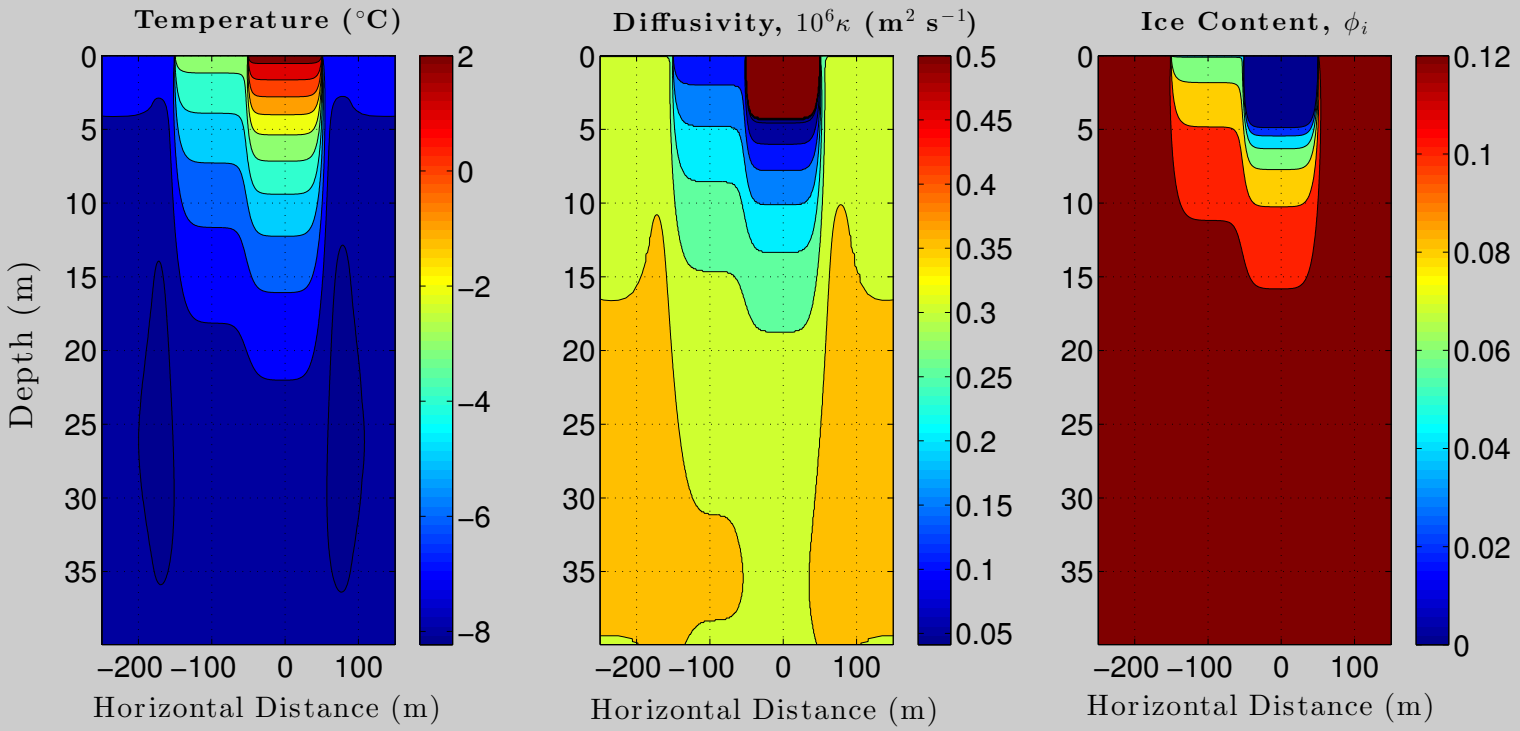
CVPM

Version 1.1 Modeling System User's Guide

April 2020



CVPM was developed by:

Gary Clow
Institute of Arctic and Alpine Research
University of Colorado
Boulder, Colorado USA

Contents

1 Overview

Introduction

The Control Volume Permafrost Model (CVPM) is a flexible heat-transfer modeling system designed for scientific and engineering studies in permafrost terrain, and as an educational tool. CVPM implements the nonlinear heat-transfer equations in 1-D, 2-D, and 3-D cartesian coordinates, as well as in 1-D radial and 2-D cylindrical coordinates. To accommodate a diversity of geologic settings, a variety of materials can be specified within the model domain, including: organic-rich materials, sedimentary rocks and soils, igneous and metamorphic rocks, pure ice, borehole fluids, and other engineering materials. A radiogenic heat-production term allows simulations to extend into deep permafrost and underlying bedrock. CVPM can be used over a broad range of depth, temperature, porosity, water saturation, and solute conditions on either the Earth or Mars. The model is suitable for applications at spatial scales ranging from centimeters to hundreds of kilometers and at timescales ranging from seconds to thousands of years, including:

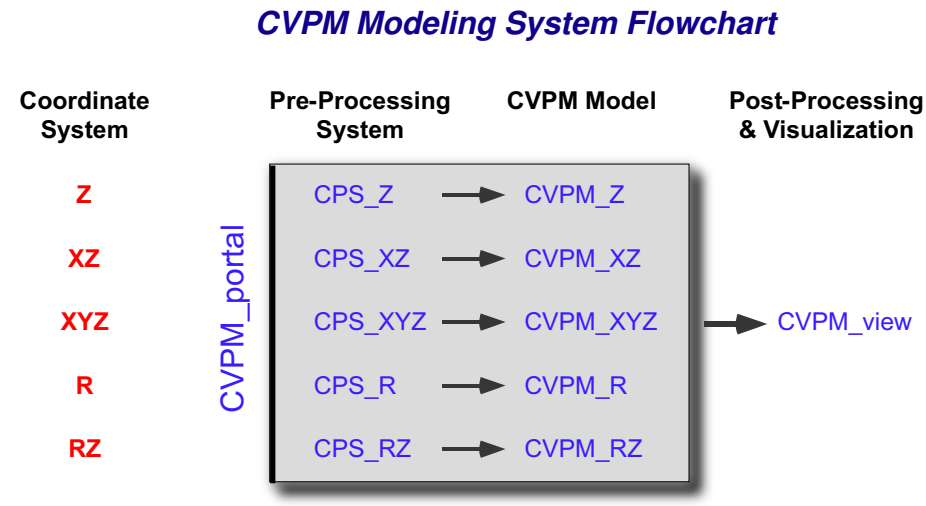
- Idealized simulations
- Geophysical inversions for subsurface material properties
- Geophysical inversions for time-dependent boundary conditions
- Climate-change impact projections
- Coupled-model applications
- Engineering applications
- Teaching

A complete description of the model physics and numerical implementation can be found in Clow (2018). The CVPM modeling system is in the public domain and is freely available for community use. CVPM is implemented entirely in the MATLAB programming language.

Modeling system components

Major components of the CVPM modeling system include:

- The preprocessing system (CPS)
- The CVPM model
- Post-processing and visualizations tools
- Utilities



CVPM Portal

Program CVPM_portal[†] provides entry to the modeling system. It will run the preprocessor (CPS_Z, CPS_XZ, CPS_XYZ, ...) for the appropriate coordinate system, the corresponding heat-transfer model (CVPM_Z, CVPM_XZ, CVPM_XYZ, ...), or both CPS and CVPM using a single call.

CPS

The CVMP preprocessor has several functions, including: (a) defining the simulation domain, (b) creating the spatial grid, (c) specifying the material properties at each grid point, (d) establishing the initial temperature field, and (e) establishing the boundary conditions. Once this information has been gathered, CPS outputs it to a file in preparation for a simulation run by CVPM.

