LA-UR-11-01952

Approved for public release;
distribution is unlimited.

Title:	User's Manual for Walkabout Version 1.0
Author(s):	Scott L. Painter
Intended for:	Internal and external software users and LANL website.

Form 836 (7/06)

USER'S MANUAL FOR WALKABOUT VERSION 1.0

Scott Painter
Computational Earth Sciences Group
Earth and Environmental Sciences Division
Los Alamos National Laboratory

APRIL 8, 2011

DISCLAIMER

ACKNOWLEDGMENTS

The author is grateful to Carl Gable for technical advice on computational geometry issues and for making some low-level computational geometry routines available for this work. The author also thanks George Zyvoloski for careful review of this manuscript.

User's Manual for
Walkabout Version 1.0

1. Introduction

Walkabout Version 1.0 (LA-CC-11-033) performs random walk particle tracking simulations of solute transport based on groundwater flow solutions that use fully unstructured control volume grids. Walkabout 1.0 is designed to work within the FEHM (Zyvoloski, 2007) code system, and accepts groundwater flow solutions from FEHM and computational mesh descriptions from LaGrit (Los Alamos Grid Toolbox, <http://lagrit.lanl.gov, 2011). Walkabout also provides input to the PLUMECALC (Robinson, et al. 2011) particle tracking post processor. Although designed to work within the FEHM system, other control-volume flow codes and mesh generators could, with appropriate reformatting of the output, provide the flow fields and mesh descriptions.

A typical workflow for Walkabout within the FEHM system would use LaGrit to generate unstructured grids. FEHM then provides a discretized representation of the steady-state flow field to Walkabout. Given this discrete solution, Walkabout then reconstructs a groundwater flow field, and performs the random walk particle tracking calculation. Output is provided in a form compatible with the PLUMECALC software. PLUMECALC may be used to efficiently post-process the particle tracking results from Walkabout to add effects of retention/retardation and arbitrary source histories. An option exists to also record particle positions versus time, thus allowing other post-processing codes or visualization systems to be used.

Other features and limitations of Walkabout are

1. Walkabout works on fully unstructured tetrahedral meshes in three spatial dimensions. Two-dimensional meshes and meshes other than tetrahedral meshes are not supported in Version 1.0.
2. A control-volume solution for steady groundwater flow is required. Finite-element solutions are not supported. Transient flow is not supported in Version 1.0.
3. All particles are launched at time zero. The PLUMECALC software may be used to postprocess the resulting particle tracks to obtain concentration for an arbitrary source history.
4. Particles are moved through the system without decay or retardation. The PLUMECALC system may be used to postprocess the particle tracks to obtain concentration with decay and matrix diffusion or other retardation/retention processes.
5. The Burnett and Frind (1987) model for dispersion coefficient is presumed.
6. Full heterogeneity in porosity, liquid density, liquid saturation index, and dispersivity is supported.

This document summarizes the technical basis for the code, describes the input and output formats, documents verification testing, and provides example test cases that exercise the full range of capabilities.

2. Technical Basis

Particle tracking methods have significant computational advantages in many situations, especially in advection-dominated systems, but clearly require accurate methods for calculating groundwater velocity. The difficulty arises because numerical solution to the flow/pressure equation typically provides flow velocities at specific locations or in some other discretized form; flow velocity at any location within the computational domain of interest must then be reconstructed from the discretized solution. How this is accomplished is highly dependent on the choice of numerical method for the flow/pressure equation and on the type of grid.

For structured grids and finite-volume methods, Pollock's method (Pollock, 1988) is widely used to establish streamlines. In Pollock's method, an approximate analytical solution is used to move a streamline trajectory from the entrance point on a computational cell to the exit point. This method works directly on fluxes computed in a finite volume approximation and avoids explicit reconstruction of the velocity field.

Streamline tracing for unstructured grids is considerably more difficult, requiring in general flow velocity to be available everywhere in the spatial region of interest. Velocity fields must, as a consequence, be reconstructed from a discretized numerical solution. Painter et al. (2011) describe a method for reconstructing velocity fields from control-volume solutions to groundwater flow. Their method is the basis for Walkabout. In contrast to most previous work on streamline tracing on unstructured grids, the emphasis is on fully unstructured grids; i.e., no restrictions are placed on the number of faces on each control volume.

2.1 Random Walk Particle Tracking

The relations used to compute the random-walk particle trajectories that simulate the advection dispersion equation have been described in detail elsewhere (e.g. Tompson and Gelhar, 1990, Labolle et al., 1996, Lichtner et al., 2002). The well-known result for particle position at time $t + \Delta t$ is

$$\mathbf{X}_p(t+\Delta t) = \mathbf{X}_p(t) + \mathbf{A}[\mathbf{X}_p(t)]\Delta t + \mathbf{B}[\mathbf{X}_p(t)]\cdot \mathbf{Z}\sqrt{\Delta t} \qquad (1)$$

where $\mathbf{X}_p$ is the particle location, $\mathbf{Z}$ is a vector of three independent random numbers (mean of 0, variance of 1), and $\mathbf{A}$ and $\mathbf{B}$ are related to the porosity θ, velocity $\mathbf{v}$, and dispersion tensor $\mathbf{D}$ as

$$\mathbf{A} = \mathbf{v} + \nabla\cdot \mathbf{D} + \frac{1}{\theta}\mathbf{D}\cdot \nabla\theta \tag{2}$$

$$\mathbf{B}\cdot \mathbf{B}^T = 2\mathbf{D} \tag{3}$$

2.2 Reconstruction and Interpolation of the Velocity Field

The velocity v is clearly needed at all points in space to simulating the random walk particle trajectories. Similarly, the dispersion tensor D, which depends on velocity, is also needed at all points in the domain of interest. The velocity field must be reconstructed from the discretized flow solution.

A subset of finite volume methods is considered, wherein the domain of interest has been partitioned into control volumes formed by perpendicular bisectors between adjacent nodes in an underlying triangulation of cell centers. Specifically, a nodal network triangulated into a triangular (2-D) or tetrahedral (3-D) element mesh is presumed to be available. A control volume V_i is constructed around each node *i* in the triangulation (Figure 1) such that V_i is bounded by the set of perpendicular bisectors between node *i* and each of its neighbors. If Delaunay triangulation is used to construction the original triangulation, then the control volumes form the Voronoi tessellation. This particular restriction on the control volume based on perpendicular bisectors is imposed explicitly in FEHM and is a subset of the integrated finite difference (Narasimhan and Witherspoon, 1987) discretizations allowed in TOUGH2. Even in the more general case of TOUGH2, external grid generation software that are usually used to construct input mesh, typically impose similar restrictions. Note that this approach makes no assumption about the shape of the finite volume cell.

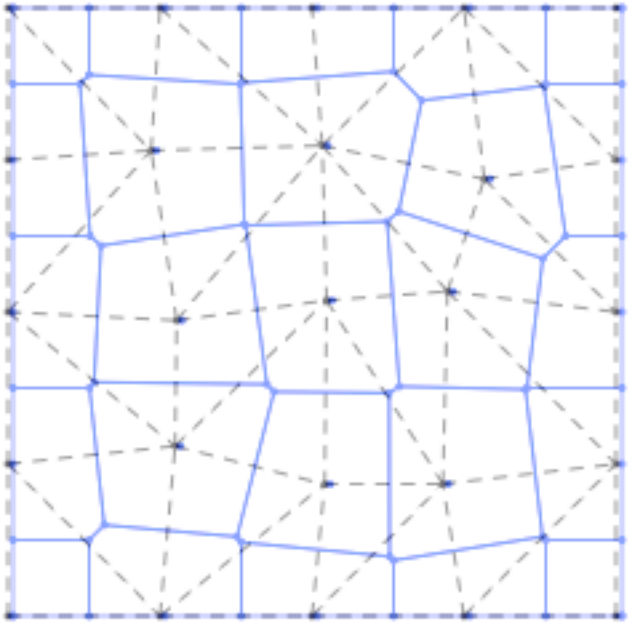

Figure 1. Example triangulation and associated control volumes for a 25-node mesh in 2D. The triangulation is shown with dashed lines. The control-volume elements are shown as solid blue lines. The control volume and element boundaries coincide on the domain boundary.

