VALIDATION AND DOCUMENTATION OF AN INDEPENDENT GROUNDWATER FLOW MODELING TOOL (xFLO) FOR USE IN PERFORMANCE CONFIRMATION OF A GEOLOGIC REPOSITORY

Prepared for

U.S. Nuclear Regulatory Commission
Contract NRC–HQ–12–C–02–0089

Prepared by

Stuart Stothoff

Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

June 2017

CONTENTS

FIGURES

TABLES

EXECUTIVE SUMMARY

xFlo is a computer code for modeling coupled thermal and hydrological processes in variably saturated and heated porous and fractured media. xFlo was developed as a replacement for the MULTIFLO computer code, which was developed in the mid-1990s and early 2000s to aid in understanding perturbations to the near-field environment surrounding a proposed underground high-level nuclear waste repository at Yucca Mountain, Nevada, following emplacement of the waste. This report documents the status of xFlo in regard to the expected capabilities for replacing MULTIFLO and provides code validation and related verification analyses needed to place xFlo Version 1.1β under formal software control.

MULTIFLO 2.0 was capable of being used to address drift- and repository-scale coupled thermal-hydrological-chemical processes that could affect the performance of the proposed repository. MULTIFLO linked two codes, METRA and GEM, that provided the capabilities for thermal-hydrologic modeling and multicomponent transport modeling. MULTIFLO was capable of being applied to assess processes such as

- Isothermal and nonisothermal movement of water through variably saturated rock as liquid and vapor (METRA)
- Evolution of groundwater compositions near and within the engineered barrier system (GEM)
- Changes in porosity and permeability of the host rock resulting from mineral alteration and the resulting effects on fluid transport (GEM)
- Transport of aqueous and gaseous radionuclides from the waste package (GEM)

MULTIFLO was designed to be a general integral finite-volume code for simulating multiphase, multicomponent transport processes in nonisothermal systems with chemical reactions and reversible and irreversible phase changes in solids, liquids, and gases in one, two, and three spatial dimensions. MULTIFLO considered single-phase liquid, two-phase gas and liquid, and single-phase gas conditions. MULTIFLO represented rocks with both matrix and fractures using equivalent continuum and dual-continuum approaches, including the active fracture model. MULTIFLO is entirely coded in FORTRAN 77.

Porting MULTIFLO to xFlo, taking advantage of capabilities of newer releases of Fortran, began prior to the Yucca Mountain license application review. The new code, xFlo, was intended to replace MULTIFLO for use in performance confirmation activities. The porting process was incomplete at the time of the license application, with xFlo replicating most of the capabilities for METRA but none of the GEM capabilities. Since that time, xFlo has been used for other projects, which has led to model revisions.

xFlo Version 1.1β is capable of modeling single-phase liquid, two-phase gas and liquid, and single-phase gas conditions, and includes single- and dual-permeability capabilities. In principle, xFlo is capable of considering multiple interacting continua, but this has never been implemented. METRA capabilities for considering an equivalent continuum, radiation boundary conditions, and outflow boundary conditions (e.g., gravity drainage) have not been implemented. xFlo permits tabular input of constitutive properties, which is not tested in this report, and this capability could be used to represent the equivalent continuum approach. The missing capabilities may all be useful for performance confirmation modeling activities.

Four test case suites that compare xFlo results with an applicable comparison solution are documented in this report to verify that xFlo correctly implements the intended mathematical models. Two test case suites were developed during earlier modeling activities in order to build confidence that the code correctly implements the balance equations in one-dimensional (1-D) and two-dimensional (2-D) systems, and this report documents these test cases. A third test case extends MULTIFLO validation Test Case 2, and the fourth test case verifies that one-way coupling between balance equations is correctly implemented in 2-D.

MULTIFLO validation Test Case 1 was identified as a fifth test case verifying that xFlo correctly implements the nonlinear constitutive relationships and strongly coupled balance equations, but developing the necessary comparison model required effort inconsistent with the project schedule.

The two earlier test cases used special-purpose MATLAB® routines to generate the input files that drive xFlo. The test cases developed for this report used a preliminary version, currently under development, of an interface coupling xFlo with the geomechanical FLAC code to generate the xFlo input files, run xFlo, and archive the results. The interface uses MATLAB routines with Microsoft® Excel® spreadsheet files to develop 1-D and 2-D models.

The test cases are intended to separately test the implementations of (i) the coupled mass and energy balance equations given the model parameters and (ii) the constitutive relationships used to derive the model parameters. The test cases demonstrate that the approach to solving the mass and energy balance equations is correctly implemented, as are the retention relationships.

The test cases identified two discrepancies between the comparison solution and xFlo results:

- The transfer flux between matrix and fracture continua in dual-permeability models does not match without scaling the matrix/fracture area. The scale factor appears to be a constant regardless of the problem.

- Based on the development of unexpected or unexpectedly large perching, xFlo may underestimate liquid flux across interfaces between media with strongly contrasting permeabilities and retention properties, even with local grid refinement at the interface and the fully upstream weighted computational scheme. Graded material properties are correctly handled in the mass balance equation and strongly contrasting properties are correctly handled in the energy balance equation (which has only weak state-dependence in the model parameters). By implication, the underestimated liquid flux may be related to extremely steep gradients developing in the state-dependent model parameters or the unexpected extent of perching may be in part simply a transient response during equilibration. The xFlo simulations exhibiting this phenomenon always terminated before equilibrium was reached, either due to input instructions or due to an unexpected internal event.

The discrepancy in transfer flux is likely to be due to a single coding error that should be relatively straightforward to resolve in future work.

Additional work is necessary to understand what conditions lead to an incorrect representation of liquid fluxes across interfaces. The particular test case was selected to be rather extreme, and prior work considering interfaces with less extreme contrasts suggests that liquid fluxes are appropriately handled or overestimated.

Additional work is necessary to understand why xFlo unexpectedly terminates for some of the simulations testing layered systems. Prior work suggests that simulations typically grind to a halt with continually decreasing time steps when convergence is an issue, and it is unusual for xFlo to simply halt with no error message. This unexpected termination is likely due to xFlo failing to trap an out-of-bound condition for a constitutive equation, perhaps related to variable switching in grid cells transitioning between partial and full saturation.

ACKNOWLEDGMENTS

This report was prepared to document work performed by the Center for Nuclear Waste Regulatory Analyses (CNWRA®) for the U.S. Nuclear Regulatory Commission (NRC) under Contract No. NRC–HQ–12–C–02–0089. The activities reported here were performed on behalf of the NRC Office of Nuclear Material Safety and Safeguards, Division of Spent Fuel Management. This report is an independent product of CNWRA and does not necessarily reflect the views or regulatory position of NRC.

The authors would like to thank P. LaPlante for his technical review and D. Pickett for his programmatic review. The authors also appreciate C. Barberino for quality assurance support and A. Ramos for providing word processing support in preparation of this document.

QUALITY OF DATA, ANALYSES, AND CODE DEVELOPMENT

DATA: All CNWRA-generated data contained in this report meet quality assurance requirements described in the Geosciences and Engineering Division Quality Assurance Manual.

ANALYSES AND CODES: The computer software xFlo Version 1.1β (Stothoff and Painter, 2016), MATLAB Version 8.6 (MathWorks®, 2015), and COMSOL Multiphysics® Version 5.1 (COMSOL®, 2015) were used in the analyses contained in this report. xFlo (Version 1.1β) is controlled under Technical Operating Procedure (TOP)–018, Development and Control of Scientific and Engineering Software; a formal Software Validation Report documenting all calculations and model input values has been completed. COMSOL Multiphysics Version 5.1 and MATLAB Version 8.6 are commercial software also controlled under TOP–018. A limited validation was performed on COMSOL Multiphysics Version 5.1 according to TOP-18 guidelines for the type of application used for xFlo testing. Documentation for theoretical analyses can be found in Scientific Notebook 1177E (Stothoff, 2017).

References:

COMSOL. "COMSOL Multiphysics, version 5.1." Stockholm, Sweden: COMSOL AB. 2015.

MathWorks. "MATLAB, version 8.6." Natick, Massachusetts: The MathWorks, Inc. 2015.

Stothoff, S. "xFlo Development and Analyses." Scientific Notebook No. 1177. San Antonio, Texas: Center for Nuclear Waste Regulatory Analyses. 2017.

Stothoff, S. and S. Painter. "xFlo Version 1.1β User's Manual: DECOVALEX–2015 Version." San Antonio, Texas: Center for Nuclear Waste Regulatory Analyses. ML16167A315. U.S. Nuclear Regulatory Commission. 2016.

CONVERSION FACTORS

Length, area, pressure and temperature conversions have been included in the text. The conversions for other parameters are provided in this table.

Name	SI Unit	Conversion
Density	1 g/cc 1 kg/m^3	= 0.036 lb_m/in^3 = 0.06243 lb_m/ft^3
Heat transfer rate	1 W	= 3.4123 Btu/hr
Length	1 cm 1 m	= 0.394 in = 3.28 ft
Permeability	1 m^2	= 1.01 × 10^{12} darcies
Area	1 m^2	= 10.8 ft^2
Power	1 W	= 3.412 Btu/hr
Pressure and Stress*	1 N/m^2 (1 Pa) 1 MPa	= 0.000145 psi = 145 psi
Specific heat	1 kJ/kg·K	= 0.2389 $Btu/lb_m·°F$
Thermal conductivity	1 W/m·K	= 0.57782 Btu/hr·ft·°F
Temperature	1 °F	= 1.8 × T °C + 32
Viscosity (dynamic)†	1 $N·s/m^2$	= 5.8016 × 10^{-6} $lb_f·hr/ft^2$
Mass	1 g	= 0.002 lb_m
Physical Constants		
Gravitational Acceleration (Sea Level)		g = 9.807 m/s^2 = 32.174 ft/s^2

*The SI name for the quantity pressure is Pascal (Pa) having units N/m^2 or $kg/m·s^2$
†Also expressed in equivalent units of kg/s·m

1 INTRODUCTION

The computer code MULTIFLO was developed by the Center for Nuclear Waste Regulatory Analyses (CNWRA®) in the mid-1990s and early 2000s to aid in understanding perturbations to the near-field environment surrounding a proposed underground high-level nuclear waste repository at Yucca Mountain, Nevada, following emplacement of the waste. MULTIFLO was designed to be a general integral finite-volume code for simulating multiphase, multicomponent transport processes in nonisothermal systems with chemical reactions and reversible and irreversible phase changes in solids, liquids, and gases in one, two, and three spatial dimensions. MULTIFLO considered single-phase liquid, two-phase gas and liquid, and single-phase gas conditions. MULTIFLO represented rocks with both matrix and fractures using equivalent continuum and dual-continuum approaches, including the active fracture model. MULTIFLO was coded in FORTRAN 77.

MULTIFLO 2.0 was capable of being used (Painter, 2005) to address drift-scale and repository-scale coupled thermal-hydrological-chemical processes that could affect the performance of the proposed repository, and could be applied to assess processes such as

- Isothermal and nonisothermal movement of water through unsaturated rock as liquid and vapor
- Evolution of groundwater compositions near and within the engineered barrier system
- Changes in porosity and permeability of the host rock resulting from mineral alteration and the resulting effects on fluid transport
- Transport of aqueous and gaseous radionuclides from the waste package

Porting MULTIFLO to a newer code, taking advantage of capabilities of newer releases of Fortran, began prior to the Yucca Mountain license application review. The new code, xFlo, was intended to replace MULTIFLO for use in performance confirmation activities. At the time of the license application, the conversion was incomplete. Since that time, xFlo has been used for other projects, which has led to model revisions. This report documents the status of xFlo Version 1.1β in regard to the expected capabilities for replacing MULTIFLO and provides information to meet software validation requirements for xFlo Version 1.1β as described in CNWRA Technical Operating Procedure TOP-018..

Repository performance confirmation activities related to groundwater span the areas of general requirements, confirmation of geotechnical and design parameters, design testing, and monitoring and testing of waste packages. Pertinent activities include

1. Subsurface Water and Rock Testing
2. Unsaturated Zone Testing
3. Seepage Monitoring
4. Thermally Accelerated Drift Near-Field Monitoring
5. Thermally Accelerated Drift In-Drift Environment Monitoring

6. Thermally Accelerated Drift In-Drift Thermal-Mechanical Monitoring (DOE would compare with THMC model results)

7. Saturated Zone Monitoring

8. Saturated Zone Fault Hydrology Testing

9. Saturated Zone Alluvium Testing

10. Seal and Backfill Testing

Based on these activities, a model that calculates nonisothermal groundwater flow and transport could be applied to examine (i) water, air, and dissolved chemical fluxes in the matrix and fractures of the unsaturated zone, in backfill, and in seals under ambient and thermally perturbed conditions; (ii) seepage and vapor transport into ventilated and unventilated mined openings and vapor redistribution within the openings under thermally perturbed conditions; and (iii) saturated zone flow and transport. Although it is recognized that performance confirmation activities related to the saturated zone may have used codes specialized for saturated conditions, other performance confirmation activities may need to consider locally saturated conditions above the water table, for example in perched zones or thermally induced saturated halos.

The xFlo simulator (Stothoff and Painter, 2016) is an integrated finite volume code. The code is implemented as an unstructured grid of cells and one-dimensional (1-D) links between cells, with a variety of constitutive relationships defining relationships between state variables and fluxes. The user is responsible for defining an appropriate grid for the problem of interest. Because xFlo is implemented as a set of 1-D links, it lends itself to a computational molecule of approximately orthogonal links, and computations are usually most accurate when the grid consists of approximately orthogonal brick elements aligned with the principal directions of flow (like a finite difference grid). Grids with orthogonal links are known to introduce substantial numerical diffusion into advection-dominated transport calculations, especially when flows are not aligned with the grid direction, and such problems typically require more complex computational strategies than provided by xFlo. However, the relatively simple approach to gridding is reasonable for resolving thermal-hydrologic problems featuring coupled nonlinear diffusion-dominated fluxes with phase changes, because the solution often forms self-sharpening fronts controlled by the nonlinear state-dependent properties.

The confirmatory analyses described in this report were selected to verify that the version of xFlo documented by Stothoff and Painter (2016) correctly implements the governing balance equations and the constitutive relationships included in the model. For this purpose, code validation includes (i) performing a series of validation tests to provide reasonable confidence that the software successfully implements the underlying theory and algorithms and (ii) documenting the analyses. In order to verify that the simulator is working correctly, it is useful to consider a suite of simulations that (i) test particular key aspects of the simulator and (ii) provide benchmarks for comparison after code modification.

The comparison problems address the following verification topics:

- Are Dirichlet, flux, and source conditions correctly implemented?
- Are fluxes correctly implemented in 1-D/two-dimensional (2-D)/three-dimensional (3-D)?
- Are constitutive relationships correctly implemented?

- Are coupled effects correctly implemented?
- Are fluxes between grid cells with different constitutive properties correctly implemented?

The confirmatory analyses largely rely on comparisons with analytical or semi-analytical results in 1-D to demonstrate that the code structure of cells and 1-D links is correctly implemented. Supplemental 2-D problems are provided to verify that the model correctly implements the governing equations on grids with more complex network structures.

2 PRIOR WORK

2.1 xFlo Validation

The xFlo model considers simultaneous solution of three balance equations (water mass, air mass, and energy). A suite of test cases were previously developed in 2014 to build confidence in xFlo calculations for modeling coupled themal-hydrologic processes. The test cases were used to confirm that xFlo correctly represents (i) Dirichlet boundary conditions, flow boundary conditions, and sources, and (ii) linear, cylindrical, and spherical flow conditions. Each test case considered a single balance equation, including isothermal unsaturated flow and thermal diffusion. For comparison with xFlo, the test cases were solved using a combination of (i) analytical and semi-analytical solutions, (ii) numerical solutions based on an intrinsic MATLAB routine, and (iii) comparison with another numerical model. For all test cases, xFlo calculated all three balance equations. This report formally documents these test cases for xFlo Version 1.1β as the test case 1 suite and test case 3.

2.2 MULTIFLO Validation

Validation testing for MULTIFLO version 1.5.2 (Painter, 2003) and 2.0 (Painter, 2005) was intended to cover the major capabilities of the code that are to be used in regulatory activities:

1. Nonisothermal multiphase flow and phase-change phenomena in partially saturated porous media

2. Flow in composite fractured/porous media using a dual continuum formulation

3. Flow in saturated porous media including compressibility effects

4. Advective and diffusive transport of chemicals in the aqueous and gaseous phase

5. Equilibrium speciation of aqueous and gaseous phase constituents

6. Kinetically controlled mineral formation and dissolution, and resulting effects on porosity, permeability, and flow

7. Unstructured grid configuration with arbitrary interblock connectivity

The validation testing considered nine test cases (eight for Version 1.5.2 and one for Version 2.0.1):

1. Self-similar response of a partially saturated one-dimensional (1-D) cylindrical domain to a cylindrical heat source. The test case used an analytical approach based on the Boltzmann transform (Doughty and Pruess, 1992) to represent nonisothermal two-phase (gas and liquid) and two component (air and water) redistribution for comparison.