CVPM

This is the main computational component of the modeling system. CVPM solves the transient thermal problem in the model domain using the control-volume method subject to time- and space-dependent boundary conditions (Patankar, 1980; Anderson et al., 1984; Minkowycz et al., 1988). All the information needed to start a simulation run is obtained from the CPS output file. CVPM in turn stores its results in an output file for subsequent post-processing and visualization.

Post-Processing, Visualization, and Utilities

At this time, a limited amount of post-processing is done within the visualization routines (view_Z, view_XZ, view_XYZ, ...) which can be accessed through program CVPM_view. Utility routines are currently available to assist with making: (a) initial condition files for radial and 2-D simulations, and (b) boundary condition files for 2-D and 3-D simulations.

[†]Explicit code names are indicated in blue throughout the user's guide. The terms CPS and CVPM are used generically.

2 Software Installation

As CVPM is written in the MATLAB programming language, a MATLAB license is required to run the model. CVPM has no other program or library dependencies. Once the location for the modeling system's working directory is established, several subdirectories should be created that will be utilized by CVPM. The directory structure of the CVPM modeling system is as follows,

Directories	Description
`wdir`[†]	working directory for the CVPM system
`wdir/docs`	document files
`wdir/source`	source codes
`wdir/utilities`	visualization and utility codes
`wdir/namelists`	namelist files
`wdir/geo`	geology (GEO) files
`wdir/tmp`	temporary (scratch) files
`wdir/ICs`	initial condition files
`wdir/BCs`	boundary condition files
`wdir/CPSout`	CPS output files
`wdir/CVPMout`	CVPM output files

[†]`wdir` is an alias for the actual working directory which can be located anywhere within the user's directory system.

Directories `wdir`, `wdir/source`, and `wdir/utilities` should be added to the MATLAB path. Program set_CVPM_paths can be used to set these paths prior to running CVPM for the first time.

3 CVPM Portal

Entry to the CVPM modeling system is provided by program CVPM_portal. This program reads a user-created file CVPM.config that provides the location of the working directory, general information about the numerical simulation(s), and the names of the experiments to be run. Below is a description of the configuration variables and sample values.

Configuration File (CVPM.config)

Variable Names	Sample Values	Description
`wdir`	'~/numer/CVPM_v1.1'	working directory for the CVPM system
`coordinate_system`	'Z'	1-D vertical
	'XZ'	2-D cartesian
	'YZ'	2-D cartesian
	'XYZ'	3-D cartesian
	'R'	1-D radial
	'RZ'	2-D cylindrical
`Gopt`	1	GEO files are in 'wdir/geo/' (text format)
	2	GEO files are in 'wdir/geo/' (MATLAB format)
	3	GEO files are in 'wdir/tmp/' (MATLAB format)
`Ropt`	1	run CPS on one or more files
	2	run CVPM on one or more files
	3	run CPS and CVPM on one or more files

`experiment`	'ESN_qb40'	namelist file for simulation #1
	'ESN_qb42'	namelist file for simulation #2
	⋮	⋮
	'ESN_qb50'	namelist file for simulation #N

Variable `Gopt` indicates the location and format of the material property (GEO) files; `Gopt` option 3 is provided for experiments seeking to find the material properties through an inversion. Variable `Ropt` controls whether only CPS is to be run, only CVPM is to be run, or CPS is to be run followed by CVPM. One or more experiments can be launched sequentially by CVPM_portal. Variable `experiment` contains the name of the CPS namelist file (without the .namelist extension) for each of the experiments. File CVPM.config should be placed in the CVPM working directory (`wdir`).

4 Preprocessing System (CPS)

CPS does everything needed in preparation for solving the numerical heat-transfer equations by CVPM. This includes establishing: (a) the limits of the model domain, (b) the location of the control-volume (CV) grid points and interfaces, (c) the material properties within each of the CVs, (d) the initial temperature at each grid point, (e) the type of boundary condition (BC) on each of the domain boundaries, and (f) the name of the file specifying the temporal and spatial dependence of the BC on each boundary. Parameters controlling the material properties are found by CPS in the user-created GEO files associated with each experiment. These files should be placed in either the '`wdir/geo`' or '`wdir/tmp`' directory, depending on how variable `Gopt` has been set in CVMP.config. The remainder of the required information for a simulation is found by CPS in a user-created namelist file (e.g., ESN_qb40.namelist) placed in the '`wdir/namelists`' directory. Below is a description of the namelist variables and sample values for a 3-D cartesian experiment. Namelist variables for the radial and 2-D cylindrical coordinate systems are completely analogous.

Namelist File, 3-D Cartesian Case (XYZ)

Variable Names	Sample Values	Description
`planet`	'earth', 'mars',	planet
`site`	'ESN',	simulation site
`coordinate_system`	'XYZ',	coordinate system
`problem_scale`	'local', 'regional'	problem scale
`min_X, max_X`	-1000, 1000,	model domain limits (unit: m)
`min_Y, max_Y`	-1500, 500,	model domain limits (unit: m)
`min_Z, max_Z`	0, 800,	model domain limits (unit: m)
`time_units`	'years' 'months' 'weeks' 'days' 'seconds'	time units

`start_time, end_time`	1970, 2015,	simulation start and end times (unit: time_units)
`computational_time_step`	0.01,	computational time step (unit: time_units)
`output_interval`	5,	output interval (unit: time_units)
`initT_opt`	1,	input initial temperature field from a file
	2,	calculate initial temp. field assuming a steady-state
	3,	use an analytic solution for the initial temp. field
`initial_condition_file`	'none'	no initial temperature file (`initT_opt` = 2 or 3)
	'ESN_1970.txt',	name of initial temperature file (`initT_opt` = 1)
`upperBC_type, upperBC_file`	'T', 'Ts_ESN_xyz.mat',	upper boundary condition type and file name
		BCtype = 'T': prescribed temperature
		BCtype = 'q': prescribed heat flux
`lowerBC_type, lowerBC_file`	'q', 'qb_40_xz.txt',	lower boundary condition type and file name
`xleftBC_type, xleftBC_file`	'q', 'qa_0_xz.txt',	X-left boundary condition type and file name
`xrightBC_type, xrightBC_file`	'q', 'qo_0_xz.txt',	X-right boundary condition type and file name
`yleftBC_type, yleftBC_file`	'q', 'qc_0_xz.txt',	Y-left boundary condition type and file name
`yrightBC_type, yrightBC_file`	'q', 'qd_0_xz.txt',	Y-right boundary condition type and file name
`source_function_opt`	'zero',	heat-production function: $S(z) = 0$
	'linear',	heat-production function: $S(z) = S_0\,(1 - z/h_s)$
	'exponential',	heat-production function: $S(z) = S_0\,\exp(-z/h_s)$
`compaction_function_opt`	'off',	compaction function: $\phi(z) = \phi_0$
	'linear',	compaction function: $\phi(z) = \phi_0\,(1 - z/h_c)$
	'exponential',	compaction function: $\phi(z) = \phi_0\,\exp(-z/h_c)$
`pressure_opt`	'off'	turn freezing-point pressure effects off
	'hydrostatic'	freezing-point pressure effects = hydrostatic
	'lithostatic'	freezing-point pressure effects = lithostatic
`solute`	'NaCl'	chemical formula of dominant pore-water solute
	'KCl'	
`implicit_factor`	0.86,	implicit/explicit factor
		0 = fully explicit, 1 = fully implicit

