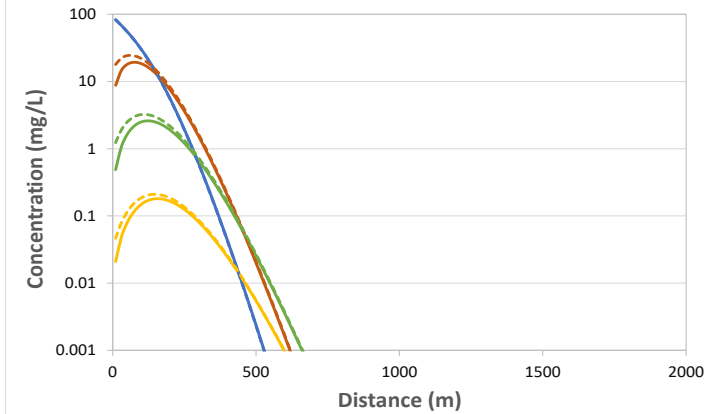
MODFLOW USG-Transport

Sorab Panday

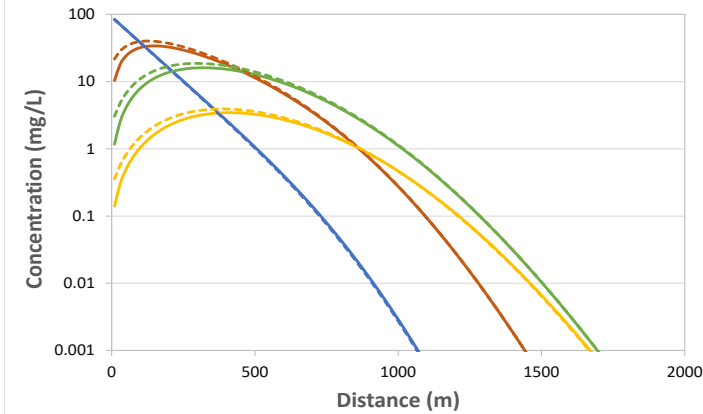
A) 1 Year After Source Release



B) 5 Years After Source Release



C) 25 Years After Source Release



D) 50 Years After Source Release

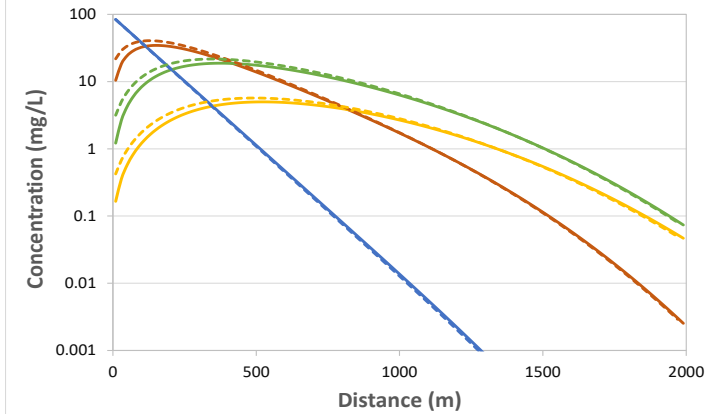


Figure Ex 14. Comparison of MODFLOW-USG MDT model output with the semi-analytical model 1 year, 5 years, 25 years, and 50 years after source release.



Example Problem for Solute Transport in LAKE Package Verification

A verification example is presented here. The example is based on the verification example provided in Merritt and Konikow (2000) and Bedekar et al (2016). The simulation considers a lake in a two-layer groundwater model simulated for a 5,000-day period. Initial conditions in the model are such that the lake stage is 50 ft below the fixed groundwater heads on the left side of the model boundary. The initial concentration of the lake is set to 100 mg/L. As the simulation begins, groundwater with zero concentration enters the lake causing dilution in the lake. As the lake fills over time lake stage and volume become stable and the lake stage rises sufficiently above the right boundary of the groundwater model to cause seepage from the lake into the groundwater system. Continued precipitation over the lake and lake seepage into ground causes dilution to continue at a low rate.

Change in lake stage and volume over time is shown in **Figure Ex 15**. Lake stage and volume calculated with MODFLOW-USG are compared with a MODFLOW simulation to verify the flow solution of the LAK package of MODFLOW-USG. A good agreement is observed between MODFLOW-USG and MODFLOW. MT3D-USGS is used with the MODFLOW model to simulate solute transport in lakes. Dilution in lake concentration is shown in **Figure 23**. Results from MODFLOW-USG lake transport are compared to results obtained using MT3D-USGS. Manual calculations were also performed to verify the correctness of the solution. Lake concentration calculated by MODFLOW-USG show a good match with manual calculations and MT3D-USGS results.