2. Steady infiltration in dual permeability media. The test case considers vertical partially saturated isothermal flow in dual permeability media (matrix and fractures) with specified saturation at top and bottom, using the Richards approximation of infinitely mobile air. Painter (2003, 2005) used a shooting algorithm to identify the steady-state flux consistent with the specified bottom pressure and top saturation for comparison.

3. Drawdown in an infinite confined aquifer. The test case considers isothermal saturated 1-D radial flow with compressibility in an aquifer. Painter (2003, 2005) used the analytical solution by Theis (1935) for comparison.

4. Equilibrium speciation in GEM. *This problem tests a capability not included in xFlo.*

5. Solute transport in dual permeability media. The test case considers advective/diffusive transport with constant flow in 1-D with flow in both matrix and fractures and an instantaneous increase in concentration at the inlet. Painter (2003, 2005) used a semi-analytical model based on the Laplace transform method for comparison.

6. Three-dimensional (3-D) advective/dispersive transport. The test case considers advective/diffusive transport with constant flow in 3-D with a rectangular patch source perpendicular to flow. Painter (2003, 2005) used an analytical solution for comparison.

7. Fully coupled flow/transport with mineral dissolution and permeability dissolution. *This problem tests a capability not included in xFlo*.

8. Unstructured grid capability. This repeats test problem 3 with an unstructured grid. *xFlo is inherently designed with an unstructured grid and therefore the test is redundant.*

9. Node (i.e., grid cell) renumbering scheme. This runs an unstructured grid simulation with and without internal node reordering to improve solver performance. *This problem tests a capability not included in xFlo*.

Currently, xFlo can consider test cases 1 and 2 without modification. Test case 3 uses a specific storage or storativity parameter, which is hard-coded into xFlo and not part of the current xFlo input, but otherwise xFlo can consider test case 3. Test cases 5 and 6 apply to solute transport; xFlo directly considers solute transport only for dissolved air but the analytical solutions could be used to test transport of air, water, and heat. Test cases 4, 7, and 9 require capabilities that are not currently coded in xFlo.

Test cases 8 and 9 test particular features of the MULTIFLO coding rather than numerical capabilities. The difference between test cases 3 and 8 (structured versus unstructured grids) is not described in the validation, but xFlo uses a fully unstructured grid and does not have a structured grid. Test case 9 tests a capability that is not included in xFlo, but node numbering is part of the input and an optimal numbering scheme can be applied (with limitations regarding boundary nodes).

Test case 2 of the MULTIFLO validation set inspired the xFlo Version 1.1β test case 2 suite, in which essentially the same solution approach is adapted to a wider variety of parameters.

Test case 6 of the MULTIFLO validation set inspired xFlo Version 1.1β test case 4. The specialization of the analytical solution into 2-D was used to consider heat transport.

Test case 1 of the MULTIFLO validation set is directly applicable and would provide a valuable test, but reproducing the analytical solution was not feasible within project constraints.
Test case 3 was not reproduced because it requires xFlo code modification. Test cases 4, 5, 7, 8, and 9 were not reproduced because of minor (case 5) or no (case 4, 7, 8, and 9) applicability.

3 TEST CASE 1 SUITE: 1-D DIFFUSIVE TRANSPORT

The test case 1 suite of solutions considers transient one-dimensional (1-D) single-continuum single-component diffusion.

The test case 1 suite uses 1-D scenarios to verify correct implementation of (i) boundary and source conditions, (ii) fluxes between grid cells, and (iii) constitutive relationships. The suite relies on the PDEPE routine provided by MATLAB using independently coded constitutive relationships. A second model, based on a semi-analytical approach and also implemented in MATLAB, is used to confirm the implementation of the PDEPE routine. The PDEPE-based model and the semi-analytical model use the same routines to describe constitutive properties and typically agree to within a fraction of a percent or better with respect to change in mass during the simulation when equivalent discretization levels are used. An analytical model might be used for special scenarios, but it is easier to make direct comparisons using the semi-analytical model and the PDEPE model because these models can account for initial mass existing in the xFlo domain. The comparison models are discussed in more detail in Section A1 of Appendix A.

The test case suite considers the thermal diffusion and Richards equations with no gravity. Three coordinate systems are used for each problem: (i) Cartesian, (ii) cylindrical, and (iii) spherical.

The thermal diffusion equation applies to a single component, energy. The test suite considers small perturbations to decouple and linearize the problem physics.

The isothermal Richards equation applies to unsaturated liquid flow, and verifies that nonlinear constitutive relationships are properly implemented.

The test problems are all 1-D in the various coordinate systems. All of the test problems were run with the PDEPE routine and xFlo. The semi-analytical model is also presented for selected problems as cross-validation. The PDEPE and xFlo simulations always used the same grid discretization unless otherwise specified. Results from the semi-analytical approach are essentially identical to the PDEPE results before the propagating front reaches the outer boundary. After that point, the outer boundary conditions are no longer the same and the solution begins to differ.

3.1 Boundary Conditions and Sources in xFlo

xFlo inherently uses an unstructured grid built on several input files, although a built-in preprocessor builds some of the input files for structured grids. All of the verification testing directly defines the unstructured grid files in order to preserve flexibility in defining gridding.

By default, xFlo imposes a zero-flux boundary condition on all boundaries and assumes that there are no sources in any cell. As currently implemented, boundary conditions are imposed using boundary condition cells that are linked to interior cells. There is no limit to the number of links to any cell in an unstructured grid, so multiple boundary condition cells can be attached to a single interior cell and each boundary condition cell can be attached to multiple interior cells. Because internal tables are dimensioned according to the maximum number of links to any cell (including boundary condition links), best practice limits memory usage by attaching each boundary condition cell to at most a small number of interior cells. One or more sources can be applied to any grid cell, but sources are ignored for boundary cells.

Values of the state variables and mole fluxes in xFlo are interpolated to the start of each time step in each boundary condition and source definition. These values remain fixed through the time step, lagging the boundary condition and source information by one time step. This has no impact when the boundary conditions are fixed in time, but is inconsistent with fully implicit time stepping. A future enhancement to xFlo may be desirable to make the boundary conditions consistent with the time stepping scheme.

There are two types of boundary conditions coded in xFlo, specified with the DIRICHLET and INFILTRATION keywords. For both boundary conditions, all three state variables are defined in the boundary condition cell and all three state variables are used to define fluxes to the interior cell. The DIRICHLET boundary condition is limited to the definition of the boundary state variables. The INFILTRATION keyword refers to a type of mixed boundary condition that also adds a specified boundary flux to the interior cell as an equivalent source term.

The preprocessor routines also define an INFILTRATION boundary condition for structured grids. The preprocessor routines convert specified volumetric water flux, pressure, and temperature into the corresponding state variables plus mole flux of air, mole flux of air plus water, and enthalpy flux as required for the boundary condition input file.

There is no explicit method for defining mixed conditions, such as a specified temperature or energy flux with no mass flux, but xFlo can accommodate mixed conditions by appropriately redefining material properties in boundary cells. This method is not documented. To turn off mass flux, the boundary cell must be assigned zero permeability (which turns off gas and liquid flow), zero porosity (which turns off vapor diffusion), and zero liquid diffusion coefficient (which turns off diffusion of air in water). To turn off thermal conduction, the boundary cell must be assigned zero dry and wet thermal conductivities.

In all of the test problems, all xFlo boundary cells were assigned zero volume and zero thickness. Note that this approach would fail for models that implement Dirichlet boundary conditions as an extremely large reservoir, but is permissible for xFlo boundary cells. In all of the test problems with the INFILTRATION boundary condition, the material properties in the boundary cells were redefined to shut off mass and energy exchange with the interior.

The source term is ignored unless it is added to interior cells. In order to test the implementation of the source condition, each boundary flux case was rerun as a source. The preprocessing routines used to compare the xFlo simulations with the PDEPE and semi-analytical approaches computed the source term by multiplying the boundary flux with the area of the boundary, and applied the result as a source term to the interior cell adjacent to the boundary. The corresponding boundary cell was omitted when the source term was used. The two procedures for applying boundary fluxes yielded identical solutions, therefore the source solutions are not shown in the following figures.

In all of the test problems, the initial conditions are uniform throughout the domain except for a ramp condition at one boundary. With the ramp condition, all of the methods start with the same amount of mass and energy within the system, regardless of discretization.

3.2 Thermal Diffusion

The first set of test problems consider a fixed and uniform moisture state with a mild thermal change imposed at one boundary. The constitutive properties vary little over the temperature range considered, and are assumed constant for the PDEPE and semi-analytical solutions.

The properties of the porous medium are based on bentonite material considered in previous modeling of coupled thermal-hydrologic processes. Table 1 provides the parameters used for verification testing of the thermal and Richards-based equations.

3.2.1 One Cartesian Dimension

The Cartesian model domain is similar to a laboratory column test. The model domain is 50 cm long. The entire model domain is filled with the porous medium, which has a uniform initial saturation throughout. An initial temperature of 6 °C is applied everywhere except for a linear ramp in temperature over 1 cm at the top boundary. The top boundary value is 16 °C when using the DIRICHLET condition and 50 °C when using the INFILTRATION condition. A DIRICHLET boundary condition is applied at the bottom boundary for both tests to maintain all state variables at the initial conditions for the duration of the test.

For the DIRICHLET test, the top boundary temperature is held fixed over time and no mass enters from the top. For the INFILTRATION and source tests, a specified energy flux of 50 $J/m^2/s$ is applied at the top boundary.

Figure 1 shows the temperature profiles at selected time instants for the DIRICHLET and INFILTRATION conditions. The xFlo, PDEPE, and semi-analytical solutions are essentially indistinguishable for this almost linear system. xFlo calculates the same solution with the source input and INFILTRATION boundary condition.

3.2.2 One Cylindrical or Spherical Dimension

The cylindrical model domain has an inner radius of 1 cm and outer radius of 51 cm, and is filled with the same porous medium as the Cartesian model. The initial temperature ramp again extends over 1 cm adjacent to the inner boundary. For the INFILTRATION and source tests, the specified energy flux is 200 $J/m^2/s$ on the inner boundary.

Figure 2 shows the temperature profiles at selected time instants for the DIRICHLET and INFILTRATION conditions. The PDEPE and xFlo solutions are essentially indistinguishable for this almost linear problem. In cylindrical coordinates, the system responds logarithmically in time, so plotted outputs are spaced logarithmically in time.

The potential in the Laplace equation varies logarithmically in cylindrical and spherical coordinate systems, and a logarithmic grid spacing (the default approach in cylindrical and spherical comparisons) yields better results than spacing the grid linearly in the radial direction. Figure 3 compares the same PDEPE solution for the DIRICHLET boundary condition with an xFlo run using linear spacing. The solutions deviate throughout the domain, and the xFlo solution calculates a smaller energy input to the domain. Using a linear grid with the PDEPE model results in a similar error. Figure 3 also shows the solution for the source input, which is identical to the solution using the INFILTRATION boundary condition.

The spherical model domain mimics the cylindrical system. The initial temperature ramp again extends over 1 cm adjacent to the inner boundary, and the inner and outer boundaries are separated by 50 cm. For the INFILTRATION and source tests, the specified energy flux is 800 $J/m^2/s$ on the inner boundary. Figure 4 demonstrates that the xFlo and PDEPE solutions match very well for the spherical discretization.

Table 1. Parameters used for verification testing in 1-D diffusion simulations			
Description	**Units**	**Bentonite**	**Steel***
van Genuchten alpha	1/Pa	3.5×10^{-8}	NA
van Genuchten lambda (= m)		0.512	NA
Liquid residual saturation		0	NA
Liquid minimum saturation		0	NA
Saturation transition (S*)		0.9995	NA
Permeability	m^2	4.9×10^{-21}	0
Porosity		0.444	0
Dry thermal conductivity	W/m-K	0.105	16.2
Wet thermal conductivity	W/m-K	0.137	16.2
Solid specific heat	J/kg-K	640	500
Solid density	kg/m^3	1,530	8,030
Diffusion coefficient for gas in liquid	m^2/s	0	0
Vapor pressure lowering		Off	Off
Klinkenberg parameter		0	0
Initial saturation		0.22	0
Initial temperature	°C	6	6
*Upstream boundary condition zone for thermal problems See Stothoff and Painter (2016) for xFlo parameter descriptions			

3.3 Isothermal Unsaturated Flow

The unsaturated flow simulations repeat the tests used for the saturated thermal problems, using the same porous media and model dimensions, and using the same strategies for applying initial conditions and boundary conditions.

The PDEPE and semi-analytical models both neglect air flow, thermal diffusion, vapor diffusion, and diffusion of dissolved air. To run the comparisons

- A large Klinkenberg parameter value is specified to make air pressure gradients very small regardless of saturation
- A uniform temperature is specified throughout to set thermal diffusion to zero
- The xFlo option for vapor-phase lowering was turned off to minimize vapor diffusion, and
- The diffusion coefficient for dissolved air in water was set to zero to eliminate dissolved-air diffusion.

3.3.1 Simulation Comparisons

At the start of the 1-D Cartesian simulation, water is introduced at the top of the column. Distance is measured from the top of the column in Figure 5, which shows the liquid saturation profiles at selected time instants for the DIRICHLET and INFILTRATION conditions. xFlo calculates the same solution with the source input and INFILTRATION boundary condition.

Figures 6 and 7 provide the simulation results for the cylindrical and spherical systems, respectively.

In general, the xFlo profiles are slightly smeared relative to the PDEPE and semi-analytical solutions, but capture the profile velocity well. The smearing is characteristic of excess diffusion. This excess diffusion is not noticeable in the thermal simulations.

3.3.2 Analysis of Results

Several possibilities were tested to understand the extra diffusion produced by the xFlo code.

- The PDEPE and xFlo results presented in Figure 5 use a cell size of 2.5 mm. The xFlo results do not change with refinement to 1.25 mm.

- Setting the binary diffusion coefficient in the gas phase to zero resulted in air interference effects near the upstream boundary.

- The PDEPE results are almost identical for vertical and horizontal column orientations.

xFlo is coded to calculate the effective liquid hydraulic conductivity using upstream-weighted liquid mobilities multiplied by distance-weighted harmonic averages of the permeabilities. The approach to estimating the effective liquid hydraulic conductivity affects the flow of water from cell to cell. Upstream weighting is known to improve simulation robustness by adding numerical diffusion, at the cost of degrading simulation accuracy.

Figure 8 shows the profiles for the Cartesian DIRICHLET case with the original approach (top plot, same as Figure 5) and with an alternative approach to defining the effective liquid hydraulic conductivity. The simulation in the top plot uses a distance-weighted harmonic mean for the intrinsic permeability multiplied by the mobility from the upstream cell. In the bottom plot, xFlo is temporarily altered for a special-purpose simulation. In this simulation, the liquid hydraulic conductivity is calculated for each cell and a distance-weighted harmonic mean is used for the effective conductivity. This approach accurately represents effective conductivity. Using upstream weighting results increased diffusion, because the mobility in the downstream cell is never larger than the mobility in the upstream cell, so using the downstream mobility to calculate effective conductivity can only reduce effective conductivity. The increased diffusion has the effect of smearing the front relative to the PDEPE solution.

