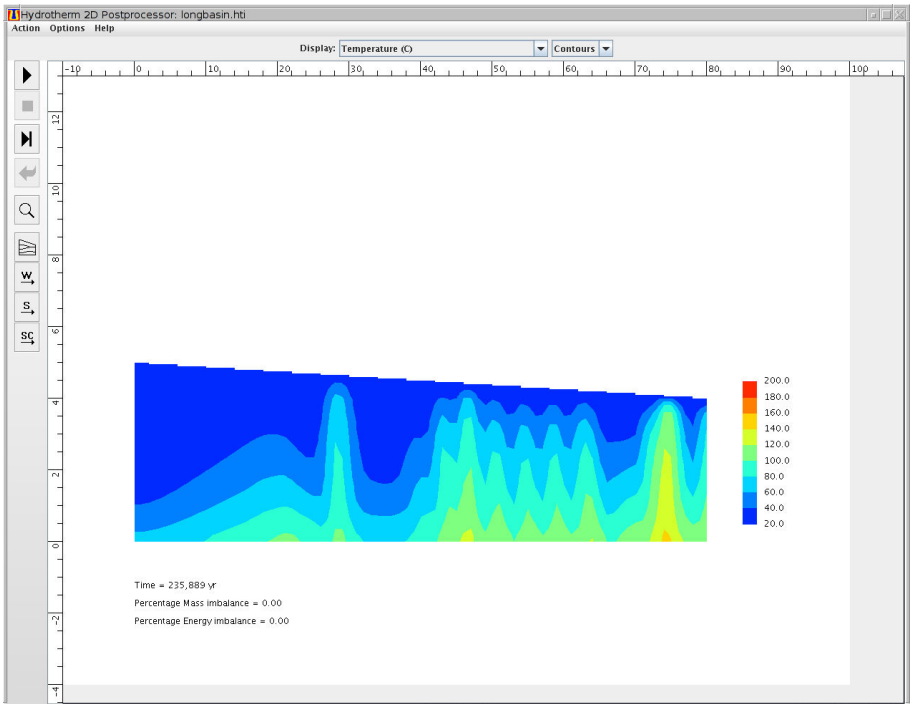
USGS
science for a changing world

Guide to the Revised Ground-Water Flow and Heat Transport Simulator: HYDROTHERM — Version 3



Techniques and Methods 6–A25

U.S. Department of the Interior
U.S. Geological Survey

Cover illustration: Temperature field in a long basin with high permeability resulting from advection and free convection in the last example of section 8.2.

Guide to the Revised Ground-Water Flow and Heat Transport Simulator: HYDROTHERM — Version 3

By Kenneth L. Kipp, Jr., Paul A. Hsieh, and Scott R. Charlton

Techniques and Methods 6–A25

U.S. Department of the Interior
U.S. Geological Survey

U.S. Department of the Interior
DIRK KEMPTHORNE, Secretary

U.S. Geological Survey
Mark D. Myers, Director

U.S. Geological Survey, Reston, Virginia 2008

For product and ordering information:
World Wide Web: http://www.usgs.gov/pubprod
Telephone: 1-888-ASK-USGS

For more information on the USGS—the Federal source for science about the Earth,
its natural and living resources, natural hazards, and the environment:
World Wide Web: http://www.usgs.gov
Telephone: 1-888-ASK-USGS

Suggested citation:
Kipp, K.L., Jr., Hsieh, P.A., and Charlton, S.R., 2008, Guide to the revised ground-water flow and heat transport simulator: HYDROTHERM — Version 3: U.S. Geological Survey Techniques and Methods 6–A25, 160 p.

Contents

Figures

Tables

Conversion Factors

SI and centimeter-gram-second to inch-pound

Multiply	By	To obtain
	Length	
millimeter (mm)	0.03937	inch (in.)
centimeter (cm)	0.3937	inch (in.)
meter (m)	3.281	foot (ft)
meter (m)	1.094	yard (yd)
kilometer (km)	0.6214	mile (mi)
kilometer (km)	0.5400	mile, nautical (nmi)
	Area	
square centimeter (cm^2)	0.001076	square foot (ft^2)
square meter (m^2)	10.76	square foot (ft^2)
square meter (m^2)	0.0002471	acre
square centimeter (cm^2)	0.1550	square inch (ft^2)
square kilometer (km^2)	0.3861	square mile (mi^2)
square kilometer (km^2)	247.1	acre
	Volume	
cubic centimeter (cm^3)	0.06102	cubic inch (in^3)
cubic meter (m^3)	35.31	cubic foot (ft^3)
cubic meter (m^3)	264.2	gallon (gal)
cubic meter (m^3)	6.290	barrel (petroleum, 1 barrel = 42 gal)
cubic meter (m^3)	0.0002642	million gallons (Mgal)
cubic meter (m^3)	1.308	cubic yard (yd^3)
cubic meter (m^3)	0.0008107	acre-foot (acre-ft)
cubic kilometer (km^3)	0.2399	cubic mile (mi^3)
	Velocity	
centimeter per second (cm/s)	3.281×10^{-2}	foot per second (ft/s)
centimeter per second (cm/s)	2.835×10^{3}	foot per day (ft/d)
centimeter per year (cm/yr)	3.281×10^{-2}	foot per year (ft/yr)
meter per second (m/s)	3.281	foot per second (ft/s)
meter per second (m/s)	2.835×10^{5}	foot per day (ft/d)
meter per year (m/yr)	3.281	foot per year (ft/yr)
	Flow rate (volumetric)	
cubic centimeter per second (cm^3/s)	3.531×10^{-5}	cubic foot per second (ft^3/s)
cubic centimeter per second (cm^3/s)	3.051	cubic foot per day (ft^3/d)
cubic meter per second (m^3/s)	35.31	cubic foot per second (ft^3/s)
cubic meter per second (m^3/s)	3.0512×10^{6}	cubic foot per day (ft^3/d)
cubic meter per second (m^3/s)	70.07	acre-foot per day (acre-ft/d)
cubic meter per second (m^3/s)	22.83	million gallons per day (Mgal/d)
cubic meter per year (m^3/yr)	0.000811	acre-foot per year (acre-ft/yr)
	Mass	
gram (g)	0.03527	ounce, avoirdupois (oz)
kilogram (kg)	2.205	pound, avoirdupois (lb)
	Flow rate (mass)	
gram per second (g/s)	1.905×10^{2}	pound per day (lb/d)

Multiply	By	To obtain
kilogram per second (kg/s)	1.905×10^{5}	pound per day (lb/d)
gram per second per cubic centimeter [(g/s)/cm^3]	6.243×10^{-4}	pound per second per cubic foot [(lb/s)/ft^3]
kilogram per second per cubic meter [(kg/s)/m^3]	6.243×10^{-2}	pound per second per cubic foot [(lb/s)/ft^3]
Volumetric flux		
millimeter per year (mm/yr)	3.281×10^{-3}	foot per year (ft/yr)
cubic centimeter per second per square centimeter [(cm^3/s)/cm^2]	2.835×10^{3}	cubic foot per day per square foot [(ft^3/d)/ft^2]
cubic meter per second per square meter [(m^3/s)/m^2]	2.835×10^{5}	cubic foot per day per square foot [(ft^3/d)/ft^2]
cubic meter per second per square kilometer [(m^3/s)/km^2]	91.49	cubic foot per second per square mile [(ft^3/s)/mi^2]
cubic meter per day per square kilometer [(m^3/d)/km^2]	0.0006844	million gallon per day per square mile [(Mgal/d)/mi^2]
meter per year (m/yr)	3.281	foot per year (ft/yr)
Mass flux		
gram per second per square centimeter [(g/s)/cm^2]	1.770×10^{3}	pound per day per square foot [(lb/d)/ft^2]
kilogram per second per square meter [(kg/s)/m^2]	1.770×10^{4}	pound per day per square foot [(lb/d)/ft^2]
Pressure		
dyne per square centimeter (dyne/cm^2)	1.450×10^{-5}	pound per square inch (lb/in^2)
dyne per square centimeter (dyne/cm^2)	2.089×10^{-3}	pound per square foot (lb/ft^2)
pascal (Pa)	1.450×10^{-4}	pound per square inch (lb/in^2)
pascal (Pa)	2.089×10^{-2}	pound per square foot (lb/ft^2)
kilopascal (kPa)	0.009869	atmosphere, standard (atm)
kilopascal (kPa)	0.01	bar
kilopascal (kPa)	0.2961	inch of mercury at 60°F (in Hg)
kilopascal (kPa)	20.89	pound per square foot (lb/ft^2)
kilopascal (kPa)	0.1450	pound per square inch (lb/in^2)
megapascal (MPa)	10.0	bar
Density		
gram per cubic centimeter (g/cm^3)	62.4220	pound per cubic foot (lb/ft^3)
kilogram per cubic meter (kg/m^3)	0.06242	pound per cubic foot (lb/ft^3)
Viscosity		
gram per centimeter per second (g/cm-s)	100	centipoise (cP)
pascal-second (Pa-s)	1,000	centipoise (cP)
Energy		
erg (erg)	9.481×10^{-11}	British Thermal Unit (BTU)
joule (J)	9.481×10^{-4}	British Thermal Unit (BTU)
erg per second (erg/s)	3.412×10^{-7}	British Thermal Unit per hour (BTU/hr)
watt (W)	3.412	British Thermal Unit per hour (BTU/hr)
megawatt (MW)	3.412×10^{6}	British Thermal Unit per hour (BTU/hr)
erg per gram (erg/g)	4.299×10^{-8}	British Thermal Unit per pound (BTU/lb)
joule per kilogram (J/kg)	4.299×10^{-4}	British Thermal Unit per pound (BTU/lb)
kilojoule per kilogram (kJ/kg)	4.299×10^{-1}	British Thermal Unit per pound (BTU/lb)
Heat flux		
milliwatt per square meter (mW/m^2)	3.170×10^{-4}	British Thermal Unit per hour per square foot [(BTU/hr)/ft^2]
erg per second per square centimeter [(erg/s)/cm^2]	3.170×10^{-4}	British Thermal Unit per hour per square foot [(BTU/hr)/ft^2]