MODFLOW USG-Transport

Sorab Panday

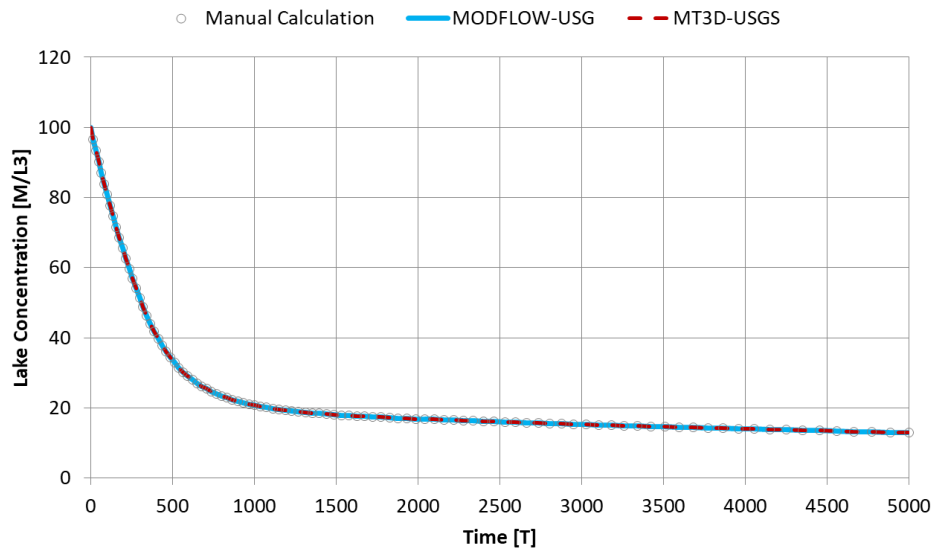


Figure Ex 15. Change in lake stage and volume.

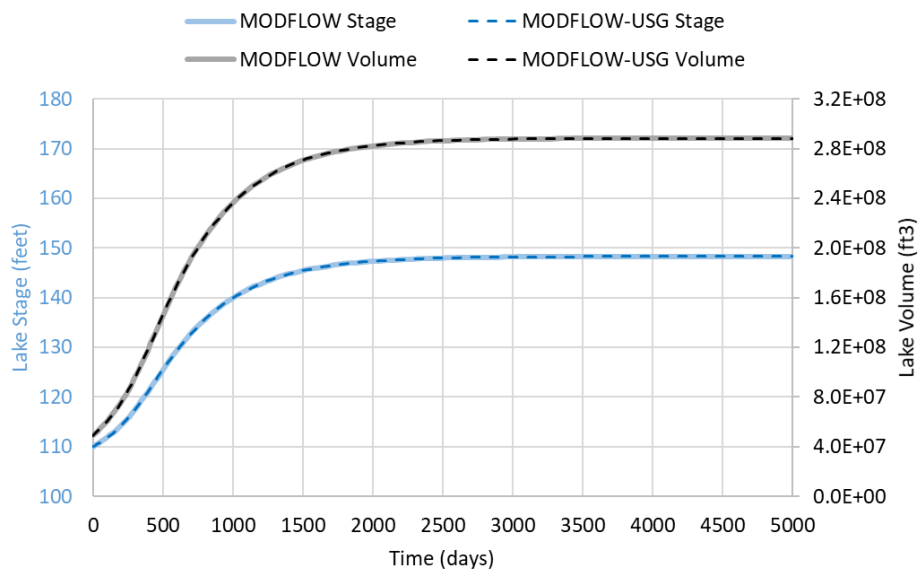


Figure Ex 16. Change in lake concentration over time.

Example Problem for PFAS Adsorption on Air-Water Interface in the Unsaturated Zone Verification

This example problem discusses one dimensional advective transport of PFAS in the vadose zone. The experiments conducted by Lyu (2018) are simulated here to evaluate the accuracy of the model in their validation. These experiments were also simulated by Silva et al. (2020) using a modified Hydrus model.



The setup simulated here consists of a 1-dimensional vertical soil column of 0.35 mm sand, 15 cm long, with a steady-state recharge from the top that gives a pore velocity of 37 cm/hr. For a porosity value of 0.33, the recharge rate is 12.21 cm/hr, and one pore volume (PV) is equal to 0.405 hours. Once the flow field was stabilized, PFOA solution was injected with the recharge water. The experiment was conducted with different soil water saturations, and different injection concentrations of PFOA. The simulations conducted here are for saturated conditions, and for a water saturation (S_w) of 0.68, with PFOA concentrations of 1, 0.1, and 0.01 mg/L.

