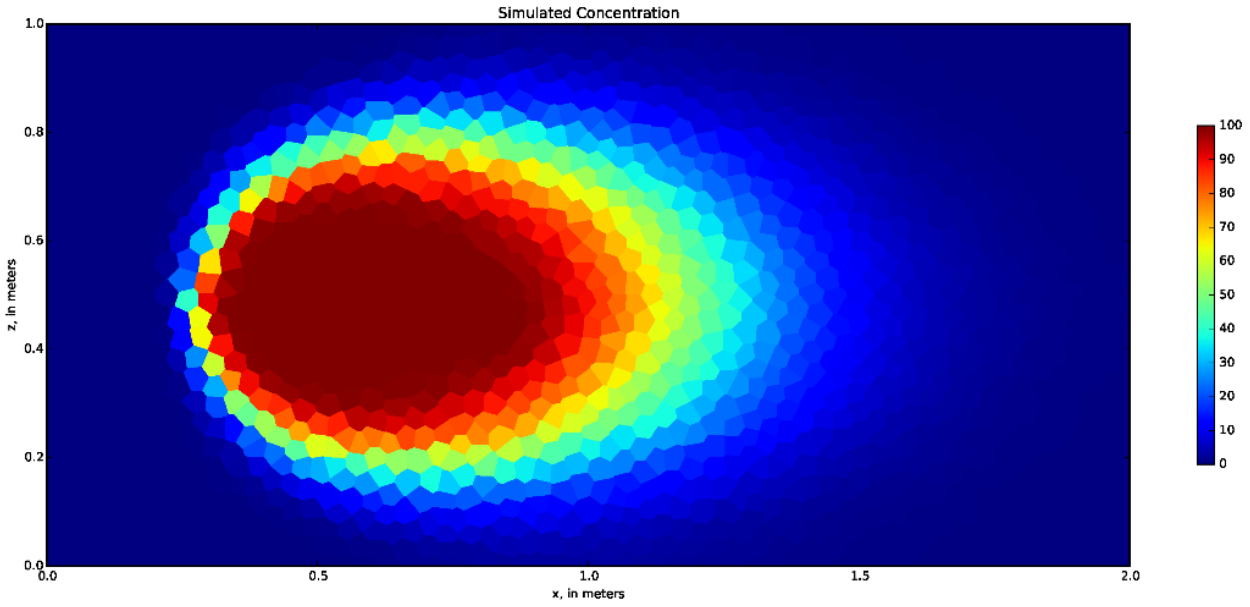
BLOCK-CENTERED TRANSPORT (BCT) PROCESS FOR MODFLOW-USG

Version 1.2.1



Author:

Sorab Panday, PhD

03 October 2018

GSI ENVIRONMENTAL

USG-Transport Version 1.2.1: The Block-Centered Transport (BCT) Process for MODFLOW-USG

By
Sorab Panday

Suggested Citations:

Panday, S., 2018; USG-Transport Version 1.2.1: The Block-Centered Transport Process for MODFLOW-USG, GSI Environmental, http:// 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.

Block-Centered Transport (BCT) Process for MODFLOW-USG Transport *(continued)*

Sorab Panday

TABLE OF CONTENTS

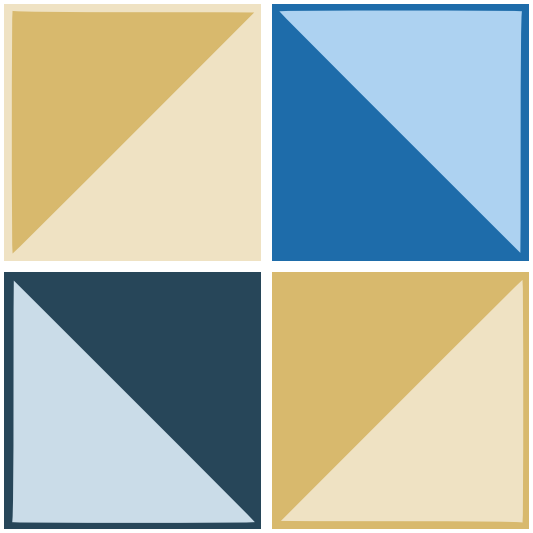
Block-Centered Transport (BCT) Process for MODFLOW-USG Transport *(continued)*

Sorab Panday

Tables

Figures

[This page intentionally left blank]



Block-Centered Transport (BCT) Process for MODFLOW-USG

Sorab Panday

PREFACE

The computer model described in this report (***USG-Transport, Version 1.2.0***) is designed to simulate heterogeneous, three-dimensional advective-dispersive chemical species transport with equilibrium and non-equilibrium retardation in the subsurface using an unstructured-grid, Control Volume Finite Difference (CVFD) framework. The transport module is developed as a process for the U.S. Geological Survey's (USGS) 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. The transport modules documented here serves as a basic transport model and can be the basis for enhanced transport simulation capabilities compatible with MODFLOW-USG.

The model is named the Block-Centered Transport (BCT) Process and is currently available in the transport version of MODFLOW-USG called USG-Transport. The code was designed and developed to be a seamless component of MODFLOW-USG. Testing indicated that the model 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.

The code for this model is currently available for downloading over the Internet from the GSI Website. The repository is accessible on the World Wide Web (www) from the web page at URL **http:// http://www.gsi-net.com/en/software/free-software/USG-Transport.html.** When the code is revised or updated in the future new versions or releases will be made available for downloading from this same site. Alternatively, the user can register at the GSI Website for emails that make users aware of code updates.

[This page intentionally left blank]

ACKNOWLEDGMENTS

Development efforts for USG-Transport were funded in part via a research project from GSI Environmental. Thanks to Christian D. Langevin and Alden M. Provost for their input to this work and preliminary review of the document. Thanks are also due to Anthony (Tony) Daus for supporting this effort.

Disclaimer

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

[This page intentionally left blank]

ABSTRACT

This report presents a transport model simulator for unstructured grids. The 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.

The transport model is fully compatible with, and integrated into MODFLOW-USG. MODFLOW-USG (Panday et al., 2013) 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.

The model uses an implicit TVD scheme to solve the conservative form of the transport equation on the basis of fluxes computed by the flow modules of MODFLOW-USG for a given time step. The conservative form of the equation provides for mass conserved solutions, the implicit time-stepping scheme eliminates time-step size restrictions, and the TVD scheme minimizes numerical dispersion without introducing oscillatory errors in the solution. A scheme has

also been provided to optionally trap localized flow imbalance errors and evaluate associated transport errors.

This report includes a description of the theoretical basis of the model, the numerical methods, the data-input requirements and output options, results of benchmark simulation problems, and an input instructions guide. The model 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 are 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 describes and documents a computer 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 model, called the Block-Centered Transport (BCT) Process for MODFLOW-USG, is fully integrated within the MODFLOW-USG Groundwater Flow model. Other processes are also available with this version, called USG-Transport. 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 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 concentration of a chemical species 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; and (5) 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, fully-implicit control volume finite difference (CVFD) scheme is used to solve the divergence form of the advection-dispersion transport equation with decay and retardation of chemicals – solution to the convective

form of the transport equation need not conserve mass. The fully implicit scheme is unconditionally stable and does 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). Also, the transport simulation immediately follows solution for flow at a given time-step.

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* includes the capability to simulate dual porosity processes so associated transport 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 with 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 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. As assumed by MODFLOW-USG, it is also assumed by the BCT process that fluid properties are homogeneous and that concentration changes do not significantly affect the fluid density or viscosity which may in turn affect the fluid velocity; 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 is 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. The code further serves as a basis for additional reactions and processes that may be included in the future, more complex nonlinear reversible reactions to simulate complex geochemical or biological processes, or multi-porosity transport mechanisms to accommodate varying degrees of sub-grid scale heterogeneities.

The report includes a detailed description of the formulation and numerical methods used to solve the solute-transport equation 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 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).

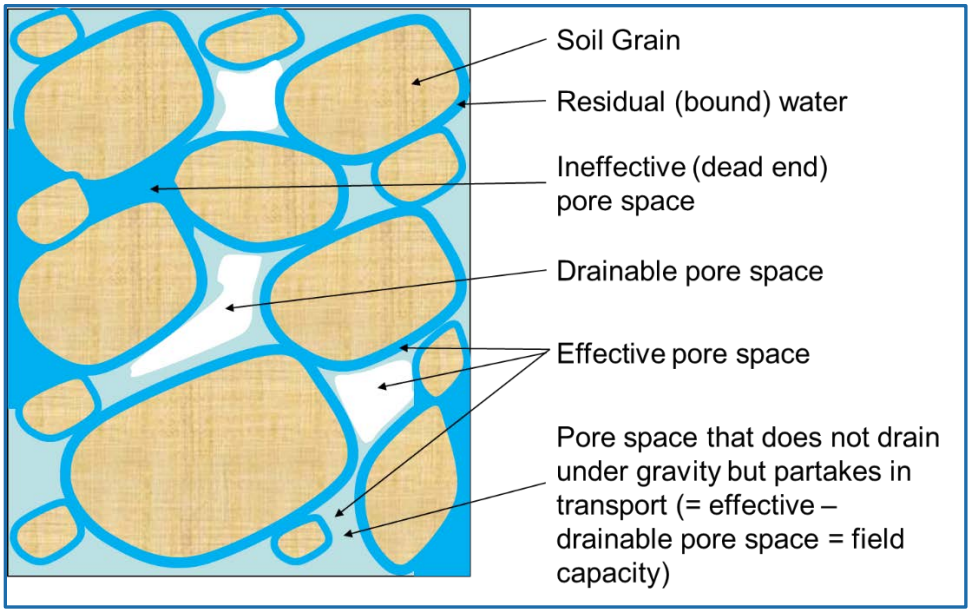


Figure 1. *Porosity Definitions*

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:

$$\phi = \phi_e + \phi_{ie} = S_y + \phi_{fc} + \phi_{ie} \tag{1}$$

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.

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 processes and reactions simulated by the BCT process as:

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

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

λ_s is the first-order decay coefficient on soil $[T^{-1}]$, = $\frac{\ln(2)}{t_{1/2s}}$ 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 $[M / (L^3 T)]$, 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 volatization 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:

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

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:

$$\theta_w = \phi_e S_w \tag{4}$$

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:

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

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:

$$\theta_w = S_y S_w + \phi_{fc} \tag{6}$$

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:

$$\frac{\partial(M)}{\partial t} = \phi_e \frac{\partial(c)}{\partial t} + S_s \frac{\partial(hc)}{\partial t} + \frac{\partial(\rho_b c_s)}{\partial t} \tag{7}$$

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:

$$\theta_w = \phi_e \tag{8}$$

Note that the head is a reference head and thus the term $S_s hc$ 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:

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

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.

It is interesting to also note that expansion of the compressible storage term (third term on the right hand side of equation 9 or second term on the right hand side of equation 7 with $S_w = 1$) yields:

$$S_w S_s \frac{\partial(hc)}{\partial t} = S_w S_s h \frac{\partial(c)}{\partial t} + \left[S_w S_s \frac{\partial(h)}{\partial t} \right] c \tag{10}$$

The term within square brackets on the right hand side of equation 6 represents the fluid flux that goes into or out of compressible storage which, when multiplied by the concentration is the mass flux into and out of compressible storage (the second term on the right hand side of equation 10). If this is provided as a boundary flux to each cell to implement confined storage effects of transient flow, it would still be missing the first term on the right hand side that considers concentration contrasts with previous water in storage.

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

$$c_s = k_d c^e \tag{11}$$

where

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

e is the nonlinear Freundlich exponent [ ].

Note that a linear adsorption isotherm is obtained by setting the exponent *e* to unity. Also 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.

Dispersion

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

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

where

a_{ijkm} is the geometric dispersivity of the medium, (*i, j, k, m = 1,2,3* for the three principal coordinate directions), $\bar{v}$ 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):

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

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 Kroneker 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:

$$a_{1111} = a_{2222} = a_{3333} = a_L \tag{14}$$

$$a_{1122} = a_{2211} = a_{1133} = a_{3311} = a_{2233} = a_{3322} = a_T \tag{15}$$

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

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:

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

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^2T^{-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^2T^{-1}, and

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

$$|v| = \sqrt{v_x^2 + v_y^2 + v_z^2} \tag{21}$$

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

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

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

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

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.

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 mass transfer rate from mobile to immobile domin [$ML^{-3}T^{-1}$], $q_{m,im}$ 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.

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}], = $\frac{\ln(2)}{t_{1/2wim}}$ where $t_{1/2wim}$ is the half-life of the chemical species within the immobile domain in water,

λ_{sim} is the 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_{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 [$M_{a,im} / (L_{im}^3 T)$].

Note that for situations of transport in a transient flow-field, an immobile domain flow equation with transfer of water between the domains is also required to quantify fluid flux exchange between domains.

If the immobile domain is not part of the flow solution for a transient dual domain transport situation (which is the case in many analyses), an assumption

is made that there is no flow between mobile and immobile domains, and thus an immobile domain flow equation is not required. In that case, the change of saturation or compressible storage within the immobile domain is zero and therefore the mass storage term for the immobile domain can be treated simply for species concentrations in water and on soil, and the special *MODFLOW* considerations of equations 5 through 9 are not required for the immobile domain giving:

$$\frac{\partial(M_{im})}{\partial t} = \frac{\partial(\phi_{im} S_{wim} c_{im})}{\partial t} + \frac{\partial(\rho_{bim} c_{sim})}{\partial t} \tag{27}$$

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

$$\theta_{wim} = \phi_{im} S_{wim} \tag{28}$$

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.

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

$$\Gamma = \varsigma(c - c_{im}) \tag{29}$$

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

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:

$$\theta_m = f_m \phi_e \tag{30}$$

And

$$\theta_{im} = f_{im} \phi_{im} \tag{31}$$

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.

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:

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

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:

$$D_{CC} = a_{CC} v_{CC} + V_s D^* \tag{33}$$

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:

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

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).

Density Dependent Flow and Transport Formulation

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*.

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:

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

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:

$$q = -(k/\mu)[\nabla(p + \rho g z) - gz\nabla\rho] \tag{36}$$

The hydraulic head can be defined as

$$h_\phi = p/(\rho g) + z \tag{37}$$

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

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

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:

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

Which is also written as:

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

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:

$$q = -K_o\left[\left(\rho/\rho_0\right)\nabla h_\varphi\right] - K_o\left(h_\varphi - z\right)\nabla\left(\rho/\rho_0\right) \tag{41}$$

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):

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

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:

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

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.

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

NUMERICAL TREATMENT

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 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. 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):

$$\frac{S_y \forall_n}{\Delta t}\left[S_w c_n^{\,t+\Delta t} - S_w c_n^{\,t}\right] + \frac{\phi_{fc} \forall_n}{\Delta t}\left[c_n^{\,t+\Delta t} - c_n^{\,t}\right] + \frac{S_w S_s \forall_n}{\Delta t}\left[hc_n^{\,t+\Delta t} - hc_n^{\,t}\right] + \frac{\forall_n}{\Delta t}\rho_b\left[c_{sn}^{\,t+\Delta t} - c_{sn}^{\,t}\right] =$$
$$\sum_{m\in\eta_n} \theta_w \tilde{D}_{nm}(c_m - c_n) - \sum_{m\in\eta_n} Q_{nm} c_{nm} - \forall_n[\lambda_w \theta_w c_n + \lambda_s \rho_b c_{sn}] - \forall_n[\mu_w \theta_w f(c_n) + \mu_s(1-\phi_e) f(c_n)] +$$
$$\theta_w D_n^{cross} + \dot{M} \tag{44}$$

where

$\forall_n$ 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.

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):

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

where

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

c_{nm}^{dwn} 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):

$$\sigma(r_{nm}) = \begin{matrix} 0 & if & r_{nm} \le 0 \\ \frac{2r_{nm}}{1+r_{nm}} & if & r_{nm} > 0 \end{matrix} . \quad (46)$$

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

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

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*,

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

$\left[L_{nm}^{ups} + L_{nm}^{2up}\right]$ 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,

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

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 2** for the definition of the geometric terms of equation (48) for various grid types.

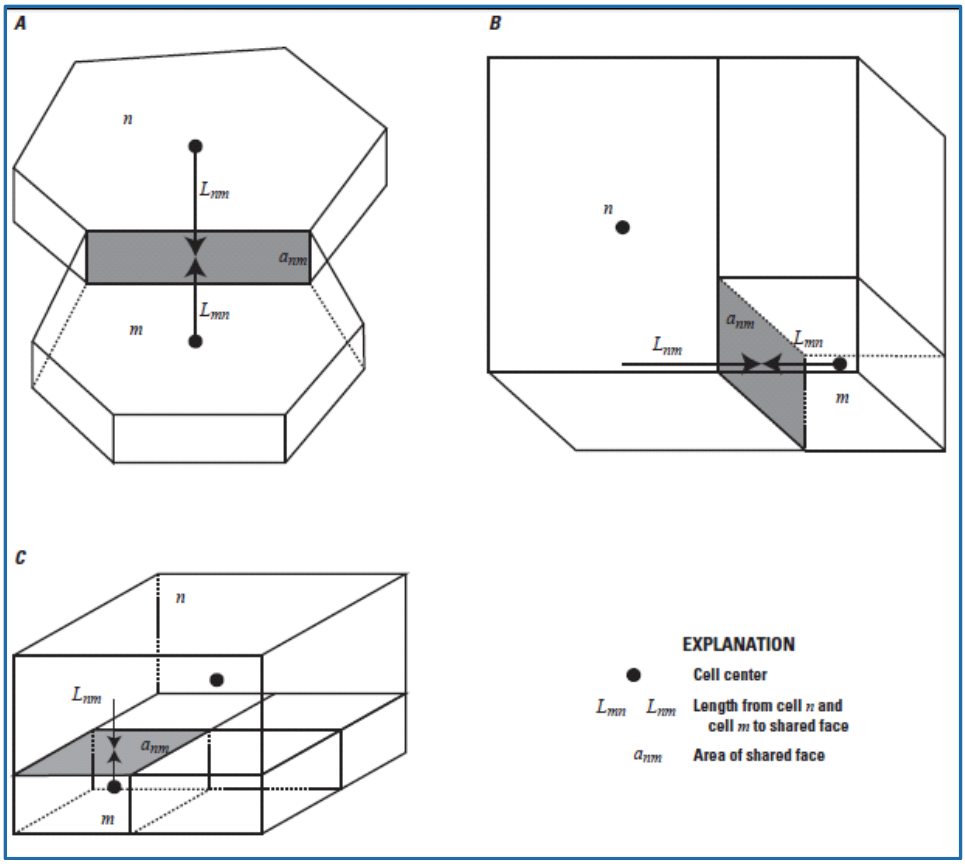


Figure 2. *Definition of Geometrical Properties for Various Grid Types*

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 3** 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.

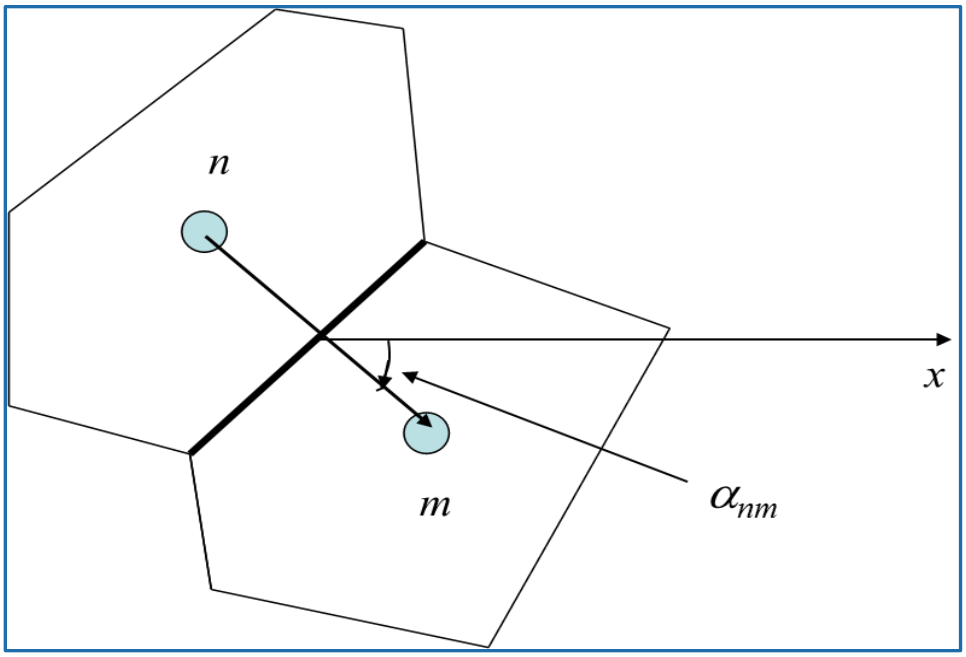


Figure 3. *Angle between x-coordinate direction and the perpendicular to face nm*

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 absolute value of the cosine of the angle between the perpendicular to the face and the x-coordinate direction (**Figure 4**).