We further consider that a control-volume method with 2-point flux approximation has been used to solve for groundwater head. Specifically, the conservation equation is written for a control volume V_i as

$$S_i|V_i|\frac{\partial h_i}{\partial t} = \sum_{j\in N(i)} Q_{ij} \qquad (3)$$

where S_i, h_i and $|V_i|$ are the specific storage [L^{-1}], head [L] and volume [L^3] in cell i, respectively. Here $N(i)$ is the list of neighbors adjacent to cell i, $Q_{ij} \equiv \int_{V_i \cap V_j} \mathbf{q}\cdot \mathbf{n}_{ij}\, ds$ is volumetric flow rate [L^3/T] through the face connecting cells i and j, and $\mathbf{n}_{ij}$ is the unit normal on the same face.

Our interest is in the velocity vector $\mathbf{v} = \mathbf{q}/\phi$ in the entire domain. However, the flow solution only provides the Q_{ij}, the normal components of $\mathbf{q}$ at each cell face. The approach proposed and tested here has two steps. In the first step, the Q_{ij} for each cell are used to construct an approximate representative value of $\mathbf{q}$ for that cell (denoted $\mathbf{q}_i$, for cell i). Second, the $\mathbf{q}_i$ are then associated with the nodes and the original triangulation (dashed lines in Figure 1) is then used to interpolate to any point in the computational domain.

The first step is to reconstruct cell-centered velocities. To this end, we (temporarily) approximate **q** as being constant in each cell. The volumetric flow rate across the face common to cells *i* and *j* then becomes $Q_{ij} = \mathbf{q}_i \cdot \mathbf{A}_{ij}$ where $\mathbf{A}_{ij} \equiv A_{ij}\mathbf{n}_{ij}$ is the vector area for the face. An analogous equation can be written for each face on the cell, thus producing the linear system for each cell *i*

$\mathbf{G}_i\mathbf{q}_i = \boldsymbol{\gamma}_i$	(4)

Here $\mathbf{G}_i$ is a $n_i \times d$ matrix, $\mathbf{q}_i$ is $d \times 1$, and γ_i is $n_i \times 1$, n_i is number of neighbors for cell *i,* and *d* is dimensionality of the space. Each row of the $\mathbf{G}_i$ matrix is the vector area for one face. The columns vector $\boldsymbol{\gamma}_i$ is the volumetric flow rate across the faces, as calculated by the control volume flow code. The column vector $\mathbf{q}_i$, the darcy flux for the cell, is the unknown.

In two dimensions, a control volume has a minimum of 3 sides and in many applications involving an unstructured grid will have more. Similarly, a control volume in 3-D has 4 or more sides. Thus, equation 4 is an $n_i \times d$ system with $n_i > d$; i.e. it is an overdetermined system. Following standard techniques, a least-squares estimate $\hat{\mathbf{q}}_i$ can be constructed for the darcy velocity $\mathbf{q}_i$

$\hat{\mathbf{q}}_i = \left(\mathbf{G}^{\mathrm{T}}\mathbf{G}\right)^{-1}\mathbf{G}^{\mathrm{T}}\boldsymbol{\gamma}$	(5)

where the subscript *i* on **G** and γ has been suppressed for readability. An estimate of the velocity for the cell is then obtained as $\hat{\mathbf{v}}_i = \dfrac{\hat{\mathbf{q}}_i}{\phi}$ where ϕ is porosity in the cell.

For the types of grids considered here, nodes are specified on the domain boundaries (see Figure 1). For nodes on Dirichlet boundaries, Equation 5 can be applied as for interior nodes. For nodes on Neumann type boundaries, the specified flux constraint must be included, which leads to a linearly constrained linear least squares problem for the boundary flux vector $\hat{\mathbf{q}}_i^b$ in the *i*-th cell

$\hat{\mathbf{q}}_i^b = \underset{\mathbf{q}_i}{\arg\min}\left\|\mathbf{G}\,\mathbf{q}_i - \boldsymbol{\gamma}\right\|$ subject to $\mathbf{B}\mathbf{q}_i = \boldsymbol{\beta}$	(6)

where $\mathbf{B}$ is a $n_i^b \times d$ matrix, and β is $n_i^b \times 1$; $n_i^b < d$ is the number of boundary faces for the cell in question. The matrix $\mathbf{B}$ and the column vector $\boldsymbol{\beta}$ are analogous to **G** and **g** but are written for the subset of faces for the cell in question that lie on the boundary. Explicit solutions are available (e.g. Amemiya, 1985)

$\hat{\mathbf{q}}_i^b = \hat{\mathbf{q}}_i - \left(\mathbf{G}^{\mathrm{T}}\mathbf{G}\right)^{-1}\mathbf{B}\left(\mathbf{B}^{\mathbf{T}}\left(\mathbf{G}^{\mathrm{T}}\mathbf{G}\right)^{-1}\mathbf{B}\right)^{-1}\left(\mathbf{B}\hat{\mathbf{q}}_i - \boldsymbol{\beta}\right)$	(7)

with Equation (5) for $\hat{\mathbf{q}}_i$. In the event that $n_i^b = d$ (i.e. for nodes at the corners of the domain) the boundary conditions alone determine the nodal velocities.

As a simple example for which an intuitively reasonable result is apparent, consider a finite-volume cell on the left boundary of a 2-D rectangular domain

with dimensions, intercell fluxes and boundary flux as shown in Figure 2. Unconstrained reconstruction from Equation 5 gives (-1/2, -3/2) for the darcy flux, equivalent to simply averaging the intercell fluxes on the left and right side to get x-velocity and averaging the top and bottom fluxes to get y-velocity. Using the constrained reconstruction Equation 7 gives (0, -3/2). Thus, the normal component of the darcy flux at the boundary is forced to honor the imposed no-flow condition at the boundary. These results could be intuited for the simple geometry shown in Figure 2; the utility of Equations 5 and 7 lies in the fact that the procedure works for finite volume cells of any shape.

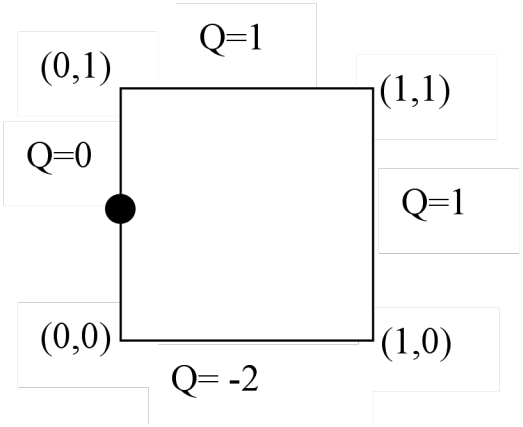

Figure 2. A control volume on the left boundary of a computational domain. Node location is shown as the black dot. The coordinates of the control volume vertices are shown in parenthesis. The volumetric fluxes Q [L^3/T] are shown for each face (positive values indicate flow into the cell). The unconstrained velocity reconstruction equation Equation 5 gives (-1/2, -3/2). The boundary constrained version Equation 7 gives (0,-3/2); i.e. it forces the no-flow condition on the left side to be honored.

Equations 5 or 7 can be applied to each cell *i* to obtain a discrete representation of the velocity field. The assumption of constant velocity in each cell, which was used to develop Equations 5 and 7, results in discontinuous velocities at cell faces. Discontinuous velocities can lead to mass balance errors and are generally inadequate for pathline tracing. However, these discontinuities can be easily removed through a simple smoothing procedure. Specifically, we reinterpret each $\mathbf{v}_i$ as being applicable at the nodal position in the original finite element mesh. Using standard barycentric coordinates, the velocities can then be interpolated without discontinuities to any position in the domain. Note also, that derivatives of the velocity, which are needed to properly calculate dispersion tensors in a random walk particle-tracking model, are readily available from the shape functions.

Dispersive and Advective Displacements

Given the ability to interpolate **v** to any point in the domain, it is a simple matter to then solve for a particle's position. A first-order predictor-corrector method

weighted to the corrector is used in Walkabout for the advective displacement. The dispersive step is first-order Euler.

The time step is adaptive based on the local value of the effective velocity (**A** term in Eq. 2) and the dispersion tensor. The time step is taken to be the smaller of t_1 and t_2, where

$$t_1 = f_c \frac{\bar{l}}{v} \tag{8}$$

$$t_2 = f_d \frac{\bar{l}^2}{4\left(\alpha v + D_m\right)} \tag{9}$$

where v is groundwater speed, D_m is molecular diffusion coefficient, $\bar{l}$ is the characteristic linear dimension of the local finite volume cell (cube root of the cell volume), and α is maximum of the longitudinal, transverse vertical, and transverse horizontal dispersivities. The parameters f_c and f_d are user-defined limits on the time step expressed as fractions of the characteristic times for advective and dispersive displacements, respectively.