The soil column was discretized uniformly into 30 numerical layers of 0.5 cm thickness each with a bottom elevation of zero and a top elevation of 15 cm. Recharge was simulated from the top end at a rate of 12.21 cm/hr, with a prescribed head boundary condition at the bottom that can control the degree of saturation of the soil column. For the saturated case, the prescribed head condition was above the top of the soil column at 20 cm. For the case of $S_w = 0.68$, The bottom head was set to -1.8 cm with van Genuchten parameters $\alpha = 12.6 \text{ cm}^{-1}$, $\beta = 1.16$, $S_r = 0.22$, and the Brooks Corey exponent = 4. The steady-state flow-fields thus generated were used for the transport simulations.

For the saturated case, there is no air-water interface, so the only adsorption that occurs is on the soil. Lyu (2018) provides a linear sorption coefficient $k_d = 0.08 \text{ cm}^3 \text{g}^{-1}$ for the 0.35 mm sand. Simulations were conducted with and without adsorption on soil to evaluate the retardation that occurs due to soil. For the unsaturated case, adsorption also occurs on the air-water interface and is related to the specific area of the air-water interface (A_{awi}) as well as an air-water interface partition coefficient (k_{awi}). A_{awi} is computed as $A_{awi} = A_{max} (1 - S_w)$ where A_{max} is given as $216 \text{ cm}^2/\text{cm}^3$ (Lyu, 2018 and Silva et al. 2020). The air-water interface adsorption coefficient was provided as linearized for the three source concentrations by Lyu (2018) as 0.0021, 0.0027, and 0.004 for PFOA concentrations of 1, 0.1, and 0.01 mg/L respectively. Silva et al. (2020) used the thermodynamic relationship with surface tension to estimate the air-water interface adsorption as a function of concentration; however, there is no change in k_{awi} for concentration values less than 1 mg/L and thus they did not see a relationship of PFAS transport with respect to source concentration. Since the Langmuir isotherm is used for k_{awi} in USG-Transport, the air-water adsorption isotherm A and B parameters can be provided as $A = 0.0021$, 0.0027, and 0.004 for PFOA concentrations of 1, 0.1, and 0.01 mg/L respectively, and $B = 0$.



For the unsaturated case, Silva et al. (2020) noted that the simulated breakthrough curves for the unsaturated cases match the laboratory study better with a dispersion coefficient of 0.7 cm. The same behavior was noted with simulations conducted with USG-Transport. Also, a diffusion coefficient of 0.01944 cm²/hr (5.4 x 10⁻⁶ cm²/s) was provided.

A comparison of simulated results versus the laboratory observations is shown for the different source concentrations in Figure **Ex 17**. The simulated results are superposed on the results of Figure 4 of Lyu (2018).