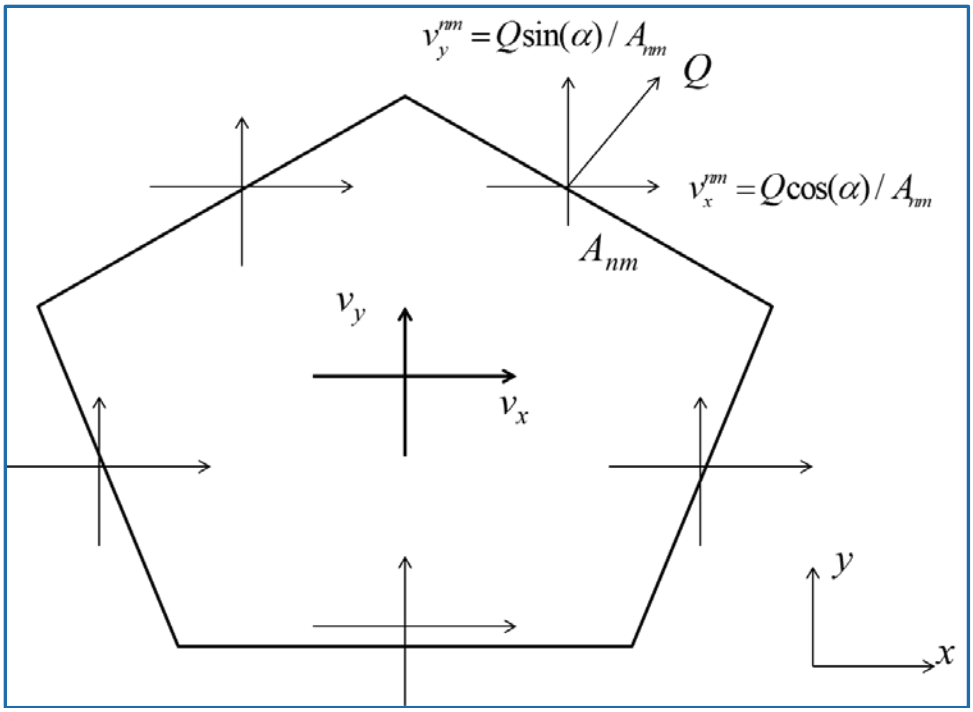


Figure 4. *Schematic for computing the average x and y- direction velocities of a cell.*

The weighting in the y-direction is performed as per the absolute value 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:

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

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

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

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**):

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

And

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

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

$$v_z = \sum_{m\in\eta_{nz}} v_z^{mn} / \sum_{m\in\eta_{nz}} m \in \eta_{nz} \tag{53}$$

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

$$v_z^{mn} = Q_{nm} / A_{nm} \tag{54}$$

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 (see **Figure 5**), as:

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

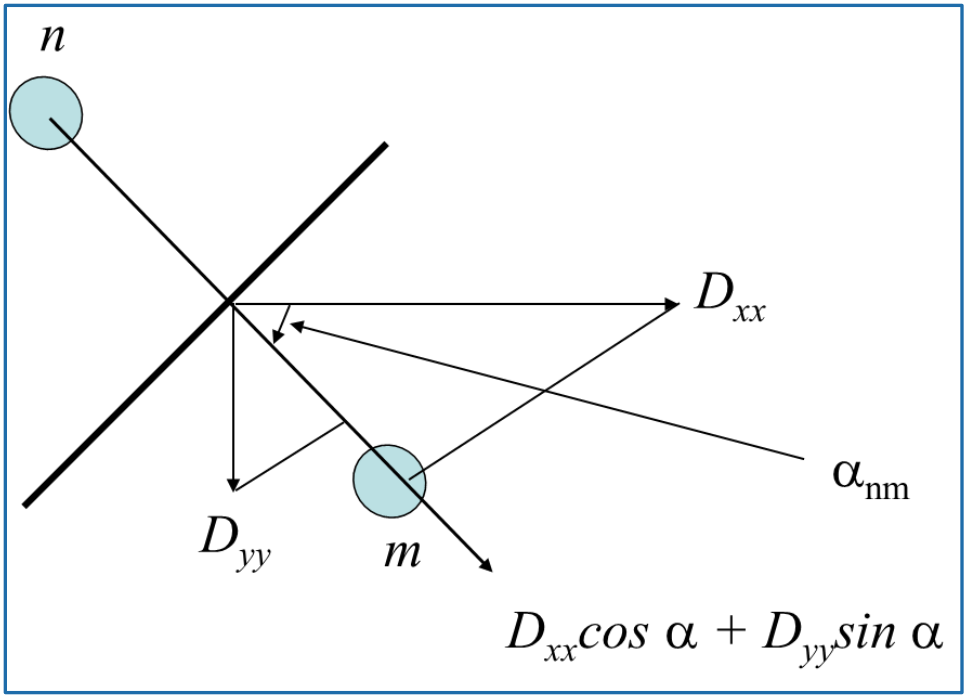


Figure 5. *Principal dispersion components of cell m across face nm*

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

$$D_{nm} = D_{zz} \tag{56}$$

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:

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

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:

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

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 6**). 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.

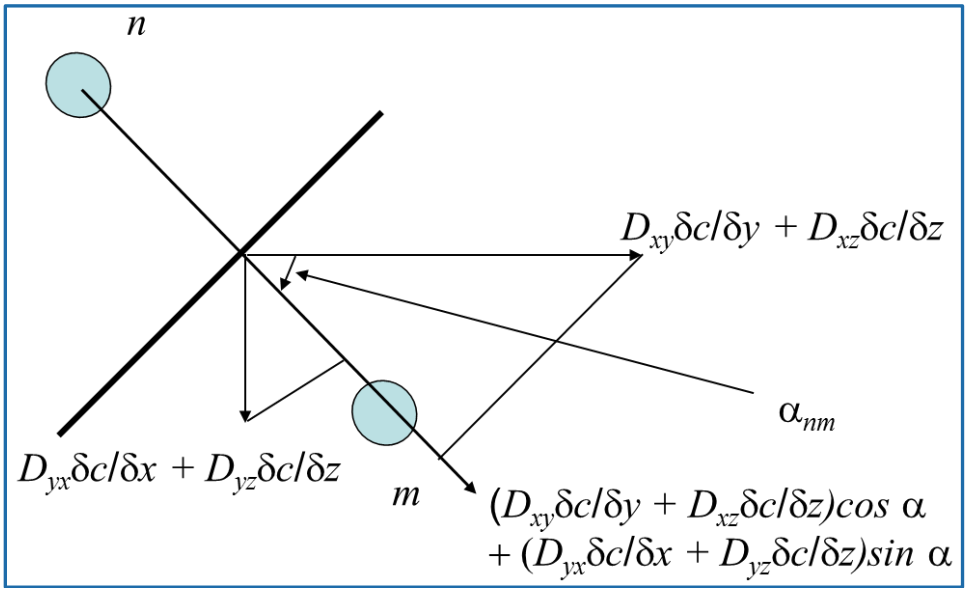


Figure 6. *Cross dispersion components across face nm*

The terms $\frac{\partial c}{\partial x}$, $\frac{\partial c}{\partial y}$ and $\frac{\partial c}{\partial z}$ are computed for each cell as follows (see **Figure 7**). 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

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

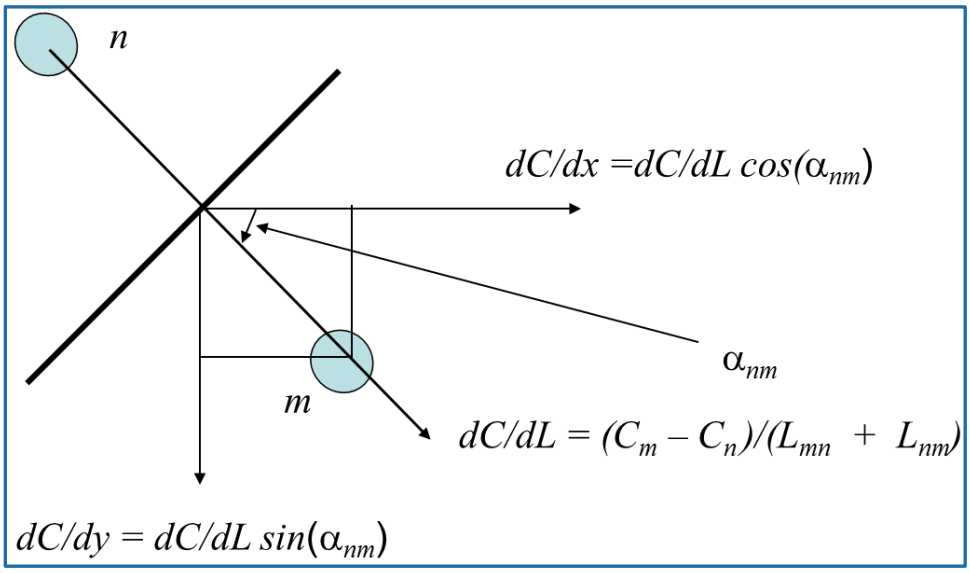


Figure 7. *dc/dx and dc/dy for each face*

The x- and y-components of this gradient are resolved as per **Figure 7**, to give:

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

and :

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

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,

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

and

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

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

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

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 KT3D (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 $-K_o\left(h_\varphi - z\right)\nabla\left(\rho/\rho_0\right)$ 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 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. Therefore, 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 the 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.

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

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 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. 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.

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 on **Figures 8,** and **9**. As noted on **Figure 8**, 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.

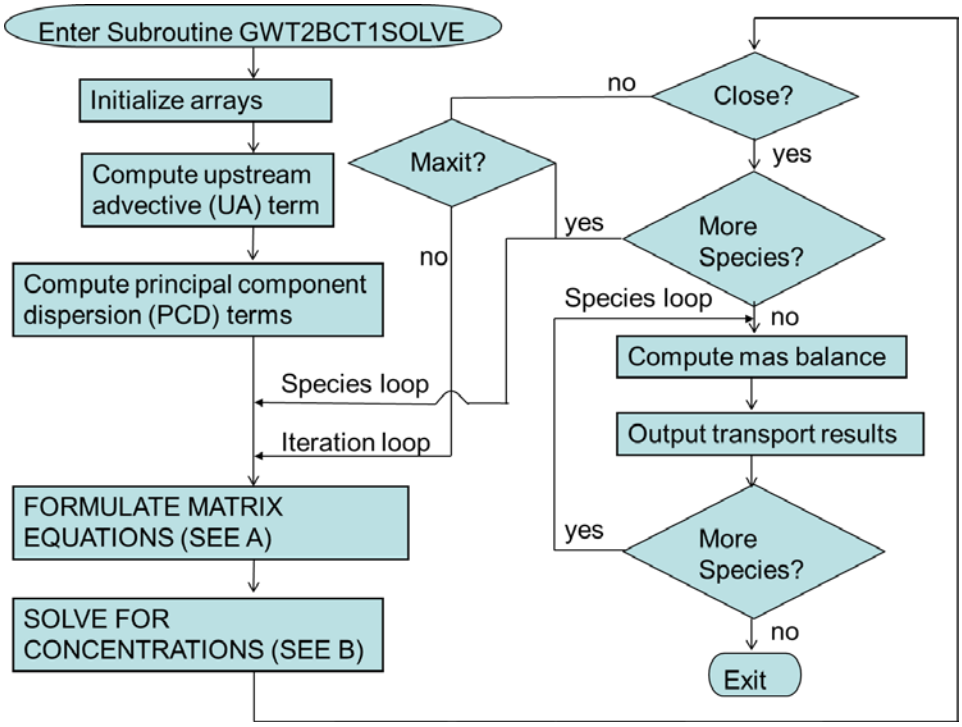


Figure 8. *Flowchart for Subroutine GWT2BCT1solve*

The flowchart for the formulation of the discretized transport equation is shown on **Figure 9**. 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 computation may be skipped for such conditions. After filling the upstream weighted terms, 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 matrix equation is then 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.

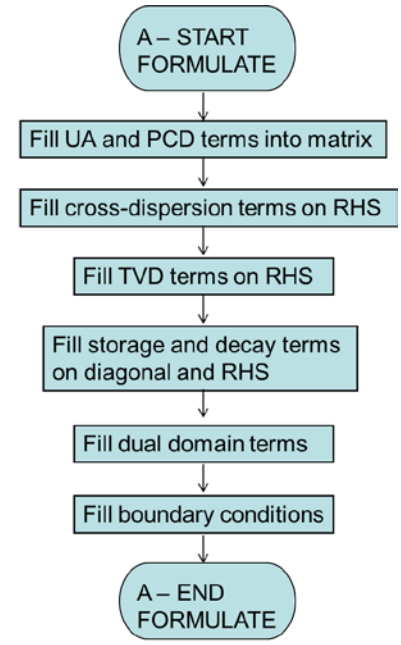


Figure 9. *Flowchart for Subroutine Formulate*

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, 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 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.

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.

The concentrations supplied in the input file are multiplied by the fluid flux at inflow nodes, to provide the respective 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.

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. 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. The binary output files use the same utility programs as used by the flow modules and hence output formats are identical. Component species mass balance output for the various supported flow boundary types 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 time-scales 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. 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 time-scale. However, field conditions may fluctuate with velocities and flow directions 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 is also critical in obtaining accurate transport behavior for complex field situations.

Solution Options

The BCT package input files provides 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. Finally, if density driven flow is simulated, the DDF input file provides the necessary input.

The physical process flags indicate 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; and whether first order or zeroth order decay are simulated and if so, whether that occurs in water, soil, or both.

Flags associated with simulation speed and accuracy include one to set the active transport domain, the cross-dispersion option flag, a TVD flag, and the number of iterations for obtaining the transport solution. These items are addressed 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 and DPT packages to inactive unwanted portions of the respective transport domain. 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 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. 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 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.

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.

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 orthogonolizations, etc. are currently the same for both flow and transport solutions.

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 feet (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 6.9315x 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 Δt = 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 11** 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 10** 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.

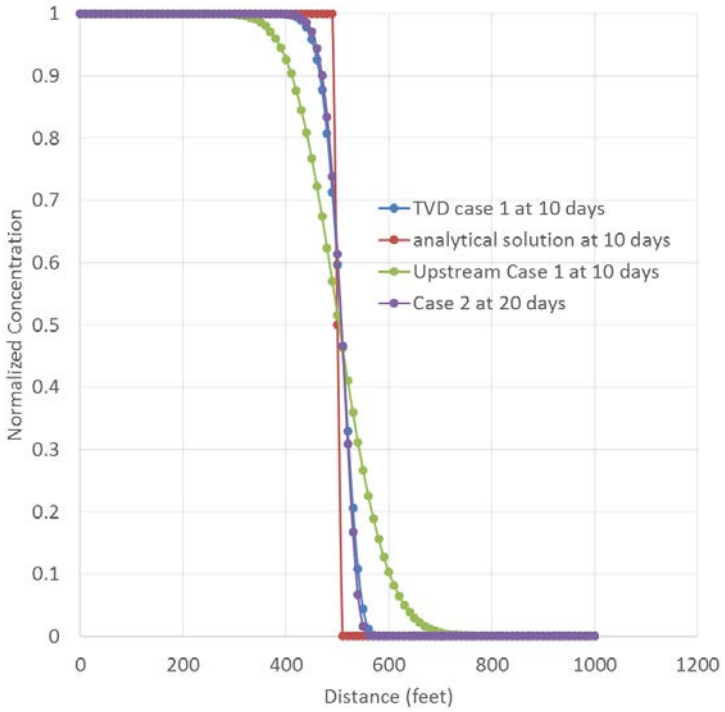


Figure 10. *Simulation Results for Advection in a One-Dimensional, Uniform Flow Field*

Results for the simulation cases 3, 4 and 5 are shown on **Figure 11** 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). 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.

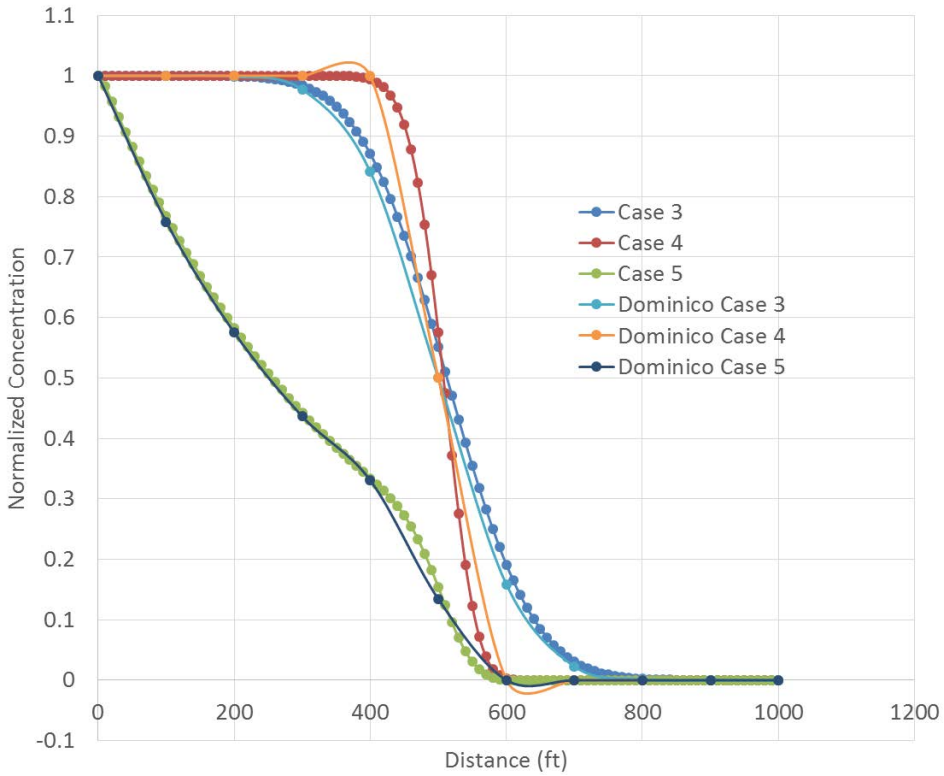


Figure 11. *Simulation Results at 20 Days for Advection, Dispersion, and Decay in a One-Dimensional, Uniform Flow Field*

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 ft^3/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 12** 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.

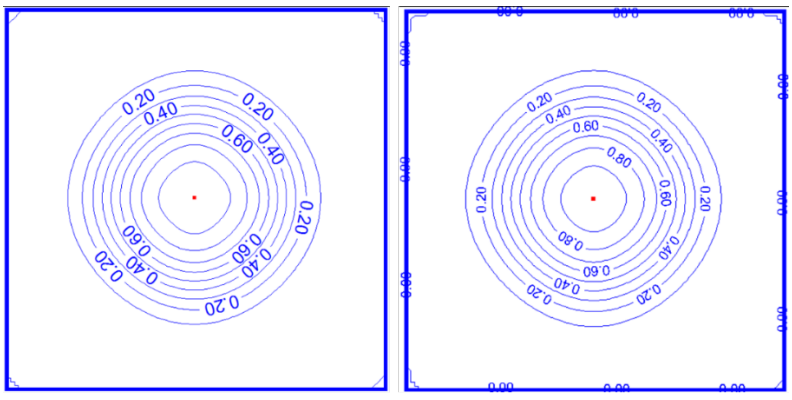


Figure 12. *Comparison of Model Results with MT3D Simulation Results for Transport in a 2-D Radial Flow Field at 5,000 days*

The same situation was also simulated without the optional cross-dispersion term activated. **Figure 13** 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.

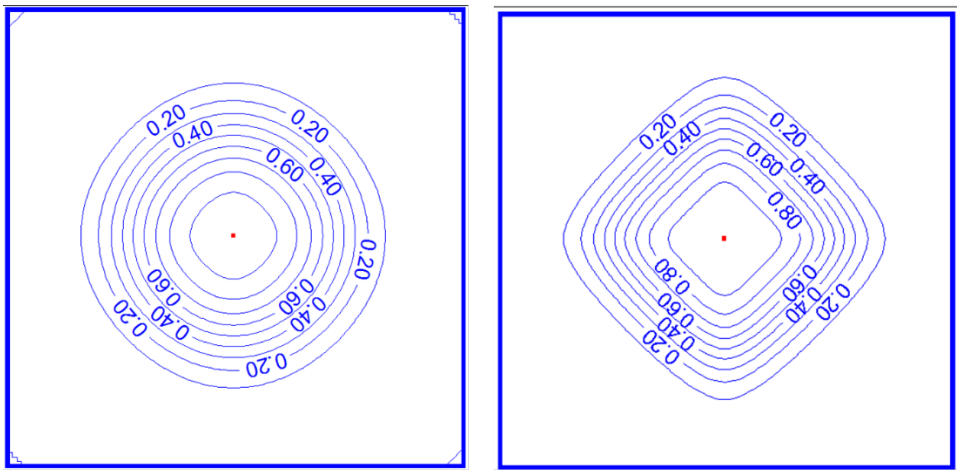


Figure 13. *Comparison of Simulation Results with and without Cross-Dispersion for Transport in a 2-D Radial Flow Field at 5,000 days*

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.

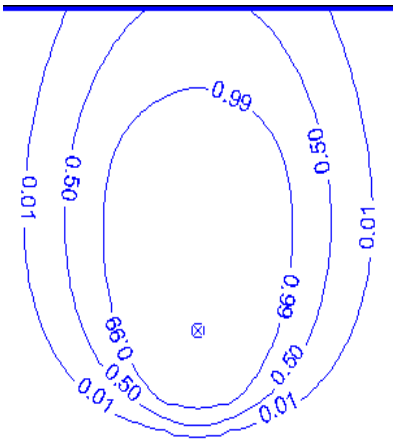


Figure 14. *Simulation Results in Layer 1 at 3,000 days for Transport through Conduit Example*

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 14** 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 15** 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 14**.

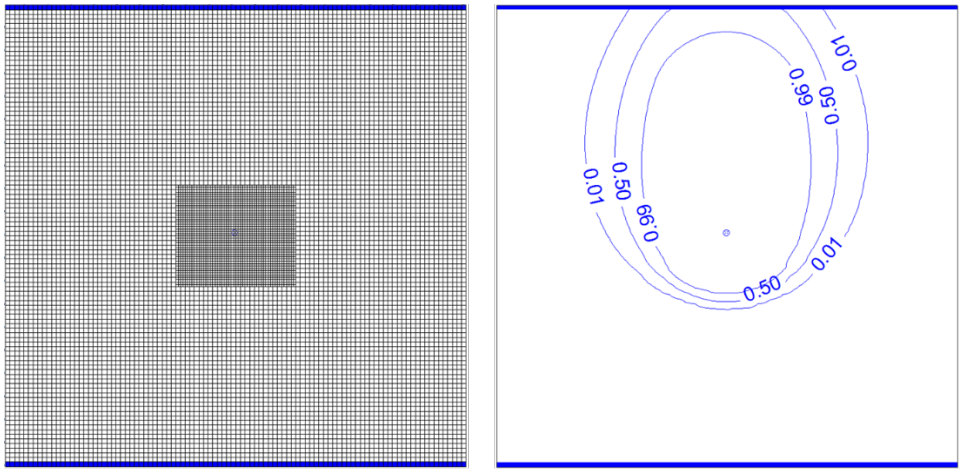


Figure 15. *Simulation Results in Layer 1 at 3,000 days for Transport through Conduit Example, Using a Nested Grid*

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 0.5 L/kg, and a mass transfer rate of 0.1 mg/L/day. The concentration of water in both mobile and immobile domains is zero at the start of the simulation.

A transport simulations 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 10 days. Subsequently, the concentration of inflow water was made to zero for a period of 20 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 16 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.

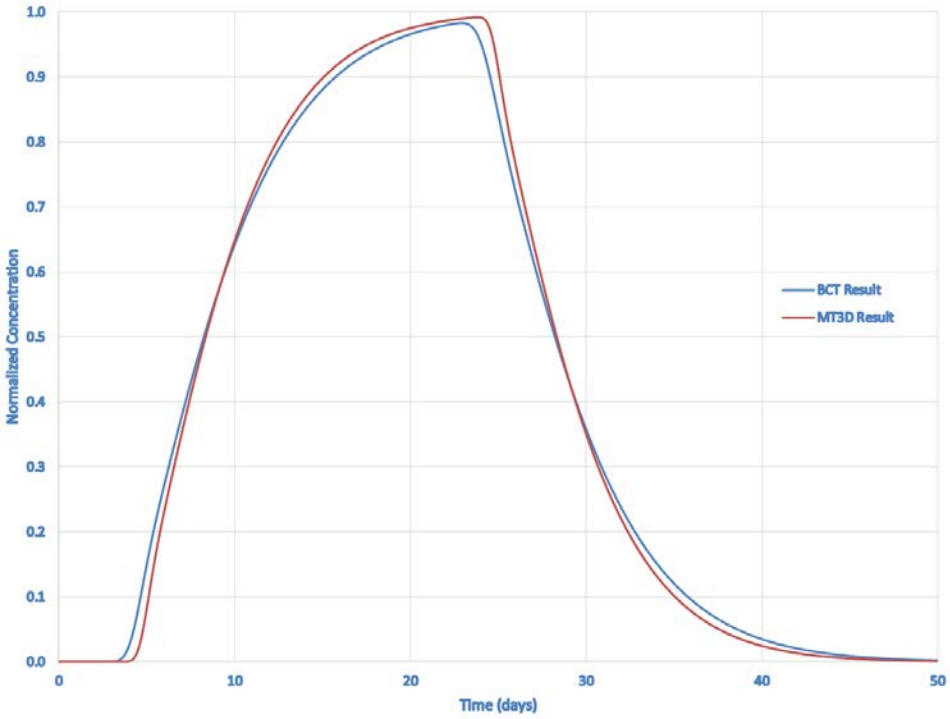


Figure 16. *Concentration at Outlet for Dual Domain Transport in a One-Dimensional, Uniform Flow Field*

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 (Qin) of 5.702m^3/d per meter with a concentration (Cin) 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 m^3/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 17 shows the results after 1 day, from the SEAWAT and the USG-Transport simulations. The results are virtually identical.