If an advective time step causes a particle to cross a no-flow boundary, the time step is recursively decreased until the step does not cause the particle to cross the boundary. Dispersive steps that cause a particle to cross a no-flow boundary are rejected.

3. Overview of Input

Walkabout requires information about the computational mesh, the flow solution, particle source locations, dispersion tensors, and numerical control parameters. Similar to the FEHM code, input is taken from a set of input files, the names of which are provided in the name file.

Conventions used here are as follows. Input parameter names, keyword types, and file types are shown in **bold font**. Literal character strings representing allowed values for keywords or filenames are shown in *italics*. The `courier font` is used when specifying input file formats. A line consisting of a single colon in the input block means that lines are skipped. An ellipsis (...) indicates that the item is to be repeated. Any text following an exclamation point in an input block is to be regarded as a comment or explanation.

3.1 Running Walkabout

Walkabout is started by typing the name of the executable file from a command line interface – no graphical user interface is provided. There is one optional command line argument giving the relative path to the **name** file.

3.2 The Name File

Names for the various input and output files are contained in the **name** file. The **name** file itself may be specified on the command line when invoking Walkabout. Otherwise it defaults to *walkabout.files* in the current working directory.

The format for the **name** file is as follows

```
filetype:filename
: ! repeat as many times as needed
```

The colon separating the **filetype** keyword and the **filename** is required. The **filetype** keywords may come in any order. A line that does not start with a valid **filetype** is ignored, thus allowing the user to add as many comments as desired to the file. Filenames may be given as full or relative filepaths as defined by the operating system. Valid **filetypes** are as follows.

control – specifies the control file, which contains control parameters and the dispersion tensor information. Optional. Defaults to *control.dat*

fehmn – specifies the FEHM geometry file. Required.

stor – specifies the FEHM stor file. Required.

ealist – specifies the element adjacency list. Required.

fin – specifies the FEHM fin file, which contains liquid fluxes. Required.

avs – specifies an FEHM AVS file, which must contain liquid saturation, porosity and liquid density. Required.

cbound – specifies a file containing a list of nodes that lie on boundaries that are closed to transport file. Optional.

trajout – specifies the output filename for trajectory output. Optional. Defaults to *walkabout.out*

sptr2 – specifies the output filename for use as PLUMECALC input. Optional. Defaults to *walkabout.sptr2*

Details for each file type are given in the appendix.

4. Verification Tests

Four verification tests are included with the Walkabout distribution.

4.1 Verification Test 1

In the first verification test, particles are released form a point source into a uniform flow aligned with the x-axis. The 1st and 2nd moments of the particle positions at time *t* have analytical solutions for this configuration. The 1st moment of the particle position is $(vt, 0, 0)$, where v is flow speed. The 2nd moment is $(2\alpha_L vt, 2\alpha_T^H vt, 2\alpha_T^V vt)$, where α_L is longitudinal dispersivity, α_T^H is horizontal transverse dispersivity, and α_T^V is vertical transverse dispersivity.

The flow speed in Test 1 is 0.839 m/day. The computational domain is 100 m x 100 m x 100 m. There are four variants of Test 1, with different dispersivity values (Table 1). Tests 1a – 1c use a regular grid with 26 nodes in each direction. Test 1d uses an unstructured grid with uniform properties (Figure 3). Note that the computational cell size is 4 m in each direction in Tests 1a – 1c. For Test1d, the smallest grid size is 1 m in the direction of flow. The grid Peclet numbers range from 1.33 to infinity for these tests.

Results from Verification Tests 1a – 1c are shown in Figures 4 – 7. In each figure, the individual points are from Walkabout and the lines are the theoretical results.

Table 1. Grid type and dispersivity values for Verification Test 1.

	Grid	α_L(m)	α_T^H (m)	α_T^V(m)
Test 1a	Regular, 26 x 26 x 26	1	0.1	0.01
Test 1b	Regular, 26 x 26 x 26	3	0.1	0.3
Test 1c	Regular, 26 x 26 x 26	0	1	1
Test 1d	Unstructured, 179925 cells	1	0.2	0.1

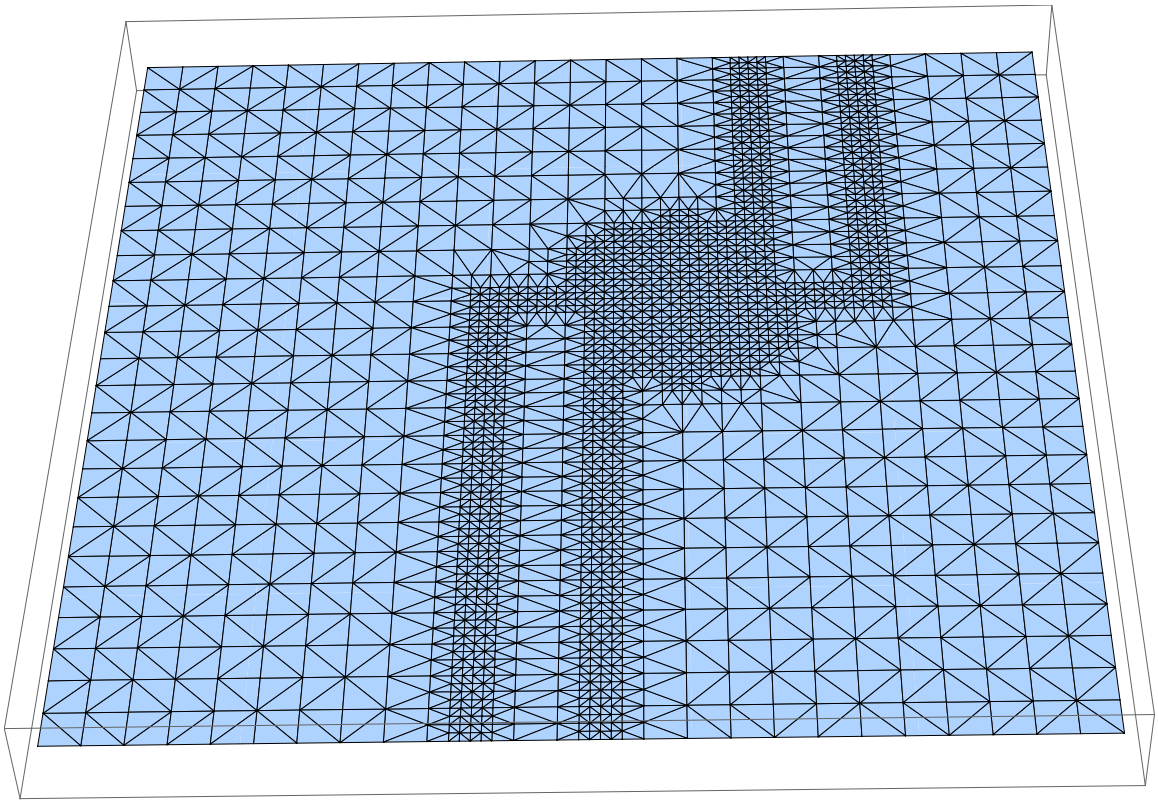

Figure 3. Horizontal slice from unstructured grid used in Test 1d. The grid is uniform in the vertical direction (not shown). Flow properties are uniform in this test. Flow is from left to right (uniform flow).

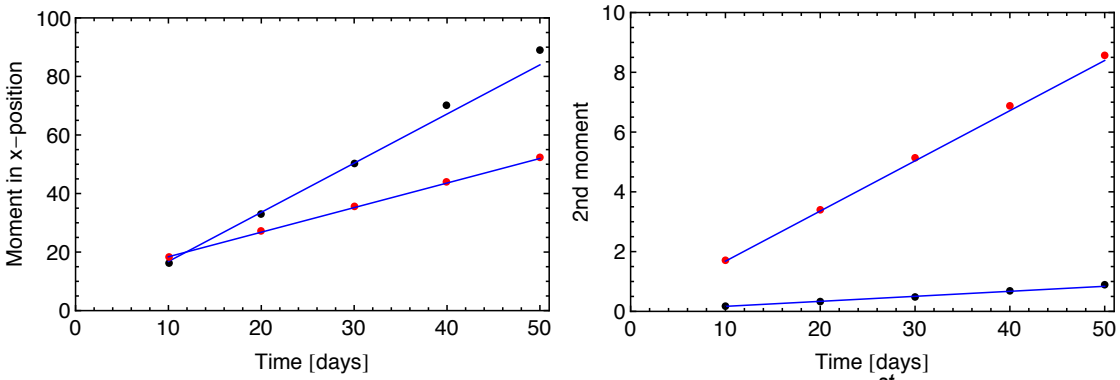

Figure 4. Result from Verification Test 1a. Shown on left are 1st (red dots) and 2nd (black dots) moments in x-position from Walkabout along with theoretical results (lines). Shown on right are 2nd moments in y-position (red dots) and z-position (black dots) along with theoretical results (lines).