Variable `planet` determines the composition of any air present in the pore spaces and the gravitational acceleration used to find the freezing-point pressure effect. Variable `site` is used as a prefix for the GEO file names CPS attempts to input after reading the namelist file. In the above case where `site` = 'ESN', CPS will attempt to input the material property parameters from files 'ESN_Xlayers.ext', 'ESN_Ylayers.ext', 'ESN_Zlayers.ext' where the extension (ext) can be either 'txt' (text file) or 'mat' (MATLAB binary file). For multidimensional local-scale problems, the initial vertical temperature profile is assumed to be identical at all XY (cartesian problems) or R (cylindrical problems) locations; for regional problems, the initial vertical temperature profile can be different at each horizontal location. The natural time unit for a problem is specified by variable `time_units`. All subsequent temporal information provided in the namelist file should use these units. Thus, if `time_units` = 'years', then CVPM will use a computational time step of 0.01 years beginning in year 1970 if `computational_time_step` = 0.01 and `start_time` = 1970. How the initial temperature field is established is controlled by variable `initT_opt`. If `initT_opt` = 1, CPS will interpolate the temperature field found in `initial_condition_file` onto the control-volume grid; the initial condition file should be placed in directory '`wdir/ICs`'. If `initT_opt` = 2, CPS will find a steady-state temperature field consistent with the boundary conditions and the material

properties. This is done iteratively as many of the bulk properties derived from the material property parameters are temperature dependent. `initT_opt` = 3 is provided for simple test cases where an analytic solution exists and is known to the CVPM system. Finally, variable `implicit_factor` controls whether to solve the numerical heat-transfer equations in a fully explicit mode, fully implicit mode, or something in-between. While running CVPM in a fully implicit mode allows for larger computational time steps, setting `implicit_factor` to an intermediate value is likely to produce a more accurate solution.

GEO Files

CVPM assumes the model domain can be divided into discrete control volumes over which the lithology is relatively uniform. Thus in sedimentary terrain, the contact between different rock units (e.g., sandstones, claystones, limestones) provide the natural location for control-volume interfaces. For most simulations, higher spatial resolution is desired than is provided by the natural rock units. To accomplish this, each rock unit can be further divided into additional control volumes. An example demonstrating this process is given by the following simple GEO file in the depth (Z) dimension (*_Zlayers.txt),

```
header 1:  simple GEO file
header 2:  organic-layer, silty clay, ice lens, siltstone, sandstone

Ztop  Zbot    dz   Mtyp   Km0    rhom   cpm0    S0    hs   phi0  phic    hc  Sr    xs0  lambda   d1   d2  n21
  0,  0.4,  0.05,   20,   1.0,   2650,  780,    0,    0,  0.40, 0.20,  2.0,  1, 0.003,   0.33,   4, 0.1,   6,
0.4,    2,   0.1,   11,   1.0,   2650,  780,  1.5,   10,  0.40, 0.05,  2.0,  1, 0.003,   0.39,  10,   2, 2.6,
  2,    4,  0.25,    3,    0,      0,    0,    0,    0,     0,    0,    0,  0,     0,      0,   0,   0,   0,
  4,    8,   0.5,   11,   1.0,   2650,  780,  1.8,   10,  0.28, 0.05,  2.0,  1, 0.003,   0.36,  30,   2,   1,
  8,   16,     1,   11,   1.0,   2650,  780,  1.8,   10,  0.28, 0.05,  2.0,  1, 0.003,   0.36,  30,   2,   1,
 16,   40,     2,   10,   4.2,   2660,  740,  0.8,   10,  0.45, 0.15,  3.0,  1, 0.003,   0.36, 180,  30,   0,
```

In this case, the upper 0.4 m consists of peat (`Mtyp` = 20) which is divided into 0.05-m thick control volumes. An ice lens (`Mtyp` = 3) occurs in the 2–4 m depth range which is divided into 0.25-m thick control volumes. The parameters expected to be present in the Z-dimension GEO file are,

Variable Names	Sample Values	Description
`Ztop`	6	depth of layer top (unit: m)
`Zbot`	10	depth of layer bottom (unit: m)
`dz`	0.5	distance between CV interfaces in this layer (unit: m)
`Mtyp`	11	material type
`Km0`	1.15	thermal conductivity of mineral grains at 0°C (unit: W m^{-1} K^{-1})
`rhom`	2650	density of mineral grains (unit: kg m^{-3})
`cpm0`	780	specific heat of mineral grains at 20°C (unit: J kg^{-1} K^{-1})
`S0`	1.8	heat-production rate extrapolated to surface (unit: μW m^{-3})
`hs`	10	heat-production length scale (unit: km)
`phi0`	0.28	porosity extrapolated to surface (range: 0–1)
`phic`	0.05	critical porosity (range: 0–1)
`hc`	2.5	compaction length scale (unit: km)
`Sr`	0.8	degree of pore saturation (range: 0–1)

`xs0`	0.003	mole fraction of solutes extrapolated to zero ice ($\phi_i = 0$)
`lambda`	0.36	interfacial melting parameter (unit: μm K$^{1/3}$)
`d1`	10	effective diameter of larger mode pores (unit: μm)
`d2`	2	effective diameter of smaller mode pores (unit: μm)
`n21`	2.6	(number of small pores) / (number of large pores)

GEO files for the other dimensions are completely analogous. The material types currently available in CVPM include,

Material Types (`Mtyp`)

Code	Material
testing	
1	properties independent of temperature
2	properties are linear functions of temperature (linearized ice)
pure ice	
3	ice (I_h)
igneous/metamorphic rocks	
4	quartz dominated
5	feldspar dominated
6	mica dominated
7	pyroxene & amphibole dominated
8	olivine dominated
9	*(reserved for future use)*
sedimentary rocks and soils	
10	sandstones
11	mudrocks (shales, claystones, siltstones, clay, silt)
12	carbonates
13	cherts
14–19	*(reserved for future use)*
organic-rich materials	
20	100% peat
21	75% peat / 25% mineral mix
22	50% peat / 50% mineral mix
23	25% peat / 75% mineral mix
24–29	*(reserved for future use)*
fluids	
30	water
31	diesel fuel arctic (DFA), JetA
32	n-butyl acetate
33	Estisol 140
34	Estisol 240
35	Isopar K
36–39	*(reserved for future use)*
metals	
40	steel drill pipe
41	stainless steel
42	cast iron

43	aluminum
44	copper
45–49	*(reserved for future use)*

CPS flag

99	use parameters found in the *_Zlayers.ext file

Parameter `Mtyp` controls which functions CVPM uses to find the specific heat and thermal conductivity of the matrix materials (Clow, 2018). For non-porous materials (ice, fluids, metals), much of the information in a GEO file is unused and can be safely set to zero. Variables in this category include: `Km0, rhom, cpm0, phi0, phic, hc, Sr, xs0, lambda, d1, d2, n21` (e.g., see the ice layer in the above sample GEO file). Finally, CPS uses `Mtyp` = 99 as a flag in *_Xlayers.ext, *_Ylayers.ext, and *_Rlayers.ext files. In this case, we're telling CPS to use whatever parameters it finds in the *_Zlayers.ext file.

A multiscale pore-size fabric is common in many sedimentary rocks. To facilitate simulations in these types of materials, CVPM currently allows for the specification of either unimodal or bimodal pore-size distributions. Parameter `d1` specifies the effective diameter of the larger mode pores while `d2` is the diameter of the smaller pores, if they exist. Variable `n21` = (n2/n1) is the ratio of the number density of smaller pores (n2) to that for larger pores (n1). For a unimodal distribution, all the pores are assumed to have an effective diameter `d1` (`n21` should be set to zero).