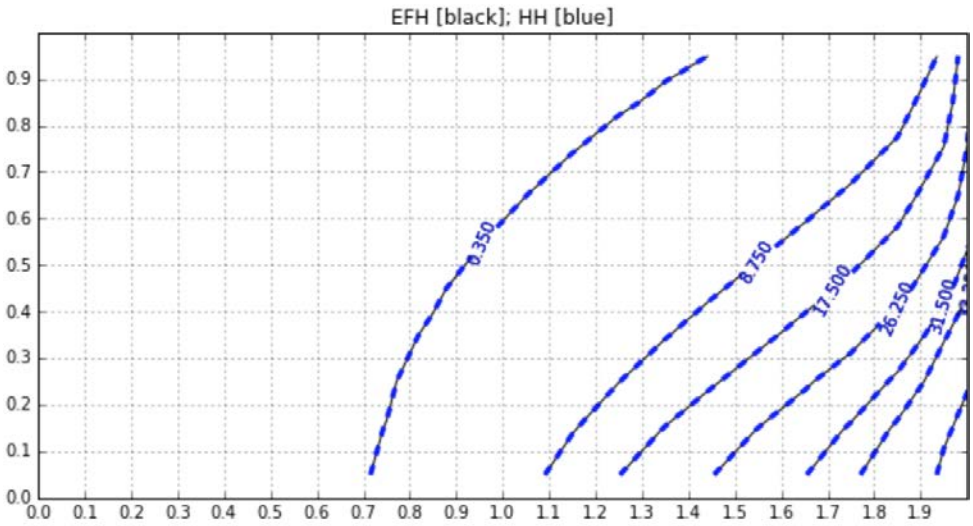


Figure 17. *Concentration profiles for Henry Problem Simulation*

The contribution of Christian D. Langevin is acknowledged in providing this example simulation problem.

SUMMARY

A Block Centered Transport (BCT) Process has been developed for *MODFLOW-USG* and is currently implemented as the *USG-Transport* version of the code. The BCT Process 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.

REFERENCES CITED

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.

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.

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.

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.

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.

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.

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_{ijkm} = 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_{Tyz}, 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}]

cim = concentration of a component species in water in the immobile domain, [ML^{-3}]

cM and cC = 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]

vi = Darcy flux in direction xi, [L/T]

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

$\overline{v}$ = 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}]

$\forall_n$ = 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 t1/2w 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 t1/2 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 t1/2wim 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 t1/2sim 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}.

ς = 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, [$ML^{-3}T^{-1}$]

DESCRIPTION OF INPUT FILES

A Block Centered Transport (BCT) process has been developed using a Control Volume Finite Difference (CVFD) scheme compatible with the flow solution of MODFLOW-USG. Input file formats and structures are also similar to those of the MODFLOW-USG flow packages, and options are available to enter data for structured finite-difference grids or for general unstructured grids. The unstructured grid input are highlighted by use of a blue font to delineate this information for convenience.

A BCT package input file is read for every transport simulation. The BCT package input includes all storage, decay, and transport related parameters necessary to conduct a species transport simulation. A boundary condition file is also included for prescribed concentration boundaries. For flow related boundaries, a third type boundary condition is applied by supplying the concentration of inflowing water as auxiliary variables in the respective boundary package input files. For outflow nodes, this concentration is not required and is ignored, if input. Further details on input of auxiliary variables for concentrations within the respective boundary files are provided in the sub-section titled “Model Input and Output” under the “Implementation and Program Design” section. Additional input required for transport simulations are discussed below.

Block Centered Transport Package Input Instructions

Input for the Block Centered Transport (BCT) Package is read from the file that is type “BCT” in the Name File. The BCT Package input instructions include flags, indices, transport related parameters and initial conditions for solution to transport of contaminants in a steady-state or transient flow field. The input is read in a free format.

FOR EACH SIMULATION

1a. ITRNSP IBCTCB MCOMP ICBNDFLG ITVD IADSORB ICT CINACT CICLOSE IDISP IXDISP DIFFNC IZOD IFOD IFMBC

These twelve variables are free format if the option "FREE" is specified in the Basic Transport Package input file; otherwise, the variables all have 10-character fields.

Read item 1b below, ONLY if IFMBC = 1 indicating that flow mass balance errors will be evaluated and incorporated into the transport solution. The data in item 1b below provides unit numbers for output of the results. Note that the NAME file should also include these unit numbers and open appropriate files for the GW and CLN domains. Suggested extensions for the files are FMG (flow mass balance for groundwater domain), TMG (transport mass balance for groundwater domain), FMC (flow mass balance for CLN domain), and TMC (transport mass balance for CLN domain).

1b. MBEGWUNF MBEGWUNT MBECLNUNF MBECLNUNT

If IFMBC in item 1a above is zero then item 1b is not read.

If UNSTRUCTURED option is used then read items 2 through 20.

A subset of the following one-dimensional variables is used to describe each node. The variables needed for each node depend on the transport options that are selected. Unneeded variables must be omitted.

The required variables for each of Items 2 – 4 and items 6 – 20 for layer 1 are read first; then for layer 2 and so forth. The next variable of Items 2 – 4 and items 6 – 20 is read after all layers for the previous variable are read. Note that there may be different number of nodes per layer (NDSLAY) for an unstructured grid.

If ICBNDFLG is zero then read item 2.

2. **[ICBUND(NDSLAY)] – U1DINT. Read one array for each layer till all layers are read.**

3. **[PRSITY(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

If IADSORB is not zero then read item 4.

4. **[BULKD(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

If IDISP is not zero then read item 5.

5. **[ANGLEX(NJA)] – U1DREL.**

If IDISP is equal to one then read items 6 and 7.

6. **[DL(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

7. **[DT(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

If IDISP is equal to two then read items 8, 9, 10, 11, 12 and 13.

8. **[DLX(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

9. **[DLY(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

10. **[DLZ(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

11. **[DTXY(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

12. **[DTYZ(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

13. **[DTXZ(NDSLAY)] – U1DREL. Read one array for each layer till all layers are read.**

Items 14 – 20 are read for the first component, followed by the next, till all MCOMP contaminant species are read.

If IADSORB is not zero then read item 14.

14. **[ADSORB(NDSLAY,ICOMP)] – U1DREL. Read one array for each layer till all layers are read.**

If IADSORB is two then read item 15.

15. **[FLICH(NDSLAY,ICOMP)] – U1DREL. Read one array for each layer till all layers are read.**

If IZOD is one or three, then read item 16.

16. **[ZODRW(NDSLAY,ICOMP)] – U1DREL Read one array for each layer till all layers are read.**

If IADSORB is not zero and IZOD is two or three, then read item 17

17. **[ZODRS(NDSLAY,ICOMP)] – U1DREL Read one array for each layer till all layers are read.**

If IFOD is one or three, then read item 18

18. **[FODRW(NDSLAY,ICOMP)] – U1DREL Read one array for each layer till all layers are read.**

If IADSORB is not zero and IFOD is two or three, then read item 19

19. **[FODRS(NDSLAY,ICOMP)] – U1DREL Read one array for each layer till all layers are read.**

20. **[CONC(NDSLAY,ICOMP)] – U1DREL. Read one array for each layer till all layers are read.**

Otherwise, if UNSTRUCTURED option is not used then read items 21 through 33 for structured input.

A subset of the following two-dimensional variables is used to describe each layer. The variables needed for each layer depend on the transport options that are selected. Unneeded variables must be omitted.

he required variables for each of Items 21 – 33 for layer 1 are read first; then for layer 2 and so forth. The next variable of Items 21 – 33 is read after all layers for the previous variable are read.

If ICBNDFLG is zero then read item 21

21. **[ICBUND(NCOL,NROW)] -- U2DINT. Read one array for each layer till all layers are read.**

22. **[PRSITY(NCOL,NROW)] -- U2DREL. Read one array for each layer till all layers are read.**

If IADSORB is not zero then read item 23

23. **[BULKD(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

If IDISP is equal to one, then read items 24 and 25

24. **[DL(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

25. **[DT(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

If IDISP is equal to two, then read items 26, 27 28, 29, 30 and 31.

26. **[DLX(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

27. **[DLY(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

28. **[DLZ(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

29. **[DTXY(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

30. **[DTYZ(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

31. **[DTXZ(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

Items 32 – 38 are read for the first component, followed by the next till all MCOMP contaminant species are read.

If IADSORB is not zero then read item 32

32. **[ADSORB(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

If IADSORB is two then read item 33

33. **[FLICH(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

If IZOD is one or three, then read item 34

34. **[ZODRW(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

If IADSORB is not zero and IZOD is two or three, then read item 35

35. **[ZODRS(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

If IFOD is one or three, then read item 36

36. **[FODRW(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

If IADSORB is not zero and IFOD is two or three, then read item 37

37. **[FODRS(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

38. **[CONC(NCOL,NROW)] -- U2DREL Read one array for each layer till all layers are read.**

EXPLANATION OF VARIABLES READ BY THE BCT PACKAGE:

ITRNSP—is a transport simulation flag

If ITRNSP = 0, then transport is not simulated.

If ITRNSP = 1, then transport is simulated immediately after every flow time step or a steady-state flow simulation.

If ITRNSP = 2, then transport is simulated using the flow field of a previously simulated flow run.

IBCTCB—is a flag and a unit number.

If IBCTCB > 0, cell-by-cell mass flux terms will be written to this unit number when "SAVE BUDGET" or a nonzero value for ICBCFL is specified in Output Control. The terms that are saved are mass storage, mass flux from constant-concentration nodes, and mass flux in or out of the respective boundaries (WEL, DRN, GHB, RIV, STR, CHD, etc.). Note that prescribed concentration boundary (PCB) fluxes can be written to a separate file indexed in the PCB package input. Also note that if the "COMPACT BUDGET" option was selected in the OC package then the compact budget formats will be used.

If IBCTCB = 0, cell-by-cell mass flux terms will not be written.

If IBCTCB < 0, cell-by-cell mass flux for constant-concentration cells will be written in the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. Cell-by-cell mass flux to storage and between adjacent cells will not be written to any file.

MCOMP—is the number of mobile component species simulated.

ICBNDFLG—is a flag that determines if the active domain for transport is the same as that for flow.

If ICBNDFLG = 0, the active domain for transport is not the same as for flow and is read.

If ICBNDFLG = 1, the active domain for transport is the same as IBOUND for flow and the transport IBOUND array does not need to be read but is set from that of flow.

ITVD—is a flag and counter.

If ITVD = 0, the upstream weighted scheme is used for simulating the advective term.

If ITVD >0, the TVD scheme is used for simulating the advective term, and the value of ITVD represents the number of TVD correction iterations applied to the solution.

IADSORB—is an adsorption flag.

If IADSORB = 0, then adsorption is not simulated.

If IADSORB = 1, then linear adsorption is simulated.

If IADSORB = 2, then Freundlich adsorption is simulated.

ICT—is a flag that determines the transport solution scheme.

If ICT = 0, then the transport solution is for water phase concentration.

If ICT = 1, then the transport solution is for total concentration.

Note that when adsorption is very large, a solution to the total concentration (ICT = 1) may be more accurate in capturing the total system mass.

CINACT—is the concentration value that will be output at inactive nodes in the simulation.

CICLOSE—is the concentration tolerance for convergence of the matrix solver.

IDISP—is a flag indicating the dispersion formula used in the model

If IDISP = 0, then dispersion is not simulated.

If IDISP = 1, then isotropic dispersion is simulated.

If IDISP = 2, then anisotropic dispersion is simulated

IXDISP—is a flag indicating if cross-dispersion is simulated

If IXDISP = 0, then cross-dispersion is not simulated.

If IXDISP = 1, then cross-dispersion is included in the model.

DIFFNC—is the molecular diffusion coefficient for contaminant in water. Note that only one diffusion coefficient is read even if there are multiple contaminants simulated. Typically, diffusion is neglected as it is a small component of the overall fate and transport.

IZOD—is a flag indicating if zero order decay is simulated

If IZOD = 0, then zero order decay is not simulated.

If IZOD = 1, then zero order decay is included in water.

If IZOD = 2, then zero order decay is included on soil if there is adsorption.

If IZOD = 3, then zero order decay is included in water and on soil if there is adsorption.

IFOD—is a flag indicating if first order decay is simulated

If IFOD = 0, then first order decay is not simulated.

If IFOD = 1, then first order decay is included in water.

If IFOD = 2, then first order decay is included on soil if there is adsorption.

If IFOD = 3, then first order decay is included in water and on soil if there is adsorption.

IFMBC—is a flag indicating if Flux mass balance errors are to be considered in the flow solution and reported for fluid and mass fluxes.

If IFMBC= 0, then flow mass balance errors are not computed and transport does not consider these errors in the solution.

If IFMBC = 1, then flow mass balance errors are computed and transport considers these errors in the solution. Also, these errors are reported for fluid flux and for mass flux in an output file.

MBEGWUNF—is the unit number for output of flow imbalance information for the groundwater domain.

MBEGWUNT—is the unit number for output of transport mass imbalance information for the groundwater domain.

MBECLNUNF—is the unit number for output of flow imbalance information for the CLN domain.

MBECLNUNT—is the unit number for output of transport mass imbalance information for the CLN domain.

ICBUND—is the boundary variable. One value is read for every model cell. Usually, these values are read one layer at a time; however, when the XSECTION option is specified, a single two-dimensional variable for the cross section is read. Note that although ICBUND may be read as one or more two-dimensional variables, it is stored internally only as a one-dimensional variable for all nodes in the domain. Note that ICBUND is read only when the inactive domain for transport is different than that for the corresponding flow simulation.

If ICBUND(N) = 0, node N is inactive for transport calculations.

If ICBUND(N) > 0, node N is an active cell for solution to the transport equation.

Note that ICBUND can be inactive for active flow nodes. However, ICBUND cannot be active for a node which is inactive for the corresponding flow simulation.

PRSITY—is the effective transport porosity of the medium.

BULKD—is the bulk density of the porous matrix.

ANGLEX—is the angle between the horizontal x-axis and the outward normal to the face between a node and its connecting nodes.

DL—is the longitudinal dispersivity for an isotropic medium

DT—is the transverse dispersivity for an isotropic medium

DLX—is the x-direction longitudinal dispersivity for an anisotropic medium

DLY—is the y-direction longitudinal dispersivity for an anisotropic medium

DLZ—is the z-direction longitudinal dispersivity for an anisotropic medium

DTXY—is the xy-direction transverse dispersivity for an anisotropic medium

DTYZ—is the yz-direction transverse dispersivity for an anisotropic medium

DTXZ—is the xz-direction transverse dispersivity for an anisotropic medium

ADSORB—is the adsorption coefficient of a contaminant species.

FLICH—is the Freundlich adsorption isotherm exponent of a contaminant species

ZODRW—is the zero-order decay coefficient in water (concentration/time)

ZODRS—is the zero-order decay coefficient on soil (concentration/time)

FODRW—is the first-order decay coefficient in water (1/time)

FODRS—is the first-order decay coefficient on soil (1/time)

CONC—is the initial concentration of each contaminant species at any location in the domain.

Prescribed Concentration Boundary Package Input Instructions

Input to the Prescribed Concentration Boundary (PCB) Package is read from the file that has type "PCB" in the Name File. Optional variables are shown in brackets. All variables are free format if the option "FREE" is specified in the Basic Package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPPCB MXL]

This optional item must start with the word "PARAMETER".

2. MXPCB IPCBCB [Option]

3. [PARNAM PARTYP Parval NLST *[INSTANCES* NUMINST]]

Repeat Item 3 combined with the indicated repetitions of Item 4 NPPCB times. Items 3 and 4 are not read if NPPCB is negative or 0.

If PARNAM is to be a time-varying parameter, the keyword "INSTANCES" and a value for NUMINST must be entered.

4a. [INSTNAM]

4b. [Layer Row Column Species_No Pconcfact [xyz]]

Omit Item 4b if unstructured grid is used (i.e., if IUNSTR = 1)

4c. Node Species_No Pconcfact [xyz]

Omit Item 4c if structured grid is used (i.e., if IUNSTR = 0)

After each Item 3 for which the keyword "INSTANCES" is not entered, read Item 4b or 4c and not Item 4a.

After each Item 3 for which the keyword "INSTANCES" is entered, read Item 4a and Item 4b or 4c for each instance.

NLST repetitions of Item 4b or 4c are required; they are read by subroutine ULSTRD. (SFAC of the ULSTRD utility subroutine applies to Pconcfact). The NLST repetitions of Item 4b or 4c follow each repetition of Item 4a when PARNAM is time varying.

FOR EACH STRESS PERIOD

5. ITMP NP

6a. Layer Row Column iSpecies_No Conc [xyz]
Omit Item 6a if unstructured grid is used (i.e., if IUNSTR = 1)

6b. Node iSpecies_No Conc [xyz]
Omit Item 6b if structured grid is used (i.e., if IUNSTR = 0)

ITMP repetitions of Item 6a or 6b are read by subroutine ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility subroutine applies to Cond.) Item 6a or 6b is not read if ITMP is negative or 0.

7. [Pname [Iname]]
(Item 7 is repeated NP times. Item 7 is not read if NP is negative or 0. Iname is read if Pname is a time-varying parameter.)

Explanation of Variables Read by the PCB Package:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

NPPCB—is the number of prescribed concentration parameters.

MXL—is the maximum number of prescribed concentration cells that will be defined using parameters.

MXPCB—is the maximum number of prescribed concentration cells in use during any stress period, including those that are defined using parameters.

IPCBCB—is a flag and a unit number.

If IPCBCB > 0, mass flux at each prescribed concentration boundary cell for the respective species will be written to this unit number when "SAVE BUDGET" or a nonzero value for ICBCFL is specified in Output Control.

If IPCBCB = 0, mass flux at prescribed concentration boundaries will not be written.

If IPCBCB < 0, mass flux at each prescribed concentration boundary cell for the respective species will be written to the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

Option—is an optional list of character values.

"**AUXILIARY abc**" or "**AUX abc**"—defines an auxiliary variable, named "abc", which will be read for each prescribed concentration condition as part of Items 4 and 6. Up to 20 variables can be specified, each of which must be preceded by "AUXILIARY" or "AUX." These variables will not be used by the Groundwater Transport Process, but they will be available for use by other processes. The auxiliary variable values will be read after the Conc variable. "NOPRINT"—specifies that lists of PCBs will not be written to the Listing File.

PARNAM—is the name of a parameter. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter. For the PCB Package, the only allowed parameter type is PCB, which defines values of the prescribed concentration value.

Parval—is the parameter value. This parameter value may be overridden by a value in the Parameter Value File.

NLST—is the number of prescribed concentration boundary conditions (one for each species that is prescribed at every PCB node) in a non-time-varying parameter. For a time-varying parameter, NLST is the number of prescribed concentration boundary conditions in each instance.

INSTANCES—is an optional keyword that designates a parameter as time varying. The keyword is not case sensitive; that is, any combination of the same characters with different case can be used. If ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, PARNAM is non-time-varying and NUMINST should not be present.

NUMINST—is the number of instances for a time-varying parameter, where each instance is a list of river reaches and associated properties. If the keyword ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, NUMINST should not be present.

INSTNAM—is the name of an instance associated with the parameter named in the corresponding Item 3. The instance name can be 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent. Instance names must be unique for a parameter, but instance names may be reused for different parameters.

Layer—is the layer number of the cell containing the prescribed concentration boundary.

Row—is the row number of the cell containing the prescribed concentration boundary.

Column—is the column number of the cell containing the prescribed concentration boundary.

Node—is the node number of the model cell that contains the prescribed concentration boundary.

iSpecies_No—is the species number of the prescribed concentration boundary.

Concfact—is the factor used to calculate the concentration from the parameter value. The conductance is the product of Concfact and the parameter value.

[xyz]—represents the values of the auxiliary variables for a PCB that have been defined in Item 2. The values of auxiliary variables must be present in each repetition of Items 4 and 6 if they are defined in Item 2. The values must be specified in the order used to define the variables in Item 2.

ITMP—is a flag and a counter.

If ITMP < 0, non-parameter PCB data from the last stress period will be reused.

If ITMP ≥ 0, ITMP will be the number of non-parameter PCB conditions read for the current stress period.

NP—is the number of parameters in use in the current stress period.

Conc—is the prescribed concentration value for species (iSpecies_No) at the boundary.

Pname—is the name of a parameter that is being used in the current stress period. NP parameter names will be read.

Iname—is an instance name that is read only if Pname is a time-varying parameter. Multiple instances of the same parameter are not allowed in a stress period.

Output Package Control for Simulating Flow and Transport: Input Instructions

Introduction

MODFLOW-USG provides adaptive time stepping and output control algorithms to enhance robustness and efficiency of highly nonlinear groundwater flow simulations. 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 or at time values when there is a change in stresses or stress periods.

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 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

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. 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.

Output Control with Adaptive Time Stepping – I/O

Input to the Output Control Option of the Groundwater and CLN Flow and Transport Processes is read from the file that is specified as type "OC" in the Name File. If no "OC" file is specified, default output control is used. Under the default, head and overall budget are written to the Listing File at the end of every stress period. If the CLN domain is active, the head and budget are also written to the Listing file when the Groundwater head and budget output is requested. Binary output for the GWF and CLN flow Processes, if active, may be to the same head or drawdown file as specified here. Alternatively, the CLN flow Process produces its own output files as discussed in the CLN input file documentation. Be it in the same or separate files, the CLN output is produced whenever the GWF output is produced, as prescribed here. If a transport simulation is conducted, concentrations and mass balance output is provided in a similar manner to output of the flow results. The default printout format for head, drawdown and concentration is 10G11.4. Output Control data may be specified as words or numeric codes. One of these methods must be used throughout any simulation.

The output control file also provides input for adaptive time stepping parameters, if adaptive time stepping is used in the simulation. Adaptive time stepping is activated by the "option" keyword "ATS" using numeric input and the "option" keyword "ATSA if alphabetic input is provided. Option keywords "NPTIMES" and "NPSTPS" further provide output control when using adaptive time stepping.

Output Control Using Words

Recognized words are shown in bold italics; these words must be entered exactly as shown except that they may be entered in either uppercase or lowercase. Optional parts of lines are shown in brackets. One or more spaces must separate each word or variable, and the total line length must not exceed 199 characters.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. Any combination of the following lines:

[**OPTIONS**]–the explanations below provide the option details.

HEAD PRINT FORMAT IHEDFM

Specifies the format for writing head to the Listing File.

HEAD SAVE FORMAT CHEDFM **[LABEL]**

Specifies the format for writing head to a file other than the Listing File. Omit this line to obtain a binary (unformatted) file. Binary files usually are smaller than text files, but they are not generally transportable among different computer operating systems or different Fortran compilers.

HEAD SAVE UNIT IHEDUN

Specifies the file unit for writing head to a file other than the Listing File.

DRAWDOWN PRINT FORMAT IDDNFM

Specifies the format for writing drawdown to the Listing File.

DRAWDOWN SAVE FORMAT CDDNFM **[LABEL]**

Specifies the format for writing drawdown to a file other than the Listing File. Omit this line to obtain an unformatted (binary) file. Binary files

usually are smaller than text files, but they are not generally transportable among different computer operating systems or different Fortran compilers.