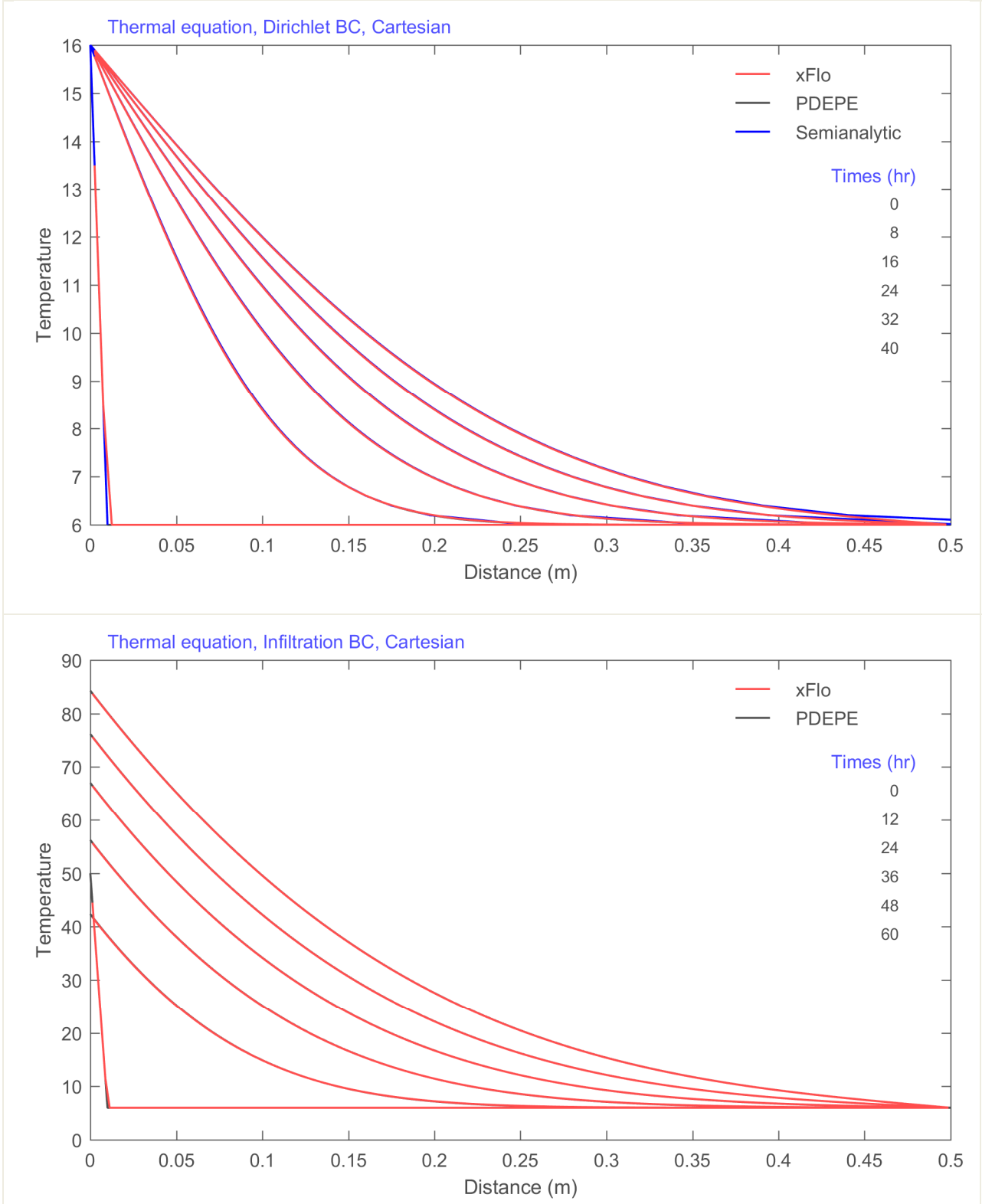

Figure 1. Calculated temperature profiles at specified output times for the Cartesian domain with a (top) DIRICHLET and (bottom) INFILTRATION boundary condition

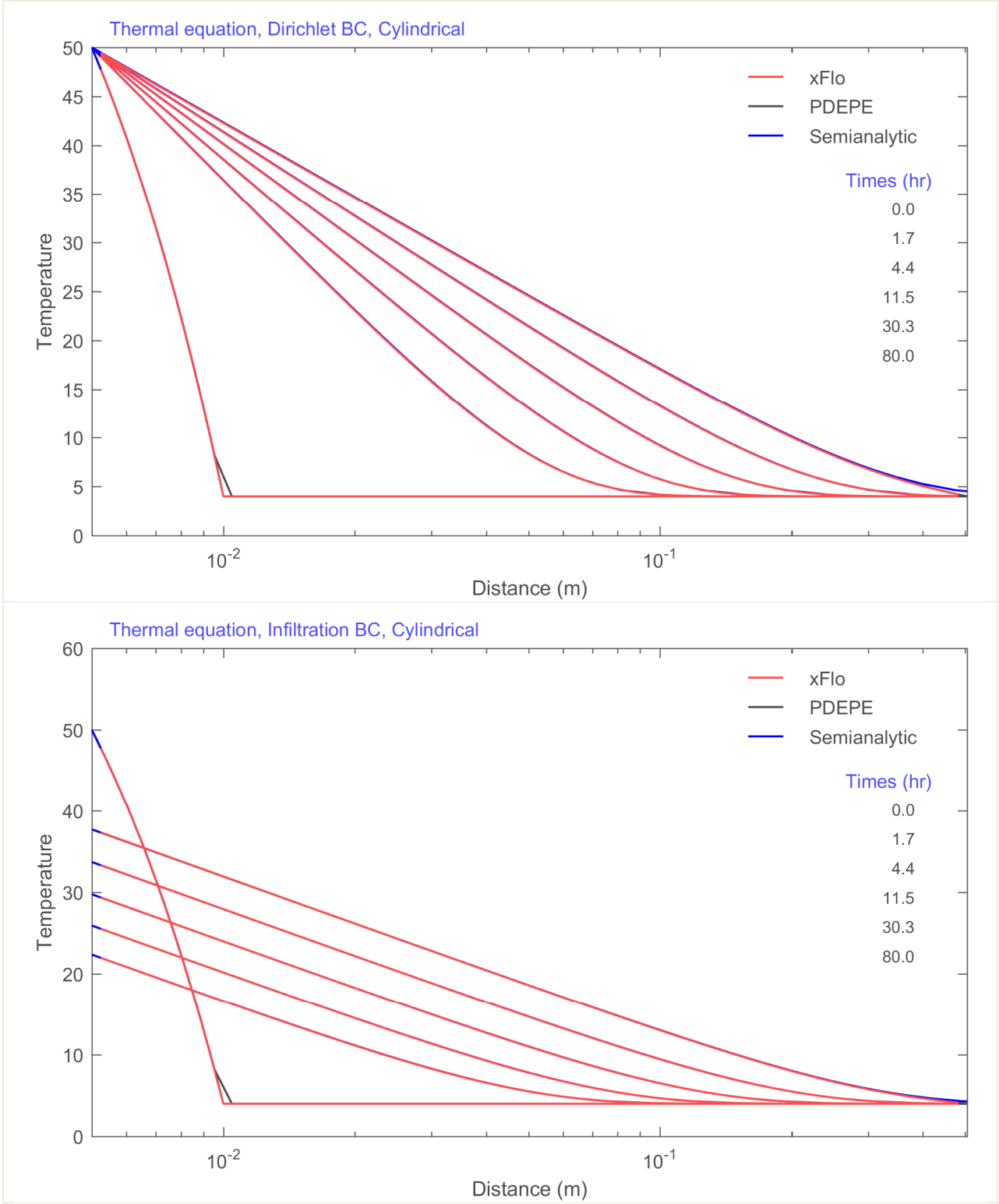

Figure 2. Calculated temperature profiles at specified output times for the cylindrical domain with a (top) DIRICHLET and (bottom) INFILTRATION boundary condition

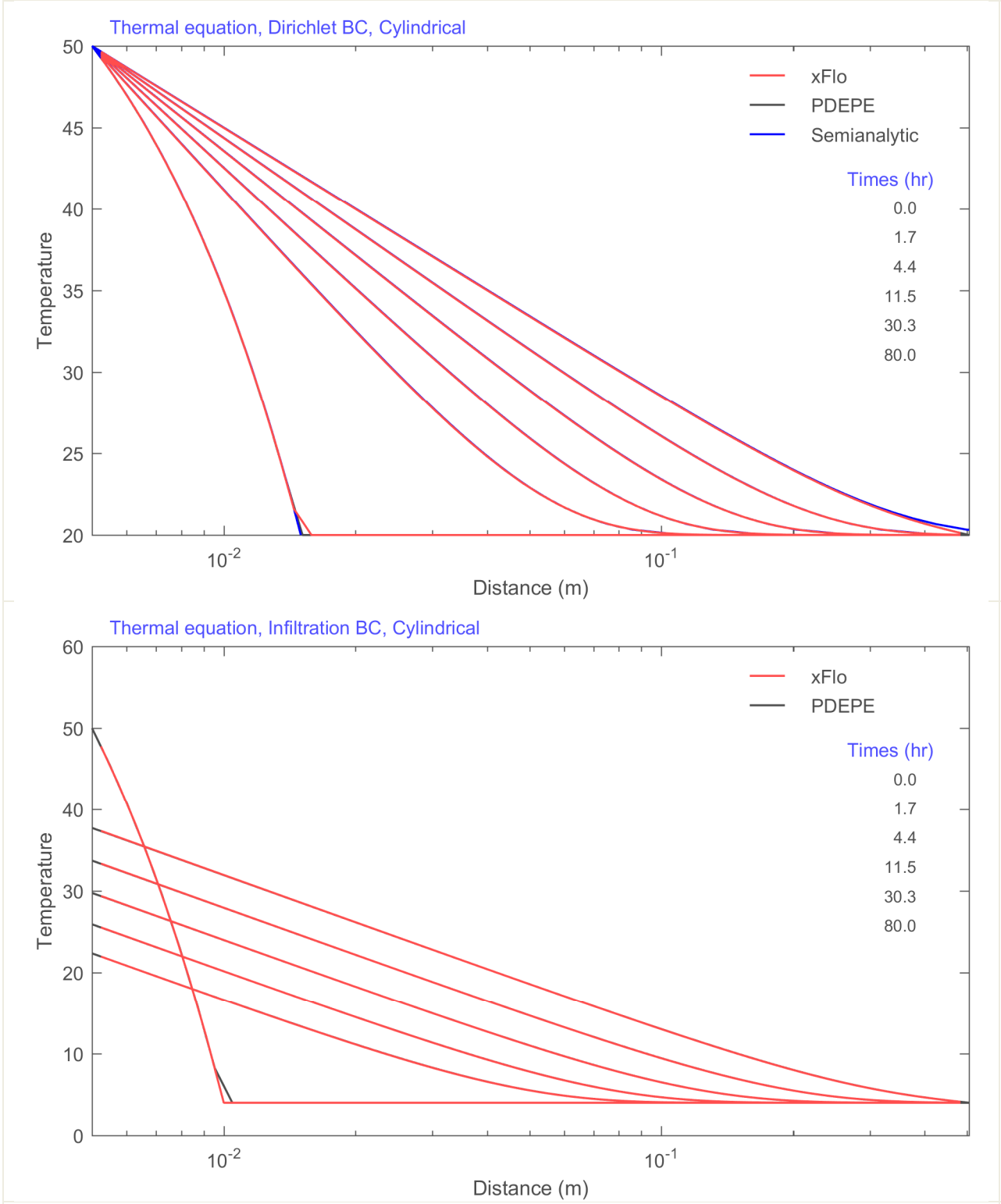

Figure 3. Calculated temperature profiles at specified output times for the cylindrical domain with a (top) DIRICHLET condition and linear grid and (bottom) source representation

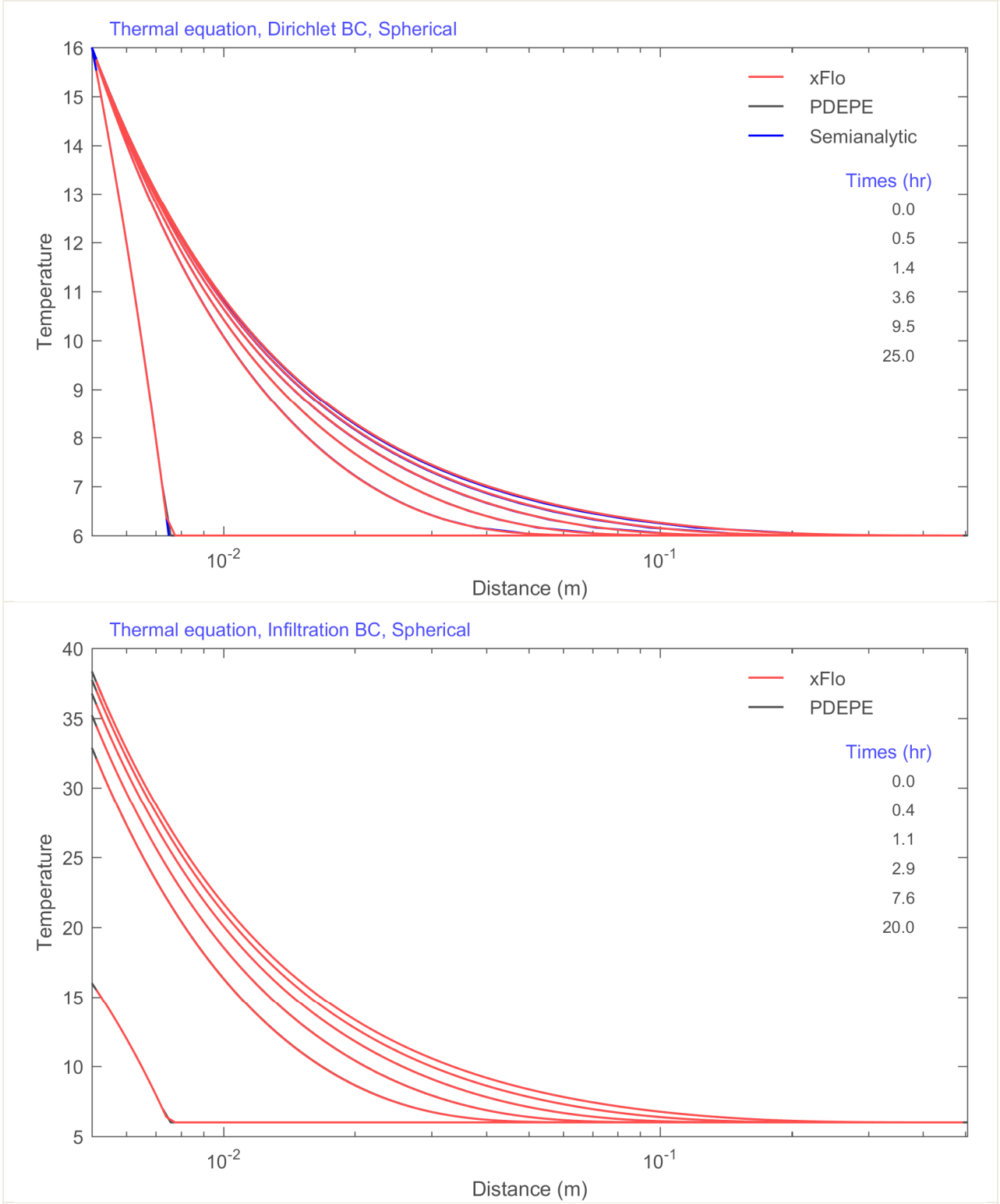

Figure 4. Calculated temperature profiles at specified output times for the spherical domain with a (top) DIRICHLET and (bottom) INFILTRATION boundary condition

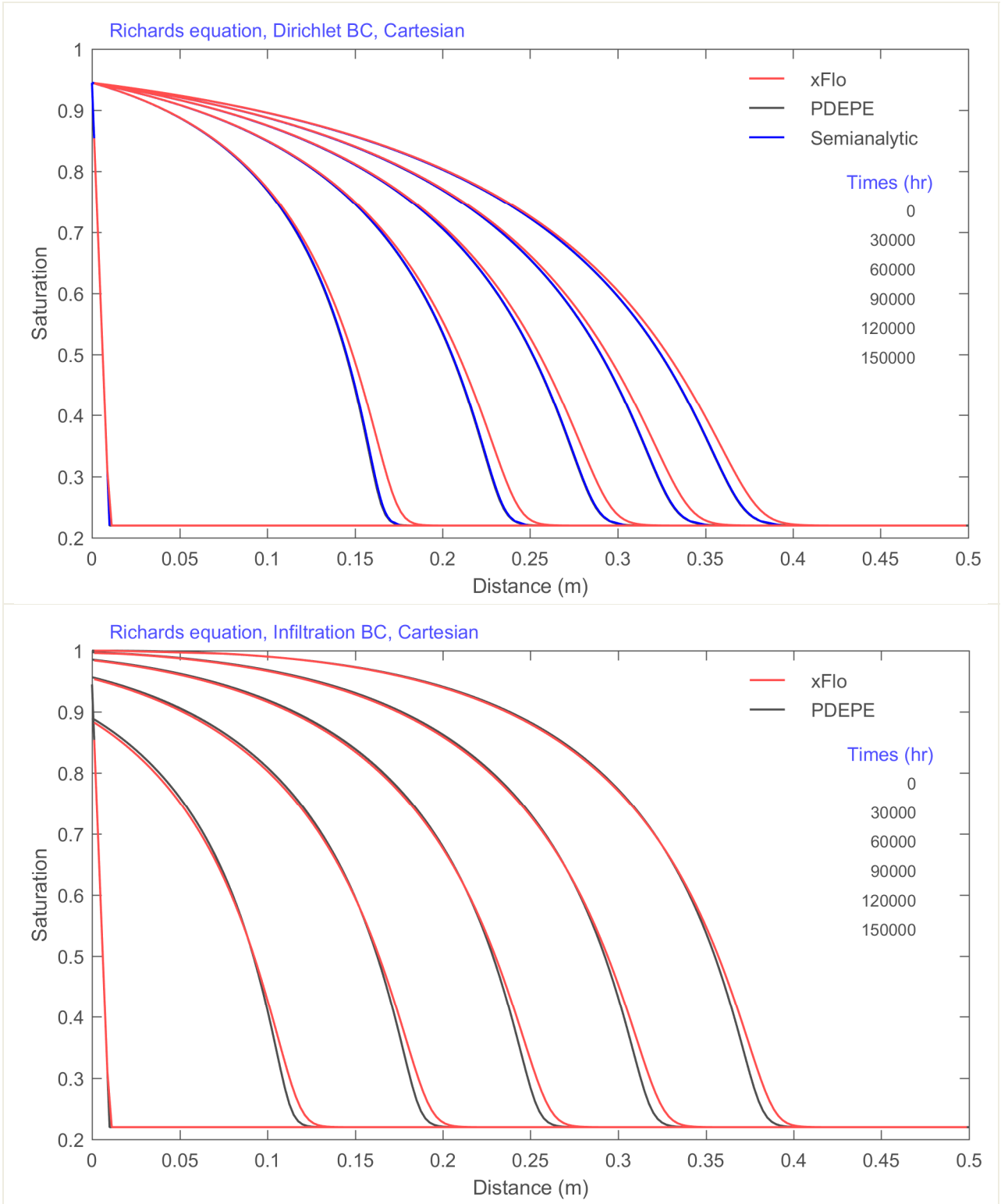

Figure 5. Calculated liquid saturation profiles at specified output times for the Cartesian domain with a (top) DIRICHLET and (bottom) INFILTRATION boundary condition