The naming convention for the CPS output file is based on the namelist file name. Thus, if the namelist file is ESN_qb40.namelist, CPS will create an output file named ESN_qb40_cps.mat in the '`wdir/CPSout`' directory.

5 CVPM Model

Once launched, CVPM runs autonomously. Variable `output_interval` in the CPS namelist file controls how often the state of the system (temperatures, thermophysical properties, etc . . .) is stored in the CVPM output file. CVPM will report when it's reached the first few output intervals to let the user know it has started but then will run quietly in the background. Again, the naming convention for the output file is based on the namelist file name. If the namelist file is ESN_qb40.namelist, CVPM will create an output file named ESN_qb40_cvpm.mat in the '`wdir/CVPMout`' directory.

Boundary Condition Files

One boundary-condition file is required for every boundary of the model domain. For 1-D and 2-D problems, all the necessary information can be specified in a text file. However for 3-D and some 2-D problems, this strategy becomes too cumbersome. To assist with the creation of BC files in these situations, the modeling system provides two utilities, makeBC_RZ and makeBC_3D, which create BC files in MATLAB binary format. Regardless of format, a user-created boundary-condition file is expected to provide: (1) a time series of the temperature or heat-flux on the boundary over the time interval specified by the `start_time` and `end_time` variables in the CPS namelist file, (2) the time units associated with the BC time series, and (3) the interpolation method to be used by CVPM to

find BC values at times between the time series points. The time units of the BC time series need not agree with the natural time units of the problem. If the units disagree, CVPM will convert the BC time-series units to agree with `time_units`. The expected units for boundary temperatures are °C while those for heat fluxes are $\mathrm{W\,m^{-2}}$. Allowed interpolation methods include 'nearest', 'linear', and 'spline'. The boundary condition files should be placed in directory '`wdir/BCs`'.

6 Visualization Routines

Visualization routines (view_Z, view_XZ, view_XYZ, ...) read the output files produced by CVPM, perform a limited amount of post-processing, and display the results. These routines can either be launched directly, or accessed through the visualization portal CVPM_view. Since these routines already extract the information out of CVPM output files, they can serve as templates for more detailed analysis and visualization, depending on the user's objectives.

7 Examples

Several test cases are built into the CVPM package. These can be used to: (1) verify the model is working after installation or modification, (2) explore how changes in the grid spacing, computational time step, or implicit/explicit factor affect the solution accuracy, and (3) provide a template for the files required to run the CVPM model for other cases. All the required test files are included with the CVPM package.

7.1 Simple Test Cases for Non-Porous Media

Several simple test cases are available for non-porous media. For many of these cases, analytic solutions are available against which the numerical solution can be compared. Thus, these tests can be used to verify whether the general model structure and numerical implementation are working. Simple CVPM tests for non-porous media include:

Test	Description
Cartesian	
`Test1_z`	1-D, steady state, simple material with fixed properties (`Mtyp` = 1)
`Test2_z`	1-D, steady state, simple composite material (`Mtyp` = 1)
`Test3_z`	1-D, steady state, properties are linearly dependent on temperature (`Mtyp` = 2)
`Test4_z`	1-D, steady state, simple material with fixed properties (`Mtyp` = 1), exponential heat source
`Test5_z`	1-D, instantaneous 5 K warming on upper boundary, simple material (`Mtyp` = 1)
`Test6_z`	1-D, 1 K/decade warming on upper boundary, simple material (`Mtyp` = 1)
`Test6ic_z`	1-D, same as Test6_z but the initial condition is provided through a file (`initT_opt` = 1)
`Test7_z`	1-D, periodic temperature on upper boundary, simple material (`Mtyp` = 1)
`Test1_xz`	2-D, same as Test1_z but in the XZ coordinate system
`Test2_xz`	2-D, same as Test2_z but in the XZ coordinate system
`Test3_xz`	2-D, same as Test3_z but in the XZ coordinate system
`Test4_xz`	2-D, same as Test4_z but in the XZ coordinate system
`Test5_xz`	2-D, same as Test5_z but in the XZ coordinate system

`Test1_xyz`	3-D, same as Test1_z but in the XYZ coordinate system
`Test2_xyz`	3-D, same as Test2_z but in the XYZ coordinate system
`Test3_xyz`	3-D, same as Test3_z but in the XYZ coordinate system
`Test4_xyz`	3-D, same as Test4_z but in the XYZ coordinate system
`Test5_xyz`	3-D, same as Test5_z but in the XYZ coordinate system
Radial	
`TestR1_r`	1-D, steady state, simple material with fixed properties (`Mtyp` = 1)
`TestR2_r`	1-D, 30 K warming on inner boundary for 1 hour, simple material (`Mtyp` = 1)
`TestR3_r`	1-D, 30 K warming on inner boundary for 60 days, simple material (`Mtyp` = 1)
`TestR20_r`	1-D, use final temperatures from TestR2_r as initial condition, no inner boundary (`Mtyp` = 1)
`TestR30_r`	1-D, use final temperatures from TestR3_r as initial condition, no inner boundary (`Mtyp` = 1)
Cylindrical	
`Test1_rz`	2-D, same as Test1_z but in the RZ coordinate system
`Test2_rz`	2-D, same as Test2_z but in the RZ coordinate system
`Test3_rz`	2-D, same as Test3_z but in the RZ coordinate system
`Test4_rz`	2-D, same as Test4_z but in the RZ coordinate system
`Test5_rz`	2-D, same as Test5_z but in the RZ coordinate system
`TestR1b_rz`	2-D, same as TestR1_r but in the RZ coordinate system
`TestR2b_rz`	2-D, same as TestR2_r but in the RZ coordinate system
`TestR3b_rz`	2-D, same as TestR3_r but in the RZ coordinate system
`TestR20b_rz`	2-D, use final temperatures from TestR2b_rz as initial condition, non inner boundary
`TestR30b_rz`	2-D, use final temperatures from TestR3b_rz as initial condition, non inner boundary

♣ Example: `Test6_z`

In this 1-D example, the initial condition is calculated by CPS using the analytic solution for this thermal problem (`initT_opt` = 3). A 1 K/decade warming is then applied to the upper boundary for 50 years. The problem domain consists of a single material with fixed thermophysical properties. The files required to run this case include:

(1) The configuration file (CVPM.config)

```
CVPM config file

working_directory   =   '~/thermal/numer/CVPM_v1.1',
coordinate_system   =   'Z',
Gopt,Ropt,          =   1, 3,
experiment          =   'Test6_z',
```

where `working_directory` needs to be set to the correct location for the user's directory system.