DRAWDOWN SAVE UNIT IDDNUN

Specifies the file unit for writing drawdown to a file other than the Listing File.

CONC PRINT FORMAT ISPCFM

Specifies the format for writing concentration of all species to the Listing File.

CONC SAVE FORMAT CSPCFM **[LABEL]**

Specifies the format for writing concentration of all species to a file other than the Listing File. Omit this line to obtain a binary (unformatted) file. Binary files usually are smaller than text files, but they are not generally transportable among different computer operating systems or different Fortran compilers.

CONC SAVE UNIT ISPCUN

Specifies the file unit for writing concentration of all species to a file other than the Listing File.

IBOUND SAVE FORMAT CBOUFM **[LABEL]**

Specifies the format for writing IBOUND to a file.

IBOUND SAVE UNIT IBOUUN

Specifies the file unit for writing IBOUND to a file.

COMPACT BUDGET [AUX or AUXILIARY]

COMPACT BUDGET indicates that the cell-by-cell budget file(s) will be written in a more compact form than is used in the 1988 version of MODFLOW (referred to as MODFLOW-88)(McDonald and Harbaugh, 1988); however, programs that read these data in the form written by MODFLOW-88 will be unable to read the new compact file. If this option

is not used, MODFLOW-2005 will write the files using the MODFLOW-88 form. The optional word **AUX** (or ***AUXILIARY***) indicates that auxiliary data that are defined in packages (see input data for the RIV, WEL, DRN, and GHB Packages) should be saved in the budget file along with budget data.

2. TIMOT (nptimes)

Data Item 2 is read only if adaptive time stepping is on, and if the number of print times (NPTIMES) is greater than zero.

FOR EACH TIME STEP FOR WHICH OUTPUT OR ADAPTIVE TIME STEPPING CONTROL IS DESIRED.

3a. PERIOD IPEROC **STEP ITSOC [**DDREFERENCE**]**

Data Item 3a is read only if adaptive time stepping is not used. Otherwise, data item 3b is read.

3b. PERIOD IPEROC **[**DDREFERENCE**]**

Data Item 3b is read only if adaptive time stepping is used. Otherwise, data item 3a is read.

4. Any combination of the following lines:

DELTAT deltat

The keyword ***DELTAT*** is followed by the numeric value *(deltat)* of the time step size with which to begin this stress period. Note that this keyword is used only if adaptive time stepping is on. Also, a value of ***DELTAT*** is required for the first stress period. If a value is not provided for later stress periods, the stress period will begin with a time-step size value equal to the last time step size value from the previous stress period.

TMINAT tminat

The keyword ***TMINAT*** is followed by the numeric value *(tminat)* of the minimum time step size for this stress period. Note that this keyword is

used only if adaptive time stepping is on. If a value of ***TMINAT*** is not provided for the first stress period, a default of 1.0e-10 is set. If a value is not provided for later stress periods, the stress period will use the value from the previous stress period.

TMAXAT tmaxat

The keyword ***TMAXAT*** is followed by the numeric value *(tmaxat)* of the maximum time step size for this stress period. Note that this keyword is used only if adaptive time stepping is on. If a value of ***TMAXAT*** is not provided for the first stress period, a default of 1.0e10 is set. If a value is not provided for later stress periods, the stress period will use the value from the previous stress period.

TADJAT tadjat

The keyword ***TADJAT*** is followed by the numeric value *(tadjat)* of the time step size adjustment factor for this stress period. Note that this keyword is used only if adaptive time stepping is on. If a value of ***TADJAT*** is not provided for the first stress period, a default of 2.0 is set. If a value is not provided for later stress periods, the stress period will use the value from the previous stress period. The time step adjustment factor is used as follows. If convergence is achieved for a time step within one-third the total number of iterations, the time step size is increased by this adjustment factor. If convergence is achieved within two-third the total number of iterations (but more than one-third the total number of iterations), then the time step size is not altered. However, if convergence required greater than two-third of the total number of iterations, then the time step size is decreased by this adjustment factor.

TCUTAT tcutat

The keyword ***TCUTAT*** is followed by the numeric value *(tcutat)* of the time step size cutting factor for this stress period. Note that this keyword

is used only if adaptive time stepping is on. If a value of ***TCUTAT*** is not provided for the first stress period, a default of 5.0 is set. If a value is not provided for later stress periods, the stress period will use the value from the previous stress period. The time step cutting factor is used to reduce the time-step size, when convergence is not achieved for a particular time step. In this case, the solution is reattempted with the reduced time step size.

HCLOSE hclose

The keyword ***HCLOSE*** is followed by the numeric value *(hclose)* which is the outer iteration tolerance for this stress period. Note that this keyword is used only if adaptive time stepping is on. A value of ***HCLOSE*** need not be provided here for the first stress period, as it will be taken from the value supplied in the SMS file. However, if provided here for any stress period, the current value will override all previous values. See SMS input instructions for more details on the parameter ***HCLOSE***.

BTOL btol

The keyword ***BTOL*** is followed by the numeric value *(btol)* which is the backtracking tolerance factor for this stress period. Note that this keyword is used only if adaptive time stepping is on. A value of ***BTOL*** need not be provided here for the first stress period, as it will be taken from the value supplied in the SMS file. However, if provided here for any stress period, the current value will override all previous values. See SMS input instructions for more details on the parameter ***BTOL***.

MXITER mxiter

The keyword ***MXITER*** is followed by the numeric value *(mxiter)* which is the maximum number of iterations for this stress period. Note that this keyword is used only if adaptive time stepping is on. A value of ***MXITER*** need not be provided here for the first stress period, as it will be taken

from the value supplied in the SMS file. However, if provided here for any stress period, the current value will override all previous values. See SMS input instructions for more details on the parameter ***MXITER***.

PRINT HEAD [list layers if all layers not desired]

Head is written to the Listing File.

PRINT DRAWDOWN [list layers if all layers not desired]

Drawdown is written to the Listing File.

PRINT CONC [list layers if all layers not desired]

Concentration is written to the Listing File.

PRINT BUDGET

Overall volumetric budget is written to the Listing File.

SAVE HEAD [list layers if all layers not desired]

Head is written to a file other than the Listing File.

SAVE DRAWDOWN [list layers if all layers not desired]

Drawdown is written to a file other than the Listing File.

SAVE CONC [list layers if all layers not desired]

Concentration is written to a file other than the Listing File.

SAVE IBOUND [list layers if all layers not desired]

IBOUND is written to a file other than the Listing File. This option is provided to allow changes in IBOUND to be recorded in simulations where IBOUND changes during a simulation.

SAVE BUDGET

Cell-by-cell budget data are written to the files that are designated in the packages that compute budget terms. If solute transport is also simulated then the SAVE BUDGET keyword also saves the mass flux terms to the appropriate budget files.

Item 2 and one or more Item-4 lines are specified for each time for which output is desired. These lines must be in the order of increasing simulation time. **Note also that the keyword CONCENTRATION may also be used instead of the shorter form CONC when specifying output control using key words.**

Explanation of Variables Read by Output Control Using Words:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

OPTIONS—are optional keywords that activate options:

ATSA indicates that adaptive time stepping will be used in the simulation with output control specified using alphabetic characters.

NPTIMES nptimes: the optional keyword NPTIMES indicates that the following numbers (nptimes) is the number of print times in the simulation. With adaptive time stepping, the time value of a time step is not known apriori and therefore, we cannot determine at which time step, a particular time value will be reached. If printout is required at particular times, the NPTIMES keyword allows input of an array of time values TIMOT (nptimes) at which output is required. The adaptive time stepping routine will adapt the simulation time to exactly match a print time so computations are performed and output as needed.

NPSTPS npsteps: the optional keyword NPSTPS indicates that the following numbers (npsteps) is the number of steps after which output is provided. When adaptive time stepping is used, the output control items that are otherwise provided for every time step are instead provided for every stress period.

FASTFORWARD ispfast, itsfast, iugfast: the optional keyword FASTFORWARD indicates that input is to be provided from a binary file, which is to be fastforwarded to the desired stress-period and time-step for input. Three numbers follow the word FASTFORWARD. The first, ispfast, is the stress period number to which to fast-forward. The second, itsfast is the time step number of that stress period, to which to fast-forward. The third, iugfast is the unit number on which to open the binary file from which the simulation's initial conditions are provided. This option is useful for restarting a simulation from somewhere in between, when output has already been provided at the point of restart in a previous simulation.

Note that with use of adaptive time stepping, it is appropriate to have output at every time step or every fixed number of time steps during model development, to check how the simulation is performing. During production runs, however, it may be more appropriate to provide time values at which output is desired, so the simulation can adapt and provide computations at those times, for output. Also, by default, output is provided and mass balance information written to the list file, at the end of every stress period. Note further that if both flags NPTIMES and NPSTPS are on, then output is provided according to both flags.

IHEDFM—is a code for the format in which heads will be printed. (Positive values indicate wrap format; negative values indicate strip format.)

0 - 10G11.4	11 - 20F5.4
1 - 11G10.3	12 - 10G11.4
2 - 9G13.6	13 - 10F6.0

3 - 15F7.1	14 - 10F6.1
4 - 15F7.2	15 - 10F6.2
5 - 15F7.3	16 - 10F6.3
6 - 15F7.4	17 - 10F6.4
7 - 20F5.0	18 - 10F6.5
8 - 20F5.1	19 - 5G12.5
9 - 20F5.2	20 - 6G11.4
10 - 20F5.3	21 - 7G9.2

CHEDFM—is a character value that specifies the format for saving heads, and can only be specified if the word method of output control is used. The format must contain 20 characters or less and must be a valid Fortran format that is enclosed in parentheses. The format must be enclosed in apostrophes if it contains one or more blanks or commas. The optional word ***LABEL*** after the format is used to indicate that each layer of output should be preceded with a line that defines the output (simulation time, the layer being output, and so forth). If there is no line specifying CHEDFM, then heads are written to a binary (unformatted) file. Binary files are usually more compact than text files, but they are not generally transportable among different computer operating systems or different Fortran compilers.

IHEDUN—is the unit number on which head will be saved.

IDDNFM—is a code for the format in which drawdown will be printed. The codes are the same as for IHEDFM.

ISPCFM—is a code for the format in which concentrations of all species will be printed. The codes are the same as for IHEDFM.

CSPCFM—is a character value that specifies the format for saving concentrations, and can only be specified if the word method of output control is used. The format must contain 20 characters or less and must be a valid Fortran format that is enclosed in parentheses. The format must be enclosed in apostrophes if it contains one or more blanks or commas. The optional word ***LABEL*** after the format is used to indicate that each layer of output should be preceded with a line that defines the output (simulation time, the layer being output, and so forth). If there is no line specifying CSPCFM, then concentrations are written to a binary (unformatted) file. Binary files are usually more compact than text files, but they are not generally transportable among different computer operating systems or different Fortran compilers.

ISPCUN—is the unit number on which concentrations will be saved.

CDDNFM—is a character value that specifies the format for saving drawdown, and can only be specified if the word method of output control is used. The format must contain 20 characters or less and must be a valid Fortran format that is enclosed in parentheses. The format must be enclosed in apostrophes if it contains one or more blanks or commas. The optional word LABEL after the format is used to indicate that each layer of output should be preceded with a line that defines the output (simulation time, the layer being output, and so forth). If there is no line specifying CDDNFM, then drawdown is written to a binary (unformatted) file. Binary files are usually more compact than text files, but they are not generally transportable among different computer operating systems or different Fortran compilers.

IDDNUN—is the unit number on which drawdowns will be saved.

CBOUFM—is a character value that specifies the format for saving IBOUND, and can only be specified if the word method of output control is used. The format must contain 20 characters or less and must be a valid Fortran format that is enclosed in parentheses. The format must be enclosed in apostrophes if it contains one or more blanks or commas. The optional word LABEL is

used to indicate that each layer of output should be preceded with a line that defines the output (simulation time, the layer being output, and so forth). If there is no line specifying CBOUFM, then IBOUND is written using format (20I4). IBOUND is never written as a binary (unformatted) file.

IBOUUN—is the unit number on which IBOUND will be saved.

TIMOT—is the array of time values at which output is requested. There are NPTIMES entries in the TIMOT array.

IPEROC—is the stress period number at which output is desired.

ITSOC—is the time step number (within a stress period) at which output is desired.

DDREFERENCE—keyword indicating that the heads at the associated stress period and time step are to be used as the reference heads for calculating drawdown for all subsequent time steps up to the next occurrence of DDREFERENCE. Prior to the first occurrence (if any) of DDREFERENCE the initial heads (STRT) will be used as the reference heads for calculating drawdown.

Example Output Control Input Using Words

```
HEAD PRINT FORMAT 15
HEAD SAVE FORMAT (20F10.3) LABEL
HEAD SAVE UNIT 30
COMPACT BUDGET
DRAWDOWN PRINT FORMAT 14
PERIOD 1 STEP 1
    PRINT HEAD 2 6
    PRINT DRAWDOWN
    PRINT BUDGET
    SAVE BUDGET
    SAVE HEAD
```

```
    SAVE CONCENTRATION
PERIOD 1 STEP 7
    SAVE HEAD 1 3 5
    PRINT DRAWDOWN
    SAVE BUDGET
PERIOD 2 STEP 5
    PRINT HEAD
    PRINT BUDGET
    SAVE BUDGET
    SAVE HEAD
```

Note that the first line cannot be blank, but after the first line blank lines are ignored when the word method is used to specify Output Control data. Indented lines are allowed because of the use of free format input.

Example Output Control with Adaptive Time Stepping Input Using Words

```
ATSA NPTIMES 2 NPSTPS 20
    5.0E3 3.13E8        !these are the 2 print times in TIMOT
    HEAD SAVE UNIT 30
    HEAD PRINT FORMAT 0
    DRAWDOWN SAVE UNIT 31
    DRAWDOWN PRINT FORMAT 0
PERIOD 1
    DELTAT 5.0E2
    TMAXAT 5.0E5
    SAVE HEAD
    SAVE DRAWDOWN
    SAVE BUDGET
    PRINT BUDGET
PERIOD 2
    DELTAT 3600.00
    TADJAT 2.5
```

SAVE HEAD
SAVE DRAWDOWN
PRINT BUDGET
PERIOD 3
SAVE HEAD
SAVE DRAWDOWN
PRINT BUDGET
PERIOD 120
SAVE HEAD
SAVE DRAWDOWN
PRINT BUDGET

Note that the first line cannot be blank, but after the first line blank lines are ignored when the word method is used to specify Output Control data. Indented lines are allowed because of the use of free format input.

Output Control Using Numeric Codes

All variables are free format if the word FREE is specified in Item 1 of the Basic Package input file; otherwise, the variables all have 10-character fields.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. IHEDFM IDDNFM IHEDUN IDDNUN ISPCFM ISPCUN [OPTIONS]

2. TIMOT (nptimes)

Data Item 2 is read only if adaptive time stepping is on, and if the number of print times (NPTIMES) is greater than zero.

Item 3 is read for each time step if adaptive time stepping is not used. If adaptive time stepping is used, item 3 is read for each stress period.

3. DELTAT TMINAT TMAXAT TADJAT TCUTAT

Data Item 3 is read only if adaptive time stepping is on.

4. INCODE IHDDFL IBUDFL ICBCFL ISPCFL

5. Hdpr Ddpr Hdsv Ddsv Cnpr Cnsv

(Item 3 is read 0, 1, or NLAY times, depending on the value of INCODE.)

Explanation of Variables Read by Output Control Using Numeric Codes:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

IHEDFM—is a code for the format in which heads will be printed. See the description above in the explanation of variables read by output control using words.

IDDNFM—is a code for the format in which drawdowns will be printed. The codes are the same as for IHEDFM.

IHEDUN—is the unit number on which heads will be saved.

IDDNUN—is the unit number on which drawdowns will be saved.

ISPCFM—is a code for the format in which concentrations will be printed. The codes are the same as for IHEDFM.

ISPCUN—is the unit number on which concentrations will be saved.

OPTIONS—are optional keywords that activate options:

ATS indicates that adaptive time stepping will be used in the simulation with output control specified using numeric characters.

NPTIMES nptimes: the optional keyword NPTIMES indicates that the following numbers (nptimes) is the number of print times in the simulation. With adaptive time stepping, the time value of a time step is not known apriori and therefore, we cannot determine at which time step, a particular time value will be reached. If printout is required at particular times, the NPTIMES keyword allows input of an array of time values TIMOT(nptimes) at which output is required. The adaptive time stepping routine will adapt the simulation time to exactly match a print time so computations are performed and output as needed.

NPSTPS npsteps: the optional keyword NPSTPS indicates that the following numbers (npsteps) is the number of steps after which output is provided. When adaptive time stepping is used, the output control items that are otherwise provided for every time step are instead provided for every stress period.

Note that with use of adaptive time stepping, it is appropriate to have output at every time step or every fixed number of time steps during model development, to check how the simulation is performing. During production runs, however, it may be more appropriate to provide time values at which output is desired, so the simulation can adapt and provide computations at those times, for output.

TIMOT—is the array of time values at which output is requested. There are NPTIMES entries in the TIMOT array.

DELTAT—is the time step size with which to begin this stress period.

TMINAT—is the minimum time step size for this stress period.

TMAXAT—is the maximum time step size for this stress period.

TADJAT—is the time step size adjustment factor this stress period. The time step adjustment factor is used as follows. If convergence is achieved for a time step within one-third of the total number of iterations, the time step size is

increased by this adjustment factor. If convergence is achieved within two-third of the total number of iterations (but more than one-third the total number of iterations), then the time step size is not altered. However, if convergence required greater than two-third of the total number of iterations, then the time step size is decreased by this adjustment factor.

TCUT—is the time step size cutting factor this stress period. The time step cutting factor is used to reduce the time-step size, when convergence is not achieved for a particular time step. In this case, the solution is reattempted with the reduced time step size.

INCODE—is the code for reading Item 3.

If INCODE < 0, Item 3 flags are used from the last time step. Item 3 is not read.

If INCODE = 0, all layers are treated the same way. Item 3 will consist of one line.

If INCODE > 0, Item 3 will consist of one line for each layer.

IHDDFL—is a head and drawdown output flag. This flag allows Item 3 flags to be specified in an early time step and then used or not used in subsequent time steps. Thus, using IHDDFL to avoid resetting Item 3 flags every time step may be possible.

If IHDDFL = 0, no heads or drawdowns will be printed or saved regardless of which Item 3 flags are specified.

If IHDDFL ≠ 0, heads and drawdowns will be printed or saved according to the Item 3 flags.

IBUDFL—is a budget print flag.

If IBUDFL = 0, overall volumetric budget will not be printed.

If IBUDFL ≠ 0, overall volumetric budget will be printed.

ICBCFL—is a flag for writing cell-by-cell flow data. Note that if transport is simulated, this flag also controls output of cell-by-cell mass flux data.

If ICBCFL = 0, cell-by-cell flow terms are not written to any file.

If ICBCFL ≠ 0, cell-by-cell flow terms are written to the LIST file or a budget file depending on flags set in the component of flow packages, that is, IWELCB, IRCHCB, and so forth.

ISPCFL—is a concentration output flag. This flag allows Item 3 flags to be specified in an early time step and then used or not used in subsequent time steps. Thus, using ISPCFL to avoid resetting Item 3 flags every time step may be possible.

If ISPCFL = 0, no concentrations will be printed or saved regardless of which Item 3 flags are specified.

If ISPCFL ≠ 0, concentrations will be printed or saved according to the Item 3 flags.

Hdpr—is the output flag for head printout.

If Hdpr = 0, head is not printed for the corresponding layer.

If Hdpr ≠ 0, head is printed for the corresponding layer.

Ddpr—is the output flag for drawdown printout.

If Ddpr = 0, drawdown is not printed for the corresponding layer.

If Ddpr ≠ 0, drawdown is printed for the corresponding layer.

Hdsv—is the output flag for head save.

If Hdsv = 0, head is not saved for the corresponding layer.

If Hdsv ≠ 0, head is saved for the corresponding layer.

Ddsv—is the output flag for drawdown save.

If Ddsv = 0, drawdown is not saved for the corresponding layer.

If Ddsv ≠ 0, drawdown is saved for the corresponding layer.

Cnpr—is the output flag for concentration printout.

If Cnpr = 0, concentration is not printed for the corresponding layer.

If Cnpr ≠ 0, concentration is printed for the corresponding layer.

Cnsv—is the output flag for concentration save.

If Cnsv = 0, concentration is not saved for the corresponding layer.

If Cnsv ≠ 0, concentration is saved for the corresponding layer.

Connected Linear Network (CLN) Process with Transport Input Instructions

Input for the Connected Linear Network (CLN) Process is read from the file that is type "CLN" in the Name File. Options have been used to expand the capability of MODFLOW-USG, to keep the code backward compatible.

FOR EACH SIMULATION

0. [OPTIONS opt, …]

This optional item must start with the keyword "OPTIONS"

1. **NCLN ICLNNDS ICLNCB ICLNHD ICLNDD ICLNIB NCLNGWC NCONDUITYP [OPTIONS2]**

These variables are free format if the option "FREE" is specified in the Basic Package input file; otherwise, the variables all have 10-character fields.

If NCLN is greater than zero, then read item 2.

2. NNDCLN(NCLN) – U1DREL

If NCLN is greater than zero and ICLNNDS is greater than zero, then read item 3. Item 3 is read NCLN times, once for each CLN segment in the simulation. The number of entries for each line of item 3 is the number of CLN cells (NNDCLN) associated with each CLN segment, as input in item 2 above.

3. CLNCON[NNDCLN(NCLN)]

The variables of item 3 are free format if the option "FREE" is specified in the Basic Package input file; otherwise, the variables all have 10-character fields.

If NCLN is equal to zero, then read items 4, 5, and 6.

4. NJA_CLN

5. IAC_CLN(NODES) - U1DINT

6. JA_CLN(NJAG) - U1DINT

7. IFNO IFTYP IFDIR FLENG FELEV FANGLE IFLIN ICCWADI

Item 7 is read for each CLN node in the domain. Therefore, item 7 is repeated NCLNNDS times for each of the NCLNNDS Connected Linear Network nodes.

If UNSTRUCTURED option is used then read item 8.

8. IFNOD IGWNOD IFCON FSKIN FLENGW FANISO ICGWADI

Item 8 is read for each CLN node to porous medium grid-block connection in the domain. Therefore, item 8 is repeated NCLNGWC times for each of the NCLNGWC connections.

Otherwise, if UNSTRUCTURED option is not used then read item 9 for structured input

9. IFNOD IGWLAY IGWROW IGWFCOL IFCON FSKIN FLENGW FANISO ICGWADI

Item 9 is read for each CLN node to porous medium grid-block connection in the domain. Therefore, item 9 is repeated NCLNGWC times for each of the NCLNGWC connections.

10a. ICONDUITYP FRAD CONDUITK

Item 10a is read for each conduit categorized in the model. Therefore, item 10a is repeated NCONDUITYP times for each of the NCONDUITYP types of circular conduit geometries in the model.

10b. IRECTYP FLENGTH FHEIGHT CONDUITK

Item 10b is read for each conduit categorized in the model. Therefore, item 10b is repeated NRECTYP times for each of the NRECTYP types of rectangular geometries in the model.

11. IBOUND(NCLNNDS) – U1DINT

12. STRT(NCLNNDS) – U1DREL

If ITRNSP is not zero then read items 13 through 16 for transport simulations.