Multiply	By	To obtain
watt per square meter (W/m^2)	0.3170	British Thermal Unit per hour per square foot [$(BTU/hr)/ft^2$]
erg per second per cubic centimeter [$(erg/s)/cm^3$]	1.610×10^{-4}	British Thermal Unit per hour per cubic foot [$(BTU/hr)/ft^3$]
watt per cubic meter (W/m^3)	1.610×10^{-3}	British Thermal Unit per hour per cubic foot [$(BTU/hr)/ft^3$]
Permeability		
square centimeter (cm^2)	1.076×10^{-3}	square foot (ft^2)
square meter (m^2)	10.76	square foot (ft^2)
Thermal conductivity		
erg per second per centimeter per degree Celsius [{(erg/s)/cm}/°C]	5.778×10^{-6}	British Thermal Unit per hour per foot per degree Fahrenheit [{(BTU/hr)/ft}/°F]
erg per second per centimeter per kelvin [{(erg/s)/cm}/K]	5.778×10^{-6}	British Thermal Unit per hour per foot per degree Fahrenheit [{(BTU/hr)/ft}/°F]
watt per meter per degree Celsius [(W/m)/°C]	0.5778	British Thermal Unit per hour per foot per degree Fahrenheit [{(BTU/hr)/ft}/°F]
watt per meter per kelvin [(W/m)/K]	0.5778	British Thermal Unit per hour per foot per degree Fahrenheit [{(BTU/hr)/ft}/°F]
Heat capacity		
erg per gram per degree Celsius [(erg/g)/°C]	2.388×10^{-8}	British Thermal Unit per pound per degree Fahrenheit [(BTU/lb)/°F]
erg per gram per kelvin [(erg/g)/K]	2.388×10^{-8}	British Thermal Unit per pound per degree Fahrenheit [(BTU/lb)/°F]
erg per cubic centimeter per degree Celsius [(erg/cm^3)/°C]	1.491×10^{-6}	British Thermal Unit per cubic foot per degree Fahrenheit [(BTU/ft^3)/°F]
joule per kilogram per degree Celsius [(J/kg)/°C]	2.388×10^{-4}	British Thermal Unit per pound per degree Fahrenheit [(BTU/lb)/°F]
joule per kilogram per kelvin [(J/kg)/K]	2.388×10^{-4}	British Thermal Unit per pound per degree Fahrenheit [(BTU/lb)/°F]
joule per cubic meter per degree Celsius [(J/m^3)/°C]	1.491×10^{-5}	British Thermal Unit per cubic foot per degree Fahrenheit [(BTU/ft^3)/°F]
kilojoule per kilogram per kelvin [(kJ/kg)/K]	2.388×10^{-1}	British Thermal Unit per pound per degree Fahrenheit [(BTU/lb)/°F]
Diffusivity		
square centimeter per second (cm^2/s)	1.076×10^{-3}	square foot per second (ft^2/s)
square meter per second (m^2/s)	10.76	square foot per second (ft^2/s)

Temperature in degrees Celsius (°C) may be converted to degrees Fahrenheit (°F) as follows:
$°F = (1.8 \times °C) + 32$
Temperature in Kelvin (K) may be converted to degrees Fahrenheit (°F) as follows:
$°F = (1.8 \times K) - 459.67$

NOTATION

Roman Characters

$\mathbf{A}$	2×2 block matrix of size $N_a \times N_a$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)]
$\mathbf{A}_{ij}$	2×2 block matrix of size $N_x \times N_z$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)]
A_h^1	part of regional boundary with a specified enthalpy boundary condition
A_h^2	part of bottom regional boundary with a heat-flux boundary condition
A_p^1	part of regional boundary with a specified pressure boundary condition
A_p^2	part of regional boundary with a precipitation-recharge boundary condition
A_T^1	part of regional boundary with a specified temperature boundary condition
A_{Pm}	surface area of the precipitation boundary (land surface) for cell m (m^2)
A_{HCm}	surface area of the basal heat flux boundary for cell m (m^2)
a_i	empirical coefficients in eqs. 2.2.1−2.2.3 for permeability as a function of depth; i=0,1 [(m^2) or (m) or (dimensionless)]
a_m	fluid or thermal capacitance coefficient for cell m (first term of eq. 3.1.2 or 3.1.4) [(m-s) or (kg/s)]
a_{ij}	element of matrix **A** [(m-s) or (kg^2/J-s) or (W/Pa) or (kg/s)]
$\mathbf{B}$	2×2 block matrix of size $N_a \times N_a$ that approximates **A** [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)]
$\mathbf{b}$	right-hand-side 2×1 block vector of the difference equations of length N_a [element units: (kg/s), (W)]
$b_i^{(n)}$	empirical coefficients (i=0,1) in eq. 2.2.4 for time interval t_{n-1} to t_n [(m^2) or (m^2/s)]
Cr	cell Courant number, $Cr = q\Delta t/\Delta x$ (dimensionless)
c_i	empirical coefficients (i=0,1,2) in eq. 2.2.12a for water saturation function and eqs. 2.2.15−16 for properties as temperature functions [(dimensionless) or ($°C^{-1}$)]
$\mathbf{D}$	2×2 block matrix of size $N \times N$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)]
$\mathbf{D}_i$	2×2 block matrix of size $N_x \times N_z$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)]
D_{ns}	numerical dispersion coefficient from spatial discretization (m^2/s)
D_{nt}	numerical dispersion coefficient from temporal discretization (m^2/s)
d	coordinate direction; x, y, or z
d	numerical data item (appropriate units)
$\mathbf{e}_\nu$	error vector of the solution at iteration ν [element units: (Pa), (J/kg)]
$\hat{\mathbf{e}}_z$	unit vector in the z-coordinate direction (dimensionless)
F	discretized spatial derivative terms and source terms from eq. 3.1.2 or 3.1.4 [(kg/s) or (W)]
F_{ij}	level of fill (integer) of matrix element a_{ij} during factorization
$\mathbf{f}$	vector of residual rates of the nonlinear finite-difference equations from eqs. 2.1.1 and 2.1.4 [element units: (kg/s), (W)]
f	factor for maximum increase of the time step (dimensionless)
g	gravitational constant (m/s^2)
h	specific enthalpy of the fluid phase (J/kg)

h_B specific enthalpy at a specified value boundary (J/kg)
h_P specific enthalpy of the precipitation (J/kg)
h_{Pm} specific enthalpy of the precipitation for cell m (J/kg)
h_m specific enthalpy of the fluid phase for cell m (J/kg)
h_r specific enthalpy of the porous-matrix solid phase (rock or sediment) (J/kg)
h^0 initial specific enthalpy distribution (J/kg)
$h_{p,ups}$ enthalpy of phase p in the upstream cell (J/kg)
$h_{p,dns}$ enthalpy of phase p in the downstream cell (J/kg)
$\mathbf{I}$ identity matrix of rank 3 (dimensionless)
i node index in the x-coordinate direction
$\mathbf{J}$ Jacobian 2×2 block matrix of size $N_a \times N_a$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)]
j node index in the y-coordinate direction
K_a effective thermal conductivity of the bulk porous medium (combined liquid, gas, and solid phases) (W/m-°C)
K_{am} is the effective thermal conductivity of the bulk porous medium for cell m (W/m-°C)
K_ν Krylov vector subspace at iteration ν
$\mathbf{k}$ porous-medium permeability tensor (m^2)
k node index in the z-coordinate direction
k_L layer index of the bottom segment of a well screen
k_U layer index of the top segment of a well screen
k_i permeability in coordinate direction i (m^2)
k_r relative permeability (dimensionless)
k_{rw} relative permeability to liquid water (dimensionless)
k_{rs} relative permeability to water vapor (dimensionless)
k_w effective permeability at a well bore (m^2)
$k_{rp,ups}$ relative permeability to phase p of the upstream cell (dimensionless)
$k_{rp,dns}$ relative permeability to phase p of the downstream cell (dimensionless)
$\mathbf{L}$ 2×2 block lower triangular matrix of size $N \times N$ (appropriate units)
$\mathbf{L}_{ij}$ diagonal 2×2 block matrix of size $N_x \times N_z$ (appropriate units)
L characteristic length scale parameter (m)
L_d characteristic length scale in direction d (m)
L_k length of well bore at layer index k (m)
l distance along well bore (m)
l_L z-coordinate of lower end of screened interval of well (m)
l_U z-coordinate of upper end of screened interval of well (m)
$\mathbf{M}$ 2×2 block matrix of size $N \times N$ that is a preconditioning matrix for $\mathbf{A}$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)]
N number of nodes in the global mesh ($= N_x \times N_y \times N_z$)
N_a number of active nodes in the global mesh

N_d number of nodes in the d-direction of the global mesh
N_x number of nodes in the x-direction of the global mesh
N_y number of nodes in the y-direction of the global mesh
N_z number of nodes in the z-direction of the global mesh
Nu Nusselt number (dimensionless)
Nu_m Nusselt number for cell m (dimensionless)
n index of time level
n_w outward normal to a well bore (dimensionless)
n repeat count for numerical data
Pe thermal Peclet number (dimensionless)
Pe_c cell thermal Peclet number (dimensionless)
Pe_d thermal Peclet number in coordinate direction d (dimensionless)
Pe_{dm} thermal Peclet number in coordinate direction d for cell m (dimensionless),
p fluid pressure in the liquid phase (Pa)
p_B fluid pressure at a specified value boundary (Pa)
p_c capillary pressure (Pa)
p_{cb} capillary pressure at the bubble point in the liquid phase (Pa)
p_{cr} capillary pressure at residual water saturation (Pa)
p_g fluid pressure in the gas phase (Pa)
p_m fluid pressure at node m (Pa)
p^0 initial fluid pressure distribution (Pa)
Q_w volumetric total production flow rate for a well (m^3/s)
$\mathbf{Q}_v$ $N \times v$ matrix of orthonormal basis vectors for the Krylov subspace, K_v,
Q_{HCm} basal heat flow rate for cell m (W)
Q_{HPm} heat flow rate at a precipitation boundary for cell m (W)
Q_{Pm} volumetric flow rate at a precipitation boundary for cell m (m^3/s)
Q_{wp} volumetric well flow rate for phase p; p=w(water) or s(steam) (m^3/s)
Q_{ws} volumetric production flow rate of steam for a well (m^3/s)
Q_{ww} volumetric production flow rate of water for a well (m^3/s)
Q_{wpi} volumetric well flow rate of phase p for cell at layer index i; p=w(water) or s(steam) (m^3/s)
$\mathbf{q}$ Darcy velocity vector (m/s)
q Darcy velocity (m/s)
q_d Darcy velocity in coordinate direction d (m/s)
q_{dm} Darcy velocity in coordinate direction d for cell m (m/s).
q_F fluid flux at a precipitation-recharge boundary (m^3/m^2-s)
q_{FP} specified fluid flux from precipitation (m^3/m^2-s)
q_H heat flux at a boundary (W/m^2)
q_{HC} conductive heat flux across a bottom boundary (W/m^2)
q_{HCm} specified basal conductive heat flux for cell m (W/m^2)

z	elevation coordinate (m)
z_d	depth below the land surface (m)