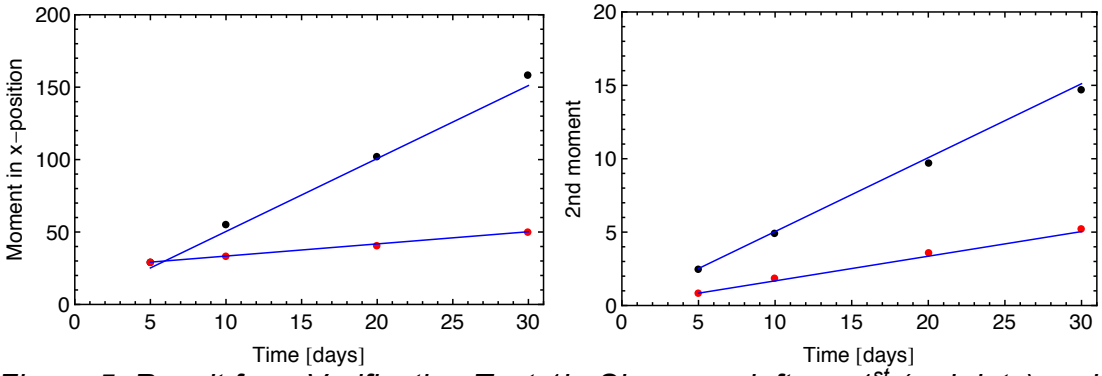

Figure 5. Result from Verification Test 1b. Shown on left are 1st (red dots) and 2nd (black dots) moments in x-position from Walkabout along with theoretical results (lines). Shown on right are 2nd moments in y-position (red dots) and z-position (black dots) along with theoretical results (lines).

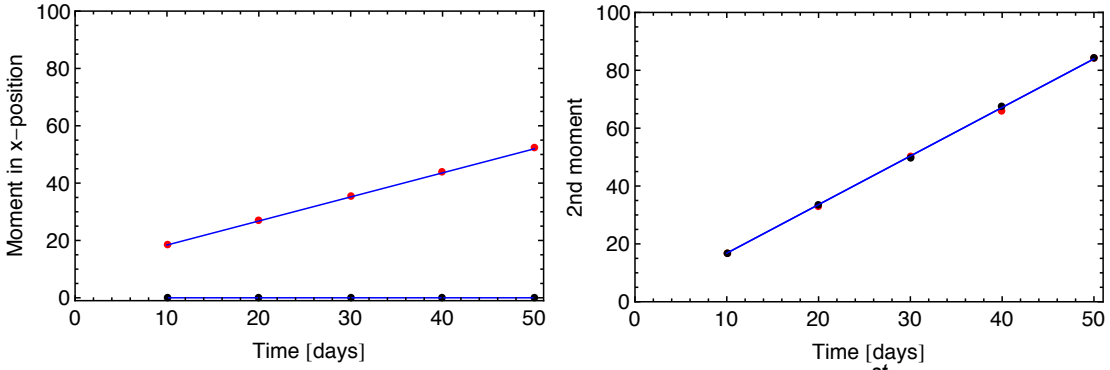

Figure 6. Result from Verification Test 1c. Shown on left are 1st (red dots) and 2nd (black dots) moments in x-position from Walkabout along with theoretical results (lines). Shown on right are 2nd moments in y-position (red dots) and z-position (black dots) along with theoretical results (lines).

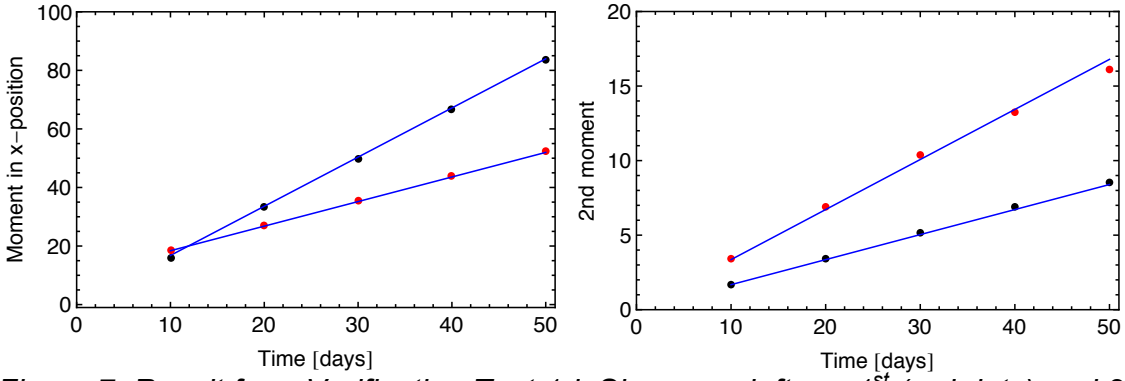

Figure 7. Result from Verification Test 1d. Shown on left are 1st (red dots) and 2nd (black dots) moments in x-position from Walkabout along with theoretical results (lines). Shown on right are 2nd moments in y-position (red dots) and z-position (black dots) along with theoretical results (lines).

4.2 Verification Test 2

Verification Test 2 is designed to test Walkabout in a nonuniform flow field. The grid is the same as Test 1d. Lower permeability is assigned to two regions, creating two non-overlapping barriers (Figure 8). A flow streamline around the two barriers is also shown in Figure 8. The travel distance along the streamline is 115.6 m. The travel time for flow tracer is 230.5 days.

In Test2, particles are released at the point (10,50,50) and tracked through the system without transverse dispersion. Two variants are considered with longitudinal dispersivity of 1 and 10 m.

Without transverse dispersion, particles should travel along the streamline upon which they are launched. Given the total travel distance and the total travel time along the streamline and adopting a Lagrangian transport perspective, the cumulative distribution of arrival times for particles at the outflow is given by Equation A-2 of Painter et al. (2008) irrespective of the streamline meander or velocity variability along the streamline. A comparison of the Walkabout results and Equation A-2 of Painter et al. (2008) is shown in Figure 9.

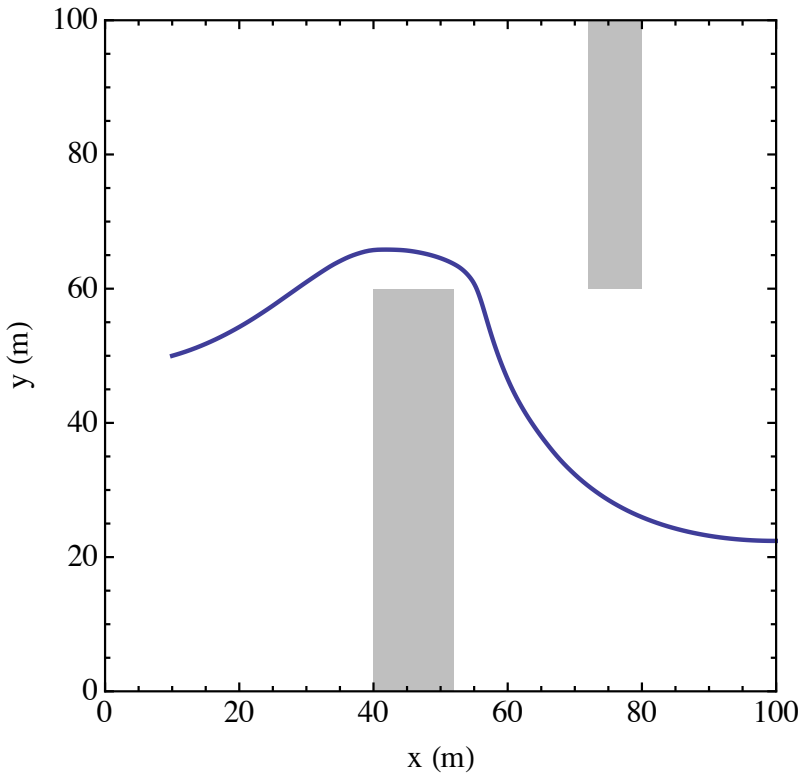

Figure 8. Horizontal projection of flow streamline for Verification Test 2. Particles are released on this streamline and tracked without transverse dispersion.