Items 13 through 16 are required for transport simulation and should not be entered for only a flow simulation.

Read Item 13 only if ICBNDFLG = 0

13. [ICBUND(NCLNNDS)] – U1DREL

Read items 14 and 15 only if IDISP is not zero.

14. **[DLL(NCLNNDS)] – U1DREL**

15. **[DLM(NCLNNDS)] – U1DREL**

Items 16 are read for the first component, followed by the next till all MCOMP contaminant species are read.

16. **[CONC(NCLNNDS)] – U1DREL**

Explanation of Variables Read by the CLN Package:

OPTIONS are keyword options. "OPTIONS" must be listed as the first keyword in order to specify any options. The following options are supported:

TRANSIENT indicates that transient IBOUND information will be read for each stress period.

PRINTIAJA will print the IA_CLN and JA_CLN arrays to the listing file. These arrays correspond with the CLNCLN flows that are written to the CLN cell-by-cell output file.

NCLN—is a flag or the number of CLN segments (a segment is defined here as a collection of linearly connected CLN nodes) simulated in the model.

If NCLN = 0, this flag indicates that the CLN domain connectivity is input in a general IA-JA manner as is used for the GWF Process.

If NCLN > 0, linear CLN segments (for instance multi-aquifer wells) or simple CLN networks are simulated and NCLN is the total number of CLN segments in the domain.

ICLNNDS—is a flag or number of CLN-nodes simulated in the model. Multiple CLN-nodes constitute a segment.

If ICLNNDS < 0, the CLN-nodes are ordered in a sequential manner from the first CLN node to the last CLN node. Therefore, only linear CLN segments

are simulated since a CLN segment does not share any of its nodes with another CLN segment.

If ICLNNDS > 0, CLN networks can be simulated and ICLNNDS is the total number of CLN-nodes simulated by the model. CLN nodes can be shared among CLN segments in the network and therefore, the CLN-nodal connectivity for the network is also required as input.

Note that if NCLN is zero, then ICLNNDS is the total number of CLN nodes in the model (even if the sign is negative).

ICLNCB—is a flag and a unit number.

If ICLNCB > 0, cell-by-cell flow terms will be written to this unit number when "SAVE BUDGET" or a nonzero value for ICBCFL is specified in Output Control. The terms that are saved are storage, and flow between adjacent cells.

If ICLNCB = 0, cell-by-cell flow terms will not be written.

If ICLNCB < 0, cell-by-cell flow for CLN cells will be written in the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

ICLNHD—is a flag and a unit number.

If ICLNHD > 0, head output for CLN-nodes will be written to this unit number.

If ICLNHD = 0, head output for CLN-nodes will not be written.

If ICLNHD < 0, head output for CLN-nodes will be written to the same unit number (IHEDUN) as used for head output for the porous matrix nodes.

ICLNDD—is a flag and a unit number.

If ICLNDD > 0, drawdown output for CLN-nodes will be written to this unit number.

If ICLNDD = 0, drawdown output for CLN-nodes will not be written.

If ICLNDD < 0, drawdown output for CLN-nodes will be written to the same unit number (IDDNUN) as used for drawdown output for the porous matrix nodes.

ICLNIB—is a flag and a unit number.

If ICLNIB > 0, IBOUND output for CLN-nodes will be written to this unit number.

If ICLNIB = 0, IBOUND output for CLN-nodes will not be written.

If ICLNIB < 0, IBOUND output for CLN-nodes will be written to the same unit number (IBOUUN) as used for IBOUND output for the porous matrix nodes.

NCLNGWC—is the number of CLN to porous-medium grid-block connections present in the model. A CLN node need not be connected to any groundwater node. Conversely, a CLN node may be connected to multiple groundwater nodes, or multiple CLN nodes may be connected to the same porous medium mode.

NCONDUITYP—is the number of circular conduit-geometry types that are present within the model.

[OPTIONS2] — include keywords and additional data that are required for different cross-sectional geometries (rectangular cross-sections are currently included), and for turbulent flow simulations using the Darcy-Weisbach equation within the CLN domain. These data are not required if Darcy-Weisbach equation is not used for any of the CLN domain cells and

may then be omitted. Additional information may also be required for formatting the cell-by-cell flow files.

RECTANGULAR nrectyp —is a flag indicating that rectangular cross-sectional geometries are included, and the value of "nrectyp" is the number of rectangular-geometry types that are present within the model.

SAVECLNCON iclncn —is a flag indicating that CLN concentration output is to be saved and the value of "iclncn" is a flag and a unit number.

If ICLNCN > 0, concentration output for CLN-nodes will be written to this unit number.
If ICLNCN = 0, concentration output for CLN-nodes will not be written.
If ICLNCN < 0, concentration output for CLN-nodes will be written to the same unit number (ISPCUN) as used for concentration output for the porous matrix nodes.

Note that if this option does not exist then CLN concentrations are not saved.

SAVECLNMAS iclnmb —is a flag indicating that CLN mass flux output is to be saved and the value of "iclnmb" is a flag and a unit number.

If ICLNMB > 0, mass flux output for CLN-nodes will be written to this unit number.
If ICLNCN = 0, mass flux output for CLN-nodes will not be written.
If ICLNCN < 0, mass flux output for CLN-nodes will be written to the same unit number (IBCTCB) as used for mass flux output for the porous matrix nodes.

Note that if this option does not exist then CLN mass fluxes are not saved.

GRAVITY grav—is the gravitational acceleration constant in model simulation units [L/T2]. The value of the constant follows the keyword GRAVITY.

VISCOSITY visk—is the kinematic viscosity of water in model simulation units [L2/T]. The value of kinematic viscosity follows the keyword VISCOSITY.

NNDCLN—is the number of CLN-nodes that are associated with each CLN segment.

CLNCON—are the CLN-node numbers associated with each CLN segment.

NJA_CLN—is the total number of connections of the CLN domain. NJA_CLN is used to dimension the sparse matrix in a compressed row storage format.

IAC_CLN—is a matrix indicating the number of connections plus 1 for each CLN node to another CLN node. Note that the IAC_CLN array is only supplied for the CLN cells; the IAC_CLN array is internally expanded to include other domains if present in a simulation.

JA_CLN—is a list of CLN cell number (n) followed by its connecting CLN cell numbers (m) for each of the m CLN cells connected to CLN cell n. This list is sequentially provided for the first to the last CLN cell. Note that the cell and its connections are only supplied for the CLN cells and their connections to the other CLN cells using the local CLN cell numbers. Also note that the JA_CLN list input may be chopped up to have every node number and its connectivity list on a separate line for ease in readability of the file. To further ease readability of the file, the node number of the cell whose connectivity is subsequently listed, may be expressed as a negative number the sign of which is subsequently corrected by the code.

IFNO—is the node number for the CLN node. CLN-nodes are numbered from 1 to the total number of CLN-nodes, NCLNNDS.

IFTYP—is the type-index for the CLN node. The type-index identifies this CLN segment type from the catalogue of CLN elements in a simulation. CLN types

include different cross-section shapes (currently only circular conduit geometries are included) of different sizes.

IFDIR—is a directional index for the CLN-node orientation.

If IFDIR = 0, the CLN-node is oriented in the vertical direction.

If IFDIR = 1, the CLN-node is oriented in the horizontal direction.

If IFDIR = 2, the CLN-node is oriented at an angle to the horizontal and the angle is read in parameter FANGLE.

Note that the parameter IFDIR is utilized only to determine a CLN-node's fractional saturation to determine transients or dry conditions.

FLENG—is the length of the CLN-node segment

FELEV—is the elevation of the bottom of the CLN-node.

FANGLE—is the angle made by a CLN-node segment from the horizontal. FANGLE is ignored if the parameter IFDIR is not equal to 2.

IFLIN—is a flag indicating flow conditions within the CLN network. A positive value of IFLIN indicates that the CLN node is treated as confined. If this is an upstream location, the relative permeability is fixed at unity and does not diminish to zero as the CLN cells dewaters and becomes dry. This is similar to the "confined flow" option in the GW domain of MODFLOW.

If IFLIN = 0, flow in the CLN network at this cell is treated as laminar, unconfined (same as if IFLIN = -1).

If the magnitude of IFLIN = 1, the CLN-node is treated as laminar flow.

If the magnitude of IFLIN = 2, the CLN-node is treated as turbulent flow using the Darcy-Weisbach equation.

If the magnitude of IFLIN = 3, the CLN-node is treated as turbulent flow using the Hazen-Williams equation.

If the magnitude of IFLIN = 4, the CLN-node is treated as turbulent flow using the Manning's equation.

ICCWADI—is a flag indicating if vertical flow correction is applied to CLN-CLN flow at this node if it is dry.

If ICCWADI = 0, flow in the CLN network at this cell is treated without vertical flow correction.

If ICCWADI = 1, vertical flow correction is applied for flow within the CLN network to this node if it is dry.

IFNOD—is the node number for the CLN node that is connected to the groundwater node.

IGWNOD—is the node number of the connecting subsurface node for unstructured grid input.

IGWLAY—is the layer number of the connecting subsurface node, for structured input.

IGWROW—is the row number of the connecting subsurface node, for structured input.

IGWCOL—is the column number of the connecting subsurface node, for structured input.

IFCON—is an index for determining the connectivity equation between CLN-node and its associated matrix node.

If IFCON = 0, the Thiem equation is used to provide the connection between CLN-node and matrix node as was done in the Multi-Node Well Package of MODFLOW-2005, without any skin effects.

If IFCON = 1, the Thiem equation is used to provide the connection between CLN-node and matrix node as was done in the Multi-Node Well Package of MODFLOW-2005, with inclusion of skin effects.

If IFCON = 2, the connection between CLN-node and matrix node is computed across a leakance term as done in the Conduit Flow Process Package of MODFLOW-2005, with leakance input to the model.

If IFCON = 3, the connection between CLN-node and matrix node is computed across a leakance term as done in the Conduit Flow Process Package of MODFLOW-2005, with skin conductivity and skin thickness input to the model and leakance computed internally as per CLN cross-sectional geometry.

If IFCON = 4, the Thiem equation is used to provide the connection between CLN-node and matrix node as was done in the Multi-Node Well Package of MODFLOW-2005, with inclusion of well efficiency.

FSKIN—This parameter determines the leakance across a skin, depending on which equation is selected to represent the flow between CLN cell and matrix.

If IFCON=0, the value of FSKIN is ignored and the skin resistance is taken as zero.

If IFCON=1, the value of FSKIN is the skin factor for a CLN-matrix connection that uses the Thiem Equation with skin resistance.

If IFCON=2, the value of FSKIN is the leakance of the sediments (skin) between the CLN and the matrix node for a CLN-matrix connection as used in the Conduit Flow Process Package of MODFLOW-2005.

If IFCON=3, the value of FSKIN is the hydraulic conductivity of the sediments (skin) between the CLN node and the matrix for computing the

CLN-matrix leakance as used in the Conduit Flow Process Package of MODFLOW-2005.

If IFCON=4, the value of FSKIN is the "well efficiency") between the CLN and matrix connection that uses the Thiem Equation with a well efficiency.

FLENGW—is the length of the connection between the CLN cell and the GW cell.

FANISO—This parameter is used in computation of leakance across a skin, depending on which equation is selected to represent the flow between CLN node and matrix.

If IFCON=0 or 1, the value of FANISO is the anisotropy factor of the porous matrix block that is connected to the CLN-node, used for computations related to the Thiem Equation. The Kx/Ky value is provided here for a vertically oriented CLN cell, and Kx/Kz is provided here for a horizontally oriented CLN cell. These anisotropies may or may not be read in the BCF or LPF packages depending on the selected simulation options, and are therefore input here to accommodate anisotropic computations for flow to wells. This input therefore provides independent control of flow to wells in anisotropic media.

If IFCON=2, the value of FANISO is not used.

If IFCON=3, the value of FANISO is the thickness of sediments (skin) between the CLN and matrix nodes for computing the CLN-matrix leakance as used in the Conduit Flow Process Package of MODFLOW-2005.

ICGWADI—is a flag indicating if vertical flow correction is applied to CLN-GW flow at this node if either cell is dry.

If ICGWADI = 0, CLN-GW flow at this cell is treated without vertical flow correction.

If ICGWADI = 1, vertical flow correction is applied for CLN-GW flow at this node if it is dry.

FLENGW—is the length of the CLN-node's connection with the groundwater node.

ICONDUITYP—is the index for the conduit type.

FRAD—is the radius of the circular conduit type.

IRECYP—is the index for the rectangular geometry type.

FWIDTH—is the width of the rectangular geometry type.

FHEIGHT—is the height of the rectangular geometry type.

CONDUITK—is the hydraulic conductivity or resistance factor of the conduit. The value that is entered depends on the equation being used to solve for flow through the conduit.

If the magnitude of IFLIN = 1, or if IFLIN = 0, the CLN flow is treated as laminar, and CONDUITK represents the laminar conductance term of the CLN segment expressed as $ConduitK = \left(\rho g / 8\mu\right)$. Note that CONDUITK times radius squared is used to compute the effective hydraulic conductivity of the conduit, as per laminar flow equation. CONDUITK has dimensions of [1/LT].

If the magnitude of IFLIN = 2, the CLN-node is treated as turbulent flow using the Darcy-Weisbach equation, and CONDUITK represents the mean roughness height of the CLN surface [L].

If the magnitude of IFLIN = 3, the CLN-node is treated as turbulent flow using the Hazen-Williams equation, and CONDUITK represents the Hazen-Williams factor for relative roughness of the CLN surface [$L^{0.37}$/T].

If the magnitude of IFLIN = 4, the CLN-node is treated as turbulent flow using the Manning's equation, and CONDUITK represents the Manning's coefficient for relative roughness of the CLN surface [$T/L^{(1/3)}$].

NCLNNDS—is the total number of CLN-nodes. This parameter is computed internally by the code.

$$NCLNNDS = ICLNNDS \qquad for \qquad ICLNNDS \geq 0$$

$$NCLNNDS = \sum_{NLFF} NNDCLN(NCLN) \qquad for \qquad ICLNNDS < 0$$

IBOUND—is the boundary array for CLN-nodes.

If IBOUND(IFN) < 0, CLN-cell IFN has a constant head.

If IBOUND(IFN) = 0, CLN-cell IFN is no flow.

If IBOUND(IFN) > 0, CLN-cell IFN is variable head.

STRT—is initial (starting) head—that is, head at the beginning of the simulation. STRT must be specified for all simulations, including steady-state simulations. One value is read for every CLN cell.

ICBUND—is the active or inactive flag for transport for the CLN-nodes.

DLL—is the dispersion coefficient along the conduit-node direction, if transport is simulated.

DLM—is the dispersion coefficient for the conduit-node flow to/from the porous matrix, if transport is simulated.

CONC—is the initial concentration of species at the conduit-node.

Evapotranspiration Package Input Instructions

Input to the Evapotranspiration (EVT) Package is read from the file that is type "EVT" in the Name File. All single-valued variables are free format if the option

"FREE" is specified in the Basic Package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPEVT]

This optional item must start with the word "PARAMETER".

The third entry in item 2a below (IETFACTOR) is required only if transport simulation is active (ITRNSP ≠ 0)

2a. NEVTOP IEVTCB IETFACTOR

Item 2b is read only if IUNSTR = 1 and NEVTOP = 2

2b. MXNDEVT

Item 2c is read only if IETFACTOR = 1 and transport simulation is active (ITRNSP ≠ 0)

2c. ETFACTOR(MCOMP)

The factors are input in free format if the option "FREE" is specified in the Basic Package input file; otherwise, the variables all have 10-character fields.

3. [PARNAM PARTYP Parval NCLU [INSTANCES NUMINST]]

Repeat Item 3 combined with the indicated repetitions of Item 4 NPEVT times. Items 3 and 4 are not read if NPEVT is negative or 0.

If PARNAM is to be a time-varying parameter, the keyword "INSTANCES" and a value for NUMINST must be entered.

4a. [INSTNAM]

4b. [MItarr Zonarr IZ]

After each Item 3 for which the keyword "INSTANCES" is not entered, read Item 4b and not Item 4a.

After each Item 3 for which the keyword "INSTANCES" is entered, read Item 4a and Item 4b for each instance.

NCLU repetitions of Item 4b are required. Each repetition of Item 4 is called a parameter cluster. The NCLU repetitions of Item 4b follow each repetition of Item 4a when PARNAM is time varying.

FOR EACH STRESS PERIOD

Read item 5a only if NEVTOP = 2

5a. INSURF INEVTR INEXDP INIEVT

Read item 5b only if NEVTOP is not 2.

5b. INSURF INEVTR INEXDP INIZNEVT

If UNSTRUCTURED option is used then read items 6 THROUGH 10.

6. [SURF(INIEVT)] -- U2DREL If INSURF ≥ 0

7. [EVTR(INIEVT)] -- U2DREL If NPEVT = 0 and if INEVTR ≥ 0

8. [Pname [Iname] [IEVTPF]] -- if NPEVT > 0 and if INEVTR > 0

Either Item 7 or Item 8 may be read, but not both. Item 8, if read, is repeated INEVTR times. Iname is read if Pname is a time-varying parameter.

9. [EXDP(INIEVT)] -- U2DREL If INEXDP ≥ 0

10. [IEVT(INIEVT)] -- U2DINT If NEVTOP = 2 and if INIEVT ≥ 0

Note in items 6 through 10 that that INIEVT is equal to the number of nodes in the top layer if INIEVT = 1 or 3. Items 6 through 10 are read for unstructured input only.

Otherwise, if UNSTRUCTURED option is not used then read items 11 through 15 for structured input.

11. [SURF(NCOL,NROW)] -- U2DREL If INSURF ≥ 0

12. [EVTR(NCOL,NROW)] -- U2DREL If NPEVT = 0 and if INEVTR ≥ 0

13. [Pname [Iname] [IEVTPF]] -- if NPEVT > 0 and if INEVTR > 0

Either Item 12 or Item 13 may be read, but not both. Item 13, if read, is repeated INEVTR times. Iname is read if Pname is a time-varying parameter.

14. [EXDP(NCOL,NROW)] -- U2DREL If INEXDP ≥ 0

15. [IEVT(NCOL,NROW)] -- U2DINT If NEVTOP = 2 and if INIEVT ≥ 0

Explanation of Variables Read by the EVT Package:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

NPEVT—is the number of evapotranspiration parameters.

NEVTOP—is the evapotranspiration (ET) option code. ET variables (ET surface, maximum ET rate, and extinction depth) are specified in layer variables, SURF, EVTR, and EXDP, with one value for each vertical column. Accordingly,

ET is calculated for one cell in each vertical column. The option codes determine the cell within a column for which ET will be calculated.

1—ET is calculated only for cells in the top grid layer.

2—The cell for each vertical column is specified by the user in variable IEVT.

3—ET is applied to the highest active cell in each vertical column. A constant-head node supplies the required ET and prevents ET from the domain. Note that if the top-most layer is inactive for unstructured grids, this option assigns all ET flux to the first active node in the layers below. Hence if there is a vertical nesting involved, with multiple active nodes underlying an inactive node then the ET is not spread over all of these underlying active underlying nodes. The quantity of water however, is conserved for the prescribed maximum ET rate.

IEVTCB—is a flag and a unit number.

If IEVTCB > 0, cell-by-cell flow terms will be written to this unit number when "SAVE BUDGET" or a nonzero value for ICBCFL is specified in Output Control.

If IEVTCB ≤ 0, cell-by-cell flow terms will not be written.

IETFACTOR—is a flag indicating if ET removes contaminant mass with water or not.

If IEVTCB > 0, contaminant mass can be removed with evapotranspiration and the factor ETFACTOR is read for each component in data item 2c. ETFACTOR determines the fraction of mass of the component that leaves with water. Thus, if ET removes water but leaves behind the solutes, the ETRACTOR is zero. At the other extreme, if all solutes leave with water then the ET factor is set to 1.

If IEVTCB = 0, ET leaves behind contaminant mass and only water is removed from the modeled system. The ETFACTOR is automatically set to zero and is not read.

If IEVTCB < 0, contaminant mass is fully removed with evapotranspiration. The ETFACTOR is automatically set to 1 and is not read.

MXNDEVT—is the maximum number of nodes on which ET is applied in a simulation. This parameter is read only when IUNSTR=1 (for a unstructured grid) with NEVTOP=2 (whereby the nodes on which ET is applied are a user input).

ETFACTOR(MCOMP)—is the fraction of mass of the component that leaves with water.

If ETFACTOR = 0 then water exits the domain via the ET boundary but all component mass is left behind.

If ETFACTOR = 1 then mass of component leaves with water as is the case in other outflow boundary conditions.

If ETFACTOR ≥ 0 or ≤ 1 then only the prescribed fraction of component mass leaves the domain with water at the ET boundary.

PARNAM—is the name of a parameter to be defined. This name can consist of 1 to 10 characters and is not case sensitive; that is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the EVT Package, the only allowed parameter type is EVT, which defines values of the maximum ET flux, variable EVTR.

Parval—is the parameter value. This parameter value may be overridden by a value in the Parameter Value File.

NCLU—is the number of clusters required to define a non-time-varying parameter or one instance of a time-varying parameter. Each repetition of Item 4b is a cluster (variables Mltarr, Zonarr, and IZ). Usually only one cluster is used to define an EVT non-time-varying parameter or an instance of a time-varying parameter; however, more than one cluster is acceptable.

INSTANCES—is an optional keyword that designates a parameter as time varying. The keyword is not case sensitive; that is, any combination of the same characters with different case can be used. If ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, PARNAM is non-time-varying and NUMINST should not be present.

NUMINST—is the number of instances for a time-varying parameter, where each instance is a list of river reaches and associated properties. If the keyword ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, NUMINST should not be present.

INSTNAM—is the name of an instance associated with the parameter named in the corresponding Item 3. The instance name can be 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent. Instance names must be unique for a parameter, but instance names may be reused for different parameters.

Mltarr—is the name of the multiplier array to be used to define the values that are determined by a parameter. The name "NONE" means that there is no multiplier array, and the values will be set equal to Parval.

Zonarr—is the name of the zone array to be used to define the cells that are associated with a parameter. The name "ALL" means that there is no zone array, and all cells are associated with the parameter.

IZ—is up to 10 zone numbers (separated by spaces) that define the cells that are associated with a parameter. These values are not used if Zonarr is specified

as "ALL." Values can be positive or negative, but 0 is not allowed. The end of the line, a zero value, or a non-numeric entry terminates the list of values.

INSURF—is the ET surface (SURF) read flag.

If INSURF≥ 0, a layer variable containing the ET surface elevation (SURF) will be read.

If INSURF < 0, the ET surface from the preceding stress period will be reused.

INEVTR—is the EVTR read flag. Its function depends on whether or not parameters are being used.

If no parameters are being used (NPEVT=0):

If INEVTR ≥ 0, a layer variable containing the maximum ET rate (EVTR) will be read.

If INEVTR < 0, the maximum ET rate from the preceding stress period will be reused.

If parameters are being used (NPEVT>0):

If INEVTR > 0, INEVTR is the number of parameters that will be used to define EVTR in the current stress period. Item 8 defines the names of the parameters.

If INEVTR < 0, EVT parameters from the preceding stress period are used.

INEVTR = 0 is not allowed. That is, when parameters are used, at least one parameter must be specified each stress period

INEXDP—is the extinction depth (EXDP) read flag.