Greek Characters

α_b	bulk vertical compressibility of the porous medium (Pa^{-1})
κ_a	effective thermal diffusivity of the bulk porous medium (combined liquid, gas, and solid phases) (m^2/s)
μ	fluid viscosity (Pa-s)
μ_s	fluid viscosity of the vapor phase (steam) (Pa-s)
μ_w	fluid viscosity of the liquid phase (water) (Pa-s)
ν	iteration counter
ρ	fluid density (kg/m^3)
ρ_m	fluid density for cell m (kg/m^3)
ρ_P	density of the precipitation (kg/m^3)
ρ_{Pm}	density of the precipitation for cell m (kg/m^3)
ρ_r	density of the porous-matrix solid phase (rock or sediment) (kg/m^3)
ρ_s	fluid density of the vapor phase (steam) (kg/m^3)
ρ_w	fluid density of the liquid phase (water) (kg/m^3)
$(\rho c)_a$	effective heat capacity per unit volume of bulk porous medium (combined liquid, gas, and solid phases) (J/m^3-°C)
$(\rho \mathbf{q})_m$	fluid mass flux for cell m ($kg/s\text{-}m^2$)
τ_{ILU}	drop tolerance for ILUT factorization (dimensionless)
τ_{SOR}	tolerance for convergence of the SOR linear equation solver (dimensionless)
τ_{GMRES}	tolerance for convergence of the GMRES linear equation solver (dimensionless)
Φ_m	hydraulic pressure at node m (Pa)
ϕ	porosity (dimensionless)
ϕ^0	porosity distribution at initial fluid pressure (dimensionless)
σ	spatial weighting factor (dimensionless)
ω	over-relaxation factor (dimensionless)

Mathematical Operators and Special Functions

∇	spatial gradient (m^{-1})
Δd_m	cell dimension in coordinate direction d for cell m (m),
Δt	current time step (s)
Δt_0	previous time step (s)
$\lvert \Delta u_{\max} \rvert$	magnitude of the maximum change calculated in u over the previous time step [(Pa) or (J/kg) or (dimensionless)]
$\Delta u^s_{\max}$	specified maximum change in u [(Pa) or (J/kg) or (dimensionless)]
Δx	cell size (m)

q_{HP}	heat flux across a precipitation-recharge boundary (W/m^2)
q_{Pm}	volumetric flux across a precipitation-recharge boundary for cell m ($m^3/s\text{-}m^2$)
q_{sf}	flow-rate intensity of a fluid-mass source (positive is into the region) ($kg/s\text{-}m^3$)
q_{sh}	flow-rate intensity of an enthalpy source (positive is into the region) (W/m^3)
q_{wp}	volumetric well flow rate per length of well bore for phase p; p=w(water) or s(steam) ($m^3/s\text{-}m$)
$\mathbf{r}$	residual vector [element units: (kg/s), (W)]
$\mathbf{r}_\nu$	current residual vector at iteration ν [element units: (kg/s), (W)]
S_e	scaled liquid-phase saturation (dimensionless)
S_g	saturation of the gas phase (steam or air) (dimensionless).
S_p	saturation of water in phase p; p=w(water) or s(steam) (dimensionless)
S_{sr}	residual saturation of water vapor (steam) (dimensionless)
S_w	saturation of the liquid phase (water) (dimensionless)
S_{wr}	residual saturation of liquid water (dimensionless)
T	temperature (°C)
T_B	temperature at a specified value boundary (°C)
T_P	temperature of the precipitation (°C)
T_k	temperature values at the ends of the intervals of constant temperature; k=1,2,3,4 (°C)
T^0	initial temperature distribution (°C)
$\tilde{T}_{cd}$	thermal conductance in coordinate direction d (W/°C)
$\tilde{T}_{pd}$	fluid transmissivity for phase p in coordinate direction d; p=w(water) or s(steam) (m-s)
t	time (s)
$\mathbf{U}$	2×2 block upper triangular matrix of size $N \times N$ (appropriate units)
$\mathbf{U}_{ij}$	diagonal 2×2 block matrix of size $N_x \times N_z$ (appropriate units)
$\mathbf{u}$	2×1 block vector of unknowns (pressure, enthalpy) of length N_a [element units: (Pa), (J/kg)]
$\mathbf{u}_{GS}$	solution vector using Gauss-Seidel algorithm [element units: (Pa), (J/kg)]
$\mathbf{u}^{(\nu)}$	vector of pressure and enthalpy values at each active node at iteration ν [element units: (Pa), (J/kg)]
u	pressure, enthalpy, or liquid saturation [(Pa) or (J/kg) or (dimensionless)]
u_m	pressure or enthalpy for cell m [(Pa) or (J/kg)]
u_m^0	initial value of pressure or enthalpy for cell m [(Pa) or (J/kg)].
u_{pm}	porous medium property (appropriate units)
$u_{pm}^{(k)}$	porous-medium property value over the temperature interval k (appropriate units)
V	simulation region
$\mathbf{v}_p$	interstitial-velocity vector for water in phase p; p=w(water) or s(steam) (m/s)
v	interstitial velocity (m/s)
$\mathbf{w}$	intermediate vector of unknowns [element units: (Pa), (J/kg)]
$\mathbf{x}$	spatial location on the boundary (m)
$\mathbf{y}_\nu$	vector chosen to minimize the Euclidean norm of current residual vector, $\mathbf{r}_\nu$ [element units: (kg/s), (W)]

Δx_i	cell dimension in coordinate direction x for index i (m)
$\|u\|$	magnitude of u [(Pa) or (J/kg) or (dimensionless)]
$\|\|u\|\|_2$	Euclidean norm of u [(Pa) or (J/kg) or (dimensionless)]
$\|\nabla T\|_m$	magnitude of the temperature gradient for cell m (°C/m)
$(\rho c)_a$	effective heat capacity per unit volume of bulk porous medium (combined liquid, gas, and solid phases) (J/m^3-°C)

Guide to the Revised Ground-Water Flow and Heat Transport Simulator: HYDROTHERM — Version 3

By Kenneth L. Kipp, Jr., Paul A. Hsieh, and Scott R. Charlton

ABSTRACT

The HYDROTHERM computer program simulates multi-phase ground-water flow and associated thermal energy transport in three dimensions. It can handle high fluid pressures, up to 1×10^9 pascals (10^4 atmospheres), and high temperatures, up to 1,200 degrees Celsius. This report documents the release of Version 3, which includes various additions, modifications, and corrections that have been made to the original simulator. Primary changes to the simulator include: (1) the ability to simulate unconfined ground-water flow, (2) a precipitation-recharge boundary condition, (3) a seepage-surface boundary condition at the land surface, (4) the removal of the limitation that a specified-pressure boundary also have a specified temperature, (5) a new iterative solver for the linear equations based on a generalized minimum-residual method, (6) the ability to use time- or depth-dependent functions for permeability, (7) the conversion of the program code to Fortran 90 to employ dynamic allocation of arrays, and (8) the incorporation of a graphical user interface (GUI) for input and output.

The graphical user interface has been developed for defining a simulation, running the HYDROTHERM simulator interactively, and displaying the results. The combination of the graphical user interface and the HYDROTHERM simulator forms the HYDROTHERM INTERACTIVE (HTI) program. HTI can be used for two-dimensional simulations only.

New features in Version 3 of the HYDROTHERM simulator have been verified using four test problems. Three problems come from the published literature and one problem was simulated by another partially saturated flow and thermal transport simulator. The test problems include: transient partially saturated vertical infiltration, transient one-dimensional horizontal infiltration, two-dimensional steady-state drainage with a seepage surface, and two-dimensional drainage with coupled heat transport.

An example application to a hypothetical stratovolcano system with unconfined ground-water flow is presented in detail. It illustrates the use of HTI with the combination precipitation-recharge and seepage-surface boundary condition, and functions as a tutorial example problem for the new user.

1. OVERVIEW OF THE SIMULATOR

The HYDROTHERM computer program described in this report simulates thermal energy transport in three-dimensional, two-phase, hydrothermal, ground-water flow systems. It can handle fluid temperatures up to 1,200 degrees Celsius (°C) and fluid pressures up to 1×10^9 pascals (Pa) [10^4 atmospheres (atm)]. Both confined and unconfined ground-water flow conditions can be represented, with unconfined flow including unsaturated-zone water flow with uniform atmospheric pressure in the soil air phase. HYDROTHERM simulates ground-water flow of only a single fluid component; pure water. The governing partial differential equations, which are solved numerically, are (1) the water-component flow equation, formed from the combination of the conservation of mass in the liquid and gas phases with Darcy's law for flow in porous media; and (2) the thermal-energy transport equation, formed from the conservation of enthalpy for the water component and the porous medium. These two equations are coupled through the dependence of advective heat transport on the interstitial fluid-velocity field, and the dependence of fluid density, viscosity, and saturation on pressure and temperature.

Finite-difference techniques are used for the spatial and temporal discretization of the equations. When supplied with appropriate boundary and initial conditions and system-parameter distributions, simulations can be performed to evaluate a wide variety of hydrothermal transport situations. Numerical solutions are obtained simultaneously for the primary dependent variables, pressure and enthalpy. Secondary dependent variables such as temperature, saturation, fluid density, fluid viscosity, and interstitial fluid velocity are computed from the primary variables.