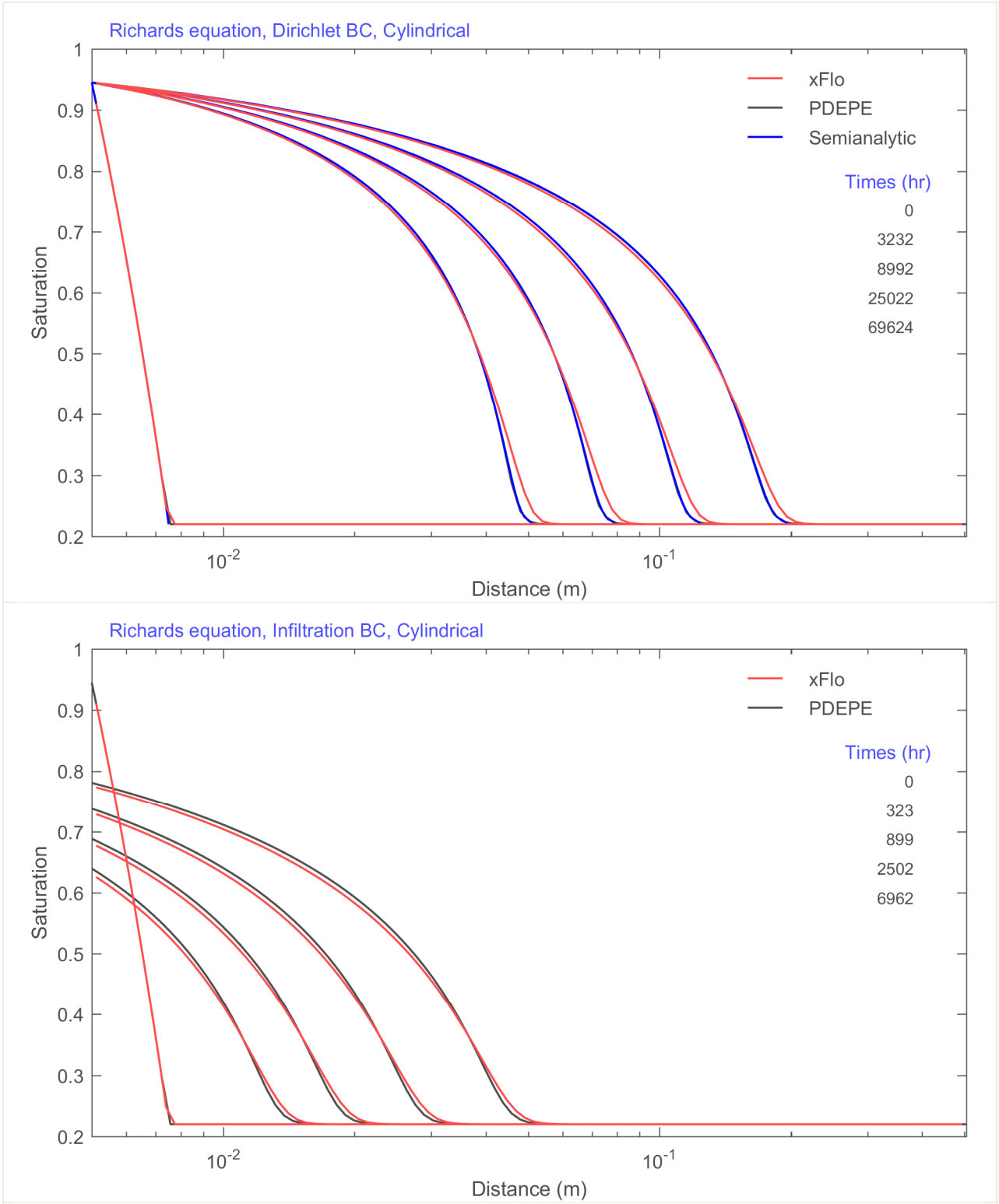

Figure 6. Calculated liquid saturation profiles at specified output times for the cylindrical domain with a (top) DIRICHLET and (bottom) INFILTRATION boundary condition

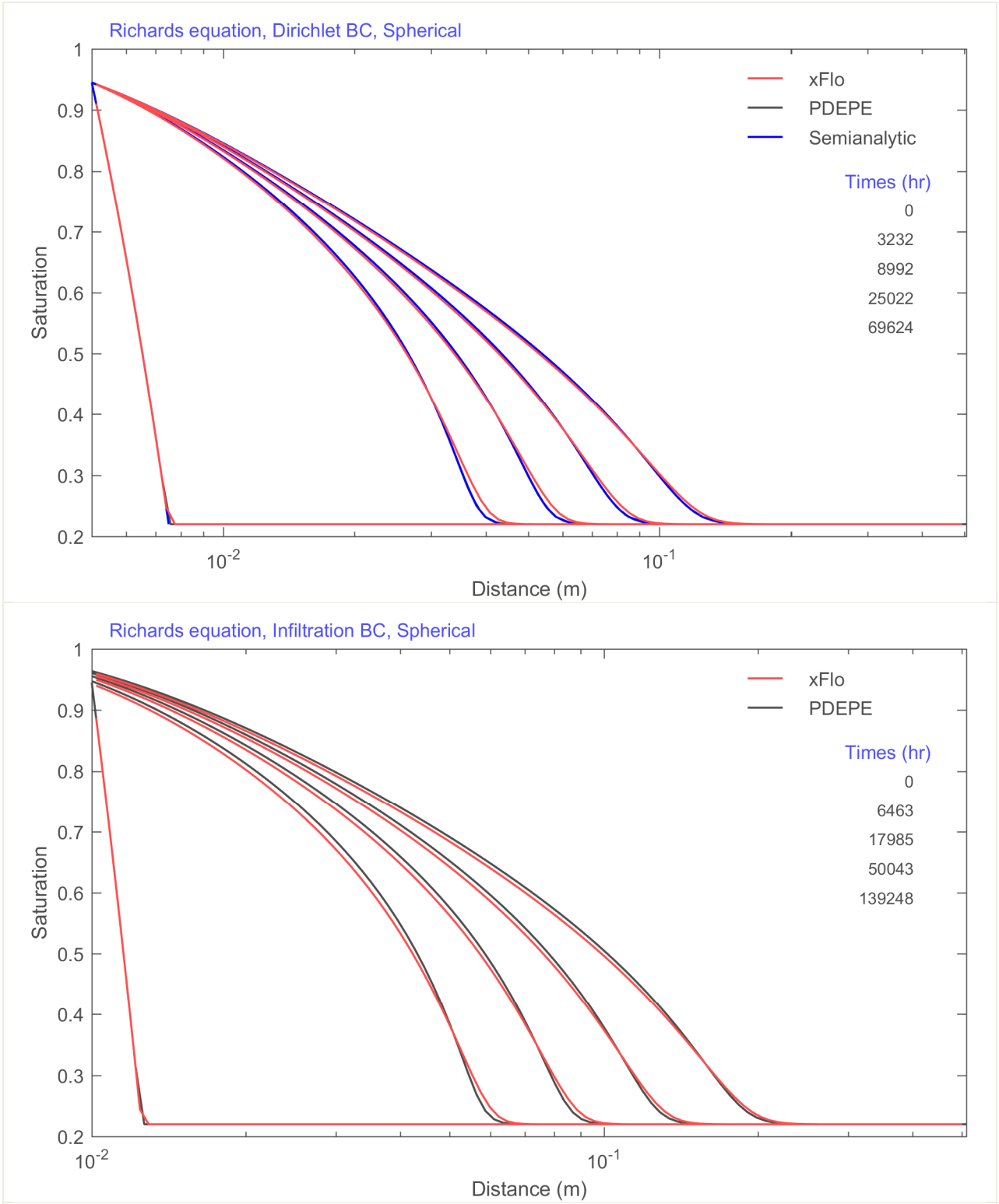

Figure 7. Calculated liquid saturation profiles at specified output times for the spherical domain with a (top) DIRICHLET and (bottom) INFILTRATION boundary condition

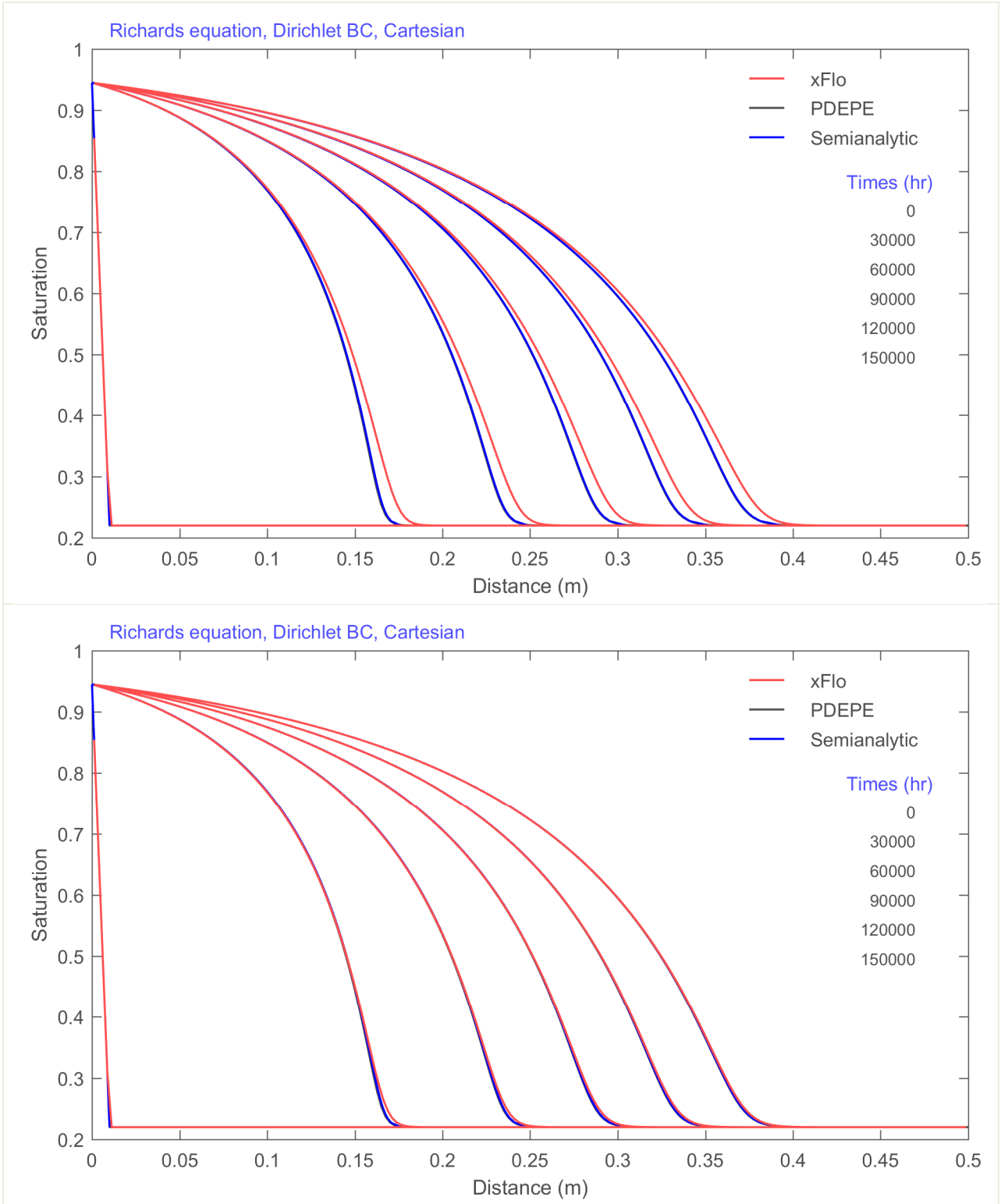

Figure 8. Calculated liquid saturation profiles at specified output times for the Cartesian domain with a DIRICHLET boundary condition and liquid mobility that is (top) upstream-weighted and (bottom) a distance-weighted arithmetic mean

4 TEST CASE 2 SUITE: 1-D DUAL-PERMEABILITY FLOW

The test case 2 suite considers steady isothermal one-dimensional (1-D) dual-permeability single-component mass balance. The test case 2 suite uses 1-D vertical scenarios to verify that (i) dual-permeability coding, (ii) transitions between saturated and unsaturated conditions, and (iii) the active fracture constitutive model are correctly implemented. At steady state, the test suite only considers fluxes within continua and between continua, isolating these components from other model components.

Comparison solutions are developed using the theoretical approach applied in MULTIFLO test case 2 (Painter, 2003, 2005). The theoretical approach is based on solving a system of ordinary differential equations (ODEs) describing both flow and the state variables in both continua, as well as transfer between the continua. Section A2 of Appendix A describes the equations and the solution approach.

The test case 2 suite focuses on liquid fluxes within the fracture and matrix continua, and liquid fluxes between the fracture and matrix continua. The test cases use selected liquid pressure Dirichlet boundary conditions to define scenarios with flow (i) from the fractures to the matrix and (ii) from the matrix to the fractures, allowing evaluation of the transfer terms. The matrix and fractures are assigned the same bottom pressure but different top pressures. The comparison solutions assume that gas pressure is in hydrostatic equilibrium at steady state (the Richards approximation) and do not consider gas fluxes, air or vapor diffusion, or heat fluxes. xFlo performs a full nonisothermal two-phase two-component simulation, but the solution is required to have zero pneumatic and temperature gradients at equilibrium because of the selected Dirichlet boundary conditions.

The solutions used to compare with xFlo are created with special-purpose routines coded in the MATLAB programming language, using the *ode45* ODE solver and *fsolve* root finder that are provided with MATLAB to integrate the equations and find solutions consistent with the boundary conditions. This approach provides highly accurate solutions by adaptive refinement where needed to resolve changing gradients. Note that system complexity is limited, because the solutions require increasingly accurate initial guesses for convergence as the system complexity increases. Some preliminary tests failed because the initial values required precision greater than machine precision.

4.1 Theory and Numerical Approach for Calculating Fluxes

xFlo considers dual-permeability flow primarily through input grid definitions supplemented by constitutive models. In xFlo's approach, each grid cell describes a single continuum and links connect grid cells. Different overlapping continua in the same spatial location are described with different grid cells, and the physics input for each grid cell includes a flag defining which continuum the grid cell represents. The user is responsible for scaling the physical volume of each grid cell by the volume fraction of its continuum. A link between grid cells at two different locations corresponds to lateral flow and a link between grid cells at the same location corresponds to local transfer flow between continua (e.g., from matrix to fracture). The routines calculating fluxes over a link are responsible for using the appropriate constitutive model for flow within a continuum and between continua. In the usual dual-permeability approach, lateral flow links only occur between cells in the same continuum. In xFlo, it is possible to define lateral flow links between different continua, which might be useful for describing local matrix redistribution due to strata-bound fractures at layer interfaces.

To calculate fluid flow for a link, xFlo requires the interfacial area, the distance from each grid cell centroid to the interface, rock properties, and fluid properties. The model uses the same calculational approach for within-matrix, within-fracture, and matrix-to-fracture flow, using the distance-weighted harmonic mean of intrinsic permeability and the fully upstream-weighted fluid mobility (relative permeability divided by viscosity). The default model uses the same calculational approach for fracture-to-matrix flow. An additional constitutive model, the active fracture model, is also available, which reduces the fracture-to-matrix flow.

xFlo generically defines liquid water fluxes between grid cells 1 and 2 in the form

$$q = KA_I \left[\frac{P_2 - P_1 + \rho g (Z_2 - Z_1)}{d_1 + d_2} \right] \quad (1)$$

where q is the transfer flux, K is the distance-weighted conductivity, A_I is the interfacial area per unit volume, d is the distance between the cell centroid and the interface, P is pressure, ρ is density, g is gravity, and Z is elevation. The A_I and d parameters are provided as input parameters for each link. Similar expressions are used for all diffusive fluxes (pressure-driven gas fluxes, thermal diffusion, vapor diffusion, diffusion of dissolved air in liquid).

Internally, xFlo calculates K/d as a rock component and a fluid component in the form

$$\frac{K}{d} = \frac{k_1 k_2}{d_2 k_1 + d_1 k_2} \lambda_u \quad (2)$$

where k_i and d_i are intrinsic permeability and distance to the cell-cell interface for cell i, and λ_u is the upstream-weighted mobility. The upstream-weighted mobility for a link is the mobility of the fluid in the cell losing water (upstream cell). Comparable expressions are used for all diffusive fluxes, with a harmonic average used for rock properties and upstream weighting used for fluid properties. Upstream weighting creates an asymmetry in the transfer function, because the mobility is typically quite different in the matrix and fractures.

In a multi-continuum formulation, bulk fluxes (i.e., at the macroscopic scale) are equal to the intrinsic fluxes (i.e., at the continuum scale) multiplied by the continuum volume fraction ϵ. xFlo does not explicitly consider ϵ. The user is responsible for ensuring that the A_I and d parameters for each link and the grid cell volume V_c for each cell is consistent with the corresponding ϵ.