(2) The namelist file (Test6_z.namelist)

```
Test6_z namelist
1 K/decade warming on the upper boundary, 1-D vertical test
simple material with fixed-properties, zero source

planet                    =  'earth',
site                      =  'Test6',
coordinate_system         =  'Z',
min_Z, max_Z              =  0, 400,
time_units                =  'years',
start_time, end_time      =  0, 50,
computational_time_step   =  0.005,
output_interval           =  5,
initT_opt                 =  3,
initial_condition_file    =  'none',
upperBC_type, _file       =  'T', 'Ts_1Kdecade.txt',
lowerBC_type, _file       =  'q', 'qb_50.txt',
source_function_opt       =  'zero',
compaction_function_opt   =  'off',
pressure_opt              =  'off',
solute                    =  'none',
implicit_explicit_factor  =  0.5,
```

(3) The GEO file (Test6_Zlayers.txt)

```
Test6 Z-layers
simple material with fixed properties

Ztop  Zbot   dz  Mtyp  K   rho    cp   S0 hs  unused ...
  0,  200, 0.5,   1,  2, 2000, 1000, 0, 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
200,  300,   1,   1,  2, 2000, 1000, 0, 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
300,  400,   2,   1,  2, 2000, 1000, 0, 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
400,  700,   5,   1,  2, 2000, 1000, 0, 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
```

(4) Upper boundary condition file (Ts_1Kdecade.txt)

```
Ts = 1K/decade warming
1-D vertical experiment

t_units        =  'years',
interp_method  =  'linear',
t, Ts          =    0, -10,
               =  100,   0,
```

(5) Lower boundary condition file (qb_50.txt)

```
qb = constant = 50 mW/m**2
1-D vertical experiment

t_units         =  'years',
interp_method   =  'linear',
t, qb           =  -60000, 50e-03,
                =       0, 50e-03,
                =   20000, 50e-03,
```

Comparing with the analytic solution, we find the maximum error in the CVPM numerical solution is less than 16 μK with the model configuration specified in the namelist and GEO files (Fig. 1).

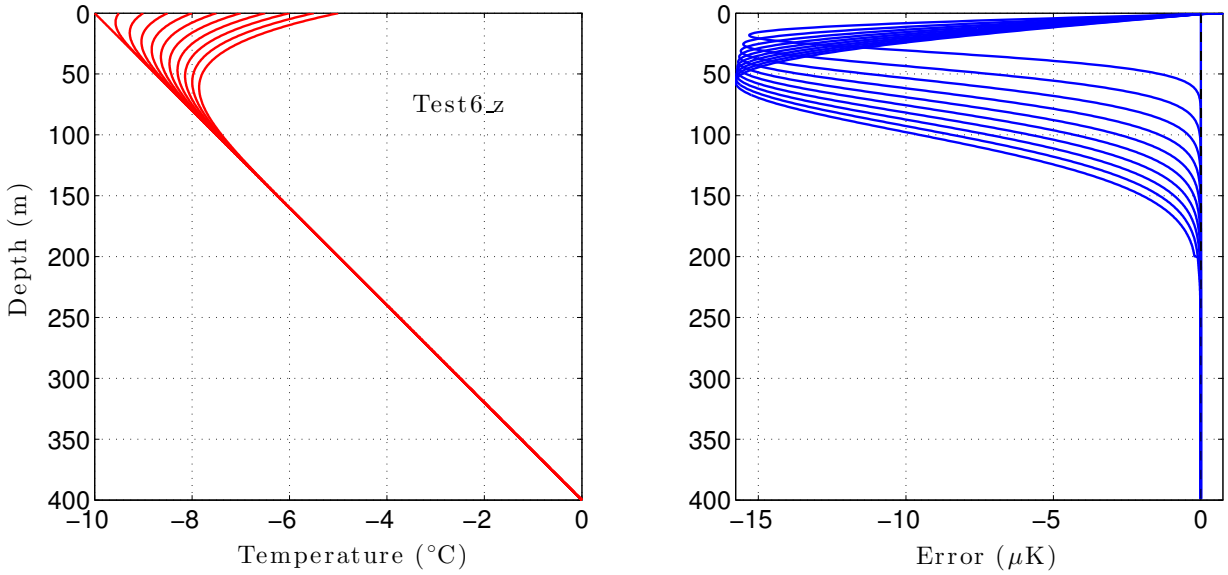


Figure 1: Temperatures predicted every 5 years over the period 0–50 years for test case `Test6_z`. Right panel shows the errors compared to the analytic solution.

♣ Example: `Test6ic_z`

This test is the same as `Test6_z` except the initial condition is provided through an input file rather than being calculated by CPS. To implement it, we provide an initial condition file and slightly modify the configuration and namelist files. The GEO and boundary condition files remain the same. The new files are:

(1) The configuration file (CVPM.config)

```
CVPM config file

working_directory   =  '~/thermal/numer/CVPM_v1.1',
coordinate_system   =  'Z',
Gopt,Ropt,          =  1, 3,
experiment          =  'Test6ic_z',
```

(2) The namelist file (Test6ic_z.namelist)

```
Test6ic_z namelist
1 K/decade warming on the upper boundary, 1-D vertical test
simple material with fixed-properties, zero source

planet                    =  'earth',
site                      =  'Test6',
coordinate_system         =  'Z',
min_Z, max_Z              =  0, 400,
time_units                =  'years',
start_time, end_time      =  0, 50,
computational_time_step   =  0.005,
output_interval           =  5,
initT_opt                 =  1,
initial_condition_file    =  'Test6_ic.txt',
upperBC_type, _file       =  'T', 'Ts_1Kdecade.txt',
lowerBC_type, _file       =  'q', 'qb_50.txt',
source_function_opt       =  'zero',
compaction_function_opt   =  'off',
pressure_opt              =  'off',
solute                    =  'none',
implicit_explicit_factor  =  0.5,
```

(3) Initial condition file (Test6_ic.txt)

```
Initial condition for Test6ic_z
1-D vertical experiment

interp_method  =     'linear',
z, T           =     0,    -10,
               =   100,   -7.5,
               =   200,   -5.0,
               =   300,   -2.5,
               =   400,      0,
```

The resulting errors are the same as for `Test6_z` (Fig. 1).

7.2 Permafrost Test Cases

Test cases are provided with the CVMP package demonstrating the full range of capabilities of the model, including the simulation of radiogenic heat production, depth-dependent compaction, and freezing-point depression due to pressure and pore-water solutes as well as to interfacial, grain-boundary, and curvature effects. For these tests, we consider the thermal response of the vertical sequence of sedimentary rocks shown in Figure 2 to changing boundary conditions.

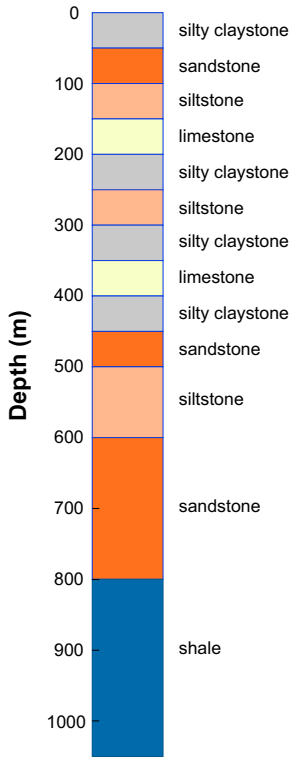


Figure 2: Vertical sequence of sedimentary rocks used for the permafrost test cases.