The HYDROTHERM simulator is a versatile computer code with various boundary-condition, source-sink, discretization, and equation-solver options. The computer code described in this documentation is Version 3. It incorporates several major additions and modifications to Versions 1 and 2 of the code documented in Hayba and Ingebritsen (1994). Primary additions and modifications include the following:

- Extension to handle unconfined ground-water flow by addition of unsaturated-zone water flow with uniform atmospheric pressure in the soil air phase,
- Addition of a precipitation-recharge boundary condition option at the land surface,
- Addition of a seepage-surface boundary condition option at the land surface,
- Addition of an optional graphical user interface (GUI) for data preparation and visualization of two-dimensional simulations,
- Addition of an iterative linear-equation solver based on a preconditioned, generalized-minimum-residual method, and
- Conversion of the program code to Fortran 90 to employ dynamic allocation of arrays.

Secondary additions and modifications include:

- Removing the limitation that a specified-pressure boundary also be a specified-temperature (enthalpy) boundary,
- Allowing permeability and porosity to be a function of depth for each rock type,
- Allowing permeability and porosity to be a function of time,
- Calculation of the dimensionless groups, Nusselt number (Nu) and thermal Peclet number (Pe), for each

cell, and
- Separation of the output files into as many as 26 individual files of variables and parameters with user-specified print intervals.

Some features have been replaced by alternative methods providing the same functionality. Superseded features include:
- Command line input of file names at the invocation of HYDROTHERM,
- Listing of version data for the most recently modified routines,
- Optionally activated code for producing output for debugging, and
- Output of execution-timing data.

A consequence of these additions and modifications is that legacy data files for HYDROTHERM Versions 1 and 2 will not run in Version 3. However, it is feasible to convert a legacy data file to run in Version 3 using the information provided in chapter 5.

1.1 APPLICABILITY AND LIMITATIONS

The HYDROTHERM code is suitable for simulating flow of liquid water and water vapor (steam) in porous media with associated thermal energy transport in two phase, three-dimensional flow systems under confined or unconfined conditions. As such, the code is applicable to the study of hydrothermal flow systems at temperatures ranging from 0 to 1,200 ° C and at pressures ranging from from 5×10^4 Pa (0.5 atm) to 1×10^9 Pa (10^4 atm). This temperature range covers that of a basaltic-magmatic hydrothermal system, and exceeds that of a silicic-magmatic hydrothermal system. The corresponding specific enthalpy ranges from 1×10^7 to 5.2×10^{10} ergs/gram (erg/g) spanning near-critical and super-critical conditions for the fluid component which is pure water. Saline ground-water systems, with salinity greater than seawater, cannot be represented. Whereas salinity effects on fluid density may be minor compared to thermal effects in two-phase, high-temperature systems, other influences of solutes may be nontrivial (Ingebritsen and Sanford, 1998, p. 80). HYDROTHERM can now handle shallow unconfined aquifer systems with precipitation recharge and associated advective addition of heat.

There is a limitation on the range of the thermodynamic tables contained in HYDROTHERM. The region for super-critical pressure (greater than 2.4×10^8 dynes per square centimeter (dyne/cm^2)[237 atm]) and low enthalpy (less than 0.5×10^9 erg/g) is not covered. This prevents applications of HYDROTHERM to high pressure, low temperature systems such as in the deep ocean floor. The simulator also can not handle situations where cold precipitation recharge penetrates several kilometers below the land surface creating a plume of cold ground water in a high pressure environment.

Three-dimensional Cartesian and axisymmetric, cylindrical-coordinate systems are available. However, the HYDROTHERM INTERACTIVE graphical user interface version is restricted to two-dimensional Cartesian or cylindrical coordinates.

Possible applications include deep and shallow hydrothermal systems, geothermal reservoirs, cooling plutons, volcanic systems with shallow magma intrusions, magmatic-hydrothermal systems, crustal-scale heat transport, and other systems with one- or two-phase thermal transport processes. Phase separation, heat transfer by a heat-pipe mechanism, and boiling-condensing with buffered pressure transmission also can be

simulated.

Because flow of the air component is not simulated and no water vapor is allowed in the air-water unsaturated zone, application of the simulator to shallow aquifers is restricted to systems where negligible vaporization of water occurs at the water table. In other words, steaming, boiling, or erupting water-table conditions can not be simulated. However, shallow, water-table aquifers at cool temperatures can be represented, with hot, two-phase zones at depth. A full two-phase, two-component simulator would be required to rigorously handle steaming water-table conditions.

A primary limitation for application to highly fractured volcanic environments is the assumption of Darcy's equation for flow in a porous medium. Fluid flow in fractured media will be realistically represented only when the scale of the region under consideration is large relative to the fracture spacing.

Evapotranspiration as a function of water-table depth below the land surface, ponding, and surface runoff of excess precipitation flux are not represented in HYDROTHERM. In addition, atmospheric pressure can not vary with location along the land surface. This could be a limitation for simulations in mountainous terrain.

HYDROTHERM heat transport equations do not take into account changes in fluid enthalpy caused by pressure-volume reversible work. This may cause artificially low temperatures to be computed when a parcel of cold water flows to depths where pressures are high and becomes substantially compressed.

The effective thermal conductivity of the ground water and the porous medium is not allowed to be a function of porosity or saturation. This simplifying assumption can be a limitation in the unsaturated zone especially.

The simulation region is spatially discretized using a cell-centered grid (chapter 3). A primary numerical limitation of this code results from the use of finite-difference techniques for spatial- and temporal-derivative approximations. Where the thermal-conduction coefficient is small, cell sizes are required to be small to minimize numerical dispersion or oscillation. Furthermore, if a region of large temperature gradients is somewhat convoluted and three-dimensional, the projection of rows and columns of cells from regions of high nodal density may cause more cells than are needed to appear in other parts of the simulation region. In addition, small cell sizes may be necessary to accurately represent spatial density variations or sharp saturation fronts in the unsaturated zone. These three factors can combine to cause an excessive number of nodes to be required for a large, field-scale model, thus making the simulations prohibitively demanding of computer storage and computation time. In such cases, a simple model of the system, useful for investigating mechanisms and testing hypotheses, may be all that is practical.

Another limitation is that finite-difference grids do not conform to boundaries that are not parallel to the coordinate axes. Stair-step approximations to angular boundaries, such as sloping land surfaces, are inconvenient to specify and can cause local variations in the ground-water flow field that are not realistic. The z-coordinate is restricted to point in the upward vertical direction; a tilted coordinate system is not possible.

Unconfined systems with nearly dry unsaturated zones can be difficult to simulate, requiring very small time steps for convergence relative to those required for reasonable time discretization errors. If flow in the unsaturated zone is relatively unimportant, reasonable time-step lengths can be used if an artificial relation of

saturation to capillary pressure is adopted. This approach results in higher saturation values than actually occur.

The HYDROTHERM simulator can be run using a graphical user interface for data input and output visualization during simulations. However, the graphical user interface is limited to simulations with two-dimensional Cartesian-coordinate or axi-symmetric cylindrical-coordinate regions. Simulations with three-dimensional regions require using text-based data-input and post-simulation output visualization. The user must use other software tools for output visualization in three dimensions.

Applications of the HYDROTHERM simulator have been described in the following published literature. Previous versions of HYDROTHERM have been used to simulate large-scale phase separation in the Lassen, California, hydrothermal system (Ingebritsen and Sorey, 1985), the natural evolution of vapor-dominated zones (Ingebritsen and Sorey, 1988), geysers (Ingebritsen and Rojstaczer, 1993, 1996), convection of water near the critical point (Ingebritsen and Hayba, 1994), the conditions leading to lead-zinc mineralization (Rowan and Goldhaber, 1995, 1996), multiphase ground-water flow near cooling plutons (Hayba and Ingebritsen, 1997), black-smoker temperatures (Jupp and Schultz, 2000), hydrothermal systems on Mars (Rathbun and Squyres, 2002; Harrison and Grimm, 2002, Abramov and Kring, 2005), convection driven by heat from magmatic dikes (Polyansky and others, 2002) and by basaltic magmatism in a petroleum basin (Polyansky and others, 2003), thermal conditions beneath the summit of Kilauea Volcano, Hawaii (Hurwitz and others, 2002; Hurwitz and Ingebritsen, 2003), hydrothermal pressurization leading to the collapse of volcanic edifices (Reid, 2004), impact-induced terrestrial hydrothermal systems (Abramov and Kring, 2004), and cooling ash-flow sheets (Hogeweg and others, 2005). Recent applications of HYDROTHERM Version 2 in Japan include geothermal flow in the Shimabara Peninsula (Fujimitsu and others, 2006), in the Kuju volcanic field of Kyushu (Kubota and others, 2006), simulation of volcanomagnetic effects (Okubo and others, 2006), and simulation of the Obama geothermal field (Saibi and others, 2006). Impact-induced hydrothermal activity at the Chicxulub crater was modeled by Abramov and Kring (2007), and the thermal structure of Merapi volcano, Indonesia was simulated by Harmako and others (2007).

The version of HYDROTHERM documented in this report has been used to simulate ground-water flow, heat transport, and water-table position within volcanic edifices (Hurwitz and others, 2003) and impact-induced terrestrial hydrothermal systems (Sanford, 2005).

1.2 PURPOSE AND SCOPE

The purpose of this guide to the revised ground-water flow and heat transport simulator is to provide the user with information about the additions and modifications incorporated into HYDROTHERM, Version 3. Chapters on the theory, numerical implementation, code description, input information, and output information are provided. In addition, there is a chapter on using the graphical-user-interface for simulation of problems with two-dimensional regions. Finally, verification test problems for the new capabilities are presented. This documentation is intended to be complete for using HYDROTHERM, Version 3. However, the documentation for HYDROTHERM Version 1 (Hayba and Ingebritsen, 1994) should be consulted for information on the original development of HYDROTHERM, on using previous versions of the input-data file, and on previous verification test problems.

Each release of the HYDROTHERM program code is identified by a version, release, and patch number in the form: (*version number.release number.patch number*). This number will change as additions, modifications, and corrections are made to the program. Modifications requiring revisions to the documentation change the version number, while minor revisions change the release number, and corrections change the patch number.