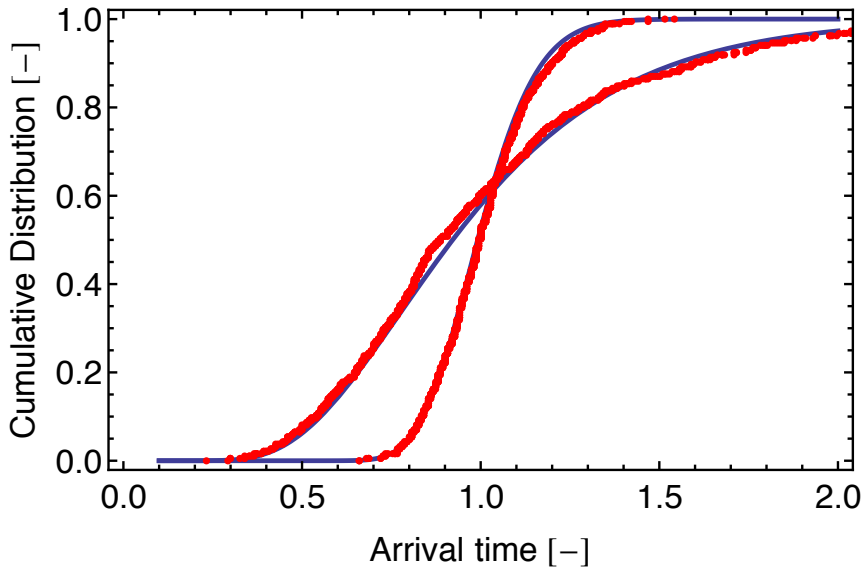

Figure 9. Results from Verification Test 2. Shown are cumulative distributions of particle arrival time from Walkabout (red dots) compared with theoretical results (blue curves) for dispersivity of 1 m and 10 m. Note the arrival time is shown normalized by the non-dispersive travel time (230.5 days).

4.3 Verification Test 3

Verification Test 3 compares Walkabout streamline tracing to the SPTR module of FEHM without dispersion. The flow configuration is the baffled flow configuration as in Verification Test 2. In this test, 8 streamlines are traced through the system with SPTR and Walkabout. Both trajectories and arrival times are compared.

Trajectories are shown in Figure 10. The trajectories overplot each other and only small differences are discernable. Note that Walkabout and SPTR use different streamline tracing algorithms. The good agreement between the two thus helps build confidence in both codes.

A cross-plot of SPTR and Walkabout arrival times is shown in Figure 11. The blue line is the line of perfect agreement. Again there is an excellent agreement between the two codes over the entire range tested.

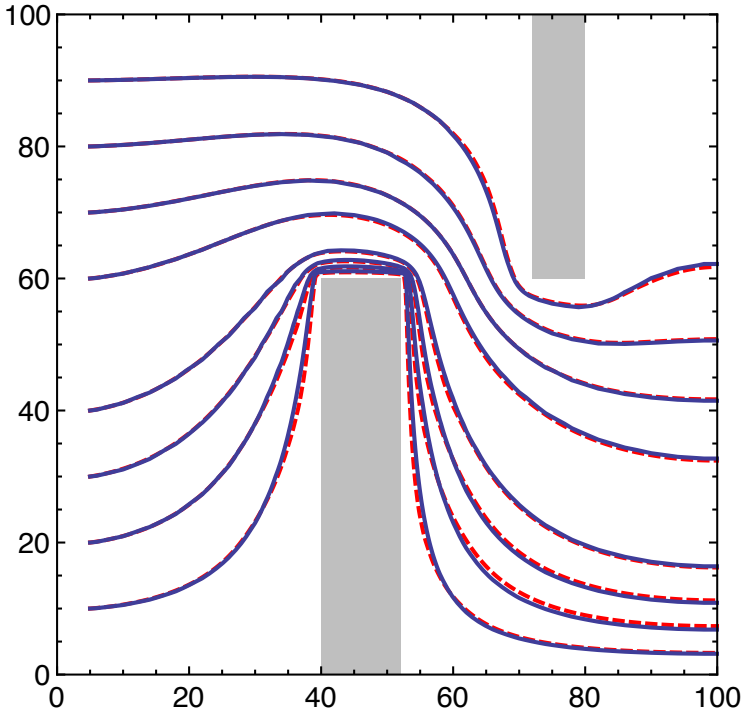

Figure 10. Results of Verification Test 3. Shown are horizontal projections of streamlines calculated by SPTR (solid blue) and Walkabout (red dashed).

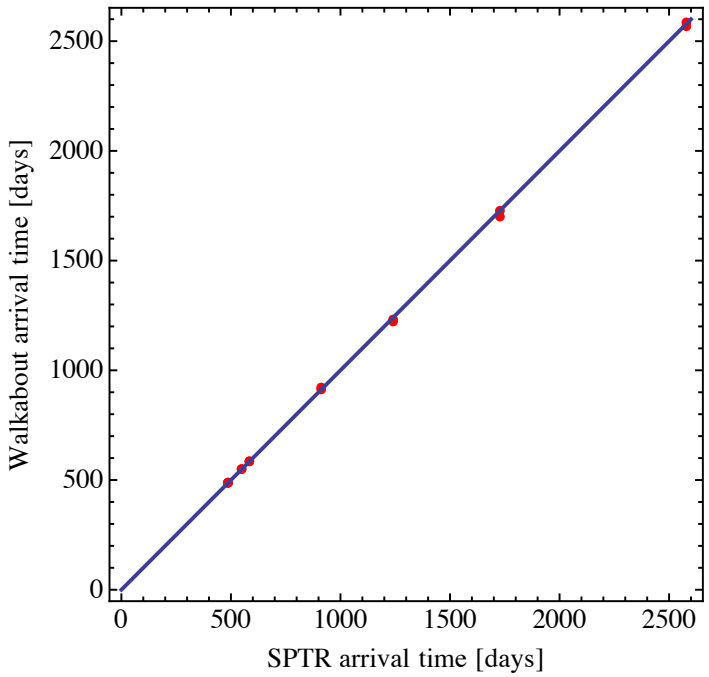

Figure 11. Result of Verification Test 3. Shown is a crossplot (red dots) of Walkabout arrival times versus SPTR times. The solid blue line has slope 1 (line of perfect agreement).

4.4 Verification Test 4

Verification Test 4 is a simple flow-through configuration with cross-section of 100 m × 100 m and length 20,000 m. The groundwater velocity is 1.96 ft/day. There are 106, 4 and 4 nodes in the x-, y-, and z- directions, respectively. The grid is uniform with 200 m spacing in the x direction except for the region from 10000 m to 11000 m where it is refined to 100 m.

For this test, Walkabout particle tracks are post-processed to create concentration versus distance at a single time, using the procedure of Robinson et al. (2010). A helper program PCLITE is included with the Walkabout distribution to postprocess the Walkabout results.

Concentration versus position at 5500 days is shown in Figure 12. The solid curve is the analytical solution from Kreft and Zuber (1978). The dots are the result of the Walkabout. The agreement is good over the entire domain. The slightly smaller concentration in the first cell is a dilution effect caused by the fact that the particles are released at x=10 m and not on the inlet boundary. Note that there are no artifacts in the refined grid region from 10,000 m to 11,000 m.

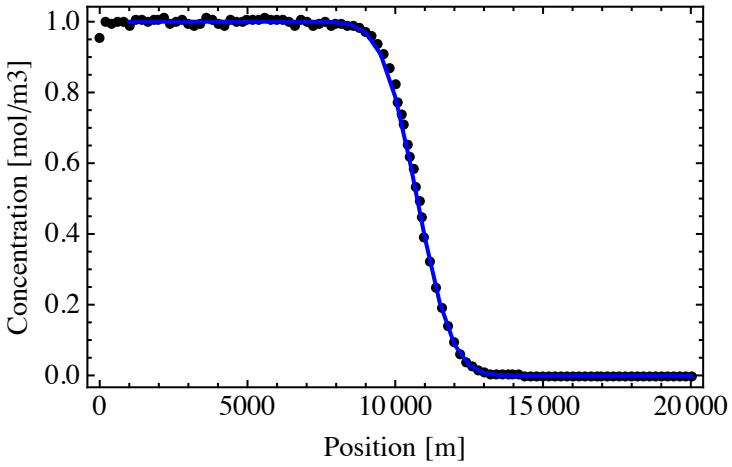

Figure 12. Result of Verification Test 4. Shown is concentration versus distance at 5500 days. Black dots are the result of Walkabout after post-processing using the procedure of Robinson et al. (2010). The blue curve is the analytical solution from Kreft and Zuber (1978).