MULTIFLO and xFlo use essentially the same gridding scheme. In the internal MULTIFLO preprocessor, each grid cell volume is multiplied by the corresponding ϵ, intrinsic flux parameters are maintained, and each link between cells in the same continuum uses an adjusted A_I that is multiplied by the average ϵ for the two cells. The internal xFlo preprocessor uses the same strategy, except that the fracture aperture thickness is assumed zero when calculating transfer between matrix and fracture.

Instead of modifying the interface area, bulk diffusive liquid fluxes can be calculated by multiplying the intrinsic permeability by ϵ, yielding

$$\frac{K_b}{d} = \frac{\epsilon_1 k_1 \epsilon_2 k_2}{d_2 \epsilon_1 k_1 + d_1 \epsilon_2 k_2} \lambda_u = \frac{k_1 k_2}{\frac{d_2}{\epsilon_2} k_1 + \frac{d_1}{\epsilon_1} k_2} \lambda_u \quad (3)$$

Dividing d by ϵ instead of multiplying A_I by ϵ provides a consistent formulation across all diffusive fluxes that automatically accounts for different cell dimensions. Because each advective flux term is simply a multiple of a diffusive flux term, this approach is also consistent for advective fluxes.

The liquid water transfer term between matrix and fracture can be written in the form

$$q = KA_{fm}\left(\frac{P_2 - P_1}{d}\right)V_c \quad (4)$$

where q is the transfer flux, K is the distance-weighted conductivity, A_{fm} is the interfacial area per unit volume, d is the distance between matrix and fracture centroids, $P_2 - P_1$ is the pressure difference between matrix and fracture, and V_c is the cell volume. The user is responsible for providing values for A_I and d for each matrix/fracture link corresponding to the A_{fm} and d parameters.

The transfer term between matrix and fracture is typically developed based on the size of the matrix blocks. For a block with dimensions L_x, L_y, and L_z, and fracture apertures much smaller than the block dimensions, the interfacial area per unit volume is

$$A_{fm} \approx 2\left(\frac{L_xL_y + L_xL_z + L_yL_z}{L_xL_yL_z}\right) = 2\left(\frac{1}{L_x} + \frac{1}{L_y} + \frac{1}{L_z}\right) \quad (5)$$

When $L_x = L_y = L_z = L$, $A_{fm} \approx 6/L$. The MULTIFLO preprocessor assumes that $L_z = \infty$ for 2-D problems and $L_y = L_z = \infty$ for 1-D problems.

The effective distance between matrix and fracture is half the effective block dimension, or

$$d \approx \frac{2}{3}\left(\frac{1}{L_x} + \frac{1}{L_y} + \frac{1}{L_z}\right)^{-1} \quad (6)$$

For a cube, $A_{fm}/d \approx 3/L^2$. Note that the presence of low-permeability coatings may further reduce the effective conductivity. Following the MULTIFLO approach, the MATLAB-based model and the routines preparing the input for xFlo multiplied A_{fm} by an area reduction factor, A_r. The area reduction factor provides a simple scaling factor for testing.

The active fracture model (Liu et al., 1998) recognizes that some fractures may carry most of the flow (active fractures) while others may not contribute significantly to flow. In the active fracture model, the active fractures are more saturated and have a higher relative permeability than the average. Liu et al. (1998) define a constitutive model that defines the fraction of active fractures using the effective fracture saturation in the form

$$f_a = S_{ef}^{\gamma} \quad (7)$$

where f_a is the fraction of fractures that are active, S_{ef} is the effective saturation of the fracture continuum, and γ is the active fracture parameter ($0 \leq \gamma < 1$). Liu et al. (1998) use scaling arguments to modify the van Genuchten capillary pressure and relative permeability relationships for the fracture continuum, and derive an alternative relationship for the transfer of fluid between matrix and fractures. Setting $\gamma = 0$ recovers the standard van Genuchten relationships. The transfer relationship is geometrically asymmetric under unsaturated

conditions, in the sense that flow from the matrix may enter all fractures but flow from the fractures only occurs for the more widely spaced active fractures. The active fracture spacing is determined using $1/f_a$. When the active fracture model is activated, the current xFlo logic applies the active fracture model to any link between two different continua.

4.2 Single-Layer System

A series of tests were run for a representative domain consisting of a homogeneous matrix and fracture. The test cases varied (i) the boundary conditions and (ii) the parameters for the transfer model. The particular test cases were selected to illustrate a range of transfer behavior, with the capabilities of the comparison model also constraining the parameter set. Most analytical solutions require that the material properties are homogeneous, but the ODE-based approach provided an opportunity to test that Equation (3) was consistent with the analytical solution by linearly increasing the fracture volume fraction by a factor of ten from top to bottom of the domain.

As discussed in Section 4.1, the modeled transfer of water between matrix and fracture is asymmetric, in part due to upstream weighting of fluid properties in the code. Even without upstream weighting, the active-fracture model inherently features asymmetric flow behavior between continua. To test that the transfer terms (with and without the active fracture conceptualization) are correctly implemented, two boundary condition cases are considered for the physical system. Both boundary conditions create vertically downward flow, but one boundary condition set creates flow predominantly from the fracture to the matrix and the other creates flow predominantly from the matrix to the fracture.

The two boundary condition cases are mirror images. In both cases, the bottom boundary condition sets liquid head to −2 m at the bottom of the domain for both the matrix and fractures, creating an unsaturated condition. The top boundary condition sets liquid head to −1.25 m in one continuum and −2.75 m in the other continuum (symmetric differences from the bottom condition), with the two cases flipping the continuum assignment. The wet matrix case sets liquid head to −1.25 m in the matrix, and the wet fracture case sets liquid head to −1.25 m in the fractures. The test properties are provided in Table 2.

Figure 9 displays the wetting saturation and capillary pressure with the active fracture γ set to 0.6 and transfer flow from matrix to fracture. Three values of A_r are used, 0.001, 0.01, and 0.1, representing increasingly strong connections. Fracture capillary pressure decreases and fracture saturation increases as A_r is increased. As indicated by the legend, solid lines indicate the xFlo solution and the darker dashed lines indicate the corresponding MATLAB-based solution. The xFlo solution follows the trend of the MATLAB-based solution quite well, but the sharp gradient at the top boundary may require finer gridding for xFlo to completely resolve.

In order to match the xFlo and MATLAB solutions, the matrix/fracture interface area in the xFlo input data was artificially increased by a constant factor. The constant adjustment factor for the matrix/fracture interface area was set by trial and error to $2/M_{\mathrm{H2O}}$ (~111 mol/kg), where M_{H2O} is the molar weight of water. The adjustment factor was consistently needed across all simulations, suggesting that there may be an internal inconsistency between molar fluxes and mass fluxes in the definition of the transfer term between matrix and fractures. This inconsistency does not appear in matrix/matrix fluxes or fracture/fracture fluxes. Further work will be needed to identify the source of the inconsistency.

Table 2. Parameters used for verification testing in 1-D single-layer dual-permeability simulations			
Description	**Units**	**Matrix**	**Fracture**
van Genuchten alpha	1/Pa	8.77×10^{-5}	1×10^{-4}
van Genuchten lambda (= m)		0.231	0.67
Liquid residual saturation		0	0
Liquid minimum saturation		0	0
Saturation transition (S*)		0.99	0.99
Permeability	m^2	1.86×10^{-13}	1×10^{-11}
Porosity		0.42	1
Volume fraction		1 - fracture	0.01 to 0.001
Dry thermal conductivity	W/m-K	1	1
Wet thermal conductivity	W/m-K	1	1
Solid specific heat	J/kg-K	1,000	1,000
Solid density	kg/m^3	2,500	2,500
Diffusion coefficient for gas in liquid	m^2/s	0	0
Vapor pressure lowering		Off	Off
Klinkenberg parameter		0	0
Initial temperature	°C	15	15
Block dimension	m	5	

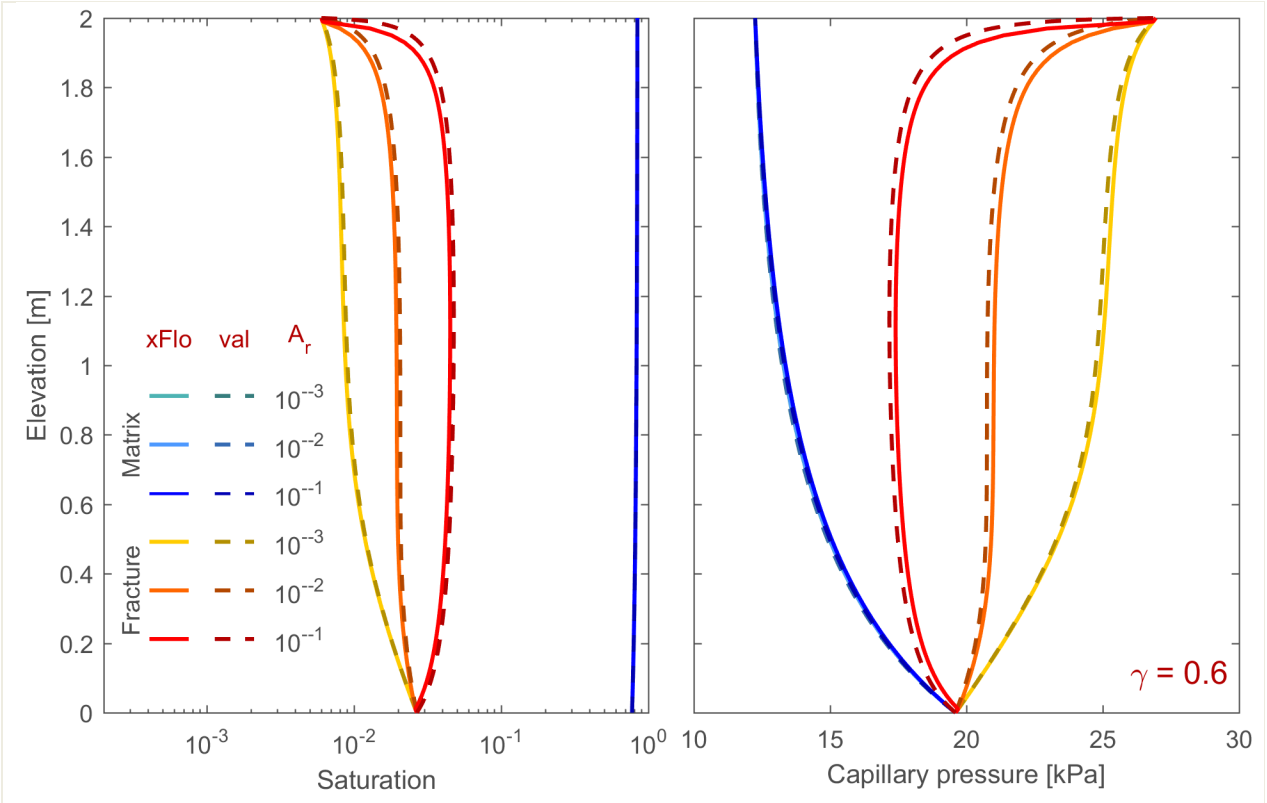

Figure 9. Comparison between xFlo and MATLAB-based validation solutions for the wet matrix case (transfer flow from matrix to fracture) with the active fracture γ set to 0.6 for simulations with A_r set to 0.001, 0.01, and 0.1. (Left) wetting saturation and (right) capillary pressure.

Figure 10 displays the wetting saturation and capillary pressure for combinations of three active fracture parameters (0, 0.3, and 0.6), three area reduction factors (0.001, 0.01, and 0.1), and two boundary condition pairs (wet matrix and wet fracture). Figure 10 uses the same line color scheme as Figure 9. The xFlo and validation solutions exhibit the same behavior with changing parameters, suggesting that the coding of the active fracture model in xFlo is generally consistent with the independently derived MATLAB model. Simulations with a spatially constant fracture volume fraction are similarly consistent. Furthermore, the use of Equation (3) to develop xFlo input also produces results consistent with the MATLAB-based model.

All xFlo wet-fracture simulations with area reduction factor set to 0.1 have slightly excessive fluid transfer from fracture to matrix (see Figure 10). This appears to be related to the inconsistency in the matrix/fracture interface area rather than the active fracture model, because the same behavior occurs regardless of the active fracture parameter.

4.3 Three-Layer System

The three-layer system considers vertical flow over two transitions between rock types with dissimilar matrix and fracture properties. The system is based on a welded tuff/nonwelded tuff/welded tuff sequence with highly contrasting layer properties. This is an idealized system; in actuality, physical transitions are typically more gradual and physical layers are typically dipping, which may induce lateral flow and funneling into local permeable features. The default matrix/fracture transition model is used for this system (the active fracture model is not used). The idealized three-layer system is intended to verify that (i) the symmetric default dual-permeability model is properly coded and (ii) the transition between partial and full saturation is properly coded.

All validation exercises for MULTIFLO (Painter, 2003, 2005) considered a single property set for the entire domain, and the three-layer system problem was created for xFlo validation. It is expected that xFlo and MULTIFLO would calculate similar system behavior and face similar numerical challenges for the same property sets, because they use similar algorithms.

Verification that the transition between partial and full saturation is correctly handled is of particular interest. In xFlo's approach, the state variables are gas saturation, gas pressure, and temperature under two-phase conditions and mole fraction of gas in liquid, liquid pressure, and temperature under fully saturated conditions. xFlo uses a variable-switching algorithm to transition between the two conditions, which may cause inconsistencies in calculated state variables at the transition. Furthermore, it is notoriously difficult to determine the position of a zero-capillary-pressure interface with some numerical models, because the change in water mass with respect to change in liquid pressure is very different for saturated and partially saturated conditions. A strongly discontinuous response to pressure change poses a difficult challenge while converging during time steps.

The flow rate in the system was selected to be large enough to develop a small perched zone in the nonwelded matrix above the transition between the nonwelded and lower welded tuff, even though the top and bottom boundaries are partially saturated. The flow rate is larger than either welded matrix can carry under gravity drainage, even fully saturated, but the nonwelded matrix and the fracture continua are each easily capable of carrying the flow under gravity drainage with partial saturation. The welded and nonwelded units have highly contrasting properties. Table 3 provides the test problem properties.

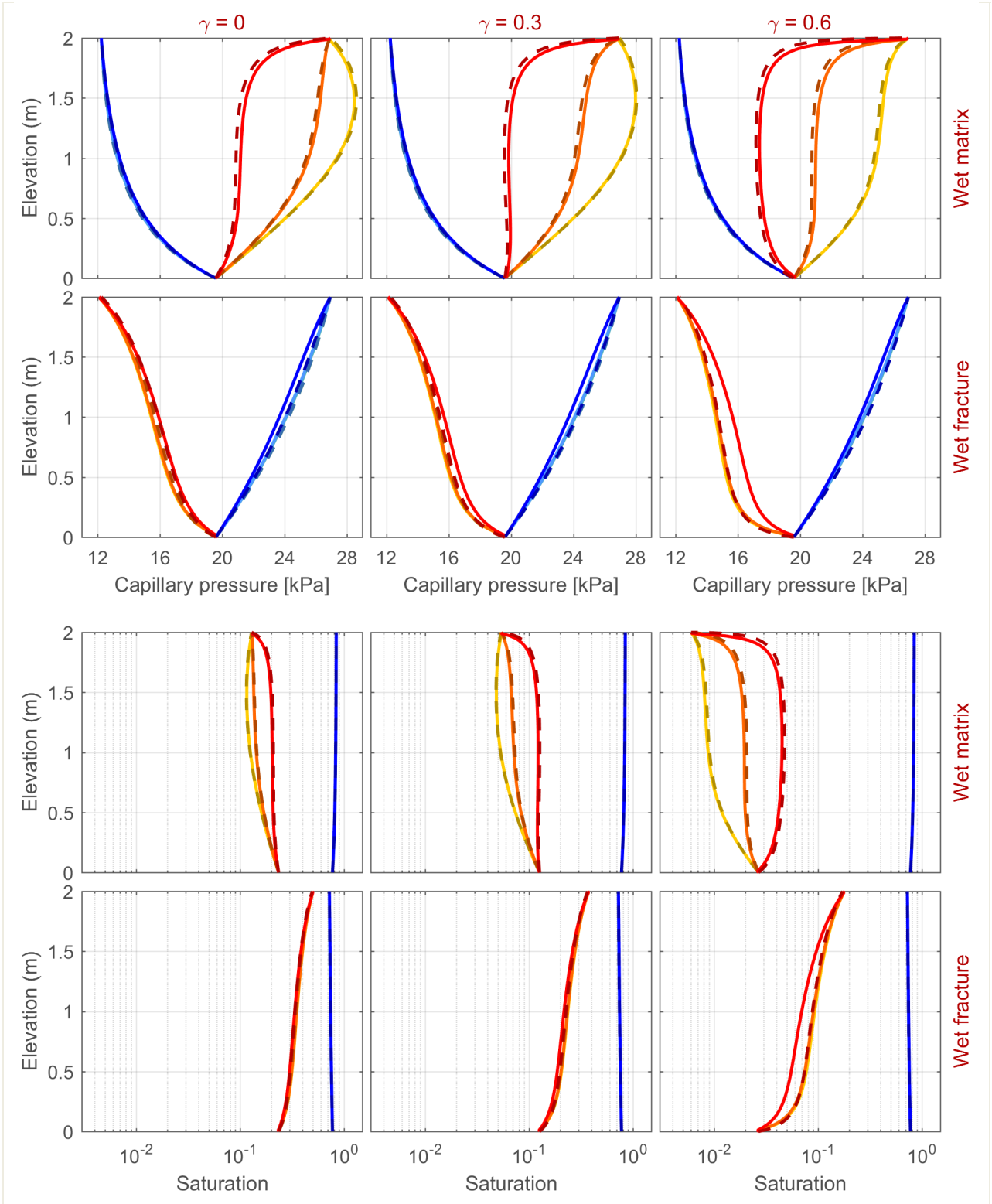

Figure 10. Matrix and fracture capillary pressure (top rows) and wetting saturation (bottom rows) for the active fracture γ set to 0, 0.3, and 0.6 (columns) for wet matrix and wet fracture conditions. The color scheme follows Figure 9.