Explanations of the mathematical equation variables appear after the first usage, and a complete table of notation appears at the front of this report. Abbreviations for the units of measure are defined at the first usage and also in the conversion table at the front of this report.

1.3 ACKNOWLEDGMENTS

The contributions of Shaul Hurwitz, Steve Ingebritsen, Dan Hayba, and Mike Sorey of the U.S. Geological Survey, who provided application problems and technical guidance that influenced program development and who helped with program testing, are gratefully acknowledged.

2. THEORY

The ground-water flow and thermal energy (heat) transport equations solved by the HYDROTHERM simulator are described by Faust and Mercer (1979a, 1979b, 1982). Version 3 of this simulator contains extensions to handle shallow, unconfined aquifers including a nonflowing air component that can exist only in the gas phase, a precipitation-recharge boundary condition at the land surface, and a seepage-surface boundary condition. The equations for these extensions will be presented in this chapter.

Terminology conventions in this report are as follows: enthalpy and specific enthalpy refer to enthalpy per unit mass; two-component, air-water, and partially saturated refer to the unsaturated zone above the water table; water vapor and steam refer to the water component in the gas phase; and fluid refers to the water component in either the liquid or the gas (vapor) phase. Explanations of the mathematical notation appear after the first usage and a complete table of notation appears at the front of this report. All variables are given in metric (SI) units of measure. However, the HYDROTHERM simulator uses centimeter-gram-second (cgs) metric units internally.

2.1 FLOW AND TRANSPORT EQUATIONS

The first two versions of the HYDROTHERM simulator incorporated a single component (water) that could exist in two phases (gas, liquid). Pressure and enthalpy were selected as the primary dependent variables because they uniquely define the thermodynamic state of the system. Extending HYDROTHERM to be able to simulate unconfined ground-water flow was done by adding a second component (air) that exists only in the gas phase and is always at atmospheric pressure (Pinder and Gray, 1977, p. 190) so it does not flow. This addition enables simulation of the partially saturated zone from the land surface down to the water table. The water table is defined as the surface of atmospheric pressure in the liquid phase. This approach allows for the same flow equation to be used in each of the saturated, two-phase, and partially saturated zones, and also allows for future extension of HYDROTHERM to become a full two-component, two-phase simulator with flow of both air and water throughout a domain.

2.1.1 Ground-Water Flow Equation

The partial-differential equation of flow of the water component is based on the following assumptions:

- The fluid is pure water which can exist in liquid and gas (vapor) phases.
- Flow is described by Darcy's law for a two-phase system.
- Capillary-pressure effects are negligible in zones of liquid water coexisting with water vapor (one-component zones).
- Capillary-pressure effects are represented by non-hysteretic functions of liquid phase saturation in zones of liquid water coexisting with air (two-component zones).
- Relative permeabilities are non-hysteretic functions of liquid phase saturation.
- No dissolved air exists in the liquid phase.
- No water vapor exists in the gas phase in zones where air is present (two-component zones).
- The air component is stagnant with no buoyant circulation.

- The porous medium is compressible except in the partially saturated zone.
- The liquid water and water vapor are compressible.
- Water density is a function of space and time through dependence on temperature and pressure.
- Water viscosity is a function of space and time through dependence on temperature and pressure.
- Porosity and permeability can be functions of spatial location.
- The coordinate system is right-handed with the z-axis pointing vertically upward.
- The coordinate system is chosen to be aligned with the principal directions of the permeability tensor so that this tensor is diagonal for anisotropic media.
- Dispersive-mass fluxes of the bulk fluid from spatial velocity fluctuations are not included.
- Fluxes of liquid water and water vapor (steam) driven by thermal gradients are not included.

The justifications and implications of most of these assumptions are given in Hayba and Ingebritsen (1994) and Ingebritsen and Sanford (1998, p. 80). One limitation on the unconfined extension is that HYDROTHERM can not simulate boiling liquid at the water table.

Pressure is chosen as the dependent variable for fluid flow, because no potentiometric-head function exists for density fields that depend on temperature. All pressures are expressed as absolute. The water-component flow equation (Faust and Mercer, 1979a, Huyakorn and Pinder, 1983) is based on the conservation of water mass in a volume element, coupled with Darcy's law for multiphase flow through a porous medium. Thus:

$$\frac{\partial}{\partial t}\left[\phi(\rho_w S_w + \rho_s S_s)\right] - \nabla \cdot \frac{\mathbf{k} k_{rw} \rho_w}{\mu_w}\left[\nabla p + \rho_w g \hat{\mathbf{e}}_z\right] - \nabla \cdot \frac{\mathbf{k} k_{rs} \rho_s}{\mu_s}\left[\nabla p_g + \rho_s g \hat{\mathbf{e}}_z\right] - q_{sf} = 0, \qquad (2.1.1)$$

where

ϕ is the porosity (dimensionless),
ρ is the fluid density (kg/m^3),
S_p is the saturation of water in phase p; p=w(water) or s(steam) (dimensionless),
$\mathbf{k}$ is the porous-medium permeability tensor (m^2),
k_r is the relative permeability (dimensionless),
μ is the viscosity (Pa-s),
p is the fluid pressure in the liquid phase (Pa),
p_g is the fluid pressure in the gas phase (Pa),
g is the gravitational constant (m/s^2),
$\hat{\mathbf{e}}_z$ is the unit vector in the z-coordinate direction (dimensionless),
q_{sf} is the flow-rate intensity of a fluid-mass source (positive is into the region) (kg/s-m^3),
t is the time (s), and
∇ is the spatial gradient (m^{-1}).

The phase subscripts w and s refer to water (liquid phase) and steam (gas phase, vapor phase) respectively. In the single-component (water) zone, $p_g = p$, because capillary pressure is assumed to be zero. In the two-component (air-water) unsaturated zone, the terms of equation 2.1.1 involving steam are not present. Equation 2.1.1 relates the rate of change of total water mass in the liquid and gas phases to the net inflow rate of water and source flow rate. In the unsaturated zone, no flow equation needs to be formulated for the air component which is assumed be at atmospheric pressure and of negligible density and, thus, not flowing.

Because any point in the simulation region may be in the single-component or two-component zone, the saturation constraint equation is generalized to:

$$S_w + S_g = 1, \tag{2.1.2}$$

where

S_w is the saturation of the liquid phase (water) (dimensionless), and

S_g is the saturation of the gas phase (steam or air) (dimensionless).

Under the assumptions presented above, S_g represents the saturation of either water vapor (steam) or air at a given point in the region. There is no provision for steam and air to coexist in the HYDROTHERM simulator.

The interstitial or pore velocity ($\mathbf{v}_p$) for the water component in phase p is obtained from Darcy's law as:

$$\mathbf{v}_p = -\frac{\mathbf{k}k_{rp}}{\phi S_p \mu_p}[\nabla p + \rho_p g \hat{\mathbf{e}}_z], \tag{2.1.3}$$

where

$\mathbf{v}_p$ is the interstitial-velocity vector for water in phase p; p=w(water) or s(steam) (m/s).

The water table is defined as the surface of atmospheric pressure and its configuration can be determined from the pressure solution. Using a partially saturated zone formulation means that the flow simulation region can extend up to the land surface and does not need to be adjusted to conform to the water-table configuration as the simulation progresses.

2.1.2 Thermal-Energy Transport Equation

The partial-differential equation of thermal-energy transport is based on the following assumptions:

- Heat-transport mechanisms comprise thermal conduction and advection only.
- Heat transport by dispersion is neglected.
- Heat transport by radiation is neglected.
- The porous matrix and the fluid phases are in thermal equilibrium.
- Heating from viscous dissipation is neglected.
- Convective heat transport by air in the gas phase is neglected.
- Thermal conduction occurs through the liquid, gas, and solid phases in parallel.
- Thermal conductivity is not a function of porosity or liquid saturation.
- Conductive heat transport through the air in the gas phase is approximated by an effective thermal conductivity specified for the solid and liquid phases.
- Thermal conductivity of the porous matrix is a function of spatial location.
- Heat capacity of the porous matrix may be a function of temperature.
- Heat capacity of the air component is neglected.
- Thermal conductivity of the porous matrix may be a function of temperature.
- Enthalpy of the porous matrix is only a function of temperature.
- Density of the porous matrix may be a function of temperature.
- Changes in fluid enthalpy by pressure volume work, reversible work, or flow work as a parcel of fluid

moves are neglected.

- Thermal expansion of the porous medium is neglected.

The justifications and implications of most of these assumptions also are given in Hayba and Ingebritsen (1994) and Ingebritsen and Sanford (1998, p. 80). Support for the assumption of thermal equilibrium between the fluid and porous solid phases comes from de Marsily (1986, p. 277) who reports equilibration times of minutes to hours. The error incurred by assuming that the effective thermal conductivity of a porous medium does not change with liquid saturation may be assessed by noting that this parameter can change by a factor of 1.2 to 6 for a typical porous media (sand) as saturation changes from residual saturation to full saturation (de Marsily (1986, p. 281). Thus a constant effective thermal conductivity is a poor approximation unless the porosity is small (less than 0.1). A consequence of neglecting pressure-volume work in the thermal-energy transport equation is that it is possible for temperatures in the liquid phase at large depths to become lower than any boundary-condition or source temperature or initial-condition temperature. Usually the minimum temperature under this approximation is only a few degrees Celsius below what it would be if pressure-volume work were not neglected.

The thermal-transport equation is based upon the conservation of enthalpy in both the fluid phases and the solid phase of the porous medium in a volume element of the region. Enthalpy is a derived property containing both internal energy and flow energy. Thus:

$$\frac{\partial}{\partial t}\left[\phi(\rho_w h_w S_w + \rho_s h_s S_s) + (1-\phi)\rho_r h_r\right] - \nabla \cdot K_a \mathbf{I} \nabla T + \nabla \cdot \phi(S_w \rho_w h_w \mathbf{v}_w + S_s \rho_s h_s \mathbf{v}_s) - q_{sh} = 0, \quad (2.1.4)$$

where

h is the specific enthalpy of the fluid phase (J/kg),

h_r is the specific enthalpy of the porous-matrix solid phase (rock or sediment) (J/kg),

ρ_r is the density of the porous-matrix solid phase (rock or sediment) (kg/m^3),

K_a is the effective thermal conductivity of the bulk porous medium (combined liquid, gas, and solid phases) (W/m-°C),

$\mathbf{I}$ is the identity matrix of rank 3 (dimensionless),

T is the temperature (°C),

q_{sh} is the flow-rate intensity of an enthalpy source (positive is into the region) (W/m^3).