The GEO file for this sequence in the vertical dimension is,

GEO file, Z-dimension (sedSeq_Zlayer.txt)

```
sedSeq_Zlayer.txt:  generic sedimentary sequence
vertical layers consisting of:  limestone, shale, silty claystone, siltstone, and fine sandstone

 Ztop  Zbot  dz Mtyp  Km0  rhom cpm0   S0   hs   phi0  phic   hc  Sr   xs0  lambda  d1   d2   n21
   0,   50,  2,  11, 1.9, 2650, 780, 1.8,  10, 0.41, 0.05, 1.4, 1, 0.003,  0.39,  10,   2, 2.55,
  50,  100,  2,  10, 4.2, 2660, 740, 0.8,  10, 0.36, 0.10, 2.4, 1, 0.003,  0.36, 177,  30,    0,
 100,  150,  2,  11, 1.9, 2650, 780, 1.8,  10, 0.37, 0.05, 2.0, 1, 0.003,  0.36,  30,   2,    0,
 150,  200,  2,  12, 3.7, 2650, 780, 0.6,  10, 0.38, 0.05, 2.0, 1, 0.003,  0.39,  10,   2,    0,
 200,  250,  2,  11, 1.9, 2650, 780, 1.8,  10, 0.41, 0.05, 1.4, 1, 0.003,  0.39,  10,   2, 2.55,
 250,  300,  2,  11, 1.9, 2650, 780, 1.8,  10, 0.37, 0.05, 2.0, 1, 0.003,  0.36,  30,   2,    0,
 300,  350,  2,  11, 1.9, 2650, 780, 1.8,  10, 0.41, 0.05, 1.4, 1, 0.003,  0.39,  10,   2, 2.55,
 350,  400,  2,  12, 3.7, 2650, 780, 0.6,  10, 0.38, 0.05, 2.0, 1, 0.003,  0.39,  10,   2,    0,
 400,  450,  2,  11, 1.9, 2650, 780, 1.8,  10, 0.41, 0.05, 1.4, 1, 0.003,  0.39,  10,   2, 2.55,
 450,  500,  5,  10, 4.2, 2660, 740, 0.8,  10, 0.36, 0.10, 2.4, 1, 0.003,  0.36, 177,  30,    0,
 500,  600, 10,  11, 1.9, 2650, 780, 1.8,  10, 0.37, 0.05, 2.0, 1, 0.003,  0.36,  30,   2,    0,
 600,  800, 10,  10, 4.2, 2660, 740, 0.8,  10, 0.36, 0.10, 2.4, 1, 0.003,  0.36, 177,  30,    0,
 800, 1200, 25,  11, 1.9, 2650, 780, 1.8,  10, 0.41, 0.05, 1.4, 1, 0.003,  0.33,   2, 0.1,    1,
```

♣ 1-D Vertical Example: A warming upper boundary (sedSeq_z)

In this 1-D vertical example, the problem domain extends from the surface to the 1000-m depth. Both the heat production and compaction functions are assumed to have an exponential form while the pore pressures are hydrostatic. Sodium chloride is the dominant pore-water solute. The domain is assumed to be initially in a steady-state condition with a surface temperature $T_s = -11°\text{C}$ and a heat flux $q_b = 60\ \text{mW}\,\text{m}^{-2}$ on the lower boundary. Temperatures on the surface are then uniformly warmed at 0.75 K/decade for 100 years. Figure 3 shows the initial values for the temperature and thermophysical properties (black lines) and their values after 100 years (colored lines). Throughout the simulation, the base of permafrost is found to be located in a limestone layer at 362.4 m while the base of ice-bearing permafrost is at 337.9 m in a silty claystone. By the end of the simulation (year = 100), the warming at the surface has penetrated to a depth of $\sim$ 150 m. Within the permafrost zone, substantial volume fractions of unfrozen water ($\phi_u > 0.1$) are predicted to occur in the fine-grained silty claystone layers while low volume fractions ($\phi_u < 0.06$) occur in the coarser siltstones and sandstones. Large temperature-gradient changes with depth reflect variations in the bulk thermal conductivity due to lithology and ice content.

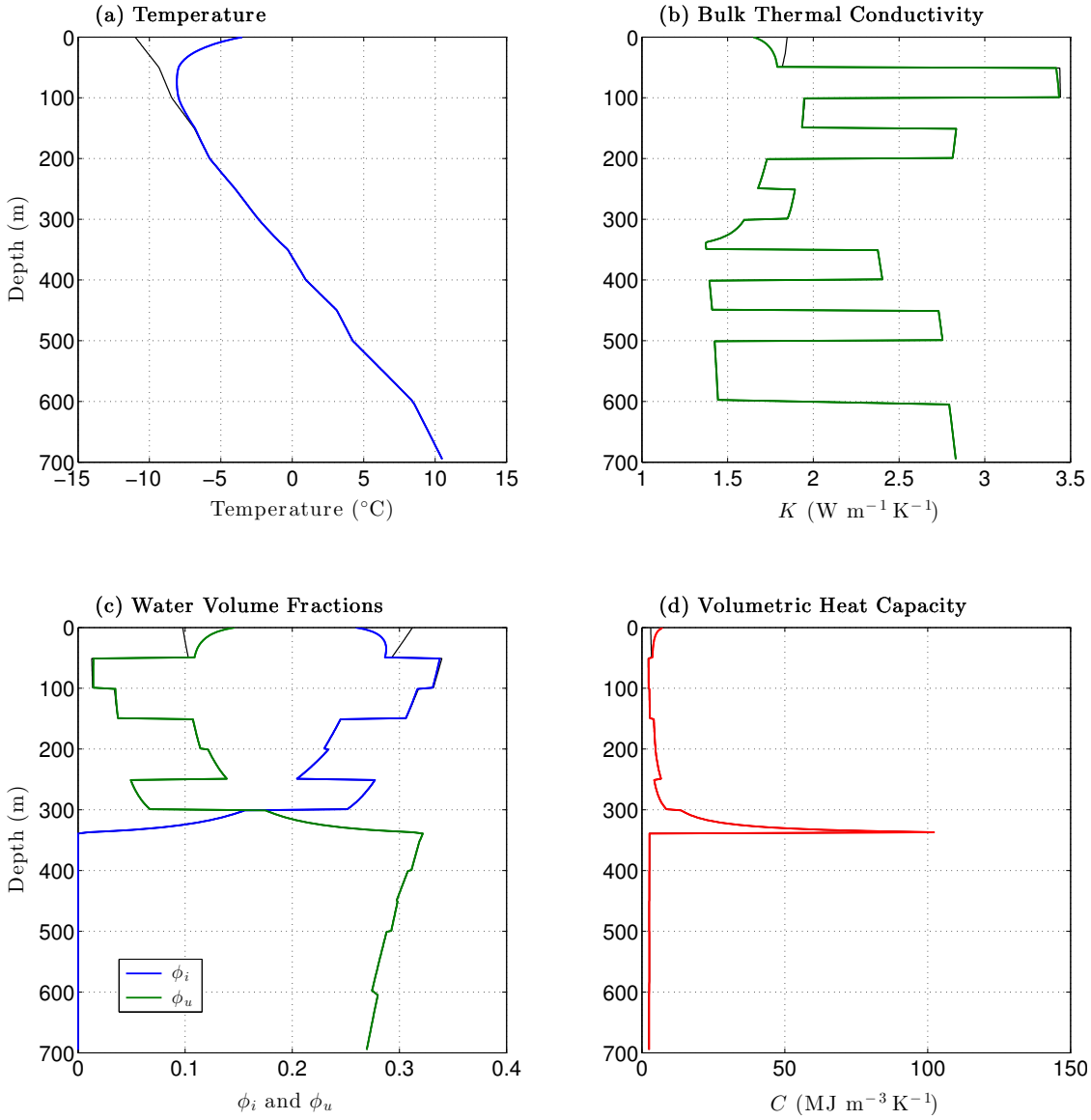


Figure 3: Temperatures and thermophysical properties for test case `sedSeq_z` after being subjected to an 0.75 K/decade warming for 100 years (colored lines). Fine black lines show the initial values. ϕ_i is the volume fraction of ice while ϕ_u is the volume fraction of unfrozen water.