5. Acceptance Tests

5.1 Acceptance Test 1

Acceptance Test 1 uses the same unstructured grid and flow barriers as in Verification Test 3. Particles are tracked with and without dispersion. The test is

intended to check for non-physical effects on a realistic flow/transport configuration involving strong permeability contrasts and unstructured grids.

Results with and without dispersion are shown in Figures 13 – 15. In producing these figures, Walkabout particle tracks were generated assuming an instantaneous release of particles. These particle tracks were then used to generate concentrations for a continuous source following the procedure of Robinson et al. (2010).

Anomalous accumulation of particles in low-permeability zones is a long-standing issue with random-walk particle tracking codes. Even in the case of relatively large dispersion (smallest grid Peclet number in Figure 15 is 0.25), no significant accumulation of particles is observed in low permeability zones.

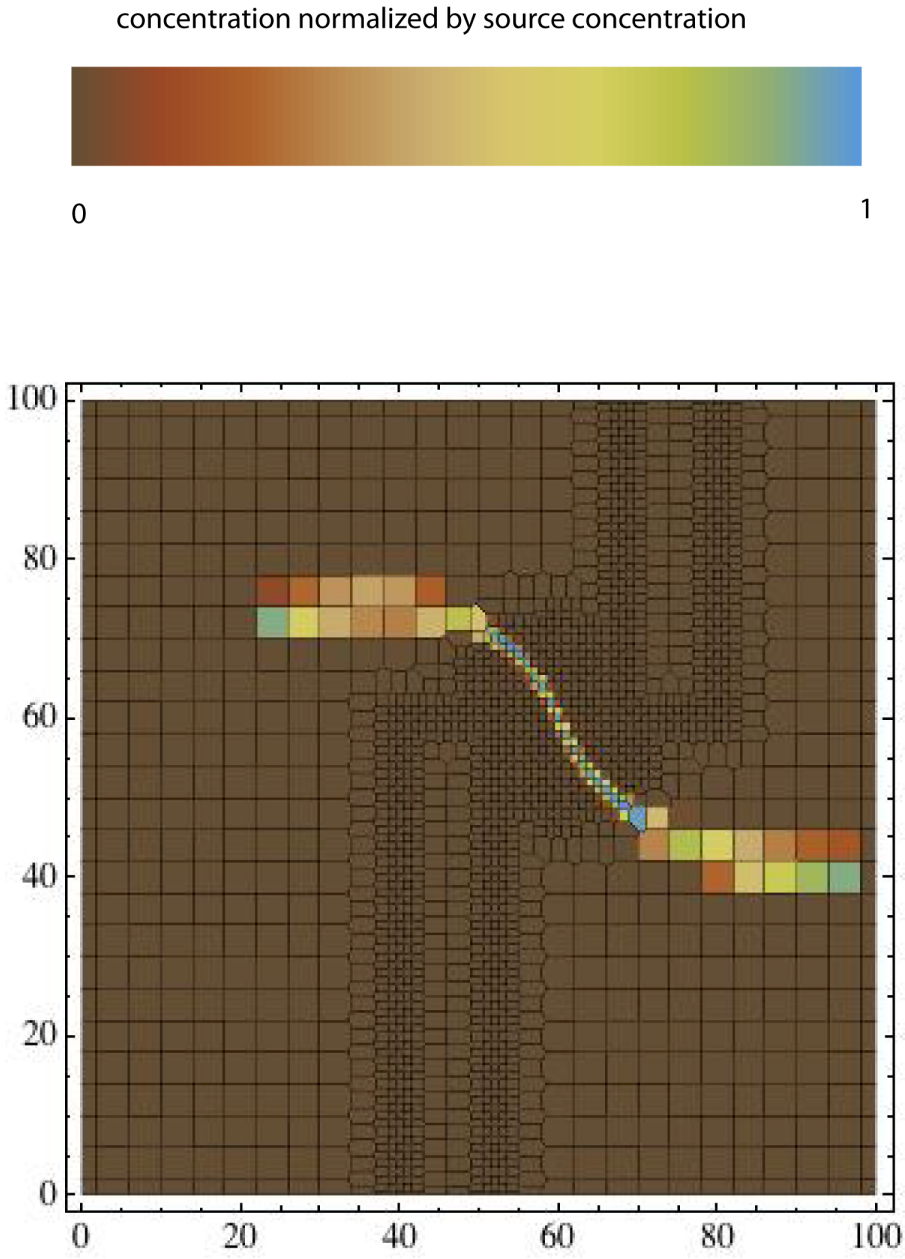

Figure 13. Horizontal slice of concentration from Acceptance Test 1 for the case without dispersion.

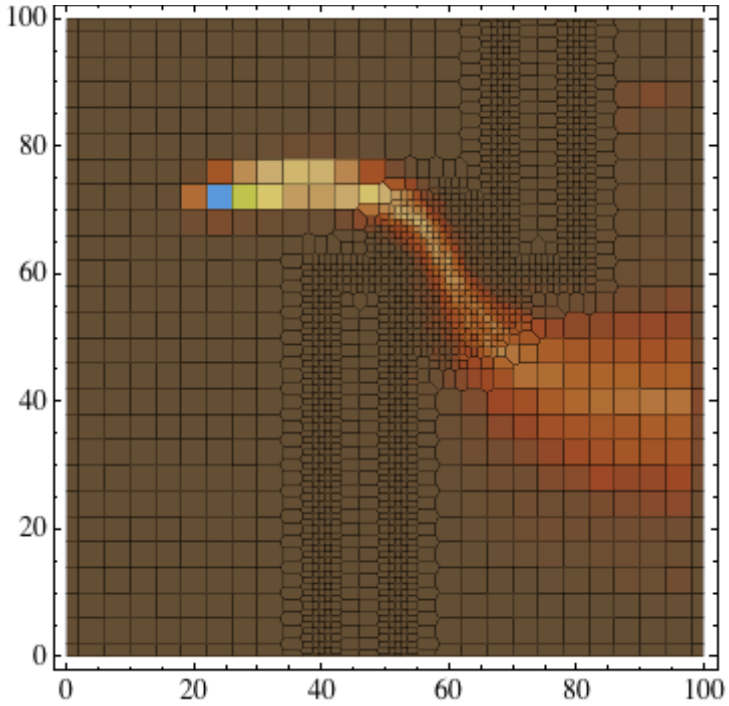

Figure 14. Horizontal slice of concentration from Acceptance Test 1. The longitudinal, transverse horizontal and transverse vertical dispersivity values are 1 m, 0.1 m and 0 m, respectively. The color map is the same as in Figure 13.

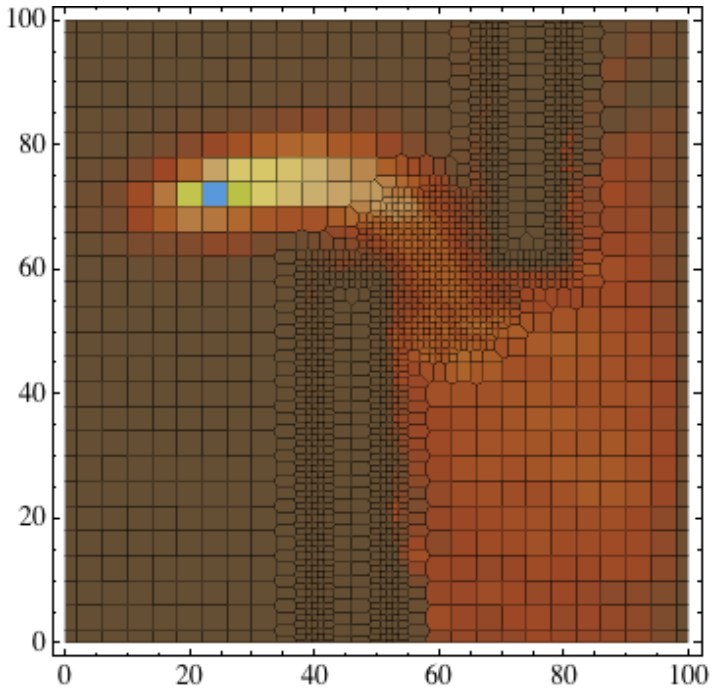

Figure 15. Horizontal slice of concentration from Acceptance Test 1. The longitudinal, transverse horizontal and transverse vertical dispersivity values are 5 m, 0.5 m and 0 m, respectively. The color map is the same as in Figure 13.

5.2 Acceptance Test 2

Acceptance Test 2 is designed to further test for anomalous mass accumulation in low permeability cells. This test uses a structured grid. There are two barriers to flow separated by a one-cell gap. Results are shown in Figure 16. No anomalous mass accumulation is apparent in this demanding test.