The phase subscripts w and s refer to water and steam respectively. Equation 2.1.4 relates the rate of change of fluid and porous-medium enthalpy to the net conductive enthalpy flux, to the net advective enthalpy flux, and to the heat source. It is written for a unit volume containing liquid, gas, and solid phases. Note that the terms involving steam are not present in air-water (two-component) zones. A detailed derivation of equation 2.1.4 is given by Faust and Mercer (1977) and Huyakorn and Pinder (1983).

2.2 PROPERTY FUNCTIONS AND TRANSPORT COEFFICIENTS

Before the component conservation equations can be solved, information about the fluid properties, porous-medium properties, and transport coefficients is required. Many of these properties are discussed in Hayba and Ingebritsen (1994). Properties associated with partially saturated flow above the water table are presented in the following sections.

2.2.1 Fluid Density Function

An equation of state for pure water is used to determine fluid density as a function of enthalpy and pressure. This equation is incorporated in HYDROTHERM in tabular form. Values of the partial derivatives of density with respect to pressure and enthalpy are also in the tables. Values in two-phase, one-component (water) zones are interpolated, using saturations, from values for saturated liquid water and saturated steam. The fluid density values were calculated from the steam-table routines of the National Bureau of Standards (Haar and others, 1984).

2.2.2 Fluid Viscosity Function

An equation of state for pure water, in tabular form, is used to determine fluid viscosity as a function of enthalpy and pressure, and the partial derivatives of viscosity with respect to pressure and enthalpy. Values in two-phase, one-component (water) zones are interpolated from values for saturated liquid water and saturated steam. Viscosity values were calculated using the formula of Watson and others (1980), as reevaluated by Sengers and Kamgar-Parsi (1984) using density values accepted by the 1982 International Association for the Properties of Steam agreement (Haar and others, 1984). For high pressures and temperatures, viscosity values were interpolated between values at 10 kilobars(kbar) estimated from plots of viscosity in relation to pressure and values at the limits of the formula of Sengers and Kamgar-Parsi (1984). Further details appear in Hayba and Ingebritsen (1994).

2.2.3 Temperature as a Function of Enthalpy and Pressure

An equation of state for pure water, in tabular form, also is used to calculate temperature as a function of enthalpy and pressure and partial derivatives of temperature with respect to pressure and enthalpy. The temperature values in the tables were calculated from the steam-table routines of Haar and others (1984).

2.2.4 Enthalpy and Density as a Function of Pressure and Temperature

The functions from the steam tables of the National Bureau of Standards (Haar and others, 1984) are used to determine enthalpy and density of water and steam as a function of temperature and pressure. They are based on equations of state for pure water. These functions are used for alternative data input of temperature instead of enthalpy.

There is a limitation on the range of the thermodynamic tables contained in HYDROTHERM. The basic range of pressure-enthalpy coverage in the tables is for pressure from 5×10^5 dyne/cm^2 (5×10^4 Pa, 0.5 atm) to 1×10^{10} dyne/cm^2 (1×10^9 Pa, 10^4 atm) and enthalpy from 1×10^7 erg/g to 5.2×10^{10} erg/g. However, the region in the tables for super-critical pressure (greater than 2.4×10^8 dyne/cm^2(237 atm)) and low enthalpy (less than 0.5×10^9 erg/g) is not covered. This prevents applications of HYDROTHERM to high-pressure, low-temperature systems such as in the deep ocean floor. The simulator also can not handle situations where cold precipitation recharge penetrates several kilometers below the land surface creating a plume of cold ground water in a high pressure environment.

2.2.5 Pressure-Enthalpy Diagram and the Unsaturated Zone

In the unsaturated zone of dry environments or in areas with deep water tables, the capillary pressure can be very high and the corresponding liquid-phase pressure can be a large negative value. The liquid-phase (hydrodynamic) pressure has characteristics of a potential energy which can range over negative and positive values. In the saturated flow region, the hydrodynamic pressure is always positive when either an absolute scale or a scale relative to atmospheric is used.

The pressure used with the pressure-enthalpy table and in the equation of state for water is the physical pressure, which on an absolute scale must always be greater than zero or on a relative scale must be greater than −1 atm. In the saturated flow region, the hydrodynamic pressure and the physical pressure can be taken as identical. Actually, the HYDROTHERM simulator is limited in thermodynamic fluid property tables to pressures greater than 0.5 atm. Therefore, the hydrodynamic pressure and the physical pressure are regarded as the same for pressures greater than 0.5 atm. This lower limit on physical pressure is used for all thermodynamic calculations in the partially saturated zone when the hydrodynamic pressure is less than 0.5 atm.

This approximation should not induce serious error into a simulation because at low hydrodynamic pressures in the partially saturated zone, there is little heat transport by advection. Conductive heat transport dominates there.

In the partially saturated zone, additional simplifying assumptions are made for properties of the liquid water component. Density, viscosity, and temperature are assumed to be independent of pressure, and saturation is assumed to be independent of enthalpy. Temperature is a function of enthalpy but is always evaluated at the minimum pressure of 0.5 atm in the unsaturated zone.

2.2.6 Permeability Functions

The spatial distribution of permeability can be specified in various ways depending on the nature and extent of the region being modeled. The simplest is uniform zones of permeability corresponding to different rock types or sediment. Variation of permeability with depth below land surface described by a function, possibly corresponding to rock type, also can be useful. Linear, logarithmic, and exponential permeability functions are available in HYDROTHERM. Permeability as a function of temperature or as a function of time also can be used to establish the permeability distribution. These latter variations introduce additional nonlinearity into the flow and heat transport equations.

Permeability variation with depth can be linear

$$k_i = a_0 + a_1 z_d \tag{2.2.1}$$

or logarithmic

$$\log_{10}(k_i) = a_0 + a_1 \log_{10}(z_d) \tag{2.2.2}$$

or log-linear

$$\log_{10}(k_i) = a_0 + a_1 z_d \tag{2.2.3}$$

where

k_i is the permeability in coordinate direction i (m^2),
a_i are empirical coefficients; i=0,1 [($m^{2)}$ or (m) or (dimensionless)], and
z_d is the depth below the land surface (m).

Equation 2.2.2 was proposed by Manning and Ingebritsen (1999) from analysis of geothermal systems, whereas equation 2.2.3 comes from from Williams and Narasimhan (1989).

Permeability as a function of time can be piecewise linear,

$$k_i(t) = b_0^{(n)} + b_1^{(n)}\, t \quad \text{for } t_{n-1} \le t < t_n, \tag{2.2.4}$$

where

$b_i^{(n)}$ are empirical coefficients (i=0,1) for time interval t_{n-1} to t_n [(m^2) or (m^2/s)].

This function can represent the decrease of permeability due to mineralization, for example. The function of time can be different for each type of porous medium. The functions are defined for the x-direction permeability, then the y- and z-direction values are obtained as constant multiples of the x-direction value. Input of a new permeability value or function for a subsequent simulation period replaces the current permeability function of time. Permeability as a function of temperature is described in section 2.2.10.

2.2.7 Relative-Permeability Functions

The shapes of typical relative-permeability curves are shown by Bear (1972, p. 460). A choice of three sets of functions in HYDROTHERM is provided to represent the relative permeability for liquid water and water vapor as a function of liquid-phase saturation. Set one contains linear functions for relative permeability:

$$k_{rw} = S_e, \tag{2.2.5a}$$

$$k_{rs} = 1 - S_e, \tag{2.2.5b}$$

where

k_{rw} is the relative permeability to liquid water (dimensionless),
k_{rs} is the relative permeability to water vapor (dimensionless), and
S_e is the scaled liquid-phase saturation (dimensionless).

The scaled saturation, S_e, is defined by

$$S_e = \frac{S_w - S_{wr}}{1 - S_{wr} - S_{sr}}, \tag{2.2.6}$$

where

S_{wr} is the residual saturation of liquid water (dimensionless), and
S_{sr} is the residual saturation of water vapor (dimensionless).

Set two contains Corey functions for relative permeability (Corey, 1957):

$$k_{rw} = S_e^4 \tag{2.2.7a}$$

$$k_{rs} = (1 - S_e^2)\left[1 - S_e\right]^2 \tag{2.2.7b}$$

Set three contains fracture flow functions for relative permeability (Sorey and others, 1980):

$$k_{rw} = S_e^4 \tag{2.2.8a}$$

$$k_{rs} = 1 - S_e^4 \tag{2.2.8b}$$

The linear and Corey relative-permeability functions are available for unconfined flow systems. However, the relative permeability to water vapor is not applicable in the air-water zones, and the residual saturation of water vapor should be specified as zero.

The user can choose the most appropriate function set for a given application. In particular, the linear functions are useful when the water-table configuration is desired but the details of flow in the partially saturated zone are unimportant. The selected set of relative-permeability functions applies to the entire simulation domain.

2.2.8 Water-Saturation Functions

In the air-water zone of partial saturation, a function is needed to relate the water saturation to the capillary pressure. Capillary pressure is the difference in pressure between the gas phase consisting of air and the liquid phase consisting of water. Thus:

$$p_c = p_g - p, \tag{2.2.9}$$

where

p_c is the capillary pressure (Pa).

The shapes of typical capillary pressure-saturation curves are shown by Bear (1972, p. 449,479). For consistency, the saturation functions associated with the linear and Corey permeability functions were incorporated for this extension to unconfined flow systems. The Cooley (1983) water-saturation function also was included because it provides a more stable calculation for conditions near the bubble-point pressure. The bubble-point pressure is essentially the minimum value of capillary pressure during drainage at which a continuous gas phase exists. The residual water saturation is the saturation below which the liquid phase is discontinuous and liquid flow ceases.
The linear water-saturation function is:

$$\frac{1 - S_w}{1 - S_{wr}} = \frac{p_c - p_{cb}}{p_{cr} - p_{cb}}, \text{ for } p_{cb} \le p_c \le p_{cr}, \tag{2.2.10a}$$

$$S_w = 1, \text{ for } p_c < p_{cb} \tag{2.2.10b}$$

$$S_w = S_{wr}, \text{ for } p_c > p_{cr} \tag{2.2.10c}$$

where

p_{cb} is the capillary pressure at the bubble point in the liquid phase (Pa), and
p_{cr} is the capillary pressure at residual water saturation (Pa).