If INEXDP ≥ 0, a layer variable containing the extinction depth (EXDP) will be read.

If INEXDP < 0, the extinction depth from the preceding stress period will be reused.

INIEVT—is the layer indicator (IEVT) read flag that is read only if the ET option (NEVTOP) is equal to two.

If INIEVT ≥ 0, an array containing the layer indicators (IEVT) will be read for a structured grid. For an unstructured grid, INIEVT is further equal to the number of nodes for which ET values are read in the simulation, with the nodes being identified in the array IEVT. For an unstructured grid, INIEVT is further equal to the number of nodes for which ET values are read in the simulation, with the nodes being identified in the array IEVT.

If INIEVT < 0, layer indicators used during the preceding stress period will be reused.

SURF—is the elevation of the ET surface. This variable is read only if INSURF ≥ 0

EVTR—is the maximum ET flux [volumetric flow rate per unit area (LT-1)]. This variable is read only if INEVTR ≥ 0 and if NPEVT=0. Contrary to the usual convention in MODFLOW, EVTR values should be specified as positive values even though they represent an outflow from the groundwater system.

Pname—is the name of a parameter that will be used to define the EVTR variable in the current stress period. Read INEVTR values if NPEVT > 0 and INEVTR > 0.

Iname—is an instance name that is read only if Pname is a time-varying parameter. Multiple instances of the same parameter are not allowed in a stress period.

IEVTPF—is an optional format code for printing the EVTR variable after it has been defined by parameters. The format codes are the same as those used in the U2DREL array reading utility subroutine.

EXDP—is the ET extinction depth. This variable is read only if INEXDP ≥ 0.

IEVT—is the layer indicator variable. For each horizontal location, IEVT indicates the layer from which ET is removed, when a structured MODFLOW grid is used (IUNSTR=0). For an unstructured grid input (IUNSTR=1), IEVT is the node number on which the ET is applied, where the list includes INIEVT number of nodes. Read only if NEVTOP is two and if INIEVT is greater than or equal to zero.

Dual Porosity Transport Package Input Instructions

Input for the Dual Porosity Transport (DPT) Package is read from the file that is type "DPT" in the Name File. The DPT Package input instructions include flags, indices, transport related parameters and initial conditions for solution to transport of contaminants in a steady-state or transient flow field, for the immobile (matrix) domain of the dual porosity system. Input for the mobile (fracture) domain is supplied with the BCT package. The input for DPT is read in a free format. Also note that only UNSRUCTURED input formats are available for the DPT package, even if the rest of the problem was set up as a STRUCTURED finite-difference system.

FOR EACH SIMULATION

1. **IDPTCB IDPTCON ICBNDIMFLG IADSORBIM IDISPIM IZODIM IFODIM** [**OPTIONS**]

These seven variables are free format if the option “FREE” is specified in the Basic Flow Package input file; otherwise, the variables all have 10-character fields.

The rest of the file contains arrays of dual porosity transport properties. The variables needed for each cell depend on the transport options that are selected. Unneeded variables must be omitted. The required variables are read for all layers before reading the next variable. Note that there may be different number of nodes per layer (NDSLAY) for an unstructured grid.

If ICBNDIMFLG is not zero then read item 2.

2. **[ICBUNDIM(NDSLAY)]** – U1DINT. Read one array for each layer.

If IDPF is zero (flow solution was not dual porosity) then read item 3.

3. **[PHIF(NDSLAY)]** – U1DINT. Read one array for each layer.

4. **[PRSITYIM(NDSLAY)]** – U1DREL. Read one array for each layer.

If IADSORBIM is not zero then read item 5.

5. **[BULKDIM(NDSLAY)]** – U1DREL. Read one array for each layer.

If IDPF is not zero (flow solution was dual porosity) then read item 6.

6. **[DLIM(NDSLAY)]** – U1DREL. Read one array for each layer.

7. **[DDTTR(NDSLAY)]** – U1DREL. Read one array for each layer.

If IDPF is zero (flow solution was not dual porosity) and the optional keyword INPUTSAT was read on item 1, then read item 8.

8. **[SIM(NDSLAY)]** – U1DREL. Read one array for each layer.

Items 9 – 15 are read for the first component, followed by the next, till all MCOMP contaminant species are read.

If IADSORBIM is not zero then read item 9.

9. **[ADSORBIM(NDSLAY,ICOMP)]** – U1DREL. Read one array for each layer.

If IADSORBIM is two then read item 10.

10. **[FLICHIM(NDSLAY,ICOMP)]** – U1DREL. Read one array for each layer.

If IZODIM is one or three, then read item 11.

11. **[ZODRWIM(NDSLAY,ICOMP)]** – U1DREL Read one array for each layer

If IADSORBIM is not zero and IZODIM is two or three, then read item 12.

12. **[ZODRSIM(NDSLAY,ICOMP)]** – U1DREL Read one array for each layer

If IFODIM is one or three, then read item 13.

13. **[FODRWIM(NDSLAY,ICOMP)]** – U1DREL Read one array for each layer

If IADSORBIM is not zero and IFODIM is two or three, then read item 14.

14. **[FODRS(NDSLAY,ICOMP)]** – U1DREL Read one array for each layer

15. **[CONC(NDSLAY,ICOMP)]** – U1DREL. Read one array for each layer.

Explanation of Variables Read by the DPT Package:

IDPTCB—is a flag and a unit number.

If IDPTCB > 0, cell-by-cell mass flux terms will be written to this unit number when "SAVE BUDGET" or a nonzero value for ICBCFL is specified in Output Control. The terms that are saved are mass storage, mass flux from constant-concentration nodes, and mass flux between adjacent cells.

If IBDPTCB = 0, cell-by-cell mass flux terms will not be written.

If IDPTCB < 0, cell-by-cell mass flux for constant-concentration cells will be written in the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. Cell-by-cell mass flux to storage and between adjacent cells will not be written to any file.

IDPTCON—is a flag and a unit number.

If IDPTCON > 0, immobile domain concentrations will be written to this unit number in a binary format when "SAVE CONC" or a nonzero value and ICONSV is specified in Output Control.

If IDPTCON = 0, immobile domain concentrations will not be saved.

If IDPTCON < 0, immobile domain concentrations will be written to the listing file.

ICBNDIMFLG—is a flag that determines if the active domain for the immobile (matrix) domain for transport is the same as that for the mobile (fracture) domain for transport.

If ICBNDIMFLG = 0, the active domain for immobile domain is not the same as for the mobile domain and is read.

If ICBNDIMFLG = 1, the active domain for the immobile domain is the same as IBOUND for the mobile domain for transport and the transport IBOUND array does not need to be read but is set from that of flow.

IADSORBIM—is an adsorption flag.

If IADSORBIM = 0, then adsorption is not simulated.

If IADSORBIM = 1, then linear adsorption is simulated.

If IADSORBIM = 2, then Freundlich adsorption is simulated.

IDISPIM—is a flag indicating the dispersion formula used in the model

If IDISPIM = 0, then dispersion is not simulated.

If IDISPIM = 1, then dispersion is simulated.

IZODIM—is a flag indicating if zero order decay is simulated

If IZODIM = 0, then zero order decay is not simulated.

If IZODIM = 1, then zero order decay is included in water.

If IZODIM = 2, then zero order decay is included on soil if there is adsorption.

If IZODIM = 3, then zero order decay is included in water and on soil if there is adsorption.

IFODIM—is a flag indicating if first order decay is simulated

If IFODIM = 0, then first order decay is not simulated.

If IFODIM = 1, then first order decay is included in water.

If IFODIM = 2, then first order decay is included on soil if there is adsorption.

If IFODIM = 3, then first order decay is included in water and on soil if there is adsorption.

OPTIONS—are optional keywords that activate options. Note that these options are required only if the flow simulation is also not dual porosity and indicates otherwise, how to handle the flow simulation parameters.

FRAHK—indicates that the hydraulic conductivity and storage terms (for transient simulations) which are input are only for the fracture (mobile) domain. Otherwise, it is assumed that the hydraulic conductivity and storage input for the flow simulation are effective hydraulic conductivity and storage terms for the medium, representative of the entire volume.

MOBILESAT—the optional keyword MOBILESAT indicates that the immobile domain saturation will be set equal to the initial mobile domain saturation. Otherwise, the immobile domain saturation is set to unity. Note that for transport in a transient flow-field, the immobile domain saturation is not varied through time to maintain mass balance of flow, since no water exchange is assumed between the domains.

INPUTSAT—the optional keyword INPUTSAT indicates that the immobile domain saturation will be input in this dataset.

ICBUNDIM—is the boundary variable. One value is read for every model cell. Usually, these values are read one layer at a time; however, when the XSECTION option is specified, a single two-dimensional variable for the cross section is read. Note that although ICBUNDIM may be read as one or more two-dimensional variables, it is stored internally only as a one-dimensional variable for all nodes in the domain. Note that ICBUNDIM is read only when inactive sections of the immobile domain are different from the corresponding inactive sections of the mobile domain.

If ICBUNDIM(N) = 0, cell N is inactive in the immobile domain (either the cell itself is inactive or the immobile domain does not exist at this cell).

If ICBUNDIM(N) > 0, cell N is active in the immobile domain.

Note that ICBUND can be inactive for active flow nodes. However, ICBUND cannot be active for a node which is inactive for the corresponding flow simulation.

PHIF—is the mobile fraction. i.e., the fraction of the total space that is occupied by the mobile domain.

PRSITYIM—is the effective transport porosity of the immobile domain.

BULKDIM—is the bulk density of the porous matrix of the immobile domain.

DLIM—is the longitudinal dispersivity coefficient for transport between mobile and immobile domains when flow is also dual porosity.

DDTR—is the mass transfer coefficient for diffusive transport between mobile and immobile domains.

SNIM—is the saturation in the immobile domain which is input when dual domain flow is not solved.

ADSORBIM—is the adsorption coefficient of a contaminant species in the immobile domain.

FLICHIM—is the Freundlich adsorption isotherm exponent of a contaminant species in the immobile domain.

ZODRWIM—is the zero-order decay coefficient in water (concentration/time) in the immobile domain.

ZODRSIM—is the zero-order decay coefficient on soil (concentration/time) in the immobile domain.

FODRWIM—is the first-order decay coefficient in water (1/time) in the immobile domain.

FODRSIM—is the first-order decay coefficient on soil (1/time) in the immobile domain.

CONCIM—is the initial concentration of each contaminant species at any location in the immobile domain

Density Driven Flow (DDF) Package Input Instructions

Input to the Density Driven Flow (DDF) Package is read from the file that has type "DDF" in the Name File. All single valued variables are free format if the option “FREE” is specified in the Basic Package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

For density driven flow, the boundary files for CHD and GHB are also modified. The modification is to allow options for input of different “head” values. Specifically, the user can input the hydraulic head or the potential head (defined above). Options govern which head is defined and conversion is done internally by the code as needed.

FOR EACH SIMULATION

0. [**#Text**]
Item 0 is optional—“#” must be in column 1. Item 0 can be repeated multiple times.

1. **RHOFRESH, RHOSTD, CSTD, ITHICKAV, IMPH**

Explanation of Variables Read by the DDF Package:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The “#” character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

RHOFRESH—is the density of freshwater

RHOSTD—is the density of standard solution

CSTD—is the concentration of standard solution

ITHICKAV—is a flag indicating if thickness weighted averaging should be used for the density term

ITHICKAV = 0 if arithmetic averaging is used.

ITHICKAV = 1 if thickness weighting averaging is used.

IMPHDD—is a flag indicating if the hydraulic head term in the density formulation is to be treated implicitly or explicitly

IMPHDD = 0 if treatment of the head term is explicit (on the right-hand side vector) maintaining symmetry of the matrix.

IMPHDD = 1 if treatment of the head term is implicit (on the left-hand side matrix) thus creating an asymmetric matrix.

ADDITIONAL CUSTOM PACKAGES

Development of USG-Transport has included further customization of the flow routines to accommodate additional capabilities and flexibility. A brief introduction to these customized solutions and associated input and output formats 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.

Transient IBOUND Package

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.*

Input to the TIB Package is read from the file that has type "TIB" in the Name File. All single valued variables are free format if the option “FREE” is specified in the Basic Package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format. Cell numbers used to identify grid-blocks that turn on or off are input in terms of the global cell number. Note that GWF cells are first numbered followed by CLN cells in the global numbering sequence. For structured GWF grids, the node numbering first follows columns, then rows and finally layers (as is the convention in MODFLOW).

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

FOR EACH STRESS PERIOD

1. NIB0 NIB1 NIBM1 NICB0 NICB1 NICBM1

Read item 2 only if NIB0 is greater than zero

2. [IBO(NIB0)] -- U1DINT

Read item 3 only if NIB1 is greater than zero

3. IB1 [OPTIONS]

Note that item 3 is read NIB1 times, one record for each node for which IBOUND is activated.

Read item 4 only if NIBM1 is greater than zero

4. IBM1 [OPTIONS]

Note that item 4 is read NIBM1 times, one record for each node for which IBOUND is made into a prescribed head.

Read item 5 only if NICB0 is greater than zero

5. [ICBO(NICB0)] -- U1DINT

Read item 6 only if NICB1 is greater than zero

6. ICB1 [**OPTIONS2**]

Note that item 6 is read NICB1 times, one record for each node for which ICBUND is activated.

Read item 7 only if NICBM1 is greater than zero

7. ICBM1 [**OPTIONS2**]

Note that item 7 is read NICBM1 times, one record for each node for which ICBUND is made into a prescribed concentration.

Explanation of Variables Read by the TIB Package:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

NIB0—is the number of IBOUND values that will be turned to zero in this stress period.

NIB1—is the number of IBOUND values that will be turned to active (one) in this stress period.

NIBM1—is the number of IBOUND values that will be turned to prescribed head (minus one) in this stress period.

NICB0—is the number of ICBUND values that will be turned to zero in this stress period.

NICB1—is the number of ICBUND values that will be turned to active (one) in this stress period.

NICBM1—is the number of ICBUND values that will be turned to prescribed concentration (minus one) in this stress period.

IB0—is the array of cell numbers for which IBOUND values will be set to zero.

IB1—is the cell number for which IBOUND is set to active (one).

OPTIONS — are keywords to indicate how head values will be set for nodes that are activated:

HEAD [hvalue] indicates that the head value is input and the number following the keyword HEAD is the value to be used as the initial head at the given cell.

AVHEAD indicates that the average head of all connected cells that are not inactive, will be used as the initial head at the given cell.

Note that if no options are provided, then the head from the previous stress period will be used at this node which is turned active. If the node was inactive in the previous stress period, an option is required to provide a head value to the cell that is made active or prescribed head, otherwise the simulation will be aborted.

IBM1—is the cell number for which IBOUND is set to prescribed head (minus one). Note that the same options follow IBMI as do IB1 for each cell for which the head will be prescribed in this stress period.

ICB0—is the array of cell numbers for which ICBUND values will be set to zero.

ICB1—is the cell number for which ICBUND is set to active (one). If IBOUND for the cell is 0 (inactive cell), then the simulation will be aborted with an appropriate error message.

OPTIONS2 — are keywords to indicate how concentration values will be set for nodes that are activated:

CONC [cvalue(mcomp)] indicates that the concentration value(s) for each species (a total of mcomp numbers) is input and the number(s) following the keyword CONC is the value to be used as the initial concentration at the given cell.

AVCONC indicates that the average concentration of all connected cells that are not inactive (based on ICBUND), will be used as the initial concentration at the given cell.

Note that if no options are provided, then the concentration from the previous stress period will be used at this node which is turned active. If the node was inactive in the previous stress period, an option is required to provide a concentration value to the cell that is made active or prescribed concentration, otherwise the simulation will be aborted.

ICBM1—is the cell number for which ICBUND is set to prescribed concentration (minus one). Note that the same options follow ICBMI as do ICB1 for each cell for which the concentration will be prescribed in this stress period.

Recharge (RCH) Package

Input to the Recharge (RCH) Package is read from the file that has type "RCH" in the Name File. All single valued variables are free format if the option “FREE” is specified in the Basic Package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

If transport is simulated with the recharge package, it is assumed by default that concentration in recharge water is zero. If it is required to be non-zero for any species of simulation, the optional keyword CONCENTRATION (or CONC) allows for further input of species that have non-zero concentration, and the concentration array for those species.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—“#” must be in column 1. Item 0 can be repeated multiple times.

1. [***PARAMETER*** NPRCH]

This optional item must start with the word “PARAMETER”.

2a. NRCHOP IRCHCB [OPTIONS1]

Item 2b is read only if UNSTRUCTURED option is used and NRCHOP =2

2b. MXNDRCH

Item 2c is read only if option “CONCENTRATION” or “CONC” is used for the simulation

2c. IRCHCONC(MCOMP)

3. [PARNAM PARTYP Parval NCLU [INSTANCES NUMINST]]

Repeat Item 3 combined with the indicated repetitions of Item 4 NPRCH times. Items 3 and 4 are not read if NPRCH is negative or 0. If PARNAM is to be a time-varying parameter, the keyword “INSTANCES” and a value for NUMINST must be entered.

4a. [INSTNAM]

4b. [Mltarr Zonarr IZ]

After each Item 3 for which the keyword “INSTANCES” is not entered, read Item 4b and not Item 4a.

After each Item 3 for which the keyword “INSTANCES” is entered, read Item 4a and Item 4b for each instance. NCLU repetitions of Item 4b are required. Each repetition of Item 4b is called a parameter cluster. The NCLU repetitions of Item 4b follow each repetition of Item 4a when PARNAM is time varying.

FOR EACH STRESS PERIOD

Read item 5a only if NRCHOP = 2

5a. INRECH INIRCH

Read item 5b only if NRCHOP is not 2.

5b. INRECH

If UNSTRUCTURED option is used, then read items 6a and 6b.

6a. [RECH(INIRCH)] -- U1DREL if NPRCH=0 and if INRECH $\geq$ 0

Note that INIRCH is equal to the number of nodes in the top layer if NRCHOP =1 or 3.

6b. [Pname [Iname] [IRCHPF]] -- if NPRCH > 0 and if INRECH > 0

Either Item 6a or Item 6b may be read, but not both. Item 6b, if read, is repeated INRECH times. Iname is read if Pname is a time-varying parameter.

Otherwise, if UNSTRUCTURED option is not used, then read items 7a anf 7b for structured input.

7a. [RECH(NCOL,NROW)] -- U2DREL if NPRCH=0 and if INRECH $\geq$ 0

7b. [Pname [Iname] [IRCHPF]] -- if NPRCH > 0 and if INRECH > 0

Either Item 7a or Item 7b may be read, but not both. Item 7b, if read, is repeated INRECH times. Iname is read if Pname is a time-varying parameter.

If UNSTRUCTURED option is used, then read item 8a

8a. [IRCH(INIRCH)] -- U1DINT If NRCHOP=2 and if INIRCH > = 0

Otherwise, if UNSTRUCTURED option is not used, then read item 8b for structured input

8b. [IRCH(NCOL,NROW)] -- U2DINT If NRCHOP=2 and if INIRCH > = 0

Item 9 is read only if the flag on option2 “INCONC” is supplied. Item 11 is read as many times as the flag IRCHCONC(ICOMP) > 0. Item 11 is the array of concentrations for each of the non-zero concentration components.

If UNSTRUCTURED option is used, then read item 9a

9a. [RCHCONC(INIRCH)] -- U1DREL

Otherwise, if UNSTRUCTURED option is not used, then read item 9b for structured input

9b. [RCHCONC(NCOL,NROW)] -- U2DREL

Explanation of Variables Read by the RCH Package:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The “#” character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

NPRCH—is the number of recharge parameters.

NRCHOP—is the recharge option code. Recharge fluxes are defined in a layer variable, RECH, with one value for each vertical column. Accordingly, recharge is applied to one cell in each vertical column, and the option code determines which cell in the column is selected for recharge.

1—Recharge is only to the top grid layer.

2—Vertical distribution of recharge is specified in layer variable IRCH.

3—Recharge is applied to the highest active cell in each vertical column. A constant-head node intercepts recharge and prevents deeper infiltration. Note that if the top-most layer is inactive for unstructured grids, this option assigns all recharge flux to the first active node in the layers below. Hence, if there is a vertical nesting involved, with multiple active nodes underlying

an inactive node, then the recharge is not spread over all of these underlying active underlying nodes. The quantity of water, however, is conserved for the prescribed recharge rate.

IRCHCB—is a flag and a unit number.

If IRCHCB > 0, cell-by-cell flow terms will be written to this unit number when "SAVE BUDGET" or a nonzero value for ICBCFL is specified in Output Control.

If IRCHCB $\leq$ 0, cell-by-cell flow terms will not be written.

OPTIONS1 are keywords that activate options:

CONCENTRATION (or CONC) indicates that species concentrations will also be input for transport simulations. If this option is not present, the concentration of all species in recharge water is assumed to be zero.

MXNDRCH—is the maximum number of nodes to which recharge is applied in a simulation. This parameter is read only when the UNSTRUCTURED option is used with NRCHOP=2 (whereby the nodes to which recharge is applied are a user input).

PARNAM—is the name of a parameter to be defined. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the RCH Package, the only allowed parameter type is RCH, which defines values of the recharge flux.

Parval—is the parameter value. This parameter value may be overridden by a value in the Parameter Value File.

NCLU—is the number of clusters required to define a non-time-varying parameter or one instance of a time varying parameter. Each repetition of Item

4b is a cluster (variables Mltarr, Zonarr, and IZ). Usually only one cluster is used to define a RCH non-time-varying parameter or an instance of a time-varying parameter; however, more than one cluster is acceptable.

INSTANCES—is an optional keyword that designates a parameter as time varying. The keyword is not case sensitive; that is, any combination of the same characters with different case can be used. If ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, PARNAM is non-time-varying and NUMINST should not be present.

NUMINST—is the number of instances for a time-varying parameter, where each instance is a list of river reaches and associated properties. If the keyword ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, NUMINST should not be present.

INSTNAM—is the name of an instance associated with the parameter named in the corresponding Item 3. The instance name can be 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent. Instance names must be unique for a parameter, but instance names may be reused for different parameters.

Mltarr—is the name of the multiplier array to be used to define cell values that are determined by a parameter. The name “NONE” means that there is no multiplier array, and the cell values will be set equal to Parval.

Zonarr—is the name of the zone array to be used to define the cells that are associated with a parameter. The name “ALL” means that there is no zone array, and all cells in the layer are associated with the parameter.

IZ—is up to 10 zone numbers (separated by spaces) that define the cells that are associated with a parameter. These values are not used if Zonarr is specified as “ALL.” Values can be positive or negative, but 0 is not allowed. The end of the line, a zero value, or a non-numeric entry terminates the list of values.

INRECH—is the RECH read flag. Its function depends on whether or not parameters are being used.

If no parameters are being used (NPRCH = 0):

If INRECH $\geq$ 0, a layer variable of recharge fluxes, RECH, is read.

If INRECH < 0, recharge rates from the preceding stress period are used.

If parameters are being used (NPRCH > 0):

If INRECH > 0, INRECH is the number of parameters that will be used to define RECH in the current stress period. Items 6b (for unstructured grids) or 7b (for structured grids) define the names of the parameters.

If INRECH < 0, recharge parameters from the preceding stress period are used.

INRECH = 0 is not allowed. That is, when parameters are used, at least one parameter must be specified each stress period.