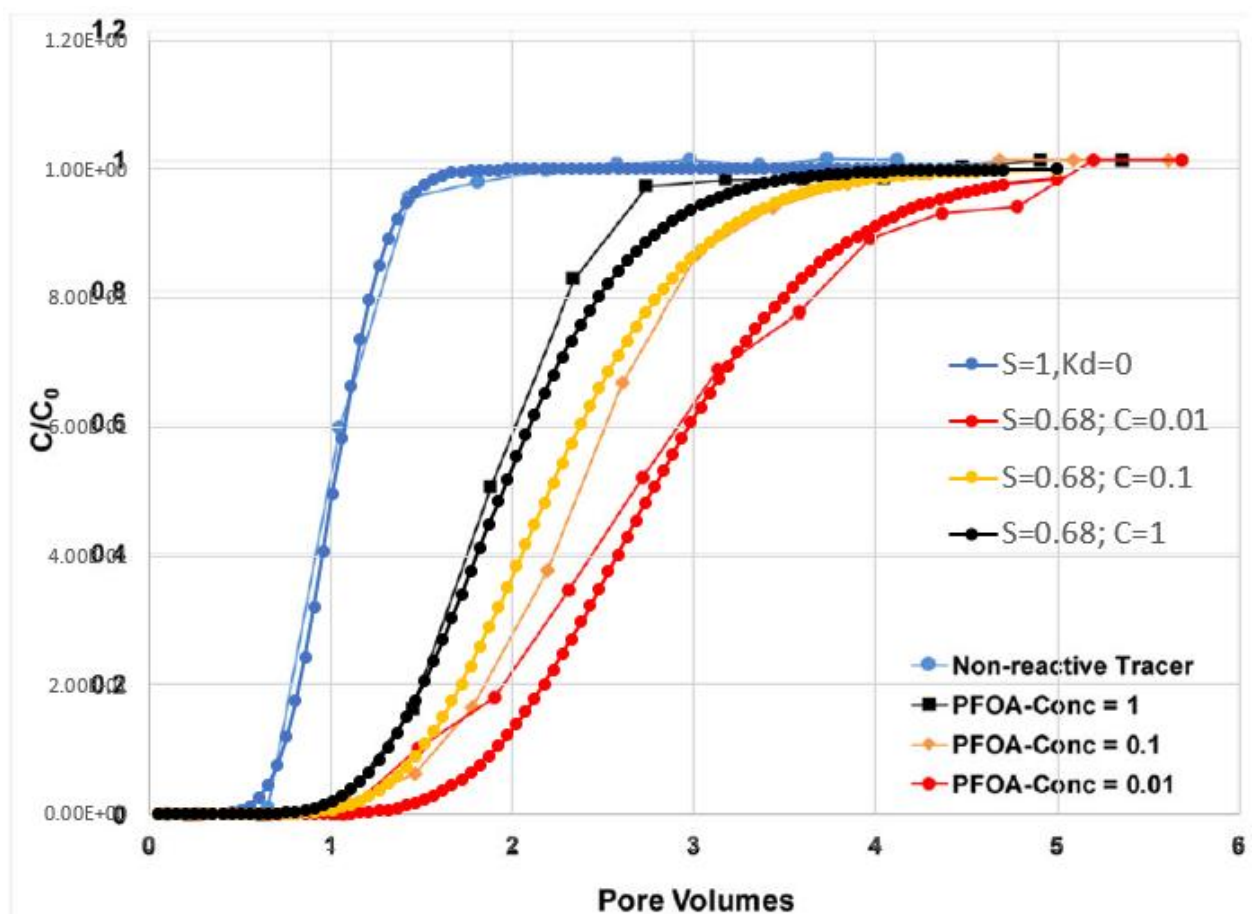


Figure **Ex 17**. Breakthrough of PFOA for different concentration cases compared to Lyu (2018) experimental results.

For all cases, it is noted that the simulations compare very well with the experimental results. Figure **Ex 18** shows a comparison of simulated results versus the laboratory observations for the different saturation conditions. Note that the case of saturation = 0.86 was not simulated, however, the simulation cases for both with and without dispersion are shown for the



unsaturated case. Again, it is noted that the simulations compare very well with the experimental results. The simulation case without soil adsorption is also shown on Figure **Ex 18**, to note the impact of the various adsorption mechanisms on the movement of PFOA. Retardation due to soil adsorption slows down the breakthrough of PFOA by less than half a pore volume, however, adsorption on the air-water interface slows it down further by more than one pore volume.