In addition to the GEO file (sedSeq_Zlayer.txt), the following files are needed to implement this 1-D test case:

(1) The configuration file (CVPM.config)

```
CVPM config file

working_directory   =   '~/thermal/numer/CVPM_v1.1',
coordinate_system   =   'Z',
Gopt,Ropt,          =   1, 3,
experiment          =   'sedSeq_z',
```

(2) The namelist file (sedSeq_z.namelist)

```
sedSeq_z namelist
0.75 K/decade warming on the upper boundary, 1-D vertical test
vertical sequence of sedimentary rocks

planet                    =  'earth',
site                      =  'sedSeq',
coordinate_system         =  'Z',
min_Z, max_Z              =  0, 1000,
time_units                =  'years',
start_time, end_time      =  0, 100,
computational_time_step   =  0.25,
output_interval           =  5,
initT_opt                 =  2,
initial_condition_file    =  'none',
upperBC_type, _file       =  'T', 'Ts_11_0p75Kdecade.txt',
lowerBC_type, _file       =  'q', 'qb_60.txt',
source_function_opt       =  'exponential',
compaction_function_opt   =  'exponential',
pressure_opt              =  'hydrostatic',
solute                    =  'NaCl',
implicit_explicit_factor  =  0.85,
```

(3) Upper boundary condition file (Ts_11_0p75Kdecade.txt)

```
Ts = 0.75 K/decade warming
1-D vertical experiment

t_units        =  'years',
interp_method  =  'linear',
t, Ts          =    0, -11,
               =  100, -3.5,
```

(4) Lower boundary condition file (qb_60.txt)

```
qb = constant = 60 mW/m**2
1-D vertical experiment

t_units        =  'years',
interp_method  =  'linear',
t, qb          =  -300000, 60e-03,
               =        0, 60e-03,
               =    20000, 60e-03,
```

♣ 2-D Cylindrical Example: A warming inner boundary (`sedSeq_drillD_rz`)

In this example, we consider the drilling of a 3000-m deep, 30-cm diameter borehole through the test sedimentary sequence (Fig. 2) over a 60-day period. The associated vertical GEO file (`sedSeq_Zlayer.txt`) is the same as for the 1-D permafrost test case `sedSeq_z`. The initial condition is extracted from a previous 1-D CVPM experiment intended to simulate evolving permafrost conditions over the last two ice-age cycles. The initial condition (`sedSeq_IAC_1980_z_finalT_rz.mat`) created by utility `makeIC_RZ`, represents the final state of that simulation. To be consistent with the initial condition, the upper boundary condition is defined such that the surface temperature T_s is -8.5°C at the onset of drilling and then warms at 0.75 K/decade. Drilling fluids pumped into the hole at 30°C thermally interact with the drill pipe and surrounding rock as they circulate to the bottom of the hole and then back to the surface. As a result of drilling processes, rocks surrounding the hole warm throughout the permafrost zone. The degree of warming depends on both depth and time as the drill bit advances into the warmer rocks below (Clow, 2015). For this test, utility `makeBC_RZ` is used to create the boundary condition (`dTa_sedSeq_drillD_rz.mat`, Fig. 4) at the borehole wall ($r = 15$ cm) which is used as the inner boundary of the cylindrical problem domain. In this example, the borehole wall warms 30–40 K at shallow depths for the duration of the drilling. At 1000 m, temperatures remain undisturbed ($\Delta T_a = 0$) until the drill bit advances past this depth on day 20. After this, temperatures initially cool ~ 3 K and then warm almost 13 K by day 60. Note that unlike the other coordinate systems, a temperature condition on the inner boundary for the 2-D cylindrical coordinate case is given by the amount of warming or cooling that has occurred on the boundary since the initial time,

$$\Delta T_a \;=\; T(z,t)|^{r=a} \;-\; T(z,0)|^{r=a} \;. \tag{1}$$

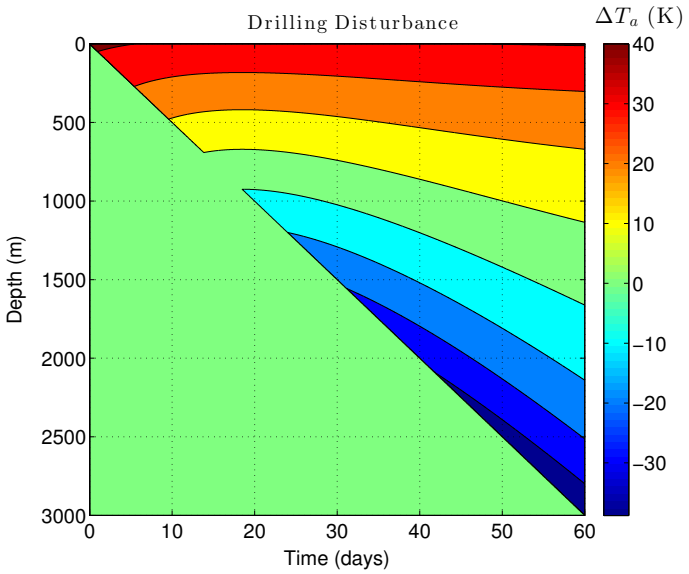


Figure 4: Boundary condition `dTa_sedSeq_drillD_rz.mat` at the borehole wall (inner boundary) used for permafrost test case `sedSeq_drillD_rz`.

For all other coordinate systems, a temperature condition on a boundary is specified by the actual temperature rather than by a temperature difference. To complete the boundary conditions for `sedSeq_drillD_rz`, the heat flux across the outer radial boundary at $r = 40$ m is assumed to be zero.

Figure 5 shows the simulated temperatures and thermophysical properties in the sedimentary sequence upon completion of drilling on day 60. As expected, the thermal drilling disturbance extends