Table 3. Parameters used for verification testing in 1-D three-layer dual-permeability simulations

Description	Units	Matrix 1	Fracture 1	Matrix 2	Fracture 2	Matrix 3	Fracture 3
van Genuchten alpha	1/Pa	2.45×10^{-7}	1×10^{-4}	8.77×10^{-5}	1×10^{-3}	2.45×10^{-7}	1×10^{-4}
van Genuchten lambda (= m)		0.451	0.67	0.231	0.67	0.451	0.67
Liquid residual saturation		0	0	0	0	0	0
Liquid minimum saturation		0	0	0	0	0	0
Saturation transition (S*)		0.99	0.99	0.99	0.99	0.99	0.99
Permeability	m^2	4.53×10^{-20}	1×10^{-11}	1.86×10^{-13}	1×10^{-13}	4.53×10^{-19}	1×10^{-11}
Porosity		0.072	1	0.42	1	0.072	1
Volume fraction		0.999	0.001	0.999	0.001	0.999	0.001
Dry thermal conductivity	W/m-K	1	1	1	1	1	1
Wet thermal conductivity	W/m-K	1	1	1	1	1	1
Solid specific heat	J/kg-K	1,000	1,000	1,000	1,000	1,000	1,000
Solid density	kg/m^3	2,500	2,500	2,500	2,500	2,500	2,500
Diffusion coefficient for gas in liquid	m^2/s	0	0	0	0	0	0
Vapor pressure lowering		Off	Off	Off	Off	Off	Off
Klinkenberg parameter		0	0	0	0	0	0
Initial temperature	°C	15	15	15	15	15	15
Block dimension	m	0.1		10		0.1	
Area reduction factor		1×10^{-3}		1×10^{-4}		1×10^{-3}	
See Stothoff and Painter (2016) for xFlo parameter descriptions							

The MATLAB-based ODE routines were able to successfully calculate comparison solutions using the intrinsic MATLAB ODE45 solver, although the shooting algorithm required extremely accurate initial conditions in order to negotiate the strong nonlinearities that develop near the interfaces. The algorithm was not able to converge when the top and bottom boundary conditions are too decoupled, such as when the overall domain length or A_r parameter were too large.

The saturations at the top and bottom of each layer that were calculated by the shooting algorithm were used to derive initial and boundary conditions for xFlo. In both the matrix and fracture, the top and bottom saturation for each layer was assumed to be linearly changing. The actual profiles were more complex, so the xFlo simulations needed to step through time. The same top and bottom saturation values were used for boundary conditions, anticipating that the profiles would evolve to the consistent fluxes. A simulation successfully completing 100 years was still far from equilibrium.

No xFlo simulation achieved an equilibrium state, often crashing after hundreds or thousands of time steps without trapping or identifying an error. The last simulation outputs for the failed runs typically showed that perched conditions had developed above one or both layer interfaces, even for simulations based on fluxes too small to develop perching at steady state. Transient perched zones may develop due to initial conditions that are out of equilibrium. Previous experience has shown that xFlo can have difficulty with the transition between partial and full

saturation, in part due to inconsistencies arising during variable switching, and it is likely that some of the computational difficulties are due to the formation of a perched zone. The perched zones may be due to xFlo underestimating liquid flux across interfaces between media with strongly contrasting permeabilities and retention properties, even though xFlo uses fully upstream weighting for mobility.

Graded material properties are correctly handled in the mass balance equation and strongly contrasting properties are correctly handled in the energy balance equation (which has only weak state-dependence in the model parameters) and in the mass balance equations under fully saturated conditions.

The MATLAB-based solutions showed that strong gradients can develop near interfaces due to the nonlinear state-dependent constitutive relationships. By implication, underestimated liquid fluxes may be related to poor resolution of extremely steep gradients developing in the state-dependent model parameters. Local grid refinement near interfaces was not successful in preventing premature simulation terminations.

Additional work is necessary to understand what conditions lead to an incorrect representation of liquid fluxes across interfaces. The particular test case was selected to be rather extreme, and prior work considering interfaces with less extreme contrasts suggests that liquid fluxes are appropriately handled or overestimated.

Additional work is necessary to understand why xFlo unexpectedly terminates for some of the simulations testing layered systems. Prior work suggests that simulations typically grind to a halt with continually decreasing time steps when convergence is an issue, and it is unusual for xFlo to simply halt with no error message. This unexpected termination is likely due to xFlo failing to trap an out-of-bound condition for a constitutive equation, perhaps related to variable switching in grid cells transitioning between partial and full saturation.

5 TEST CASE 3: 2-D HEAT FLUX IN A HETEROGENEOUS CYLINDER

Test case 3 considers transient two-dimensional (2-D) single-continuum heat fluxes. The test case suite verifies that xFlo correctly implements diffusion-dominated balance equations in cylindrical coordinates and appropriately represents spatial variation in material properties. The test suite considers energy transport from a heater to a cold sink in a test column under partially saturated conditions. Test case 3 uses a commercially available simulator, COMSOL (see Section A3 of Appendix A), to provide a comparison steady-state thermal diffusion solution. xFlo solves the equations for transient nonisothermal two-phase two-component flow and transport, but the assigned material properties require that mass fluxes are zero in the solid components of the cylinder and are very small in the porous medium filling the cylinder.

The test problem is based on a heated column test considered during previous modeling activities (Stothoff et al., 2015). The test problem considers a Teflon® cylinder filled with bentonite pellets (Figure 11). The Teflon cylinder is wrapped with steel restraining bands and the cylinder and bands are surrounded by a layer of insulation. The steel bands are placed at different distances along the column, with Teflon-filled gaps between the bands. The bottom plug for the column is a steel heater, and the top plug is a porous plate (represented as Teflon for the test case). This problem is simplified from the actual experiment by removing a layer of foam and neglecting heat loss due to radiation.

The column is initially at the ambient temperature of 22 °C. The column is heated from the bottom, and is held at the ambient temperature at the top and outside the insulation. The heater is represented by a fixed temperature along most of the bottom of the steel plug.

The different material types provide a strong contrast in thermal conductivity values (Table 4). Steel is orders of magnitude more thermally conductive than the bentonite and Teflon, which are about an order of magnitude more conductive than the insulation. The same bentonite thermal conductivity is used for wet and dry conditions to ensure that the COMSOL and xFlo models used the same thermal properties.

The results presented in Figure 12 are 500 hr after setting the heater temperature to 140 °C, and the temperature fields have achieved steady state. The xFlo and COMSOL models agree very well for this thermal problem, suggesting that xFlo properly represents (i) 2-D cylindrical coordinates and (ii) spatially variable material properties.

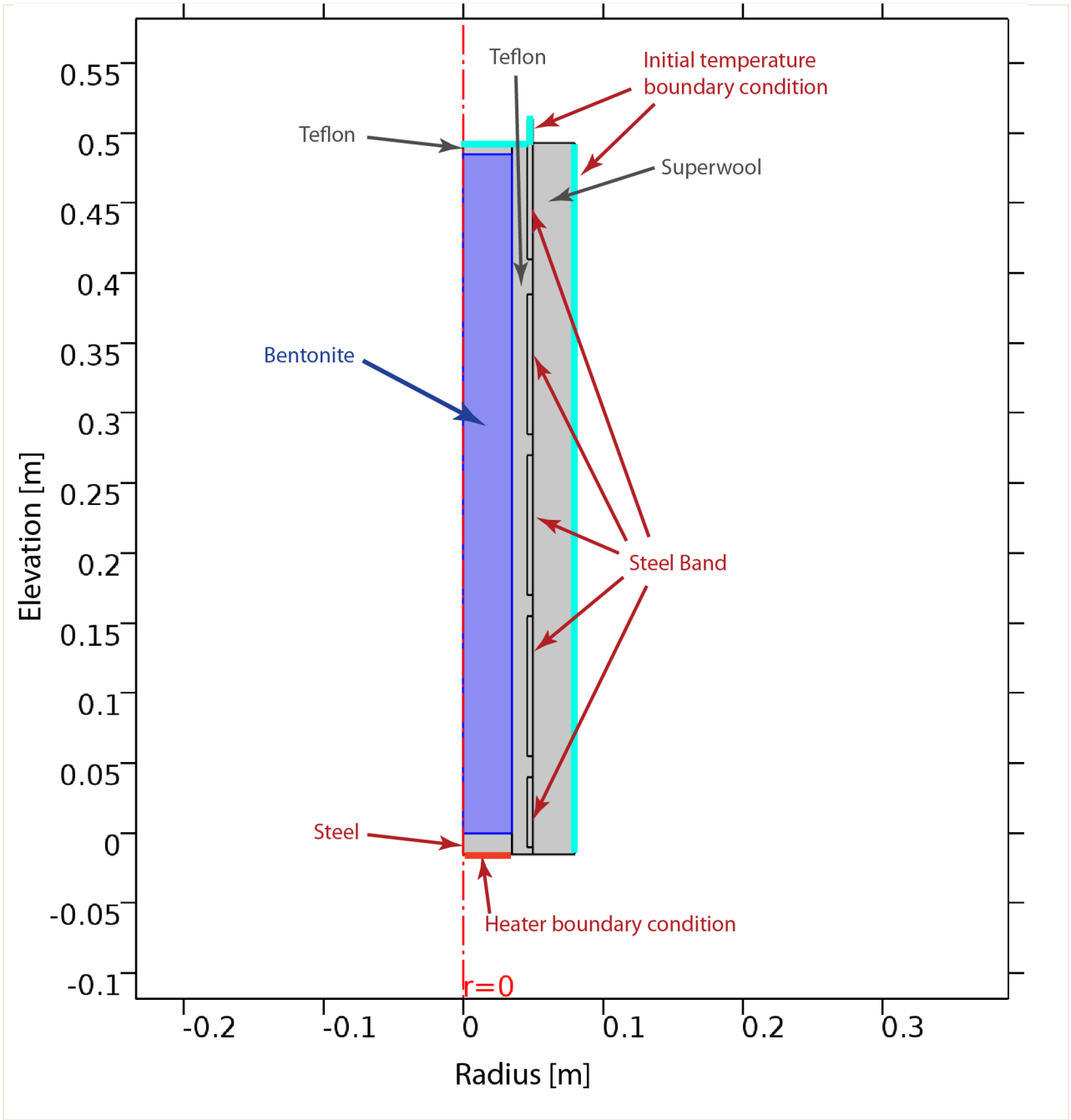

Figure 11. Model domain, parameter zones, and boundary conditions

Table 4. Parameters used for verification testing in 2-D cylindrical thermal models					
Description	**Units**	**Bentonite**	**Steel**	**Teflon**	**Superwool**
van Genuchten alpha	1/Pa	3.5×10^{-8}	NA	NA	NA
van Genuchten lambda (= m)		0.512	NA	NA	NA
Liquid residual saturation		0	NA	NA	NA
Liquid minimum saturation		0	NA	NA	NA
Matrix permeability	m^2	4.9×10^{-21}	0	0	0
Matrix porosity		0.444	0	0	0
Dry thermal conductivity	W/m-K	0.3	16.2	0.25	0.04
Wet thermal conductivity	W/m-K	0.3	16.2	0.25	0.04
Solid specific heat	J/kg-K	640	500	1,300	733
Solid density	kg/m^3	1,530	8,030	2,160	128
Diffusion coefficient for gas in liquid	m^2/s	0	0	0	0
Vapor pressure lowering		Off	Off	Off	Off
Klinkenberg parameter		0	0	0	0
Initial saturation		0.22	NA	NA	NA
Initial temperature	°C	22	22	22	22
See Stothoff and Painter (2016) for xFlo parameter descriptions					

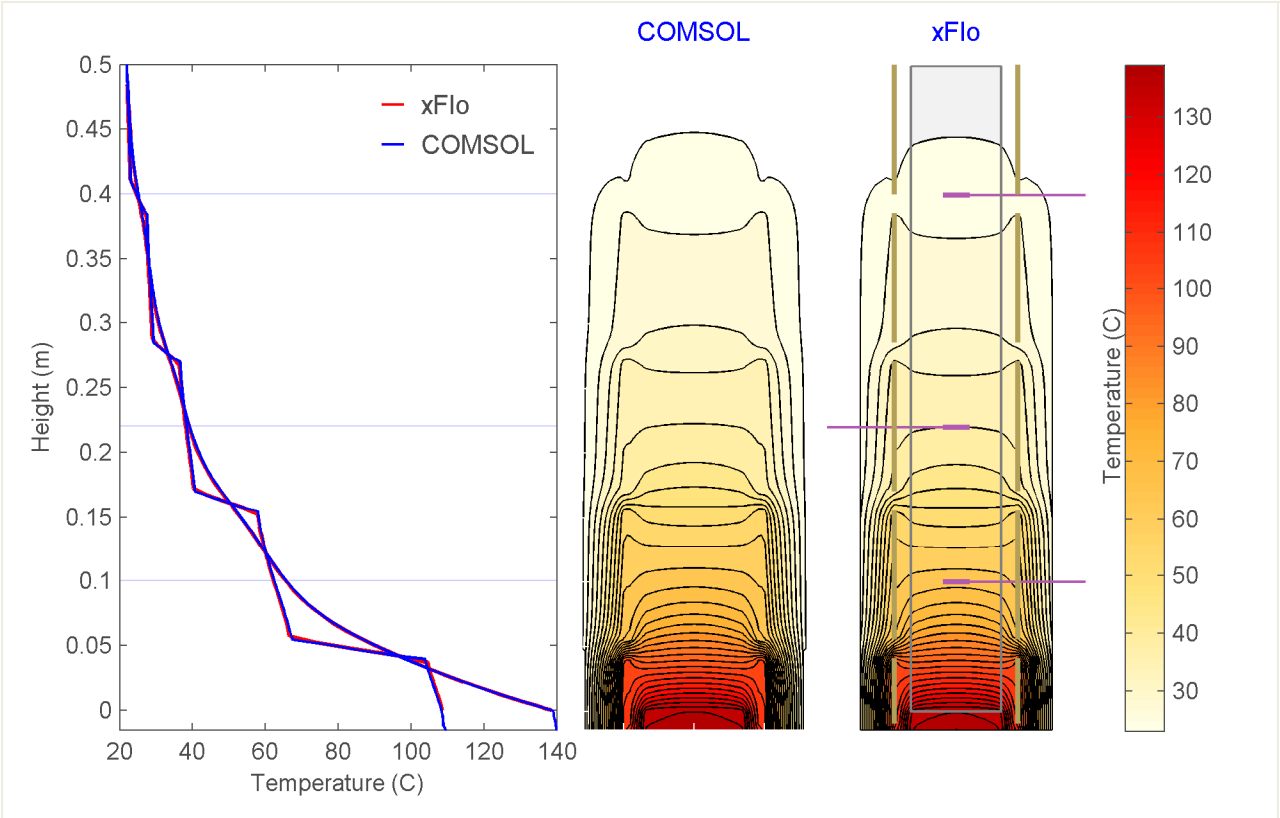

Figure 12. Calculated column-test temperatures after 500 hr. (Left) Vertical profiles along the centerline (smoothly varying curves) and steel bands (stair-step curves) for xFlo and COMSOL. (Right) Temperature contours (4 °C contour interval). Purple lines represent probes in the physical experiment.

6 TEST CASE 4: 2-D HEAT RELEASE FROM A STRIP SOURCE

Test case 4 considers transient two-dimensional (2-D) single-permeability single-component mass and energy balance. The test case suite verifies that xFlo correctly implements balance equations when each grid cell has connections in multiple space dimensions and transport includes advection as well as diffusion. The test case suite builds on test case 3, which considered diffusion in two dimensions without advection. The test suite considers energy transport from a strip source subject to steady flow of water under fully saturated conditions. Test case 4 uses an analytical solution for transient solute transport in a semi-infinite medium with uniform and constant velocity, subject to a strip source with an instantaneous jump in solute concentration. Heat transport is analogous to solute transport, except that heat also conducts through the solid phase. In the test cases, water flows from top to bottom due to gravity. The flow field essentially uniform because the temperature perturbation is too small to cause significant density or viscosity changes. Two cases are considered, identical except for the intrinsic permeability of the medium, to consider diffusion- and advection-dominated conditions.