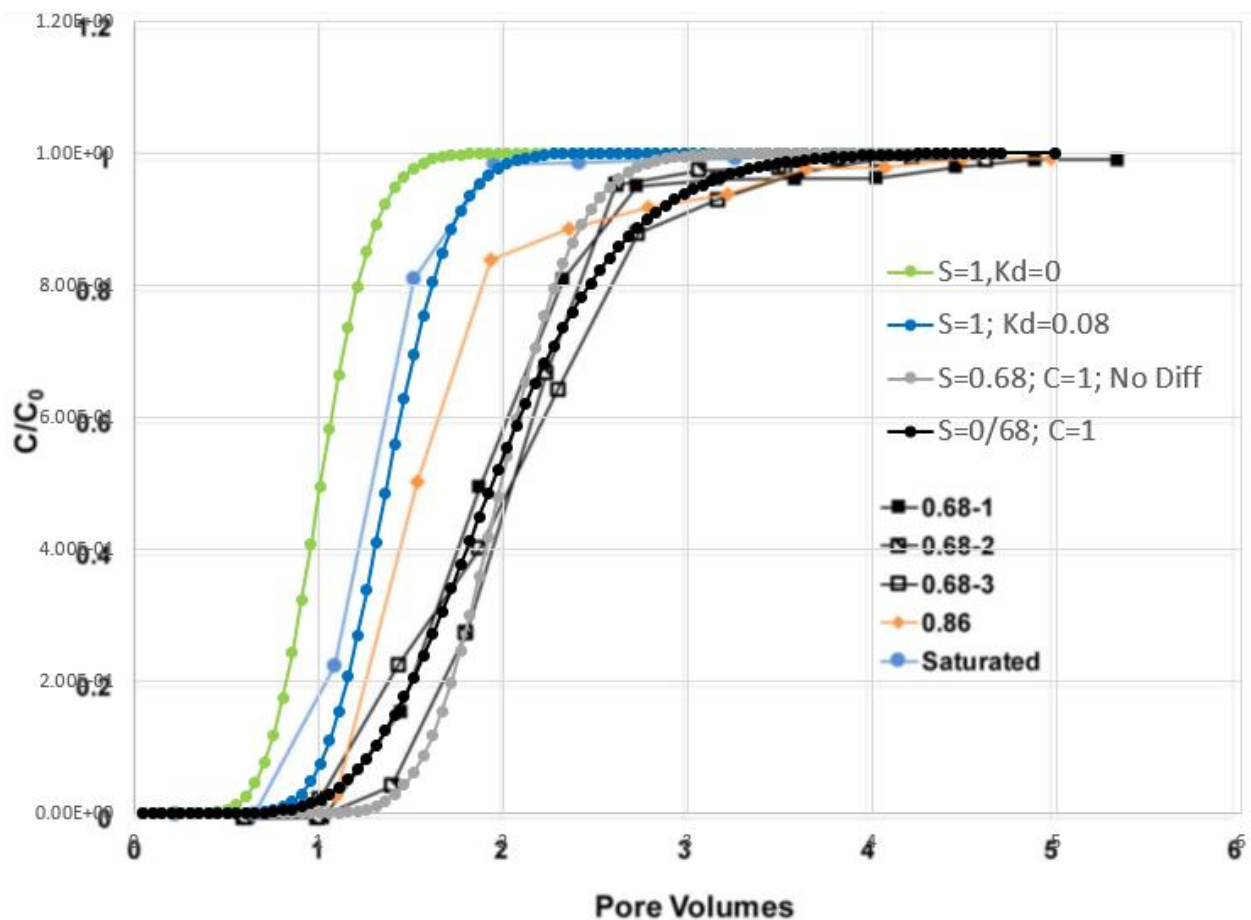


Figure **Ex 18**. Breakthrough of PFOA for different saturation cases compared to Lyu (2018) experimental results.

SUMMARY

Enhancements to the *MODFLOW-USG* software are currently implemented as *MODFLOW USG-Transport* and available as open source, public domain software courtesy of GSI Environmental. The enhancements include the BCT Process for solute transport simulations, which uses an unstructured-grid, Control Volume Finite Difference (CVFD) framework. The



model simulates three-dimensional solute transport for multiple chemical constituents in a heterogeneous steady-state or transient flow-field generated by the groundwater flow (GWF) and connected linear network (CLN) packages of *MODFLOW-USG*. Model capabilities, formulation and numerical methods have been presented in detail, and the features, capabilities and accuracy are demonstrated using example problems. The BCT package is a basic unstructured-grid transport model which can be the foundation for enhanced transport simulation capabilities compatible with *MODFLOW-USG*. The BCT package can also serve as the basis for implementing other numerical transport schemes into an unstructured grid framework.

Flow simulation enhancements are also included in *USG-Transport*. These include simulation of turbulent flow in the CLN domain, unsaturated flow above the water table, dual porosity flow considerations, various customized flow boundary conditions, and general convenient features to help with model efficiency and robustness.

REFERENCES CITED

- Bear, J., Nichols, E., Kulshrestha, A., Ziagos, J., 1994. Effect of Contaminant Diffusion into and out of Low-Permeability Zones. Lawrence Livermore National Laboratory Report UCRL-ID-115626.
- Bear, J., 2012. *Hydraulics of Groundwater*, Courier Dover Publications, 2012, ISBN: 0486136167, 9780486136165, 592 pages.
- Bear, J., 1988. *Dynamics of Fluids in Porous Media*, Dover Publications, Inc., NY.
- Bedekar, V., Morway, E.D., Langevin, C.D., and Tonkin, M., 2016, MT3D-USGS version 1: A U.S. Geological Survey release of MT3DMS updated with new and expanded transport capabilities for use with MODFLOW: U.S. Geological Survey Techniques and Methods 6-A53, 69 p., <http://dx.doi.org/10.3133/tm6A53>
- Burnett, R.D., and Frind, E.O., 1987. Simulation of contaminant transport in three dimensions, 2. Dimensionality effects, *Water Resources Research*, 23(4):695- 705.
- Chapman, S.W., B.L. Parker, T.C. Sale, and L.A. Doner, 2012. Testing high resolution numerical models for analysis of contaminant storage and release from low permeability zones, *Journal of Contaminant Hydrology*, 126-137: 106-116.