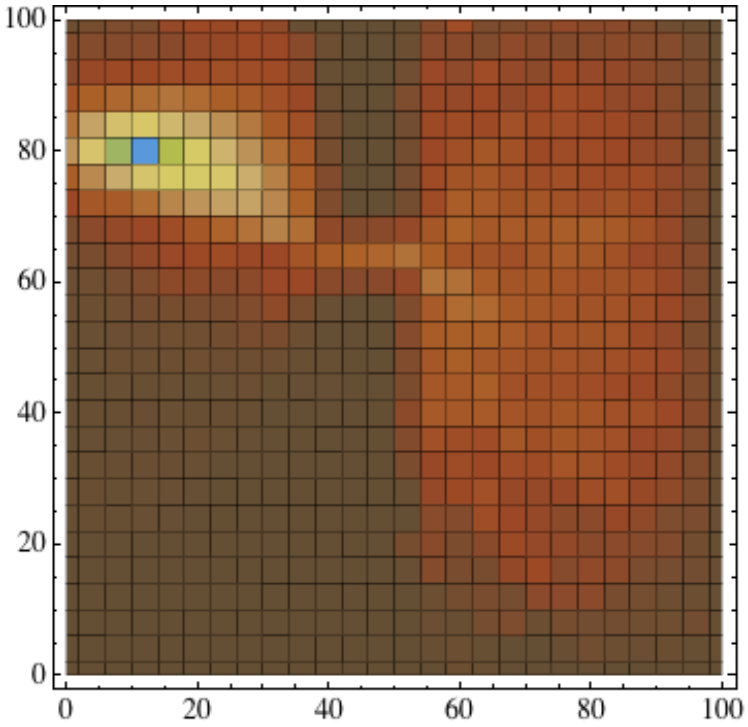

Figure 16. Horizontal slice of concentration from Acceptance Test 1. The longitudinal, transverse horizontal and transverse vertical dispersivity values are 5 m, 0.5 m and 0 m, respectively. The color map is the same as in Figure 13.

6 Known Issues

There are two known issues with Walkabout Version 1.0.

First, boundary facets for cells on a specified no-flow boundary must be aligned with the principal coordinate axes. This limitation is a consequence of the available information from the LaGrit software, which identifies boundary facets as “top”, “bottom”, “left”, etc., but does not provide a normal vector for each facet.

The second issue that the user should be aware of is that purely advective scenarios can, under certain conditions, produce spiky concentrations when post-processed by PLUMECALC or similar software. This phenomenon is demonstrated in Figure 17. At the end of each time step, Walkabout checks to see whether a particle has left the current finite volume cell. If the particle has left the cell, then the exit time is recorded for post-processing by PLUMECALC. When only advection is modeled, the spatial step is constant and the time spent is rounded to the next larger integer multiple of the fixed time step for some cells and to the next smaller integer multiple of the time step for other cells, thus creating the spiky pattern shown in Figure 17. When averaged over several cells, the concentration is correct. In addition, any process that adds a random component to the time spent in a cell (e.g. dispersion, matrix diffusion, spread in the initial positions of the particles) reduces or eliminates this artifact. The artifact may also be eliminated by choosing the advective time step to make the number of time steps within each cell exactly an integer (Figure 18).

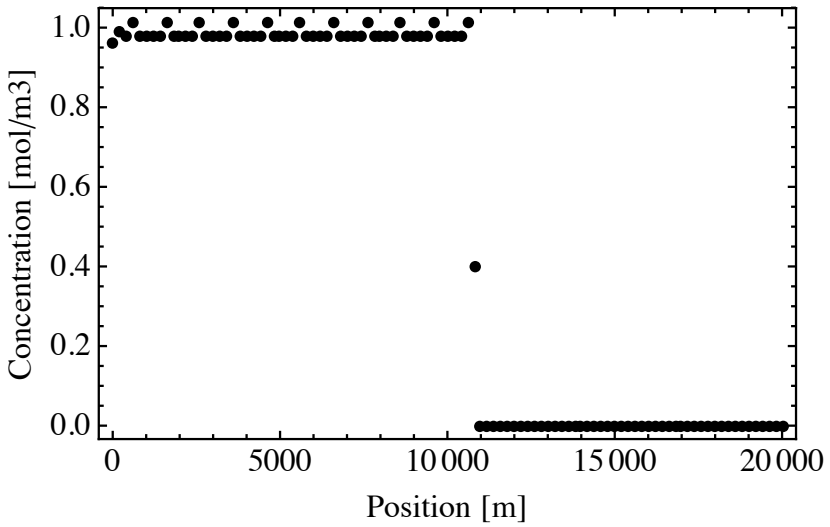

Figure 17. Concentration calculated by post-processing the results of Walkabout using the algorithm of Robinson et al. (2010) showing an concentration artifact associated with purely advective transport. Rounding of the particle dwell time in the cell because of the finite-sized step size causes concentration in every 5th cell to be overestimated with the rest underestimated slightly. The position of the front is in the correct position and the 5-cell average is correct in this case. Any process that adds a random component to the dwell time in a cell reduces or eliminates the artifact. It may also be eliminated by careful selection of the time step control parameters.

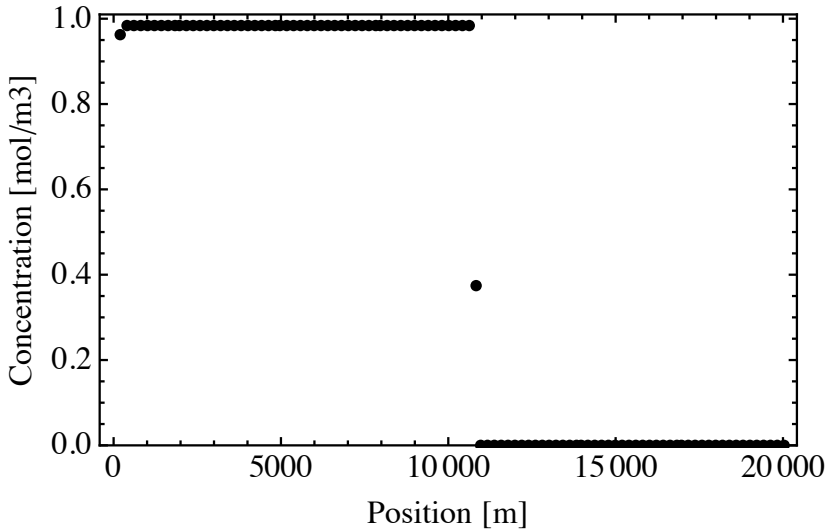

Figure 18. Same as in Figure 17, but with time step chosen to make the number of steps in each cell an integer. The artifact of Figure 17 is eliminated in this case.

7 Conclusions

Walkabout performs random walk particle tracking simulations on structured or fully unstructured grids. It is designed to work with LaGrit, FEHM and PLUMECALC. Based on verification and acceptance test results shown here, Walkabout Version 1 is sufficiently developed and tested to deploy in radionuclide transport calculations.

References

T. Amemiya. Advanced econometrics. Harvard University Press. 1985.

R.D. Burnett and E. O. Frind. Simulation of contaminant transport in three dimensions, 2, Dimensionality effects, Water Resour. Res., 23, 695– 705, 1987.

A. Kreft and A. Zuber, “On the Physical Meaning of the Dispersion Equation and its Solutions for Different Initial and Boundary Conditions.” Chemical Engineering Science, 33, 1471-1480. 1978.

T.N. Narasimhan and P. A. Witherspoon, An integrated finite difference method for analyzing fluid flow in porous media, Water Resour. Res., 12(1), 57–64, doi:10.1029/WR012i001p00057. 1976.

B.A. Robinson, Z.V. Dash, and G. Srinivasan, “A particle tracking transport method for the simulation of resident and flux-averaged concentration of solute plumes in groundwater models.”, Comput. Geosci. 14(14), 779-792, DOI: 10.1007/s10596-010-9190-6. 2010.

S. Painter, V. Cvetkovic, J. Mancillas, O. Pensado, “Random Walk Particle Tracking Methods for Simulating Transport with Retention and First-order Transformation.” Water Resources Research, 44, doi:10.1029/2007WR005944. 2008.

S. Painter, C.W. Gable and S. Kelkar. “Pathline Tracing on Unstructured Control Volume Grids.” submitted 2011.

D.W. Pollock. Semi-analytical computation of path lines for finite-difference models. Ground Water, 26(6):743–750, 1988.