INIRCH—is the IRCH read flag, which is read only if NRCHOP is two:

If INIRCH $\geq$ 0, a layer variable of layer numbers (IRCH) is read for a structured grid. For an unstructured grid, INIRCH is further equal to the number of nodes for which recharge values are read in the simulation, with the nodes being identified in array IRCH.

If INIRCH < 0, the variable (IRCH) used in the preceding stress period is reused.

INCONC indicates that concentrations will be input for species with non-zero concentrations in recharge water for transport simulations. If this option is not provided, concentrations of components in recharge water will not be read for this stress period and will be assumed to be the same as for the previous stress period.

RECH—is the recharge flux (LT^{-1}). Read only if INRECH is greater than or equal to zero and if NPRCH = 0.

Pname—is the name of a parameter that will be used to define the RECH variable in the current stress period. Read INRECH values if NPRCH > 0 and INRECH > 0.

Iname—is an instance name that is read only if Pname is a time-varying parameter. Multiple instances of the same parameter are not allowed in a stress period.

IRCHPF—is an optional format code for printing the RECH variable after it has been defined by parameters. The format codes are the same as those used in the U2DREL array reading utility subroutine.

IRCH—is the layer number that defines the layer in each vertical column where recharge is applied when a structured MODFLOW grid is used. For an unstructured grid, IRCH is the node number to which the recharge is applied, where the list includes INIRCH number of nodes. Read only if NRCHOP is two and if INIRCH is greater than or equal to zero.

Segmented Evapotranspiration (ETS) Package

The Segmented Evapotranspiration (ETS) 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.]

Input to the Segmented Evapotranspiration (ETS) Package is read from the file that is type "ETS" in the Name File. All single-valued variables are free format if the option “FREE” is specified in the Basic Package input file; otherwise, the

non-optional variables have 10-character fields and the optional variables are free format.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [***PARAMETER*** NPETS]

This optional item must start with the word "PARAMETER".

2a. NETSOP IETSCB NPETS NETSEG IESFACTOR

Item 2b is read only for an unstructured grid and NETSOP = 2

2b. MXNDETS

Item 2C is read only if transport is active and IESFACTOR=1

2b. ESFACTOR(MCOMP)

3. [PARNAM PARTYP Parval NCLU [INSTANCES NUMINST]]

Repeat Item 3 combined with the indicated repetitions of Item 4 NPETS times. Items 3 and 4 are not read if NPETS is negative or 0.

If PARNAM is to be a time-varying parameter, the keyword "INSTANCES" and a value for NUMINST must be entered.

4a. [INSTNAM]

4b. [Mltarr Zonarr IZ]

After each Item 3 for which the keyword "INSTANCES" is not entered, read Item 4b and not Item 4a.

After each Item 3 for which the keyword "INSTANCES" is entered, read Item 4a and Item 4b for each instance.

NCLU repetitions of Item 4b are required. Each repetition of Item 4 is called a parameter cluster. The NCLU repetitions of Item 4b follow each repetition of Item 4a when PARNAM is time varying.

FOR EACH STRESS PERIOD

Read item 5a if NETSOP = 2 or if NETSEG > 1.

5a. INETSS INETSR INETSX [INIETS [INSGDF]]

Read item 5b if NETSOP is not equal to 2 and NETSEG =1

5b. INETSS INETSR INETSX [INSGDF]

If UNSTRUCTURED option is used, then read items 6 through 12.

6. [ETSS(INIETS)] -- U2DREL If INETSS ≥ 0

7. [ETSR(INIETS)] -- U2DREL If NPETS = 0 and if INETSR ≥ 0

8. [Pname [Iname] [IETSPF]] -- if NPETS > 0 and if INETSR > 0

Either Item 7 or Item 8 may be read, but not both. Item 8, if read, is repeated INETSR times. Iname is read if Pname is a time-varying parameter.

9. [ETSX(INIETS)] -- U2DREL If INETSX ≥ 0

10. [IETS(INIETS)] -- U2DINT If NETSOP = 2 and if INIETS ≥ 0

Read items 11 and 12 below only if NETSEG > 1. (i.e., there are more than one segment to the ET function), and INSGDF ≥ 0

11. [PXDP(INIETS)] -- U2DREL If NETSET > 1 and INSGDF ≥ 0

12. [PETM(INIETS)] -- U2DINT If NETSET > 1 and INSGDF ≥ 0

Note in items 6 through 12 that INIETS is equal to the number of nodes in the top layer if NETSOP = 1 or 3. Items 6 through 12 are read for unstructured input only.

If NETSEG > 1, (NETSEG − 1) repetitions of items 11 and 12 are read. If NETSEG > 2, items 11 and 12 are read for the uppermost segment intersection, followed by repetitions of items 11 and 12 for successively lower intersections.

Otherwise, if UNSTRUCTURED option is not used, then read items 13 through 19 for structured input

13. [ETSS(NCOL,NROW)] -- U2DREL If INETSS ≥ 0

14. [ETSR(NCOL,NROW)] -- U2DREL If NPETS = 0 and if INETSR ≥ 0

15. [Pname [Iname] [IETSPF]] -- if NPETS > 0 and if INETSR > 0

Either Item 15 or Item 16 may be read, but not both. Item 13, if read, is repeated INETSR times. Iname is read if Pname is a time-varying parameter.

16. [ETSX(NCOL,NROW)] -- U2DREL If INETSX ≥ 0

17. [IETS(NCOL,NROW)] -- U2DINT If NETSOP = 2 and if INIETS ≥ 0

Read items 11 and 12 below only if NETSEG > 1. (i.e., there are more than one segment to the ET function), and INSGDF ≥ 0

18. [PXDP(NCOL,NROW)] -- U2DREL If NETSET > 1 and INSGDF ≥ 0

19. [PETM(NCOL,NROW)] -- U2DINT If NETSET > 1 and INSGDF ≥ 0

If NETSEG > 1, (NETSEG − 1) repetitions of items 18 and 19 are read. If NETSEG > 2, items 18 and 19 are read for the uppermost segment intersection, followed by repetitions of items 18 and 19 for successively lower intersections.

Explanation of Variables Read by the ETS Package:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

NPETS—is the number of evapotranspiration parameters.

NETSOP—is the evapotranspiration (ET) option code. ET variables (ET surface, maximum ET rate, and extinction depth) are specified in layer variables, SURF, ETSR, and EXDP, with one value for each vertical column. Accordingly, ET is calculated for one cell in each vertical column. The option codes determine the cell within a column for which ET will be calculated.

1—ET is calculated only for cells in the top grid layer.

2—The cell for each vertical column is specified by the user in variable IETS.

3—ET is applied to the highest active cell in each vertical column. A constant-head node supplies the required ET and prevents ET from occurring from the domain. Note that if the top-most layer is inactive for unstructured grids, this option assigns all ET flux to the first active node in the layers below. Hence if there is a vertical nesting involved, with multiple active nodes underlying an inactive node then the ET is not spread over all of these active underlying nodes. The quantity of water however, is conserved for the prescribed ET rate if the cell can provide the ET rate without becoming dry.

IETSCB—is a flag and a unit number.

If IETSCB > 0, cell-by-cell flow terms will be written to this unit number when "SAVE BUDGET" or a nonzero value for ICBCFL is specified in Output Control.

If IETSCB $\leq$ 0, cell-by-cell flow terms will not be written.

NETSEG—is the number of segments used to define the relation of evapotranspiration rate to hydraulic head in the interval where the evapotranspiration rate is variable.

IESFACTOR—is flag indicating that soute removal fraction with ET will be read when transport is active.

If IESFACTOR =1, solute removal fraction with ET flux is read in ESFACTOR array for each component simulated.

If IESFACTOR = 0, All solutes are left behind when ET removes water and the ESFACTOR array is not read.

MXNDETS—is the maximum number of nodes on which ET is applied in a simulation. This parameter is read only for an unstructured grid with NETSOP=2 (whereby the nodes on which ET is applied are a user input).

ESFACTOR(MCOMP)—is the array for input of fraction of solute (for all 1 to MCOMP solutes) that is extracted with ET. A value of zero means all solutes are left behind with ET flux only extracting the water. A value of one indicates that solutes leave in an advective manner with the flux of water as it would be extracted by a well. A fraction allows for partial extraction of solutes with the evapotranspiration flux, to allow for partial volatility of solute.

PARNAM—is the name of a parameter to be defined. This name can consist of 1 to 10 characters and is not case sensitive; that is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the ETS Package, the only allowed parameter type is ETS, which defines values of the maximum ET flux, variable ETSR.

Parval—is the parameter value. This parameter value may be overridden by a value in the Parameter Value File.

NCLU—is the number of clusters required to define a non-time-varying parameter or one instance of a time-varying parameter. Each repetition of Item 4b is a cluster (variables Mltarr, Zonarr, and IZ). Usually only one cluster is used to define an ETS non-time-varying parameter or an instance of a time-varying parameter; however, more than one cluster is acceptable.

INSTANCES—is an optional keyword that designates a parameter as time varying. The keyword is not case sensitive; that is, any combination of the same characters with different case can be used. If ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, PARNAM is non-time-varying and NUMINST should not be present.

NUMINST—is the number of instances for a time-varying parameter, where each instance is a list of river reaches and associated properties. If the keyword ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, NUMINST should not be present.

INSTNAM—is the name of an instance associated with the parameter named in the corresponding Item 3. The instance name can be 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent. Instance names must be unique for a parameter, but instance names may be reused for different parameters.

Mltarr—is the name of the multiplier array to be used to define the values that are determined by a parameter. The name “NONE” means that there is no multiplier array, and the values will be set equal to Parval.

Zonarr—is the name of the zone array to be used to define the cells that are associated with a parameter. The name “ALL” means that there is no zone array, and all cells are associated with the parameter.

IZ—is up to 10 zone numbers (separated by spaces) that define the cells that are associated with a parameter. These values are not used if Zonarr is specified as

"ALL." Values can be positive or negative, but 0 is not allowed. The end of the line, a zero value, or a non-numeric entry terminates the list of values.

INETSS—is the ET surface (ETSS) read flag.

If INETSS≥ 0, a layer variable containing the ET surface elevation (ETSS) will be read.

If INETSS < 0, the ET surface from the preceding stress period will be reused.

INETSR—is the ETSR read flag. Its function depends on whether or not parameters are being used.

If no parameters are being used (NPETS=0):

If INETSR ≥ 0, a layer variable containing the maximum ET rate (ETSR) will be read.

If INETSR < 0, the maximum ET rate from the preceding stress period will be reused.

If parameters are being used (NPETS>0):

If INETSR > 0, INETSR is the number of parameters that will be used to define ETSR in the current stress period. Item 15 defines the names of the parameters.

If INETSR < 0, ETS parameters from the preceding stress period are used.

INETSR = 0 is not allowed. That is, when parameters are used, at least one parameter must be specified each stress period

INETSX—is the extinction depth (ETSX) read flag.

If INETSX ≥ 0, a layer variable containing the extinction depth (ETSX) will be read.

If INETSX < 0, the extinction depth from the preceding stress period will be reused.

INIETS—is the layer indicator (IETS) read flag. It is read if the ET option (NETSOP) is equal to two or if NETSEG > 1. If NETSEG = 1 and NETSOP is not equal to two, INIETS is ignored and IETS is not read.

If INIETS ≥ 0, an array containing the layer indicators (IETS) will be read for a structured grid. For an unstructured grid, INIETS is further equal to the number of nodes for which ET values are read in the simulation, with the nodes being identified in the array IETS.

If INIETS < 0, layer indicators used during the preceding stress period will be reused.

INIETS—is the layer indicator (IETS) read flag that is read only if the ET option (NETSOP) is equal to two or if NETSET > 1. If NETSEG = 1 and NETSOP is not equal to two, INIETS is ignored and IETS is not read.

If INIETS ≥0, an array containing the layer indicators (IETS) will be read for a structured grid. For an unstructured grid, INIETS is further equal to the number of nodes for which ET values are read in the simulation, with the nodes being identified in the array IETS.

If INIETS < 0, layer indicators used during the preceding stress period will be reused.

INSGDF—is the segment definition read flag. It is read only if NETSEG > 1.

If INSGDF ≥ 0, two layer variables to define PXDP and PETM for each of (NETSEG – 1) segment intersections are read from items 10 and 11, respectively, of the ETS1 input file.

If INSGDF < 0, PXDP and PETM from the preceding stress period will be reused.

ETSS—is the elevation of the ET surface. This variable is read only if INETSS $\geq$ 0

ETSR—is the maximum ET flux [volumetric flow rate per unit area (LT^{-1})]. This variable is read only if INETSR $\geq$ 0 and if NPETS=0. Contrary to the usual convention in MODFLOW, ETSR values should be specified as positive values even though they represent an outflow from the groundwater system.

Pname—is the name of a parameter that will be used to define the ETSR variable in the current stress period. Read INETSR values if NPETS > 0 and INETSR > 0.

Iname—is an instance name that is read only if Pname is a time-varying parameter. Multiple instances of the same parameter are not allowed in a stress period.

IETSPF—is an optional format code for printing the ETSR variable after it has been defined by parameters. The format codes are the same as those used in the U2DREL array reading utility subroutine.

ETSX—is the ET extinction depth. This variable is read only if INETSX $\geq$ 0.

IETS—is the layer indicator variable. For each horizontal location, IETS indicates the layer from which ET is removed when a structured MODFLOW grid is used. For an unstructured grid, IETS is the node number to which the ET is applied, where the list includes INIETS number of nodes. Read only if NETSOP is two and if INIETS is greater than or equal to zero.

PXDP—is a proportion of the extinction depth (dimensionless), measured downward from the ET surface, which, with PETM, defines the shape of the relation between the evapotranspiration rate and head. The value of PXDP must be between 0.0 and 1.0, inclusive. Repetitions of PXDP and PETM are read in sequence such that the first occurrence represents the bottom of the first segment, and subsequent repetitions represent the bottom of successively

lower segments. Accordingly, PXDP values for later repetitions (representing lower segments) should be greater than PXDP values for earlier repetitions.

PETM—is a proportion of the maximum evapotranspiration rate (dimensionless) which, with PXDP, defines the shape of the relation between the evapotranspiration rate and head. The value of PETM should be between 0.0 and 1.0, inclusive. Repetitions of PXDP and PETM are read in sequence such that the first occurrence represents the bottom of the first segment, and subsequent repetitions represent the bottoms of successively lower segments. Accordingly, PETM values for later repetitions (representing lower segments) generally would be less than PETM values for earlier repetitions.

Drain with Return Flow (DRT) Package

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.]

Input to the DRT1 Package is read from the file that has type “DRT” in the name file. Optional variables are shown in brackets. All variables are free format if the option “FREE” is specified in the Basic Package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—“#” must be in column 1. Item 0 can be repeated multiple times.

1. MXADRT IDRTCB NPDRT MXL [Option]

2. [PARNAM PARTYP Parval NLST]

Item 3a is read only for a structured grid

3a. Layer Row Column Elevation Condfact [LayR RowR ColR Rfprop] [xyz]

Item 3b is read only for an unstructured grid

3b. Node Elevation Condfact [NodeR Rfprop] [xyz]

NLST repetitions of Item 3 records are required; they are read by module ULSTRD (Harbaugh and others, 2000). (SFAC of the ULSTRD utility module applies to Condfact.)

Repeat Items 2 and 3 for each parameter to be defined (that is, NPDRT times). Items 2 and 3 are omitted if NPDRT = 0.

FOR EACH STRESS PERIOD

4. ITMP NP

Item 5a is read only for a structured grid

5a. Layer Row Column Elevation Cond [LayR RowR ColR Rfprop] [xyz]

Item 5b is read only for an unstructured grid

5b. Node Elevation Cond [NodeR Rfprop] [xyz]

ITMP repetitions of Item 5 records are read by module ULSTRD (Harbaugh and others, 2000) if ITMP > 0. (SFAC of the ULSTRD utility module applies to Cond). Item 5 is not read if ITMP$\leqslant 0$.

6. Pname

(Item 6 is repeated NP times. It is not read if NP $\leqslant 0$.)

Explanation of Variables Read by the DRT Package:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

MXADRT—is the maximum number of drain-return cells in use during any stress period, including those defined using parameters. **Recipient cells are not included in MXADRT**.

IDRTCB—is a flag and a unit number.

> If IDRTCB > 0, it is the unit number to which DRT1-Package cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control (Harbaugh and others, 2000). IDRTCB must be a unit number associated with a file listed with type "DATA (BINARY)" or "DATAGLO(BINARY)" in the name file.
>
> If IDRTCB = 0, DRT1-Package cell-by-cell flow terms will not be written.
>
> If IDRTCB < 0, drain leakage for each drain-return cell and return flow to each recipient cell will be written to the LIST file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

NPDRT—is the number of drain-return parameters.

MXL—is the maximum number of drain-return cells that will be defined using parameters. Recipient cells are not included in MXL. Option—is an optional list of character values.

Option—is an optional list of character values

> "AUXILIARY abc" or "AUX abc"—defines an auxiliary variable (Harbaugh and McDonald, 1996a, p. 9, item 4), named "abc," which will be read for

each drain as part of items 3 and 5. Up to five variables can be specified, each of which must be preceded by "AUXILIARY" or "AUX." These variables will not be used by the Ground-Water Flow Process, but they will be available for use by other processes. The auxiliary variable values will be read after the Cond variable.

"CBCALLOCATE" or "CBC"—indicates that memory should be allocated to store cell-by-cell flow for each drain in order to make these flows available for use in other packages.

"RETURNFLOW"—activates the return-flow option of the DRT1 Package. If "RETURNFLOW" is listed as an option, LayR, and, optionally, RowR, ColR, and Rfprop are read from items 3 and (or) 5.

PARNAM—is the name of a parameter to be defined. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the DRT1 Package, the only allowed parameter type is "DRT," which defines values of the drain hydraulic conductance.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity Process input file or by a value generated by the Parameter-Estimation Process.

NLST—is the number of drain-return cells included in the parameter.

Node—is the layer number of the cell containing the drain.

Layer—is the layer number of the cell containing the drain.

Row—is the row number of the cell containing the drain.

Column—is the column number of the cell containing the drain.

Elevation—is the elevation of the drain.

Condfact—is the factor used to calculate drain hydraulic conductance from the parameter value. The conductance (L^2/T) is the product of Condfact and the parameter value.

NodR—is a flag and, if greater than 0, a node number. If auxiliary variables are being read, NodR must be greater than zero, so that Rfprop is read. NodR is not read if "RETURNFLOW" is not listed as an option in item 1.

If NodR > 0, it is the node number of the recipient cell.

If NodR = 0, there is no return flow for the drain cell, and Rfprop are not read.

LayR—is a flag and, if greater than 0, a layer number. If auxiliary variables are being read, LayR must be greater than zero, so that RowR, ColR, and Rfprop are read. LayR is not read if "RETURNFLOW" is not listed as an option in item 1.

If LayR > 0, it is the layer number of the recipient cell.

If LayR = 0, there is no return flow for the drain cell, and RowR, ColR, and Rfprop are not read.

RowR—is the row number of the recipient cell. RowR is not read if "RETURNFLOW" is not listed as an option in item 1.

ColR—is the column number of the recipient cell. ColR is not read if "RETURNFLOW" is not listed as an option in item 1.

Rfprop—is the return-flow proportion. Valid values are in the range 0.0 to 1.0, inclusive. Rfprop is the proportion of the drain flow, if any, calculated for the drain-return cell simulated as returning to the recipient cell. If Rfprop equals 0.0, the return-flow capability is deactivated for the cell. Rfprop is not read if "RETURNFLOW" is not listed as an option in item 1.

[xyz]—is up to five auxiliary variables for a drain-return cell that have been defined in item 1. The auxiliary variables must be present in each repetition of items 3 and 5 record if they are defined in item 1.

ITMP—is a flag and a counter.

> If ITMP < 0, non-parameter drain-return data from the last stress period will be reused.
>
> If ITMP ≥0, ITMP will be the number of non-parameter drain-return cells read for the current stress period.

NP—is the number of drain-return parameters in use in the current stress period.

Cond—is the hydraulic conductance of the interface between the aquifer and the drain.

Pname—is the name of a parameter being used in the current stress period. NP parameter names will be read.

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).

Input to the QRT1 Package is read from the file that has type “QRT” in the name file. Optional variables are shown in brackets. All variables are free format if the option “FREE” is specified in the Basic Package input file; otherwise, the non-

optional variables have 10-character fields and the optional variables are free format.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. MXAQRT MXRTCELLS IQRTCB NPQRT MXL [Option]

2. [PARNAM PARTYP Parval NLST]

Item 3a is read only for a structured grid

3a. Layer Row Column SinkQfact NumRT [Rfprop] [xyz]

Item 3b is read only for an unstructured grid

3b. Node RTNum SinkQfact NumRT [Rfprop] [xyz]

> NLST repetitions of Item 3 records are required; they are read by module ULSTRD (Harbaugh and others, 2000). (SFAC of the ULSTRD utility module applies to SinkQ.)

Repeat Items 2 and 3 for each parameter to be defined (that is, NPQRT times). ***Items 2 and 3 are omitted if NPQRT = 0.***

FOR EACH STRESS PERIOD

4. ITMP NP

Item 5a is read only for a structured grid

5a. Layer Row Column SinkQ NumRT [Rfprop] [xyz]

Item 5b is read only for an unstructured grid

5b. Node SinkQ NumRT [Rfprop] [xyz]

ITMP repetitions of Item 5 records are read by module ULSTRD (Harbaugh and others, 2000) if ITMP > 0. (SFAC of the ULSTRD utility module applies to SinkQ). Item 5 is not read if ITMP$\leqslant 0$.

6. NodQRT(NumRT,IsinkQ) (U1DINT)

ITMP repetitions of Item 6 records are read by module U1DINT if ITMP > 0, once for each of the IsinkQ extraction points or sinks simulated. The array contains the NumRT return flow cells associated with sink IsinkQ. Item 6 is not read if ITMP$\leqslant 0$

7. Pname

(Item 7 is repeated NP times. It is not read if NP $\leqslant 0$.)

Explanation of Variables Read by the QRT Package:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

MXAQRT—is the maximum number of "Sink-return" sink cells in use during any stress period, including those defined using parameters. **Recipient cells are not included in MXAQRT**.

MXRTCELLS—is the maximum number of recipient cells that may exist in the simulation during any stress period. MXRTCELLS is used to dimension the array of recipient cells for all sinks and should be larger than or equal to the number of recipient cells that may be listed in the simulation during any stress period.

IQRTCB—is a flag and a unit number.

If IQRTCB > 0, it is the unit number to which QRT1-Package cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control (Harbaugh and others, 2000). IQRTCB must be a unit number associated with a file listed with type "DATA(BINARY)" or "DATAGLO(BINARY)" in the name file.

If IQRTCB = 0, QRT1-Package cell-by-cell flow terms will not be written.

If IQRTCB < 0, Sink leakage for each Sink-return cell and return flow to each recipient cell will be written to the LIST file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

NPQRT—is the number of Sink-return parameters.

MXL—is the maximum number of "Sink-return" sink cells that will be defined using parameters. Recipient cells are not included in MXL. Option—is an optional list of character values.

Option—is an optional list of character values

"AUXILIARY abc" or "AUX abc"—defines an auxiliary variable (Harbaugh and McDonald, 1996a, p. 9, item 4), named "abc," which will be read for each Sink as part of items 3 and 5. Up to five variables can be specified, each of which must be preceded by "AUXILIARY" or "AUX." These variables will not be used by the Ground-Water Flow Process, but they will be available for use by other processes. The auxiliary variable values will be read after the Cond variable.