The analytical solution was reported by Cleary and Ungs (1978), and is publicly available in collections by Segol (1994) and Wexler (1992). Note that the analytical solution could also be applied to transport of dissolved air in the liquid phase, vapor in the gas phase, and energy in the gas phase. The test case 1 suite verifies that the coupled equations are correctly assembled and solved in one-dimensional (1-D) systems. Verifying that the coupled equation assembly process is correctly implemented in multiple space dimensions uses a single scenario without loss of generality, because the same routines are used for assembly and solution regardless of dimensionality.

The numerical simulation considers a bounded domain with specified composition (mole fraction of air in water, liquid pressure, and temperature) on the top and bottom, and no flux on the sides. The domain takes advantage of symmetry by representing the vertical plane passing through the center of the strip as a no-flow boundary. The initial composition is uniform throughout the domain. The top and bottom boundaries are maintained at the same composition, except the temperature is increased by 1 °C at the strip source.

The governing equation for the analytical solution is

$$R\frac{\partial C}{\partial t} + v_x\frac{\partial C}{\partial x} + v_y\frac{\partial C}{\partial y} - D_x\frac{\partial^2 C}{\partial x^2} - D_y\frac{\partial^2 C}{\partial y^2} - \lambda C = 0 \quad (6)$$

where C is concentration, v_x and v_y are fluid velocity in the x and y coordinate directions, D_x and D_y are the dispersion coefficients in the x and y coordinate directions, R is a retardation coefficient, and λ is a decay coefficient. The form reported by Wexler (1992) assumes $v_y = 0$. The domain is semi-infinite in the x coordinate direction and unbounded in the y coordinate direction. All parameters are isotropic and homogeneous.

The initial condition is $C = 0$ throughout the domain. The initial condition is maintained at the lower (upstream) boundary in the x coordinate direction except for a strip of width $2a$ centered on $y = 0$, which is instantaneously changed to a condition of $C = C_0$ at time zero. The concentration at the strip exponentially decays with a time constant of α.

The analytical solution is

$$C(x,y,t) = \frac{C_0 x}{4(\pi D_x)^{1/2}} \exp\left(\frac{v_x x}{2D_x}\right) \int_{\tau=0}^{\tau=t/R} \tau^{-3/2} f(\tau) d\tau \quad (7)$$

$$f(\tau) = \exp\left[-\left(\frac{{v_x}^2}{4D_x} + \lambda R - \alpha R\right)\tau - \frac{x^2}{4D_x\tau}\right]\left[\operatorname{erf}\left(\frac{a - y + v_y\tau}{2\sqrt{D_y\tau}}\right) + \operatorname{erf}\left(\frac{a + y - v_y\tau}{2\sqrt{D_y\tau}}\right)\right] \quad (8)$$

The analytical solution applies to changes in temperature from an initial state, with $\lambda = \alpha = 0$ and heat capacity taking the place of the retardation coefficient.

The Peclet number is typically used to represent the ratio of advective to diffusive flux, where $\mathrm{Pe} = Lv_x/D_x$. The particular parameters for the problem were selected to represent a transition from diffusion-dominated to moderately advection-influenced, considering two cases with Pe different by an order of magnitude (4.58 versus 45.8).

For the test case, the physical system was assumed fully saturated and the velocity field was assumed to be vertically downward, with flow solely due to gravity. The thermal balance equation, assuming thermal equilibrium between liquid and solid, was determined by summing the thermal balance equations for liquid and solid, yielding

$$(C_w\epsilon_w + C_s\epsilon_s)\frac{\partial\delta}{\partial t} + C_w q_w \frac{\partial\delta}{\partial x} - (K_w\epsilon_w + K_s\epsilon_s)\left(\frac{\partial^2\delta}{\partial x^2} + \frac{\partial^2\delta}{\partial y^2}\right) = 0 \quad (9)$$

where δ is change in temperature from the initial value, C_w and C_s are water and solid heat capacities, K_w and K_s are the water and solid thermal conductivities, ϵ_w and ϵ_s are the water and solid volume fractions, and q_w is the liquid volumetric flux. Equation (9) corresponds to Equation (6), with δ corresponding to C. The x direction is vertically downward. The volumetric flux is calculated by setting the top and bottom pressures to the same value. Velocity is controlled by scaling the intrinsic permeability. The x coordinate direction is vertically downward. The strip source was assumed perpendicular to gravity, so $v_y = 0$. Thermal conductivity was set to the same constant in both coordinate directions.

The two comparison solutions presented in Figure 13 illustrate the xFlo solution with the background color scheme and red contours, and illustrate the analytical solution with white contours. The solution is presented at time 0.3 L/v_x on the left and time 0.4 L/v_x on the right, prior to any significant contact between the plume and the computational domain boundaries. The physical system is identical in both cases, with the intrinsic permeability (hence v_x and Pe) increased by an order of magnitude from left to right. Physical properties for the two cases are listed in Table 5.

The xFlo solution clearly responds appropriately to the change in parameters and represents the overall solution reasonably well. The two figures demonstrate that xFlo correctly implements the balance equations when each grid cell has connections in multiple space dimensions.

The xFlo solutions are overly diffusive in both cases, demonstrated by the outer xFlo contours leading the outer analytical contours. Nonphysical numerical diffusion is the expected outcome for the computational scheme, which uses upstream weighting to increase numerical robustness for highly nonlinear systems. Numerical dispersion is typically increased when the flow

directions are not parallel and perpendicular to the grid directions. Additional work would be necessary to implement transport with less numerical dispersion. However, numerical dispersion is typically less important in applications with strongly varying physical properties, because these problems tend to exhibit self-sharpening fronts.

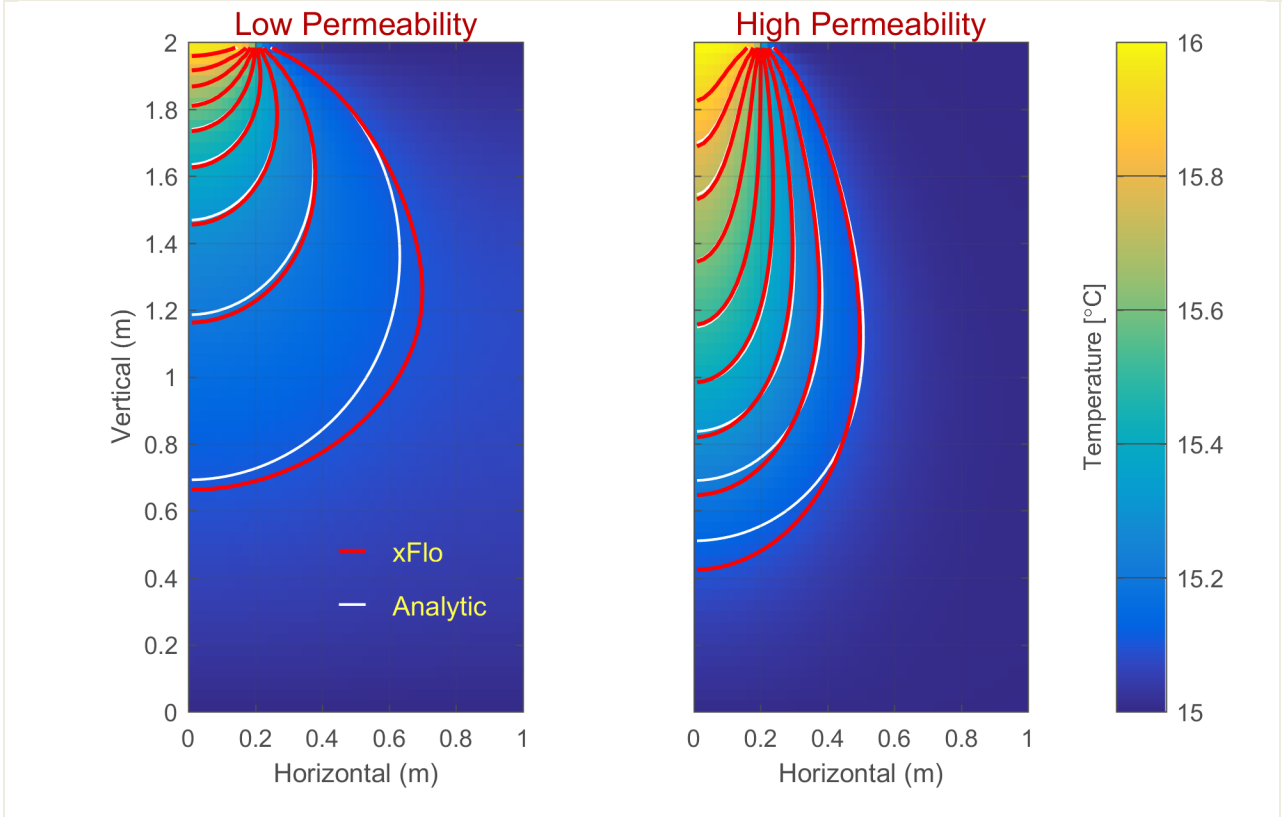

Figure 13. xFlo-calculated (background and red contours) and analytical (white contours) change in temperature due to a strip source in a vertical flow field. Contour spacing is 0.1 °C. (Left) At time 0.3 L/v_x for a Peclet number of 4.58. (Right) At time 0.4 L/v_x for a Peclet number of 45.8.

Table 5. Parameters used for verification testing in strip source simulations			
Description	**Units**	**Small Pe**	**Large Pe**
van Genuchten alpha	1/Pa	8.77×10^{-5}	8.77×10^{-5}
van Genuchten lambda (= m)		0.231	0.231
Liquid residual saturation		0	0
Liquid minimum saturation		0	0
Saturation transition (S*)		0.99	0.99
Permeability	m^2	1×10^{-13}	1×10^{-12}
Porosity		0.2	0.2
Dry thermal conductivity	W/m-K	1.56	1.56
Wet thermal conductivity	W/m-K	1.56	1.56
Solid specific heat	J/kg-K	880	880
Solid density	kg/m^3	2,300	2,300
Diffusion coefficient for gas in liquid	m^2/s	0	0
Klinkenberg parameter		0	0
Initial temperature	°C	15	15
Strip temperature	°C	16	16
Strip width	m	0.4	0.4
See Stothoff and Painter (2016) for xFlo parameter descriptions			

7 SUMMARY

The computer code MULTIFLO was developed in the mid-1990s and early 2000s to aid in understanding perturbations to the near-field environment surrounding a proposed underground high-level nuclear waste repository at Yucca Mountain following emplacement of the waste. MULTIFLO was designed to be a general integral finite-volume code for simulating multiphase, multicomponent transport processes in nonisothermal systems with chemical reactions and reversible and irreversible phase changes in solids, liquids, and gases in one, two, and three spatial dimensions. MULTIFLO was coded in FORTRAN 77. MULTIFLO 2.0 was capable of being used (Painter, 2005) to address drift-scale and repository-scale coupled thermal-hydrological-chemical processes that could affect the performance of the proposed repository.

Porting MULTIFLO to a newer code, taking advantage of capabilities of newer releases of Fortran, began prior to the Yucca Mountain license application review. The new code, xFlo, was intended to replace MULTIFLO for use in performance confirmation activities. At the time of the license application, the conversion was incomplete. Since that time, xFlo has been used for other projects, which has led to model revisions. This report documents the status of xFlo in regard to the expected capabilities for replacing MULTIFLO and provides information for putting xFlo under TOP-018 software control.

The xFlo simulator (Stothoff and Painter, 2016) is an integrated finite volume code based on MULTIFLO. xFlo reproduces most of the functionality of METRA, the thermal-hydrologic component of MULTIFLO, but does not include the multicomponent transport functionality of GEM, the geochemical and phase change component of MULTIFLO. The most notable components of METRA that would be useful for performance confirmation but have not been reproduced include anisotropy and several types of boundary conditions (e.g., radiation and gravity outflow).

The confirmatory analyses described in this report were selected to verify that the version of xFlo documented by Stothoff and Painter (2016) correctly implements the governing balance equations and the constitutive relationships included in the model. For this purpose, code validation includes (i) performing a series of validation tests to provide reasonable confidence that the software successfully implements the underlying theory and algorithms, and (ii) documenting the analyses. In order to verify that the simulator is working correctly, it is useful to consider a suite of simulations that (i) test particular key aspects of the simulator and (ii) provide benchmarks for comparison after code modification.

The validation testing considered four test case suites. Two test case suites were performed to build confidence in xFlo capabilities as part of earlier modeling activities, and these test cases are documented for the first time in this report. xFlo model input for these two cases was developed with special-purpose MATLAB routines. The other two test case suites were developed specifically for this report, and the xFlo model input for these test cases was created with a new Excel- and MATLAB-based interface that is being developed to couple xFlo with FLAC in order to perform coupled thermal-hydrologic-mechanical simulations.

Test case 1 consists of a set of one-dimensional (1-D) transient single-permeability scenarios in linear, cylindrical, and spherical coordinates. These scenarios are used to verify that liquid and energy transport balance equations and the associated boundary conditions are correctly implemented and accurate solutions are obtained. The comparison solutions use (i) the PDEPE solver provided by MATLAB with independently coded routines and (ii) a semi-analytical model. All three solutions are obtained with different methods, but agree well. Test case 1

demonstrates that the basic equation sets are correctly implemented in 1-D. The constitutive models in xFlo are minimally addressed in test case 1.

Test case 2 consists of a set of 1-D steady dual-permeability scenarios. These scenarios are intended to address the implementation of constitutive relationships for two-phase retention and relative permeability, including transfer between matrix and fracture continua. xFlo does not directly consider steady state, and must step through time until an equilibrium is reached. The comparison solutions use the ode45 solver provided by MATLAB with independently coded routines describing the constitutive relationships, using a shooting algorithm to derive consistent boundary conditions. The ode45 solver adaptively refines the grid where needed for accuracy. The approach is based on test case 2 for MULTIFLO validation (Painter, 2003, 2005). A single-layer system is used to verify that xFlo properly accounts for (i) gradual changes in fracture volume fraction and (ii) the active-fracture model for liquid transfer between matrix and fracture. The xFlo simulations agree well with the comparison solutions, provided that the interface area between matrix and fracture is scaled. The same scale factor appears to apply for all scenarios. Additional work is needed to identify and rectify the discrepancy.

Test case 2 also considers a three-layer test domain using a dual-permeability formulation. This system uses properties similar to welded and nonwelded tuffs found at Yucca Mountain, which have dramatically differing permeabilities and retention properties. The test problem was intentionally designed to be extremely challenging. The particular properties generate a perched zone in the matrix of the nonwelded middle layer, but not in the fractures or the welded matrix. No xFlo simulation reached a steady state, with many of the simulations abruptly terminating without an error message. The test case reveals two issues may be present: (i) fluxes across variably saturated interfaces, and (ii) convergence at the transition between saturated and unsaturated conditions. xFlo appears to have difficulty properly representing fluxes across the very sharp change in properties at layer transitions, even with substantial grid refinement at the transition. Partial simulations suggest that fluxes are blocked at some interfaces. Furthermore, a simplification that smoothes the van Genuchten (1980) constitutive relationship for retention near the transition point, which is intended to improve convergence rates and overall robustness, created substantially different results and associated code appears to be responsible for some crashes. This test case is documented as a starting point for identifying issues in further work.

Test case 3 considers quasi-steady energy balances in cylindrical coordinates. The model system is based on a laboratory column heater test, in which a bentonite buffer material contained in a Teflon cylinder with steel reinforcement is heated from the bottom. The physical system features strong contrasts in thermal properties, and equilibrates within a few days. The steady-state comparison solution was developed using the commercially available COMSOL software. The comparison builds confidence that xFlo correctly represents diffusion in cylindrical coordinates with strongly contrasting material properties. The three-layer problem in test case 2 has strongly contrasting properties that cause simulations to fail, but xFlo has no difficulty with test case 3. The implication is that xFlo may not adequately represent the nonlinear material properties used in test case 2, even though the balance equations are correctly implemented.

Test case 4 considers advective transport in two dimensions. The scenarios are intended to demonstrate that the advective transport resulting from coupled equations is correctly implemented in more than one spatial dimension. The xFlo-calculated solutions compare well to analytical solutions for two scenarios, although the numerical solution has smearing in the front due to numerical dispersion. Numerical dispersion is expected for the xFlo scheme, as a

consequence of the upstream weighting that provides more robust solutions with highly nonlinear balance equations. Note that numerical dispersion may not significantly influence the solution for self-sharpening fronts, which often occur in strongly nonlinear thermal-hydrologic scenarios. Numerical dispersion is minimized when gridding is aligned with the flow direction.

The first MULTIFLO test case, which used an analytical approach based on the Boltzmann transform (Doughty and Pruess, 1992) to represent nonisothermal two-phase (gas and liquid) and two-component (air and water) redistribution, was also considered for the test case suite. This test problem would be very useful for checking that strongly coupled and nonlinear processes are correctly implemented. The project schedule could not accommodate fully implementing the analytical solution, which is considerably more complex than the other comparison models. It is recommended that this additional test case be considered in future work.

The test cases demonstrate that xFlo correctly implements its set of governing balance equations and its available set of boundary conditions. The tests show that there is a discrepancy in the model related to transfer between matrix and fracture in dual-permeability simulations. The tests also suggest that, under some conditions, xFlo has difficulty with the transition between full and partial saturation, and has difficulty calculating liquid fluxes across interfaces with sharply contrasting retention properties under partial saturation. These difficulties and discrepancies are documented to provide guidance on current model limitations and to provide a starting point for addressing model limitations in future work. Of course, the significance of these difficulties and discrepancies very much depends on the specifics of the problem; therefore, the need to address these issues will depend on the potential applications being considered for the model.