The Corey water-saturation function is (Brooks and Corey, 1966):

$$\frac{S_w - S_{wr}}{1 - S_{wr}} = \left[\frac{p_{cb}}{p_c} \right]^2, \text{ for } p_c \geq p_{cb}, \tag{2.2.11a}$$

$$\frac{S_w - S_{wr}}{1 - S_{wr}} = 1, \text{ for } p_c < p_{cb}. \tag{2.2.11b}$$

In HYDROTHERM, the exponent is set at 2 although it is a parameter in the general Corey function.

The Cooley water-saturation function is (Cooley, 1983):

$$\frac{S_w - S_{wr}}{1 - S_{wr}} = \frac{c_2}{\left[\frac{p_c - p_{cb}}{p_{cb}} \right]^{c_1} + c_2}, \text{ for } p_c \geq p_{cb}, \tag{2.2.12a}$$

$$\frac{S_w - S_{wr}}{1 - S_{wr}} = 1, \text{ for } p_c < p_{cb}, \tag{2.2.12b}$$

where

c_1 is an empirical coefficient (dimensionless), and

c_2 is an empirical coefficient (dimensionless).

The Cooley function has been converted to use pressure instead of head to be compatible with the HYDROTHERM dependent variable of pressure. The ranges of the empirical coefficients are for c_1 [1–4] and for c_2 [1–40], as discussed by Winter (1983, p. 1206). This function also is associated with the Corey relative permeability function. Hysteresis is neglected in all of these water-saturation functions, and saturation is assumed to be independent of temperature (enthalpy).

Corresponding to the linear relative permeability function, the linear saturation function is useful when the water-table configuration is desired but the details of flow in the partially saturated zone are unimportant. In very dry porous media, as the saturation approaches its residual value, the slopes of the water-saturation functions of Brooks and Corey (1966) and Cooley (1983) become infinite. This leads to numerical difficulty for the simulation. One way to avoid this difficulty is to limit the minimum saturation to a value somewhat larger than the residual saturation.

2.2.9 Porous-Medium Porosity Function

In HYDROTHERM, the porous medium is slightly compressible, so porosity is a function of pressure and can be expressed as a linear function in single-component (liquid and vapor) zones. Thus,

$$\phi = \phi^0 \left(1 + \alpha_b (p - p^0) \right) \tag{2.2.13}$$

where

ϕ^0 is the porosity at the initial fluid pressure (dimensionless),

p^0 is the initial fluid pressure distribution (Pa).
α_b is the bulk vertical compressibility of the porous medium (Pa^{-1}),

This equation applies to both confined and unconfined saturated flow, but in the zone of partial saturation, the porous medium is assumed to be incompressible, so porosity is constant there.

2.2.10 Porous-Medium Properties as a Function of Temperature

Properties that can vary with temperature include: permeability, porosity, effective thermal conductivity, heat capacity, density, and bulk compressibility. Three empirical functions can be used to describe variation in porous-medium properties with temperature. The function of temperature can be selected to be piecewise constant

$$u_{pm} = u_{pm}^{(k)}, \text{ for } T_{k-1} \le T < T_k \tag{2.2.14}$$

or linear

$$u_{pm} = c_0 + c_1 T \tag{2.2.15}$$

or logarithmic

$$\log (u_{pm}) = c_0 + c_1 T \tag{2.2.16}$$

where

u_{pm} is a porous-medium property (appropriate units),
$u_{pm}^{(k)}$ is a porous-medium property value over the temperature interval k (appropriate units),
T_k are the temperature values at the ends of the intervals of constant temperature; k=1,2,3,4 (°C), and
c_i are empirical coefficients; i=0,1 [(dimensionless) or ($°C^{-1}$)].

These functions for temperature dependence can have up to three segments defined over the temperature range by specifying four sets of property and temperature values. Note that these functions introduce additional non-linearity into the system equations 2.1.1 and 2.1.4.

2.3 SOURCE TERMS

Simulated ground-water flow into and out from a model region can occur across open boundaries, including boundaries of well bores. It is possible to replace specified-flux boundary conditions by equivalent source terms in the differential equations. This is done usually at the discretization step. Thus, there is a need to have both line and point sources available in the simulator.

2.3.1 Well Model

Well bores are usually idealized as line sources in three-dimensional regions because the radius of the well is very small compared to the size of the model region or of even a single discretized cell. Only vertical well bores are allowed in the HYDROTHERM simulator. No horizontal or slanting well bores can be defined. Similarly, a well is idealized as a point source in two-dimensional areal regions. Including a well in a two-dimensional vertical-slice region could be done, but it is not physically realistic because the actual flow field near the well is three-dimensional. In this case, the well would represent a planar source extending perpendicular to the two-dimensional region. This usage may be appropriate for modeling flow in a vertical

fracture.

A well can be used for fluid injection or withdrawal, with associated heat injection or production. The well bore can communicate with the aquifer along a set of defined segments along the line source which extends in the z-coordinate direction at the given x-y location of the well. That is, the well may be screened or be an open hole over several intervals of its depth.

In the case of an injection well, usually only one phase is injected or the phase composition of the mixture is known. The total flow rate being injected for a given phase is specified by

$$Q_{wp} = \int_{l_L}^{l_U} q_{wp} dl, \quad \text{with } p = w, s, \tag{2.3.1}$$

where

Q_{wp} is the volumetric well flow rate for phase p (positive is injection) (m^3/s),
l is the distance along the well bore (m),
l_L is the z-coordinate of the lower end of the screened interval of the well (m), and
l_U is the z-coordinate of the upper end of the screened interval of the well (m).
q_{wp} is the volumetric well flow rate per length of well bore for phase p; p=w for water and s for steam (positive is injection) (m^3/s-m),

Equation 2.3.1 forms a constraint on the distribution of well flow, q_{wp}.

In the case of a production well, the phase composition of the produced fluid reflects the conditions calculated by HYDROTHERM in the vicinity of the wellbore, and thus can not be controlled directly. The imposed constraint condition can be the fluid (water and steam) volumetric production rate,

$$Q_w = Q_{ww} + Q_{ws}, \tag{2.3.2}$$

where

Q_w is the volumetric total production flow rate for the well (m^3/s),
Q_{ww} is the volumetric production flow rate of water for the well (m^3/s), and
Q_{ws} is the volumetric production flow rate of steam for the well (m^3/s).

Alternatively, the imposed constraint condition can be the specified volumetric production rate for liquid water, Q_{ww}. The former condition is used in HYDROTHERM. At a given point along the well bore, the flow rate per unit length, $q_{wp}(l)$, must be apportioned to the two phases and the total flow of each phase must be distributed along the well bore. However the $q_{wp}(l)$ depend on local pressures and saturations at the well bore which are results of the simulation. The correct distribution of flow rate must satisfy Darcy's law. So, for a capillary pressure of zero,

$$\frac{q_{ws}(l)}{q_{ww}(l)} = \frac{\left[\frac{k_w k_{rs}}{\mu_s}\right]}{\left[\frac{k_w k_{rw}}{\mu_w}\right]}, \tag{2.3.3}$$

where

k_w is the effective permeability at the well bore (m^2).

Equation 2.3.3 says that the phase flow rates per unit length along the well bore are in proportion to their relative mobilities. The total fluid flow rate (liquid plus gas phases) for the well is given by

$$Q_w = \int_{l_L}^{l_U} \left[\left(\frac{k_w k_{rw}}{\mu_w} \right) + \left(\frac{k_w k_{rs}}{\mu_s} \right) \right] \frac{\partial p}{\partial n_w} \, dl, \tag{2.3.4}$$

where

n_w is the outward normal to the well bore (dimensionless).

Allocation of the specified flow rate along the well bore by fluid mobility is obtained by assuming that the pressure gradient normal to the well bore is uniform with depth along the well screen (Aziz and Settari, 1979). Then the allocation of the total fluid flow rate is

$$\frac{q_{ws}(l)}{Q_w} = \frac{\left[\frac{k_w k_{rs}}{\mu_s} \right]}{\int_{l_L}^{l_U} \left[\left(\frac{k_w k_{rw}}{\mu_w} \right) + \left(\frac{k_w k_{rs}}{\mu_s} \right) \right] dl} \tag{2.3.5a}$$

and

$$\frac{q_{ww}(l)}{Q_w} = \frac{\left[\frac{k_w k_{rw}}{\mu_w} \right]}{\int_{l_L}^{l_U} \left[\left(\frac{k_w k_{rw}}{\mu_w} \right) + \left(\frac{k_w k_{rs}}{\mu_s} \right) \right] dl}. \tag{2.3.5b}$$

If the screened interval is not continuous from l_L to l_U, the mobility, $\left(\frac{k_w k_{rs}}{\mu_s} \right)$, is set to zero over the appropriate subintervals.

After the flow rate has been allocated, heat-injection rates are determined from the associated enthalpy or temperature values for the injected fluid. Heat-withdrawal rates are determined by the ambient enthalpy (temperature) in the aquifer over the depth range that communicates with the given well bore.

2.3.2 Point Sources

Point sources (or sinks) may be specified in the simulation domain to represent localized fluid and heat flux conditions. They can represent wells in two-dimensional areal regions or specified-flux boundary conditions. Point sources inject fluid with associated injection enthalpy or temperature defined by the user. Point sinks produce fluid with associated enthalpy (temperature) determined by ambient conditions.

2.4 BOUNDARY CONDITIONS

The types of boundary conditions available for HYDROTHERM include (1) specified pressure, (2) specified enthalpy or specified temperature, (3) specified fluid flux, (4) specified heat flux, and (5) seepage surface. The first two are specified-value type of conditions, the second two are specified-flux type of conditions, while the last one is a combination type of condition.