"CBCALLOCATE" or "CBC"—indicates that memory should be allocated to store cell-by-cell flow for each Sink in order to make these flows available for use in other packages.

"RETURNFLOW"—activates the return-flow option of the QRT1 Package. If "RETURNFLOW" is listed as an option, NumRT, and optionally, Rfprop are read from items 3 and (or) 5.

PARNAM—is the name of a parameter to be defined. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the QRT1 Package, the only allowed parameter type is "SinkQ," which defines values of the volumetric flux rate for the sink.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity Process input file or by a value generated by the Parameter-Estimation Process.

NLST—is the number of Sink-return cells included in the parameter.

Node—is the layer number of the cell containing the Sink.

Layer—is the layer number of the cell containing the Sink.

Row—is the row number of the cell containing the Sink.

Column—is the column number of the cell containing the Sink.

SinkQfact—is the factor used to calculate the volumetric flux rate of the Sink from the parameter value. The volumetric flux rate of the Sink is the product of SinkQfact and the parameter value.

NumRT—is a flag and, if greater than 0, a layer number. If auxiliary variables are being read, NumRT must be greater than zero, so that Rfprop is read. NumRT is not read if "RETURNFLOW" is not listed as an option in item 1.

If NumRT > 0, it is the number of recipient cells.

If NumRT = 0, there is no return flow for the Sink cell, and Rfprop are not read.

Rfprop—is the return-flow proportion. Valid values are in the range 0.0 to 1.0, inclusive. Rfprop is the proportion of the Sink flow, if any, that is returned to the recipient cells. If Rfprop equals 0.0, the return-flow capability is deactivated for the cell. Rfprop is not read if "RETURNFLOW" is not listed as an option in item 1.

[xyz]—is up to five auxiliary variables for a Sink-return cell that have been defined in item 1. The auxiliary variables must be present in each repetition of items 3 and 5 record if they are defined in item 1.

ITMP—is a flag and a counter.

> If ITMP < 0, non-parameter Sink-return data from the last stress period will be reused.
>
> If ITMP ≥ 0, ITMP will be the number of non-parameter Sink-return cells read for the current stress period.

NP—is the number of Sink-return parameters in use in the current stress period.

SinkQ—is the volumetric flux (L^3/T) from the Sink.

NodQRT(NumRT,IsinkQ) —is the node number of the NumRT return flow nodes associated with sink IsinkQ.

Pname—is the name of a parameter being used in the current stress period. NP parameter names will be read.

Arrays for QRT module:

QRTF(1,ii) = Node number (or Layer Number for structured grid)

QRTF(2,ii) = (Row Number for structured grid)

QRTF(3,ii) = (Column Number for structured grid)

QRTF(4,ii) = Extraction stress (SinkQ)

QRTF(5,ii) = Number of nodes on which return occurs (NumRT)

QRTF(6,ii) = Return fraction (Rfprop)

QRTF(6,ii) = Actual extraction (may be less than SinkQ due to autoflowreduce)

NodQRT(NumRT, IsinkQ) – Keep a cumulative list in a 1-D array of maximum length MXRTCELLS.

QRTFLOW(NumRT, IsinkQ) – Also kept in a cumulative list in a 1-D array.

QRT is attached to IUNIT(41)

Sparse Matrix Solver (SMS) Package

The SMS Package has been modified to include only active nodes in the linear solution routines. This enhancement speeds up computations when there are many inactive nodes (with IBOUND=0) within a model domain. Input for the Sparse Matrix Solver (SMS) Package is read from the file that is type "SMS" in the Name File.

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. This feature is included here using the DAMPBOT option.

The SMS Package input instructions include flags, indices and tolerances for the nonlinear solution of the unconfined groundwater flow equation as well as for the matrix solution scheme selected for solution of the matrix equations. Guidance for selection of appropriate solution schemes is provided in the main document under the section "Guidance for Applying MODFLOW-USG". Recommended values for of numerical parameters used by the SMS package are provided below. The input is read in as free format.

FOR EACH SIMULATION

1a. OPTIONS

1b. HCLOSE HICLOSE MXITER ITER1 IPRSMS NONLINMETH LINMETH [OPTIONS2]

If NONLINMETH ≠ 0 and OPTIONS is not specified then read item 2

2. THETA AKAPPA GAMMA AMOMENTUM NUMTRACK BTOL BREDUC RESLIM

If LINMETH = 1 and OPTIONS is not specified then read item 3 for the χMD solver

3. IACL NORDER LEVEL NORTH IREDSYS RRCTOL IDROPTOL EPSRN

If LINMETH = 2 and options is not specified then read item 4 for the PCGU solver

4. IPC ISCL IORD RCLOSEPCGU

Explanation of Variables Read by the SMS Package:

OPTIONS— are keywords that activate default solver options:

> SIMPLE indicates that default solver input values will be defined that work well for nearly linear models. This would be used for models that do not include nonlinear stress packages and models that are either confined or

consist of a single unconfined layer that is thick enough to contain the water table within a single layer.

MODERATE indicates that default solver input values will be defined that work well for moderately nonlinear models. This would be used for models that include nonlinear stress packages and models that consist of one or more unconfined layers. The “MODERATE” option should be used when the “SIMPLE” option does not result in successful convergence.

COMPLEX indicates that default solver input values will be defined that work well for highly nonlinear models. This would be used for models that include nonlinear stress packages and models that consist of one or more unconfined layers representing complex geology and sw/gw interaction. The “COMPLEX” option should be used when the “MODERATE” option does not result in successful convergence.

The values of solver parameters for the various options are shown below in Tables 1 and 2.

HCLOSE—is the head change criterion for convergence of the outer (nonlinear) iterations, in units of length. When the maximum absolute value of the head change at all nodes during an iteration is less than or equal to HCLOSE, iteration stops. Commonly, HCLOSE equals 0.01.

HICLOSE—is the head change criterion for convergence of the inner (linear) iterations, in units of length. When the maximum absolute value of the head change at all nodes during an iteration is less than or equal to HICLOSE, the matrix solver assumes convergence. Commonly, HICLOSE is set an order of magnitude less than HCLOSE.

MXITER—is the maximum number of outer (nonlinear) iterations -- that is, calls to the solution routine. For a linear problem MXITER should be 1.

ITER1—is the maximum number of inner (linear) iterations. The number typically depends on the characteristics of the matrix solution scheme being used. For nonlinear problems, ITER1 usually ranges from 60 to 600; a value of 100 will be sufficient for most linear problems.

IPRSMS—is a flag that controls printing of convergence information from the solver:

0 – print nothing

1 – print only the total number of iterations and nonlinear residual reduction summaries

2 – print matrix solver information in addition to above

NONLINMETH—is a flag that controls the nonlinear solution method and under-relaxation schemes

0 – Picard iteration scheme is used without any under-relaxation schemes involved

> 0 – Newton-Raphson iteration scheme is used with under-relaxation. Note that the Newton-Raphson linearization scheme is available only for the upstream weighted solution scheme of the BCF and LPF packages.

< 0 – Picard iteration scheme is used with under-relaxation.

The absolute value of NONLINMETH determines the underrelaxation scheme used.

1 or -1 – Delta-Bar-Delta under-relaxation is used.

2 or -2 – Cooley under-relaxation scheme is used.

Note that the under-relaxation schemes are used in conjunction with gradient based methods, however, experience has indicated that the Cooley under-

relaxation and damping work well also for the Picard scheme with the wet/dry options of MODFLOW.

LINMETH—is a flag that controls the matrix solution method

1 – the χMD solver of Ibaraki (2005) is used.

2 – the unstructured pre-conditioned conjugate gradient solver of White and Hughes (2011) is used.

OPTIONS2— are keywords that activate options:

SOLVEACTIVE — indicates that the linear solvers will only solve for active model cells, i.e. model cells with IBOUND ≠ 0. With this option, matrices passed to the linear solver are compressed by ignoring all inactive (IBOUND=0) model cells. This option improves the simulation time in proportion to the number of inactive cells in the model. This option is recommended for models that have a large number of inactive model cells. This keyword, if invoked, should be added on the first line of this package before other optional keywords. This option is available with the χMD (LINMETH=1) and the PCGU (LINMETH=2) solvers.

DAMPBOT — indicates that the bottom damping procedure of MODFLOW-NWT will be applied to the solution. Note that the USGS release of MODFLOW-USG Version 1.4 includes this bottom averaging procedure as a default.

THETA—is the reduction factor for the learning rate (under-relaxation term) of the delta-bar-delta algorithm. The value of THETA is between zero and one. If the change in the variable (head) is of opposite sign to that of the previous iteration, the under-relaxation term is reduced by a factor of THETA. The value usually ranges from 0.3 to 0.9; a value of 0.7 works well for most problems.

AKAPPA—is the increment for the learning rate (under-relaxation term) of the delta-bar-delta algorithm. The value of AKAPPA is between zero and one. If the

change in the variable (head) is of the same sign to that of the previous iteration, the under-relaxation term is increased by an increment of AKAPPA. The value usually ranges from 0.03 to 0.3; a value of 0.1 works well for most problems.

GAMMA—is the history or memory term factor of the delta-bar-delta algorithm. Gamma is between zero and 1 but cannot be equal to one. When GAMMA is zero, only the most recent history (previous iteration value) is maintained. As GAMMA is increased, past history of iteration changes has greater influence on the memory term. The memory term is maintained as an exponential average of past changes. Retaining some past history can overcome granular behavior in the calculated function surface and therefore helps to overcome cyclic patterns of non-convergence. The value usually ranges from 0.1 to 0.3; a value of 0.2 works well for most problems.

AMOMENTUM—is the fraction of past history changes that is added as a momentum term to the step change for a nonlinear iteration. The value of AMOMENTUM is between zero and one. A large momentum term should only be used when small learning rates are expected. Small amounts of the momentum term help convergence. The value usually ranges from 0.0001 to 0.1; a value of 0.001 works well for most problems.

NUMTRACK—is the maximum number of backtracking iterations allowed for residual reduction computations. If NUMTRACK = 0 then the backtracking iterations are omitted. The value usually ranges from 2 to 20; a value of 10 works well for most problems.

BTOL—is the tolerance for residual change that is allowed for residual reduction computations. BTOL should not be less than one to avoid getting stuck in local minima. A large value serves to check for extreme residual increases, while a low value serves to control step size more severely. The value usually ranges from 1.0 to 10^6; a value of 10^4 works well for most problems but lower values like 1.1 may be required for harder problems.

BREDUC—is the reduction in step size used for residual reduction computations. The value of BREDUC is between zero and one. The value usually ranges from 0.1 to 0.3; a value of 0.2 works well for most problems.

RESLIM—is the limit to which the residual is reduced with backtracking. If the residual is smaller than RESLIM, then further backtracking is not performed. A value of 100 is suitable for large problems and residual reduction to smaller values may only slow down computations.

For the χMD solver (Ibaraki, 2005):

IACL—is the flag for choosing the acceleration method.

0 – Conjugate Gradient – select this option if the matrix is symmetric.

1 – ORTHOMIN

2 - BiCGSTAB

NORDER—is the flag for choosing the ordering scheme.

0 – original ordering

1 – reverse Cuthill McKee ordering

2 – Minimum degree ordering

LEVEL—is the level of fill for ILU decomposition. Higher levels of fill provide more robustness but also require more memory. For optimal performance, it is suggested that a large level of fill be applied (7 or 8) with use of drop tolerance.

NORTH—is the number of orthogonalizations for the ORTHOMIN acceleration scheme. A number between 4 and 10 is appropriate. Small values require less storage but more iteration may be required. This number should equal 2 for the other acceleration methods.

IREDSYS—is the index for creating a reduced system of equations using the red-black ordering scheme.

0 – do not create reduced system

1 – create reduced system using red-black ordering

RRCTOL—is a residual tolerance criterion for convergence. The root mean squared residual of the matrix solution is evaluated against this number to determine convergence. The solver assumes convergence if either HICLOSE (the absolute head tolerance value for the solver) or RRCTOL is achieved. Note that a value of zero ignores residual tolerance in favor of the absolute tolerance (HICLOSE) for closure of the matrix solver.

IDROPTOL—is the flag to perform drop tolerance.

0 – do not perform drop tolerance

1 – perform drop tolerance

EPSRN—is the drop tolerance value. A value of 10^{-3} works well for most problems.

For PCGU Solver (White and Hughes, 2011):

IPC— an integer value that defines the preconditioner.

IPC = 0, No preconditioning.

IPC = 1, Jacobi preconditioning.

IPC = 2, ILU(0) preconditioning.

IPC = 3, MILU(0) preconditioning.

IPC=3 works best for most problems.

ISCL is the flag for choosing the matrix scaling approach used.

0 –no matrix scaling applied

1 –symmetric matrix scaling using the scaling method by the POLCG preconditioner in Hill (1992).

2 –symmetric matrix scaling using the ℓ^2 norm of each row of **A** ($\mathbf{D}_R$) and the ℓ^2 norm of each row of $\mathbf{D}_R\mathbf{A}$.

ISCL must be 1 or 2 if the ILU(0) or MILU(0) preconditioners are used (IPC = 2 or 3) with matrix reordering (IORD > 0).

IORD is the flag for choosing the matrix reordering approach used.

0 – original ordering

1 – reverse Cuthill McKee ordering

2 – minimum degree ordering

If reordering is used, reverse Cuthill McKee ordering has been found to be a more effective reordering approach for the test problems evaluated.

RCLOSEPCGU—a real value that defines the flow residual tolerance for convergence of the PCGU linear solver. This value represents the maximum allowable residual at any single node. Value is in units of length cubed per time, and must be consistent with MODFLOW-USG length and time units. Usually a value of 1.0×10^{-1} is sufficient for the flow-residual criteria when meters and seconds are the defined MODFLOW-USG length and time.

SMS Output:

Diagnostic output from the SMS package includes the solver iteration details (or summary), residual reduction information, and the nonlinear (outer) iteration summary. Residual reduction information includes the incoming and final

accepted residuals, the backtracking count (IBCOUNT) and a flag (IBFLAG) indicating the backtracking status for each nonlinear iteration as indicated below.

IBFLAG = 0 – backtracking was not performed

= 1 – backtracking was performed and the routine exited because the required residual reduction was achieved.

= 2 – backtracking was performed and the routine exited because the maximum number of backtracks, NUMTRACK was exceeded.

=3 – backtracking was performed and the routine exited because the new residual was less than the residual reduction limit, RES_LIM.

=4 – backtracking was performed and the routine exited because the largest step-size was less than the convergence limit of HCLOSE.

The residual reduction information is provided if the solver print-flag IPRSMS is greater than or equal to one.

The nonlinear iteration summary includes the total number of nonlinear iterations required for convergence and the maximum head change at each nonlinear iteration (immediately after a matrix solve), along with the node number where the maximum head change occurs. This information, along with the backtracking information may be used to tune the parameters for the delta-bar-delta algorithm and the residual reduction scheme to obtain convergence or improve robustness of the simulations. The summary at each iteration of the matrix solver is also provided in the output, if the value of IPRSMS is equal to two. This information is useful in tuning solver behavior if the matrix solver is stalling or failing.

Table 1. *Parameter Values for the Various Default XMD Solver Options*

Parameter	Option		
	Simple	Moderate	Complex
IACL	1	2	2
NORDER	0	0	1
LEVEL	3	3	5
NORTH	5	5	7
IREDSYS	1	1	1
IDROPTOL	0	1	1
RRCTOL	0.00E+00	0.00E+00	0.00E+00
EPSRN	1.00E-03	1.00E-03	1.00E-05
THETA	1.00E+00	9.00E-01	8.00E-01
AKAPPA	0.00E+00	1.00E-04	1.00E-04
GAMMA	0.00E+00	0.00E+00	0.00E+00
AMOMENTUM	0.00E+00	0.00E+00	0.00E+00
NUMTRACK	0	0.00E+00	2.00E+01
BTOL	0.00E+00	0.00E+00	1.05E+00
BREDUC	0.00E+00	0.00E+00	1.00E-01
RES_LIM	0.00E+00	0.00E+00	2.00E-03

Table 2*. Parameter Values for the Various Default PCGU Solver Options*

Parameter	Option		
	Simple	Moderate	Complex
IPC	2	2	2
ISCL	2	2	2
IORD	2	2	2
RCLOSEPCGU	1.00E-04	1.00E-04	1.00E-04
THETA	1.00E+00	9.00E-01	8.00E-01
AKAPPA	0.00E+00	1.00E-04	1.00E-04
GAMMA	0.00E+00	0.00E+00	0.00E+00
AMOMENTUM	0.00E+00	0.00E+00	0.00E+00
NUMTRACK	0	0.00E+00	2.00E+01
BTOL	0.00E+00	0.00E+00	1.05E+00
BREDUC	0.00E+00	0.00E+00	1.00E-01
RES_LIM	0.00E+00	0.00E+00	2.00E-03

River (RIV) Package

Input to the River (RIV) Package is read from the file that has file type "RIV" in the Name File. Optional variables are shown in brackets. All variables are free format if the option “FREE” is specified in the Basic Package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

Optional variables are used in the RIV Package to identify species concentrations for transport simulations. The concentration in the river is input to the code as an AUXILIARY variable. The name identifies the species for which the auxiliary variable is designated. For example, C01 identifies the variable as the concentration of species 1; C02 identifies the variable as the concentration of species 2; and in general, Cxx identifies the variable as the concentration of species “xx”.

An option is also provided to compute the net conductance of a river bed and the cell on which it is connected. The default operation of the RIV package in MODFLOW is such that the river-bed conductance which is input, is used to transmit flow between the river boundary and the groundwater cell. However, if the river bed 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.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [***PARAMETER*** NPRIV MXL]

This optional item must start with the word "PARAMETER".

2. MXACTR IRIVCB [Option]

3. [PARNAM PARTYP Parval NLST [***INSTANCES*** NUMINST]]

Repeat Item 3 combined with the indicated repetitions of Item 4 NPRIV times. Items 3 and 4 are not read if NPRIV is negative or 0.

If PARNAM is to be a time-varying parameter, the keyword "INSTANCES" and a value for NUMINST must be entered.

4a. [INSTNAM]

4b. [Layer Row Column Stage Condfact Rbot [xyz]]

Omit Item 4b if an unstructured grid is used

4c. Node Stage Condfact Rbot [xyz]

Omit Item 4c if a structured grid is used

After each Item 3 for which the keyword "INSTANCES" is not entered, read Item 4b or 4c and not Item 4a.

After each Item 3 for which the keyword "INSTANCES" is entered, read Item 4a and Item 4b or 4c for each instance.

NLST repetitions of Item 4b or 4c are required; they are read by subroutine ULSTRD. (SFAC of the ULSTRD utility subroutine applies to Condfact). The NLST repetitions of Item 4b or 4c follow each repetition of Item 4a when PARNAM is time varying.

FOR EACH STRESS PERIOD

5. ITMP NP

6a. Layer Row Column Stage Cond Rbot [xyz]

Omit Item 6a if an unstructured grid is used

6b. Node Stage Cond Rbot [xyz]

Omit Item 6b if a structured grid is used

ITMP repetitions of Items 6a or 6b are read by subroutine ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility subroutine applies to Cond.) Items 6a or 6b are not read if ITMP is negative or 0.

7. [Pname [Iname]]

(Item 7 is repeated NP times. Item 7 is not read if NP is negative or 0. Iname is read if Pname is a time-varying parameter.)

Explanation of Variables Read by the RIV Package:

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The

"#" character must be in column 1. Lines beginning with # are restricted to the first lines of the file. Text is written to the Listing File.

NPRIV—is the number of river parameters.

MXL—is the maximum number of river reaches that will be defined using parameters.

MXACTR—is the maximum number of river reaches in use during any stress period, including those that are defined using parameters.

IRIVCB—is a flag and a unit number.

> If IRIVCB > 0, cell-by-cell flow terms will be written to this unit number when "SAVE BUDGET" or a nonzero value for ICBCFL is specified in Output Control.
>
> If IRIVCB = 0, cell-by-cell flow terms will not be written.
>
> If IRIVCB < 0, river leakage for each reach will be written to the listing file when "SAVE BUDGET" or a nonzero value for ICBCFL is specified in Output Control.

Option—is an optional list of character values.

> “AUXILIARY abc” or “AUX abc”—defines an auxiliary variable, named "abc", which will be read for each river reach as part of Items 4 and 6. Up to 20 variables can be specified, each of which must be preceded by "AUXILIARY" or "AUX." These variables will not be used by the Ground-Water Flow Process Package, but they will be available for use by other processes. The auxiliary variable values will be read after the Rbot variable. Note that if the auxiliary variable name is “Cxx”, it identifies the variable as the concentration of species “xx”.Thus, C01 is concentration of species number 1, C02 is concentration of species number 2, and so forth (up to 99 species).
>
> “NOPRINT”—specifies that lists of river reaches will not be written to the Listing File.

"MERGE_BED_K1"—specifies that the bed conductance that is specified in this file (in Cond or Condfact) is to be merged with the vertical conductance of the groundwater cell from the bottom of the river till the center of the cell, if the bottom of the river is above the cell center. This option allows for independent control of the river bed type from the vertical conductance of the cell on which it sits.

PARNAM—is the name of a parameter. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter. For the RIV Package, the only allowed parameter type is RIV, which defines values of riverbed hydraulic conductance.

Parval—is the parameter value. This parameter value may be overridden by a value in the Parameter Value File.

NLST—is the number of river reaches in a non-time-varying parameter. For a time-varying parameter, NLST is the number of reaches in each instance.

INSTANCES—is an optional keyword that designates a parameter as time varying. The keyword is not case sensitive; that is, any combination of the same characters with different case will be equivalent. If ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, PARNAM is non-time-varying and NUMINST should not be present.

NUMINST—is the number of instances for a time-varying parameter, where each instance is a list of river reaches and associated properties. If the keyword ***INSTANCES*** is present, it must be followed by a value for NUMINST. If ***INSTANCES*** is absent, NUMINST should not be present.

INSTNAM—is the name of an instance associated with the parameter named in the corresponding Item 3. The instance name can be 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with

different case will be equivalent. Instance names must be unique for a parameter, but instance names may be reused for different parameters.

Layer—is the layer number of the cell containing the river reach.

Row—is the row number of the cell containing the river reach.

Column—is the column number of the cell containing the river reach.

Node—is the node number of the model cell that contains the river reach.

Stage—is the head in the river.

Condfact—is the factor used to calculate riverbed hydraulic conductance from the parameter value. The conductance is the product of Condfact and the parameter value.

Rbot—is the elevation of the bottom of the riverbed.

[xyz]—represents the values of the auxiliary variables for a river reach that have been defined in Item 2. The values of auxiliary variables must be present in each repetition of Items 4 and 6 if they are defined in Item 2. The values must be specified in the order used to define the variables in Item 2.

ITMP—is a flag and a counter.

> If ITMP < 0, non-parameter river data from the last stress period will be reused.
>
> If ITMP ≥ 0, ITMP will be the number of non-parameter reaches read for the current stress period.

NP—is the number of parameters in use in the current stress period.

Cond—is the riverbed hydraulic conductance.

Pname—is the name of a parameter that is being used in the current stress period. NP parameter names will be read.

Iname—is an instance name that is read only if Pname is a time-varying parameter. Multiple instances of the same parameter are not allowed in a stress period.