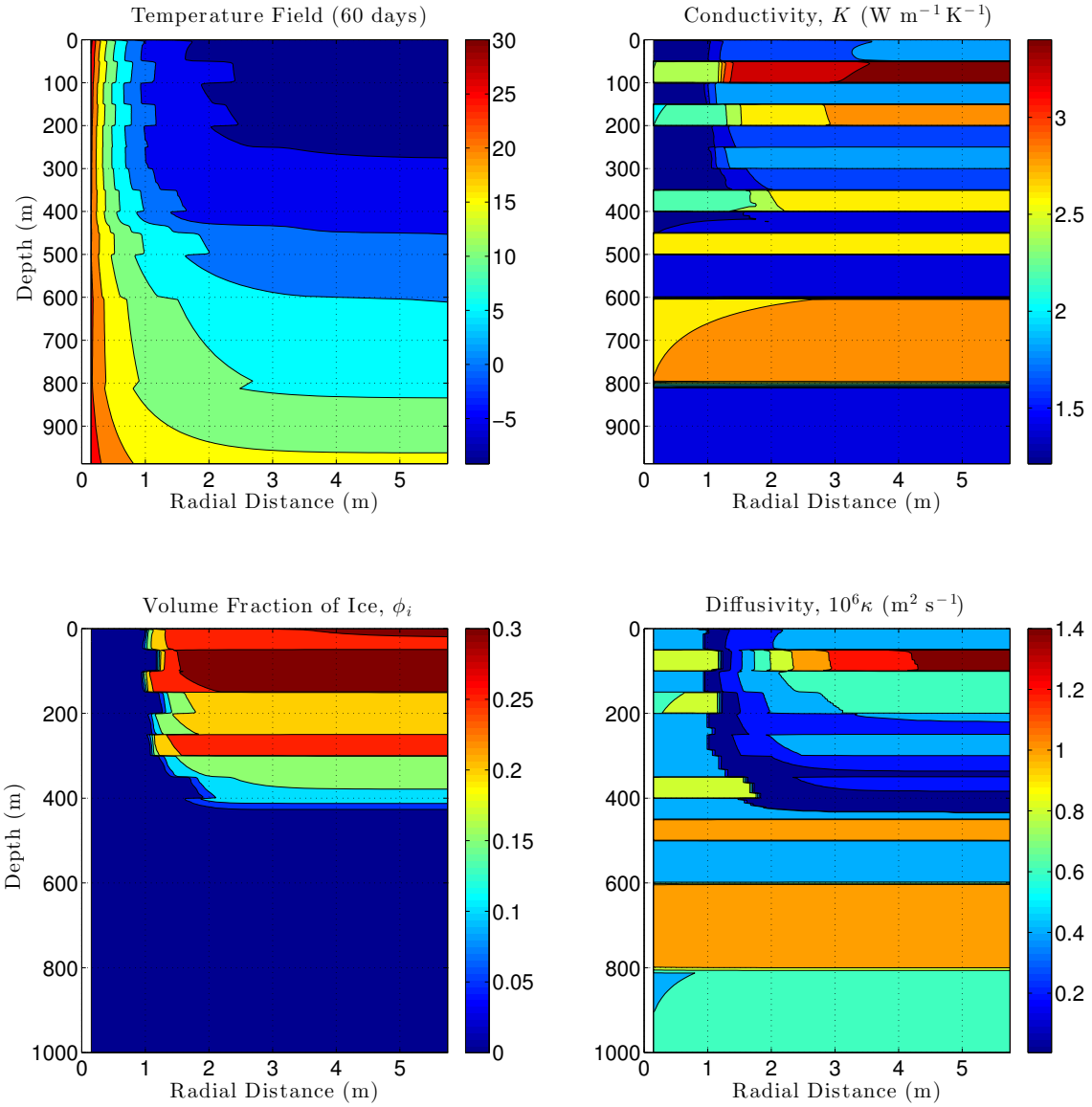


Figure 5: Simulated temperatures and thermophysical properties in the test sedimentary rock sequence (Fig. 2) upon completion of drilling a 3000-m deep, 30-cm diameter borehole, permafrost test `sedSeq_drillD_rz`. In this test, the problem domain extends from the borehole wall ($r = 15$ cm) out to $r = 40$ m where the radial heat flux is zero.

further from the hole in the higher conductivity sandstone and limestone layers than in the siltstone and claystone layers. By day 60, sufficient heat has been pumped into the permafrost to melt all the interstitial ice within 1–2 m of the hole. As a result, the thermal conductivities and diffusivities have also dropped significantly within 1–2 m of the borehole. Thermal diffusivities approach very low values in the vicinity of the pore-ice melting front due to the large volumetric heat capacities there.

In addition to the initial condition, inner boundary-condition, and vertical GEO files (sedSeq_IAC_1980_z_finalT_rz.mat. dTa_sedSeq_drillD_rz.mat, sedSeq_Zlayer.txt), the following files are needed to run the 2-D cylindrical permafrost test case:

(1) The configuration file (CVPM.config)

```
CVPM config file

working_directory   =  '~/thermal/numer/CVPM_v1.1',
coordinate_system   =  'RZ',
Gopt,Ropt,          =  1, 3,
experiment          =  'sedSeq_drillD_rz',
```

(2) The namelist file (sedSeq_drillD_rz.namelist)

```
sedseq_drillD_rz namelist
warming on inner boundary due to hot drill fluids, RZ cylindrical test
vertical sequence of sedimentary rocks

planet                    =  'earth',
site                      =  'sedSeq',
coordinate_system         =  'RZ',
problem_scale             =  'local',
borehole_depth            =  3000,
min_R, max_R              =  0.15, 40,
min_Z, max_Z              =  0, 1000,
time_units                =  'days',
start_time, end_time      =  0, 60,
computational_time_step   =  0.2,
output_interval           =  2,
initT_opt                 =  1,
initial_condition_file    =  'sedSeq_IAC_1980_z_finalT_rz.mat',
upperBC_type, _file       =  'T', 'Ts_8p5_0p75Kdecade_rz.txt',
lowerBC_type, _file       =  'q', 'qb_60_rz.txt',
innerBC_type, _file       =  'T', 'dTa_sedSeq_drillD_rz.mat',
outerBC_type, _file       =  'q', 'qo_0_rz.txt',
```

```
source_function_opt        =  'exponential',
compaction_function_opt    =  'exponential',
pressure_opt               =  'hydrostatic',
solute                     =  'NaCl',
implicit_explicit_factor   =  0.99,
```

(3) The radial GEO file (sedSeq_Rlayers.txt)

```
sedSeq R-layers
use properties found in Z-layers file

 Rmin    Rmax     dr   Mtyp   unused ...
0.15,   0.20,  0.025,   99,   0, 0, 0, 0, 0, 0,   0,   0,   0,   0,   0,   0,   0,
 0.2,     1,   0.05,   99,   0, 0, 0, 0, 0, 0,   0,   0,   0,   0,   0,   0,   0,
   1,     2,    0.1,   99,   0, 0, 0, 0, 0, 0,   0,   0,   0,   0,   0,   0,   0,
   2,     5,    0.2,   99,   0, 0, 0, 0, 0, 0,   0,   0,   0,   0,   0,   0,   0,
   5,    10,    0.5,   99,   0, 0, 0, 0, 0, 0,   0,   0,   0,   0,   0,   0,   0,
  10,    20,      1,   99,   0, 0, 0, 0, 0, 0,   0,   0,   0,   0,   0,   0,   0,
  20,    40,      2,   99,   0, 0, 0, 0, 0, 0,   0,   0,   0,   0,   0,   0,   0,
```

(4) Upper boundary condition file (Ts_8p5_0p75Kdecade_rz.txt)

```
Ts = 0.75 K/decade warming across entire surface
2-D vertical experiment

t_units         =  'years',
interp_method   =  'linear',
R               =          0,     1,    75,  1000,
t, Ts(R)        =  0,    -8.5,  -8.5,  -8.5,  -8.5,
                =  100,  -1.0,  -1.0,  -1.0,  -1.0,
```

(5) Lower boundary condition file (qb_60_rz.txt)

```
qb = constant = 60 mW/m**2
2-D vertical experiment

t_units         =  'years',
interp_method   =  'linear',
R               =            0,       1,      75,    1000,
t, Ts(R)        =  0,    60e-03,  60e-03,  60e-03,  60e-03,
                =  100,  60e-03,  60e-03,  60e-03,  60e-03,
```

(6) Outer boundary condition file (qo_0_rz.txt)

```
qo = constant = 0 mW/m**2
2-D vertical experiment

t_units        =  'years',
interp_method  =  'linear',
Z              =        0,   500,   10000,
t, Ts(Z)       =  0,    0,     0,      0,
               =  100,  0,     0,      0,
```

References

Anderson, D.A., Tannehill, J.C., and Pletcher, R.H.: Computational Fluid Mechanics and Heat Transfer. Hemisphere Publishing Corp., New York, 1984.

Clow, G.D.: A Green's function approach for assessing the thermal disturbance caused by drilling deep boreholes in rock or ice, *Geophys. J. Int.*, 203, 1877–1895, https://doi.org/10.1093/gji/ggv415, 2015.

Clow, G.D.: CVPM 1.1: a flexible heat-transfer modeling system for permafrost, *Geosci. Model Dev.*, 11, 4889–4908, https://doi.org/10.5194/gmd-11-4889-2018, 2018.

Minkowycz, W.J., Sparrow, E.M., Schneider, G.E., and Pletcher, R.H.: Handbook of Numerical Heat Transfer. John Wiley & Sons, Inc., New York, 1988.

Patankar, S.V.: Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corp., New York, 1980.