8 REFERENCES

Cleary, R.W., and M.J. Ungs, M. J. "Analytical Models for Groundwater Pollution and Hydrology." Water Resources Program Report 78-WR-15. New Jersey: Princeton University. 1978.

Doughty, C. and K. Pruess. "A Similarity Solution for Two-Phase Water, Air, and Heat Flow Near a Linear Heat Source in a Porous Medium." *Journal of Geophysical Research*. Vol. 97. pp. 1,821–1,838. 1992.

Liu, H.H., C. Doughty, and G.S. Bodvarsson. "An Active Fracture Model for Unsaturated Flow and Transport in Fractured Rocks." *Water Resources Research.* Vol. 34, No. 10. pp. 2,633–2,646. 1998.

Painter, S. "Software Validation Report for MULTIFLO Version 2.0.1." San Antonio, Texas: Center for Nuclear Waste Regulatory Analyses. 2005.

Painter, S. "Software Validation Report for MULTIFLO Version 1.5.2." San Antonio, Texas: Center for Nuclear Waste Regulatory Analyses. 2003.

Ségol, G. "Classic Groundwater Simulations: Proving and Improving Numerical Models." Englewood Cliffs, New Jersey: Prentice Hall. 1994.

Stothoff, S., C. Manepally, G. Ofoegbu, B. Dasgupta, and R. Fedors. "Modeling Thermohydrological-Mechanical Behavior of Granular Bentonite in a Laboratory Column Test." IM 17860.09.001.410. ML15324A128. San Antonio, Texas: Center for Nuclear Waste Regulatory Analyses. 2015.

Stothoff, S. and S. Painter. "xFlo Version 1.1β User's Manual." San Antonio, Texas: Center for Nuclear Waste Regulatory Analyses. ML16167A315. 2016.

Theis, C.V. "The Relation Between the Lowering of the Piezometric Surface and the Rate and Duration of Discharge of a Well Using Ground-Water Storage." *Transactions, American Geophysical Union*. Vol. 16. pp. 519–524. 1935.

van Genuchten, M.T. "A Closed-Form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils." *Soil Science Society of America Journal*. Vol. 44. pp. 892–898. 1980.

Wexler, E.J. "Analytical Solutions for One-, Two-, and Three-Dimensional Solute Transport in Groundwater Systems with Uniform Flow." Techniques of Water-Resources Investigations 03-B7. Denver, Colorado: United States Geological Survey. 1992.

APPENDIX A

CONFIRMATORY MODELS

A CONFIRMATORY MODELS

Analytical and semi-analytical approaches are particularly useful for testing that sources and flux calculations are correctly implemented in a numerical model. Some verification tests in this report rely on the PDEPE routine provided by MATLAB, using independently coded routines describing the constitutive relationships included in xFlo. An independent second model, based on a semi-analytical approach and also implemented in MATLAB, was used to confirm the implementation of the PDEPE routine. The semi-analytical model and the PDEPE-based model use the same routines to describe constitutive properties and typically agree to within a fraction of a percent or better with respect to change in mass during the simulation when equivalent discretization levels are used.

Verification testing of the MULTIFLO model, which xFlo was developed to replace, used analytical approaches relying on solving ordinary differential equations (ODEs) describing self-similar conditions or steady state conditions. MATLAB offers a variety of ODE solvers suitable for solving systems of first-order equations, and the equations considered in the analytical approaches are readily transformed into a format suitable to the ODE solvers. This approach is usually more accurate than models based on solving partial differential equations (PDEs), because the MATLAB ODE solvers preserve accuracy by adapting their step size based on the solution characteristics. While it is possible to use equally fine grids in fixed-grid models such as xFlo and MULTIFLO, the grid must be specified a priori rather than adapting to the solution.

A commercially available numerical model, COMSOL, is used to test that xFlo considers complex contrasts in parameter values.

A1 1-D Transient Systems

A1.1 PDEPE Model

MATLAB provides the PDEPE solver for one-dimensional (1-D) partial differential equations. The PDEPE solver is built on MATLAB solver routines for ordinary differential equations. To use the PDEPE solver, the user provides (i) a grid, (ii) functions that define the capacitance, fluxes, and sources give the state variables and their spatial derivative, (iii) a function defining initial conditions, and (iv) a function defining boundary conditions. The routines provided to the PDEPE solver are small special-purpose functions developed specifically to describe the test problems. The constitutive relationships and associated parameters were set to the same values that were provided to xFlo.

The routines provided to the PDEPE solver only consider one state variable. This approach was adopted for simplicity and ease of comparison, and is not an inherent limitation of the PDEPE routine. Each problem was defined to test a single state variable in xFlo.

A1.2 Semi-Analytical Model

Stothoff and Pinder (1992) developed a semi-analytical approach to modeling isothermal two-phase flow. For 1-D flow, the approach reduces to moving a discretized set of saturation values using their characteristic velocity. The characteristic velocity is determined using the relationship

$$v(\theta) = \frac{q_p - q_m}{\theta_p - \theta_m} \quad (1)$$

where v is the velocity [L/T], q is flux [L/T], θ is volumetric water content [–], and subscripts p and m represent discretized values on the plus and minus side of the characteristic. The approach is easily applied to transport of dissolved species and heat using

$$v(c) = \frac{q_p - q_m}{M(c_p) - M(c_m)} \quad (2)$$

$$v(T) = \frac{q_p - q_m}{E(T_p) - E(T_m)} \quad (3)$$

where the dissolved species equation is written in terms of mass flux [M/L^2/T] and total mass [M/L^3], and the heat transport equation is written in terms of energy flux [J/L^2/T] and total energy [J/L^3].

The original approach was developed in the context of two-dimensional (2-D) contours, where the calculated velocity is normal to the contour and the values for q were developed using an independent numerical model. The approach is well suited to a 1-D context where the terms can be evaluated using a moving grid, and is especially powerful for semi-infinite domains. Although the approach requires few grid cells to be quite accurate and inherently preserves monotonicity, the approach is not inherently mass-conservative.

The approach can be extended to 2-D and three-dimensional (3-D) radial coordinates. In this case

$$r^{N-1} v(\theta) = \frac{r_p^{N-1} q_p - r_m^{N-1} q_m}{\theta_p - \theta_m} \quad (4)$$

where N is the spatial dimension (1, 2, or 3). This result was verified numerically using the pdepe solver.

The approach is very straightforward in 1-D problems with an initially uniform θ and a specified value of θ on the boundary starting at some time. The θ profile is discretized into a set of values ramping from θ_b to θ_0 over a short distance, where θ_b and θ_0 are the boundary and initial values for θ. This is equivalent to forming a finite element model with linear elements. When comparing the semi-analytical model to a numerical model, the ideal ramp distance corresponds to the center of the numerical cell adjacent to the boundary.

Assuming that θ varies linearly between particular values of θ leads to, for "element" p between the i and $i+1$ discretized values of θ,

$$\theta_p = \frac{\theta_{i+1} + \theta_i}{2} \quad (5)$$

$$\frac{d\theta_p}{dx} = \frac{\theta_{i+1} - \theta_i}{x_{i+1} - x_i} \quad (6)$$

Accordingly, $q_p\left(\theta_p, d\theta_p/dx\right)$ is readily calculated for linear and nonlinear problems. All characteristic velocities are determined once q values are available for all of the elements.

It is reasonable to use explicit updating for the position of the characteristics, ensuring that (i) the separation between characteristics does not increase or decrease by more than a small tolerance and (ii) the time step does not increase dramatically from one step to the next (e.g., limited to 1.1 times the previous value).

The semi-analytical approach is more complicated with a flux boundary condition, because θ_b is not a fixed value. The θ_b characteristic may move in to or out of the domain, necessitating adjusting the boundary value. A globally mass-conservative approach assigns a new value of θ_b such that total mass change in the system exactly balances the input flux over the time step. The globally mass-conservative approach works well when there is a single flux boundary or other extremum to adjust, but does not assure that the local boundary flux is locally consistent with θ_b and is underconstrained if the profile has multiple flux boundaries.

A local approach assigns a new value of θ_b by calculating the new value at the boundary that is necessary to locally conserve flux. The boundary flux is used to calculate the velocity of the current boundary value. When the velocity of the current boundary value is out of the domain, simply linearly interpolating to the boundary location provides a locally conservative new boundary estimate.

When the velocity of the current boundary value is inward, a two-step process is necessary: (i) create a new profile for the region between the boundary and relocated boundary and (ii) merge the new region with the old region while conserving mass. The region between the boundary and relocated boundary is given a gradient consistent with the boundary flux. Assuming that the interior is designated with the p index in Equation (4), the velocity for θ_b is calculated by setting q_m to the boundary condition flux and setting $\theta_b = \theta_m$. When the velocity for θ_b is nonzero, assuming that the flux in the element next to the boundary equals the boundary flux provides a condition for estimating θ_b,

$$q_b = q\left(\frac{\theta_p + \theta_b}{2} \approx \theta_b, \frac{\theta_p - \theta_b}{x_p - x_b}\right) \quad (7)$$

Linearly projecting the value of θ_0 is analytically correct for linear equations and adequate for nonlinear problems when the new position of θ_b is close to the boundary. For example, unsaturated zone flux can be written

$$q = -K(\theta_b)\left[\frac{\varphi\left(\theta_p\right) - \varphi(\theta_b)}{x_p - x_b} + \frac{z_p - z_b}{x_p - x_b}\right] \quad (8)$$

Simple rearrangement provides an explicit value for $\varphi\left(\theta_p\right)$.

The new profile has two cells, the small new cell next to the boundary and the previous cell. The final boundary value is calculated by merging the two cells, calculating a new boundary value such that total mass is conserved.

The local approach converges to a value for θ_b that is consistent with the flux condition, but does not assure global mass balance.

Regridding the "element" closest to the boundary is desirable if $|\theta_b - \theta_1|$ becomes too large or small, where θ_1 is the characteristic value adjacent to the boundary. A practical approach uses a fixed value of $\Delta\theta$ between all internal cells except at the boundary. The element adjacent to the boundary is split when $|\theta_b - \theta_1| > 1.8\Delta\theta$, setting the newly created characteristic value to $\theta_1 + \Delta\theta$ and using linear interpolation to determine the spatial location for the new characteristic. The element adjacent to the boundary is merged when $|\theta_b - \theta_1| < \Delta\theta/2$, removing the θ_1 characteristic and resetting θ_b using Equation (7). Both regridding approaches conserve global mass.

Regridding is also desirable when the number of cells in the profile becomes too large or too small as boundary cells are added or removed. Increasing the number of cells is performed by splitting each cell in two, linearly interpolating to conserve mass. Decreasing the number of cells is performed by merging each set of three cells into two cells. The inner cell is merged into a point that has the average θ value of its endpoints. The position of the point is determined such that the total mass in the two remaining cells is the same as the total mass in the three original cells. Removing cells only occurs when the number of cells in the profile is a multiple of three.

A1.3 Semi-Analytical and PDEPE Model Comparisons

The two models were compared for advection/diffusion problems, thermal diffusion problems, and Richards-equation problems. The thermal diffusion and Richards-equation comparison problems were comparable to the verification tests described in Section 3. The two very different approaches provide essentially identical results until the moving front contacts the downstream boundary, providing strong confirmation that the PDEPE model is providing very good solutions. No plots comparing the results are provided here because the solutions simply plot on top of each other.

The semi-analytical model uses explicit updating for the position of the characteristics, and requires very small time steps when diffusive fluxes are important. In general, the PDEPE routine is faster than the semi-analytical model, especially for diffusion-dominated problems. The semi-analytical model is competitive for advection-dominated problems (especially in large domains), but is very slow for diffusion-dominated cylindrical and spherical problems.

A2 Steady-State ODEs

ODEs are often solved for the transient response of coupled variables. MATLAB provides a number of ODE solvers suited to different types of ODEs written in the form $\boldsymbol{y}' = f(t, \boldsymbol{y})$ or $\boldsymbol{M} \cdot \boldsymbol{y}' = f(t, \boldsymbol{y})$, where $\boldsymbol{y}' = d\boldsymbol{y}/dt$. Steady state systems of advection-dispersion equations in one spatial coordinate also can be solved as systems of ODEs, replacing the time coordinate with the spatial coordinate.

The steady-state 1-D flow equation

$$\frac{dq}{dx} = -\frac{d}{dx}K\left(\frac{dP}{dx} + \rho g\frac{dz}{dx}\right) = S \quad (9)$$

can be written in MATLAB's format as

$$\frac{dP}{dx} = -\frac{q}{K} - \rho g\frac{dz}{dx} \quad (10)$$

$$\frac{dq}{dx} = S \quad (11)$$

where P is pressure, q is flux, K is hydraulic conductivity, ρ is density, g is the acceleration due to gravity, S represents sources and sinks, z is elevation, and x is the spatial coordinate. In this format, $\boldsymbol{y} = \{P, q\}$.

The ODE solvers integrate the system of equations from one boundary to the other. This procedure requires that both P and q are specified at the starting boundary. This may be available, for example if integration starts at a water table and there is a known flux in the domain. Usually a boundary condition is specified at each boundary, so only one of the two variables are known during integration. A shooting method is typically used to determine the missing parameter value. The shooting method starts with an initial guess for the missing parameter and integrates the solution to the far boundary. Usually there is a discrepancy between the calculated solution and the specified boundary condition at the far boundary. The shooting method adjusts the initial guess until there is no longer a discrepancy. The intrinsic MATLAB routine called *fzero* finds zeros of a function, and can be used in the shooting method.

The same procedure can be generalized for systems of equations. In a dual-permeability model, a steady-state 1-D flow equation is written for the matrix and fracture continua as

$$\frac{d}{dx}K_m\left(\frac{dP_m}{dx} + \rho g\frac{dz}{dx}\right) = \lambda\left(P_f - P_m\right) \quad (12)$$

$$\frac{d}{dx}K_f\left(\frac{dP_f}{dx} + \rho g\frac{dz}{dx}\right) = \lambda\left(P_m - P_f\right) \quad (13)$$

where the m and f subscripts represent matrix and fracture. These are transformed into a system of four equations, coupled by the transfer between matrix and fracture. In this case, the shooting procedure determines two boundary conditions at the starting boundary.

The procedure can also be generalized to consider additional components to a flux. For example, vapor flux may be significant in an unsaturated medium. In this case,

$$q = -K\left(\frac{dP}{dx} + \rho g\frac{dz}{dx}\right) - D\frac{d\rho_v}{dx} \quad (14)$$

where ρ_v is the vapor density and D is a diffusion coefficient. For a system of fluxes and state variables, flux of component k can be written

$$q_k = -\sum_i K_{ki}\frac{dv_i}{dx} \quad (15)$$

where v_i is a state variable and K_{ki} is the corresponding coefficient. It may be that each variable v is a function of other state variables u_j (e.g., ρ_v is a function of pressure and temperature). By the chain rule,

$$\frac{dv_i}{dx} = \sum_j \frac{\partial v_i}{\partial u_j}\frac{du_j}{dx} \tag{16}$$

This leads to a system of flux definition equations in the form

$$\boldsymbol{q} = -\boldsymbol{K} \cdot \boldsymbol{J} \cdot \frac{d\boldsymbol{u}}{dx} \tag{17}$$

where $\boldsymbol{K}$ is a matrix with the K_{ki} entries and $\boldsymbol{J}$ is a matrix with the $\partial v_i/\partial u_j$ entries.

Written in this format, the system of ODEs for a steady state solution can be derived for whatever state variables are convenient. For example, in a system of air, water, and energy, it may be convenient to use a state variable of $u_1 = \log(\mathrm{X}_{\mathrm{L}}^{\mathrm{a}})$, where $\mathrm{X}_{\mathrm{L}}^{\mathrm{a}}$ is the mole fraction of air in liquid, to ensure nonnegativity even for very small values.

To maintain accuracy with very different material property coefficients, it is often useful to divide each flux definition equation by the corresponding coefficient on the diagonal of $\boldsymbol{K}$ or $\boldsymbol{K} \cdot \boldsymbol{J}$.

A3 COMSOL Model

The COMSOL Multiphysics® simulator is a general-purpose simulator for partial differential equations, including 2-D cylindrical problems, with a powerful graphical user interface (GUI). To use the COMSOL simulator, the user develops the entire problem within the GUI. The GUI has toolboxes specialized for particular types of problems, is very robust for defining engineered components, and ensures that all parameters are consistent with the toolbox requirements. For certain problems, such as heat transfer, the COMSOL simulator offers capabilities that xFlo does not currently support (e.g., radiation). Tests with COMSOL are limited to the heat transfer toolbox.

The flexible capabilities offered by COMSOL allows for a good check that xFlo accounts for spatial variability. The COMSOL model was not used for essentially 1-D conditions because the PDEPE and semi-analytical models provided better flexibility.

REFERENCE

Stothoff, S.A., and G.F. Pinder. "A Boundary Integral Technique for Multiple-Front Simulation of Incompressible, Immiscible Flow in Porous Media." *Water Resources Research*. Vol 28, No.8. pp. 1,067–1,076. 1992.