- Falta, R.W., and W. Wang, 2017. A Semi-Analytical Method for Simulating Matrix Diffusion in Numerical Transport Models, *Journal of Contaminant Hydrology* 197: 39-49.
- Forsyth, P.A., A.J.A. Unger, and E.A. Sudicky, 1998. Nonlinear iteration methods for nonequilibrium multiphase subsurface flow, *Advances in Water Resources*, 21, 433-449.
- Guo, W., and Langevin, C.D., 2002. User's Guide to SEAWAT: A Computer Program for Simulation of Three-Dimensional Variable-Density Groundwater Flow, *Techniques of Water-Resources Investigations* 6-A7.
- Harbaugh, A.W., 2005, MODFLOW-2005, the U.S. Geological Survey modular groundwater model -- the Groundwater Flow Process: U.S. Geological Survey Techniques and Methods 6-A16.
- Hect-Mendez J., N. Molina-Giraldo, P. Blum, and P. Bayer, 2010. Evaluating MT3DMS for Heat Transport Simulation of Closed Geothermal Systems, *Groundwater*, 48:5, 741-756, doi: 10.1111/j.1745-6584.2010.00678.x.
- Huang, K., B. P. Mohanty, F. J. Leij, and M. Th. van Genuchten. Solution of the nonlinear transport equation using modified Picard iteration. *Advances in Water Resources*, 21: 237-249, 1998.
- Hughes, J.D., Langevin, C.D., and Banta, E.R., 2017, Documentation for the MODFLOW 6 framework: U.S. Geological Survey Techniques and Methods, book 6, chap. A57, 40 p.
- Langevin, C.D., S. Panday, and A.M. Provost, 2020. Hydraulic-Head Formulation for Density-Dependent Flow and Transport, *Groundwater* 58(3), May-June 2020, P. 349-362. doi: 10.1111/gwat.12967.
- Langevin, C.D., Hughes, J.D., Banta, E.R., Niswonger, R.G., Panday, Sorab, and Provost, A.M., 2017, Documentation for the MODFLOW 6 Groundwater Flow Model: U.S. Geological Survey Techniques and Methods, book 6, chap. A55, 197 p., <https://doi.org/10.3133/tm6A55>.
- Langevin, C.D., Shoemaker, W.B., and Guo, W., 2003. MODFLOW-2000, the U.S. Geological Survey Modular Groundwater Model—Documentation of the SEAWAT-2000 Version with



- the Variable-Density Flow Process (VDF) and the Integrated MT3DMS Transport Process (IMT), U.S. Geological Survey Open-File Report 03-426, 43p.
- Lyu, Y., M.L. Brusseau, W. Chen, N. Yan, X. Fu, and X. Lin, 2018, Adsorption of PFOA at the Air-Water Interface during Transport in Unsaturated Porous Media, *Environ Sci Technol.* 2018 July 17; 52(14): 7745–7753. doi:10.1021/acs.est.8b02348.
- Merritt, M.L., and Konikow, L.F., 2000, Documentation of a computer program to simulate lake-aquifer interaction using the MODFLOW ground-water flow model and the MOC3D solute-transport model: U.S. Geological Survey Water-Resources Investigations Report 00-4167, 146 p.
- Muskus, N., and R.W. Falt, 2018, Semi-Analytical Method for Matrix Diffusion in Heterogeneous and Fractured Systems with Parent-Daughter Reaction, *Journal of Contaminant Hydrology* 218: 94-109.
- Panday, S., V. Bedekar, and C.D. Langevin, 2017. Impact of Local Groundwater Flow Model errors on Transport and a Practical Solution for the Issue, *Groundwater*, November 2017. doi: 10.1111/gwat.12627.
- Panday, Sorab, Langevin, C.D., Niswonger, R.G., Ibaraki, Motomu, and Hughes, J.D., 2013, MODFLOW-USG version 1: An unstructured grid version of MODFLOW for simulating groundwater flow and tightly coupled processes using a control volume finite-difference formulation: U.S. Geological Survey Techniques and Methods, book 6, chap. A45, 66 p.
- Panday, S., and P. S. Huyakorn, 2008. "MODFLOW SURFACT: A State-of-the-Art Use of Vadose Zone Flow and Transport Equations and Numerical Techniques for Environmental Evaluations." *Vadose Zone Journal*. Vol. 7, No. 2, pp. 610-631. May 2008.
- Peaceman, D.W., 1983. Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability, *Society of Petroleum Engineers Journal*, P 531-543.
- Pruess, K., and Y.S. Wu, 1993. A new semi-analytical method for numerical simulation of fluid and heat flow in fractured reservoirs, *SPE Advanced Technology Series* 1(2): 63-72.