G. A. Zyvoloski (2007) FEHM: A control volume finite element code for simulating subsurface multi-phase multi-fluid heat and mass transfer. Los Alamos Unclassified Report LA-UR-07-3359.

Appendix A. Input and Output Formats

A.1 The control file

The control file contains two keyword blocks that allow the user to specify dispersion tensors and particle start locations. In addition, it contains numerical parameters that control the functioning of the particle tracking simulation.

A.1.1 *INITIAL* keyword block

Particle starting positions are specified in the *INITIAL* keyword block. The keyword block may be placed anywhere in the **control** file. The form of the *INITIAL* keyword block is as follows

```
INITIAL  ! case sensitive
dist_type
num_part ! conditional input depends on dist_type
location_info ! conditional input depends on dist_type
:  ! repeat as needed
```

The dist_type keyword specifies how particles are to be distributed. Allowed values are *MANUAL*, *UNIFORM*, and *RANDOM* (case sensitive).

If *MANUAL* is specified then a particle location must be specified for each particle. In this case, the number of particles should be entered in the **num_part** field and start coordinates for each particle should be entered (**location_info** field, one x,y,z triplet on each line).

If *RANDOM* is specified, the particles are to be distributed randomly in a user-specified rectangular box. In this case, the number of particles should be entered in the **num_part** field and the **location_info** has two lines representing locations of minimum (first line) and maximum (second line) coordinates of the bounding box for the start region.

The *UNIFORM* keyword is similar to the *RANDOM* keyword except that particles will be distributed uniformly in the region of interest. In this case, three integers should be entered in the **num_part** representing spacing in the x, y, and z directions. The **location_info** lines are identical to the *RANDOM* case.

A.1.2 *DTENSOR* keyword block

Dispersion tensor information is given in the *DTENSOR* keyword block. The keyword block may be placed anywhere in the **control** file. The form of the *INITIAL* keyword block is as follows

```
DTENSOR
region_specifier !
type ! tensor type
dispersivities ! conditional on type
  : ! repeat above three lines as many times as needed
END ! terminates the keyword block
```

The **region_specifier** comes in two forms. If three integers (**min max stride)** are given, then the region comprises the nodes between **min** and **max** (inclusive) with the specified stride. The value 0 for **max** is interpreted as the last node in the grid. This format is identical to FEHM's format for specifying nodes in a region. If **region_specifier** starts with a nonnumeric character, then it is interpreted as a filename that contains zone information. In this case, the first field of the **region_specifier** is the filename and the second field of the **region_specifier** is the zone number in the specified zone file. The zone file should be in LaGrit's format.

The **type** keyword is specifies the type of the dispersion tensor. In Version 1.0, the only allowed value is *BF* denoting the Burnett and Frind tensor. The dispersivity_values required for the Burnett and Frind tensor are (in order) longitudinal dispersivity, horizontal transverse dispersivity, vertical transverse dispersivity, and molecular diffusion coefficient. The dispersivities have units of m. The molecular diffusion coefficient has units of m^2/day.

A.1.3 Numerical control parameters

Several numerical control parameters may also be specified in the control file. Defaults exist for each parameter. The parameters take scalar values may come in any order. The form is

```
parameter_name parameter_value
```

where **parameter_name** is one of the following

dtmax – maximum allowed value of time step in days. Optional. Defaults to 1000

dt0 – initial time step in days. Optional. Defaults to 0.01

maxstretch – maximum allowed time step relative to previous time step (dimensionless). Must be greater than 1. Optional. Defaults to 1.2

maxsteps – maximum number of time steps in the simulation. A particle is terminated if maxstep timesteps are taken. Optional defaults to 100000

dxtarget – Courant factor, maximum allowed time step relative to time required to advect across the cell. Optional. Defaults to 0.1

dttarget – maximum allowed time step as fraction of characteristic time to disperse across cell. Optional. Defaults to 0.1

toutfreq – number of time steps between output of trajectory data. If 0 is specified, no trajectory output is written. Optional. Defaults to 0

A.1.4 Example **control** file

Examples of control files are given in Figures A-x and A-y. In the example in Figure A-x, 10000 particles are released randomly in the region between (10,-50,-50) and (10,50,50). The dispersion coefficient in this example is

```
Title – the title goes here
dtmax 100  !maximum step size days
dxtarget 0.1  !relative to grid size
dttarget 0.1  ! relative to characteristic dispersion time
maxstretch 1.3
maxsteps 100000  ! maximum number of steps allowed
dt0 0.1  ! initial step days
toutfreq 0  !no trajectory output

INITIAL
 RANDOM
 10000
 10.0 -50.0 -50.0
 10.0  50.0  50.0

DTENSOR
1 0 0
BF  !burnett and frind tensor
 40. 0.0 0.0 0.0  !dispersivity values
END
```

Figure A-1. Example **control** file.

```
Title Goes Here
dtmax 100  !maximum step size days
dxtarget 0.1  !relative to grid size
dttarget 0.1  ! relative to characteristic dispersion time
maxstretch 1.3
maxsteps 100000  ! maximum number of steps allowed
dt0 0.1  ! initial step days
toutfreq 0  !no trajectory output

INITIAL
 RANDOM
 10000
 10.0 -50.0 -50.0
 10.0  50.0  50.0

DTENSOR
1 0 0
BF  !burnett and frind tensor
 40. 0.0 0.0 0.0  !dispersivity values
END
```

Figure A-2. Example **control** file.

A.2 Files describing geometry of the mesh

Three files describing geometry of the computational grid are required, and a fourth is optional. It is anticipated that these files will be produced by the LaGrit software, although any grid generation software could be used provided the output is converted to the required format.

A.2.1 The **fehmn** file

The **fehmn** file provides geometry information (location of nodes and lists of nodes that compose each element). See the LaGrit and FEHM manuals for details. In Walkabout Version 1.0, the **fehmn** file must be provided in ASCII format.

A.2.2 The **stor** file

The **stor** file provides information about nodal connectivity and interface areas. See the LaGrit and FEHM manuals for details. LaGrit options exist to produce vector areas, scalar areas, or ratio of scalar area to distance for each node-to-node connection. The latter option is required by Walkabout. In Walkabout Version 1.0, the **stor** file must be provided in ASCII format.

A.2.3 The **ealist** file

The **ealist** file provides information about element adjacency, which should not be confused with nodal connectivity. Element adjacency is not needed by FEHM,

but is required by Walkabout. See the LaGrit manual for details on how to produce the element adjacency lists.

A.2.4 The **cbound** file

The **cbound** (closed boundary) file provides a list of nodes on boundaries that are closed to transport. It has the same format as the LaGrit outside zone file, but excludes outflow boundaries. A simple strategy for producing the **cbound** file is to use LaGrit to produce a list of all outside nodes, and then remove those nodes associated with outflow boundaries.

In Version 1, external faces of cells on a boundary must aligned with the principal directions in the coordinate system. That is, boundary faces must be top, bottom, left, right, back or front. See the LaGrit manual. It is important to recognize, that this restriction only applies to cell faces on boundaries. Cell faces internal to the model have no such restrictions. A node/cell may have more than one no-transport boundary face associated with it, in which case it would appear more than once in the **cbound** file.

For nodes on a no-transport boundary, Walkabout first attempts to reconstruct the nodal velocity using the unconstrained algorithm, Eq. 5. If this procedure results in inflow into the domain at the boundary node, the reconstructed velocity is used as is. If the unconstrained procedure results in outflow on a no-transport boundary, then the velocity reconstruction is repeated using the constrained procedure Eq. 7 to enforce the no-flow condition on the cell's boundary face. Particles are not allowed to disperse across boundaries that are closed to transport.

A.3 Files containing FEHM results

Two files containing required FEHM results are required.

Internodal liquid fluxes are read from the **fin** file, an FEHM restart file in ASCII format. If the liquid fluxes are not found in the fin file, Walkabout will terminate.

Porosity, liquid saturation and liquid density are read from an ASCII **avs** file produced by FEHM. The **avs** file is produced by the FEHM contour macro. See the FEHM manual for details.

A.4 Output files

Two output files are always produced, a log file that echoes back input parameters and a reduced SPTR2 file for PLUMECALC. The latter may be read directly by PLUMECALC.

In addition, to the two files always produced, a trajectory file (**trajout**) may be produced if the *toutfreq* parameter is set greater than 0 (see section A.1). The **trajout** file has format

```
!header line
!header line
npart !number particles
 ntimes !number time steps reported for this particle
    t1 x1 y1 z1 ! spatial position at time t1
    : ! repeat above line for a total of ntimes
 : ! repeat each particle block for a total for npart particles
```