Silva, J.A.K., J. Simunek, and J. E. McCray, A Modified HYDRUS Model for Simulating PFAS Transport in the Vadose Zone, *Water* 2020, 12, 2758; doi:10.3390/w12102758. www.mdpi.com/journal/water.

Vinsome, P., Westerveld, J., 1980. A simple method for predicting cap and base rock heat losses in thermal reservoir simulators. *J Can Pet Technol* 19.

Voss, C.I., and A.M. Provost, 2010. SUTRA version 2.2: A Model for Saturated-Unsaturated, Variable-Density Ground-Water Flow with Solute or Energy Transport, Water-Resources Investigations Report 02-4231.

Zheng, C., and Wang, P.P., 1999, *MT3DMS: A modular three-dimensional multi-species transport model for simulation of advection, dispersion and chemical reactions of contaminants in groundwater systems; Documentation and user's guide*: Contract report SERDP-99-1: U.S. Army Engineer Research and Development Center, Vicksburg, MS, 169 p.

LIST OF SYMBOLS

$2up$ = second point upstream cell (the upstream cell to grid-block ups), []

$a_{ijk m}$ = the geometric dispersivity of the medium (i,j,k,m = 1,2,3), [L]

a_I = dispersivity modulus along direction I, [L]

a_{IJ} = dispersivity modulus normal to plane IJ

a_L and a_T = longitudinal and transverse dispersivity values for isotropic dispersion, [L]

a_{Lx} , a_{Ly} and a_{Lz} = longitudinal dispersivity values for each of the respective coordinate directions for anisotropic dispersion, [L]

a_{Txy} , a_{Tyx} , and a_{Txz} = transverse dispersivity values for anisotropic dispersion, where a_{Tij} is the modulus for transverse dispersivity to "i" in the direction of "j" (which is equal to the modulus for transverse dispersivity to "j" in the direction of "i" due to symmetry) [L]

a_{Lxy} , a_{Lyz} , and a_{Lxz} = cross-direction longitudinal dispersivity moduli ($= a_{Lyx}$, a_{Lzy} , and a_{Lzx} respectively due to symmetry), [L]



a_{cc} = longitudinal dispersivity along the CLN cell, [L]

a_{MC} = longitudinal dispersivity between GWF and CLN cells, [L]

A_{nm} = perpendicular flow area of face between cells n and m, [L²]

c = concentration of a component species in water, [ML^{-3}], (in water in mobile domain, for dual domain simulation)

c_s = adsorbed concentration of component species, [M / M_s]

c_{sim} = adsorbed concentration of component species in the immobile domain [M / M_s]

c' = concentration of inflow water at a boundary, [ML^{-3}]

c_{im} = concentration of a component species in water in the immobile domain, [ML^{-3}]

c_M and c_C = species concentrations of the matrix and CLN cell respectively, [ML^{-3}],

c_n = concentration of component species in water for cell n, [ML^{-3}]

c_{sn} = concentration of component species in soil for cell n, [ML^{-3}]

c_{nm}^{ups} = concentration of the upstream grid-block between cells n and m, [ML^{-3}]

c_{nm}^{dwn} = concentration of the downstream grid-block between cells n and m, [ML^{-3}]

c_{nm}^{2up} = concentration of grid-block 2up, [ML^{-3}]

D_{ij} = hydrodynamic dispersion tensor, [L²/T]

D_{xx} , D_{yy} , and D_{zz} = principal components of the dispersion tensor, [L^2T^{-1}]

D^* = effective molecular diffusion coefficient in water, [L^2T^{-1}]

D_{CC} = longitudinal dispersion coefficient along the CLN cell, [L^2T^{-1}]

$\tilde{D}_{nm}$ = inter-cell dispersion conductance term between cells n and m resulting from the principal components of the dispersion tensor, [L^3T^{-1}]

D_n^{cross} = cross-dispersion flux at cell n, [$ML^{-3}T^{-1}$]



D_{nm} = inter-block dispersion term between cells n and m, [$ML^{-3}T^{-1}$]

e = nonlinear Freundlich exponent, []

fm = fraction of the volume that constitutes the mobile domain or the mobile domain fraction, []

fim = fraction of the volume that constitutes the immobile domain (fim = 1 - fm), []

$f(c_n)$ = smooth function that reduces the zero-order decay rate smoothly to zero when the concentration approaches zero, []

$f(Pe, \delta)$ = function which introduces the effect of transfer by diffusion between adjacent streamlines at a microscopic level in the general anisotropic hydrodynamic dispersion expression, []

h = hydraulic head, [L]

k_d = the adsorption coefficient, [L^3M^{-1}]

L_{CC} = length dimension of the CLN cell, [L]

LMC = length scale for interaction between the CLN and matrix cells taken as the hydraulic radius of the CLN cell plus the effective cell radius of the GWF cell, [L]

$[L_{nm} + L_{mn}]$ = perpendicular distance between cells n and m, [L]

$[L_{nm}^{ups} + L_{nm}^{2up}]$ = perpendicular distance between cells ups and 2up, [L]

L_{nm} and L_{mn} = perpendicular distances between the respective cell centers and the n-m interface, [L]

M = total mass per unit volume, of a component species in water and on soil, [ML^{-3}], (in water in mobile domain, for dual domain simulation)

Mim = total mass per unit volume, of a component species in water and on soil within the immobile domain [ML_{im}^{-3}]

$\dot{M}$ = source term for the component species, representing other source and sink processes, [$ML^{-3}T^{-1}$]



Pe = Peclet number, []

Q = fluid flux, [L^3T^{-1}]

Q_{nm} = net flux between cells n and m, [L^3T^{-1}]

S_w = saturation of water, [], (in water in mobile domain, for dual domain simulation)

S_{wim} = water saturation in the immobile domain []

t = time, [T]

Δt = time step size, [T]

v_i = Darcy flux in direction x_i , [L/T]

v_x , v_y , and v_z = components of the Darcy velocity vector along the x, y, and z axes, [LT^{-1}]

$\bar{v}$ = average velocity, [LT^{-1}]

V_s = fraction of the total volume of the CLN cell that is saturated during unconfined conditions, []

v_{cc} = velocity of flow along the CLN cell, [LT^{-1}]

v_{MC} = flux per unit area from CLN cell to GWF cell, [LT^{-1}]

V_n = volume of grid block n

ϕ_e = effective (transport) porosity, []

ϕ = total porosity, []

ϕ_{im} = effective porosity within the immobile domain, []

η_{nz} = the set of all faces that are normal to the horizontal direction

λ_w = first-order decay coefficient in water, [T^{-1}], = $\frac{\ln(2)}{t_{1/2w}}$ where $t_{1/2w}$ is the half-life of the chemical species in water



λ_s = first-order decay coefficient on soil [T^{-1}], = $\frac{\ln(2)}{t_{1/2s}}$ where $t_{1/2}$ is the half-life of the chemical species adsorbed on the soil

λ_{wim} = first-order decay coefficient within the immobile domain in water, [T^{-1}], = $\frac{\ln(2)}{t_{1/2wim}}$ where $t_{1/2wim}$ is the half-life of the chemical species within the immobile domain in water

λ_{sim} = first-order decay coefficient on soil, [T^{-1}], = $\frac{\ln(2)}{t_{1/2sim}}$ where $t_{1/2sim}$ is the half-life of the chemical species within the immobile domain adsorbed on the soil

μ_w = zero-order decay coefficient in water, [$ML^{-3}T^{-1}$]

μ_s = zero-order decay coefficient on soil, [$ML^{-3}T^{-1}$]

μ_{wim} = zero-order decay coefficient within the immobile domain in water, [$ML^{-3}T^{-1}$]

μ_{sim} = zero-order decay coefficient within the immobile domain on soil, [$ML^{-3}T^{-1}$]

θ_w = moisture content, []

θ_{wim} = moisture content in the immobile domain, []

θ_m = total mobile moisture content as used in MT3D, []

θ_{im} = total immobile moisture content as used in MT3D, []

ρ_b = bulk density of the porous medium, [M_s / L^3]

ρ_{bim} = bulk density of the porous medium within the immobile domain, [M_s / L^3]; Γ = mass transfer rate from mobile to immobile domain, [$ML^{-3}T^{-1}$]

$\sigma(r_{nm})$ = flux limiter [], which depends on the smoothness sensor, r_{nm} .

ζ = first-order mass transfer rate coefficient, [T^{-1}] which defines the diffusive exchange of species between mobile and immobile domains

Γ is the mass transfer coefficient from mobile to immobile domain, [$ML^{-3}T^{-1}$]

Γ_{MC}^* = mass exchange between the GWF cell and the CLN cell, [$ML^{-3}T^{-1}$]