2.4.1 Specified Pressure, Enthalpy, and Temperature Boundary Conditions

The first type of boundary condition, known as a Dirichlet boundary condition, is a specified fluid pressure condition for the ground-water flow equation, and a specified enthalpy or specified temperature condition for the heat-transport equation. These conditions can be specified independently as functions of location and they also can vary independently with time. Mathematically,

$$p = p_B(\mathbf{x}, t), \text{ for } \mathbf{x} \text{ on } A_p^1, \text{ and} \tag{2.4.1a}$$

$$h = h_B(\mathbf{x}, t), \text{ for } \mathbf{x} \text{ on } A_h^1, \text{ or} \tag{2.4.1b}$$

$$T = T_B(\mathbf{x}, t), \text{ for } \mathbf{x} \text{ on } A_T^1, \tag{2.4.1c}$$

where

p_B is the fluid pressure at the specified value boundary (Pa),
h_B is the specific enthalpy at the specified value boundary (J/kg),
T_B is the temperature at the specified value boundary (°C),
$\mathbf{x}$ is the spatial location on the boundary (m),
A_p^1 is the part of the regional boundary with a specified pressure boundary condition,
A_h^1 is the part of the regional boundary with a specified enthalpy boundary condition, and
A_T^1 is the part of the regional boundary with a specified temperature boundary condition.

In the original HYDROTHERM simulator (Versions 1 and 2), the specified pressure and the specified temperature (enthalpy) boundary conditions were bound together. Whereas this may be satisfactory to simulate heat transport at open boundaries with ground-water inflow, it is unrealistic at open boundaries with outflow. On boundary surfaces across which fluid flow occurs, the advective transport of heat is assumed to dominate over any conductive transport. The associated boundary condition for the heat-transport equation is advective flux of enthalpy. For inflow boundary conditions, the enthalpy (or temperature) of the incoming fluid is specified. For outflow boundary conditions, the enthalpy of the outgoing fluid is that of the ambient fluid at the boundary, and is determined by solution of the heat-transport equation. This combination of flow and heat transport boundary conditions will be referred to as specified pressure and associated enthalpy (temperature) boundary conditions. Fluid entering the region advects heat in at the associated enthalpy (or temperature), while fluid leaving the region advects heat out at the local boundary enthalpy (or temperature) which will vary as the simulation progresses. Therefore, Version 3 of HYDROTHERM allows for the specified pressure and specified temperature (enthalpy) boundary conditions to be independently distributed.

Specified pressure and specified temperature boundary conditions are often used at the land-surface boundary of a model. Lateral boundaries are commonly specified as hydrostatic pressure and a geothermal

temperature profile for the associated temperature. At an outflow boundary, a boundary condition of specified hydrostatic pressure and associated temperature distribution can become unrealistic. As the boundary temperature changes, the fluid density changes, so the pressure distribution no longer satisfies hydrostatic equilibrium. This situation can induce artificial vertical flows along the boundary. One possible remedy would be to extend the simulation domain so the problematic boundary is located away from the region of substantial temperature changes. Then the assumption of a hydrostatic pressure boundary condition can be expected to hold during the simulation period.

2.4.2 Specified-Flux Boundary Conditions

The default boundary condition for the HYDROTHERM simulator is no fluid flux and no heat flux across the boundary surfaces. Fluxes of fluid and heat, known as Neumann boundary conditions, can be specified over certain parts of the boundary as functions of time and location. The available specified-flux boundary conditions are precipitation-recharge flux and basal heat flux. The former represents infiltrating precipitation and can be applied only along the top boundary of the simulation region which is usually the land surface. The latter represents conductive geothermal heating from below the simulation region and can be applied only along the bottom boundary.

2.4.2.1 Precipitation-Recharge Boundary Condition. In connection with unconfined ground-water flow, precipitation recharge at the land surface is represented by a specified fluid-flux boundary condition with associated advective heat flux into the region. The volumetric precipitation (water) flux over the land-surface boundary and the temperature associated with that flux are specified. The temperature is converted to enthalpy at atmospheric pressure, which determines the advective flux of heat into the region. The flux is defined as net recharge to the simulation region, that might result from precipitation minus evapotranspiration and/or surface runoff, and can vary in space and time during the simulation. Evapotranspiration as a function of water-table depth below the land surface, ponding, and surface runoff of excess precipitation flux are not represented in HYDROTHERM. In addition, atmospheric pressure can not vary with location along the land surface. This could be a limitation for simulations in mountainous terrain. The precipitation recharge flux and the advective heat flux do not vary in response to saturation changes or to temperature changes at or near the land surface. The maximum precipitation flux is limited in HYDROTHERM by the value of the saturated vertical permeability (hydraulic conductivity). Excess precipitation flux causes a warning to be printed and termination of the simulation. Under these assumptions, the precipitation boundary flux is given by

$$q_F = q_{FP}(\mathbf{x},t), \text{ for } \mathbf{x} \text{ on } A_p^2, \tag{2.4.2a}$$

where

q_F is the fluid flux at the precipitation-recharge boundary (m^3/m^2-s),
q_{FP} is the specified fluid flux from precipitation (m^3/m^2-s), and
A_p^2 is the part of the regional boundary with a precipitation-recharge boundary condition.

The thermal flux from precipitation is assumed to be purely advective with heat content obtained from the specified associated temperature of the precipitation flux. The density of the precipitation is calculated to be that of water at atmospheric pressure and the specified associated temperature. Thus,

$$q_H = q_{HP}(\mathbf{x},t) = h_P(T_P)\rho_P(T_P)q_{FP}(\mathbf{x},t), \text{ for } \mathbf{x} \text{ on } A_P^2, \tag{2.4.2b}$$

where

q_{HP} is the heat flux across the precipitation-recharge boundary (W/m^2),
h_P is the specific enthalpy of the precipitation (J/kg),
T_P is the temperature of the precipitation (°C),
ρ_P is the density of the precipitation (kg/m^3),

Note that h_P and ρ_P are functions of T_P and, thus, are functions of location along the boundary and of time. For time periods of no precipitation recharge flux, the corresponding thermal energy transport boundary condition becomes that of no heat flux across the land surface boundary. A more realistic boundary condition of interphase heat transfer at the land surface to or from the atmosphere is not available. In general, the precipitation recharge flux flows down through the partially saturated zone to the water table.

2.4.2.2 Basal Heat-Flux Boundary Condition. Heat flux through the bottom of the simulation region (basal heat flux) is represented by a specified heat-flux boundary condition with no coincident fluid flux. This heat flux results from thermal conduction through geologic media below the simulation region. The flux often represents geothermal heat flow and is, thus, upward or into the simulation region, although a downward flux can be specified if needed. The basal heat-flux can be specified only along the bottom boundary of the simulation region. Although the heat flux can be specified to vary in space and time during the simulation, it does not respond to changes in the temperature gradient normal to the bottom boundary within the simulation region. Under these assumptions, the basal heat flux is given by:

$$q_H = q_{HC}(\mathbf{x}, t), \text{ for } \mathbf{x} \text{ on } A_h^2, \tag{2.4.3}$$

where

q_{HC} is the conductive heat flux across the bottom boundary (W/m^2),
A_h^2 is the part of the bottom regional boundary with a heat-flux boundary condition.

2.4.3 Seepage-Surface Boundary Condition

For shallow, unconfined ground-water flow in regions with substantial topography, such as mountains or volcanic edifices, the possibility of a seepage surface is included in the simulator. A seepage surface is a boundary condition at the land surface along which atmospheric pressure exists and along which only outflow of ground-water occurs. Outflowing water from such a boundary condition on a sloping land surface might form springs with downhill, surface-water runoff. However, HYDROTHERM cannot simulate overland surface-water flow. The location of a seepage surface is a portion of the land surface. What is unknown is the point or line where the water table meets the seepage surface (Bear, 1972, p. 261). This meeting point is determined as part of the flow solution and it may move with time. The representation of a seepage surface in HYDROTHERM involves a boundary condition over a specified part of the land surface, which can switch between specified pressure and no fluid flux as the simulation progresses. The portion of the boundary surface that is seeping has a boundary condition of atmospheric pressure, while the portion of the boundary surface that is not seeping has a boundary condition of no fluid flux. These portions can change with time. A seeping portion is switched to non-seeping if the seepage rate becomes inflow to the region, and a non-seeping portion is switched to seeping if the pressure rises above atmospheric.

One simplifying assumption is that the atmospheric pressure along a seepage surface is uniform and constant. Variation of atmospheric pressure with land-surface elevation is not available in HYDROTHERM. This may be a limitation in mountainous regions.

The boundary condition for the heat-transport equation at a seepage surface also switches between two states. For a seeping portion, the boundary condition is that of advective heat transport out from the region at the enthalpy (temperature) of the boundary. For a non-seeping portion, the boundary condition is that of no heat flux. Thus, interphase transfer of heat to or from the atmosphere and the ground water and porous medium is not represented at the non-seeping portion of the seepage boundary. A seepage-surface boundary cannot be assigned a constant temperature, nor will it necessarily maintain its initial-condition temperature.

A limitation of the simulator is that a given segment of boundary can have either a precipitation-recharge flux boundary condition or a seepage-boundary condition but not both together. Along a sloping land surface, such as a mountainside or volcanic cone, the boundary condition is often a seepage surface in combination with precipitation recharge. The seepage surface occupies the lower portion of the boundary, but the contact location between the two boundary conditions is determined as the simulation progresses. Some simulators (Forster and Smith, 1988) iteratively adjust the location of the contact. This boundary condition is non-linear because the position of the contact between the seeping and non-seeping (precipitation recharge) portions of the boundary is initially unknown and moves as the simulation progresses. Switching between recharge and seepage-boundary conditions is provided in a simplified fashion. Iterative adjustment is avoided in HYDROTHERM by using a switching method described in section 3.4.3.

2.5 INITIAL CONDITIONS

HYDROTHERM only solves the transient forms of the ground-water flow and the heat-transport equations; thus, initial conditions are necessary to begin a simulation. Several options are available for specification of initial conditions.

For the ground-water flow equation, an initial-pressure distribution within the region needs to be specified. The simplest initial condition is hydrostatic, with a uniform pressure specified at a given depth or elevation. A more complex condition is an initial pressure distribution specified along the top of the simulation region. Then, an initial hydrostatic pressure distribution will be calculated based on the density field resulting from the initial temperature field. The most general initial condition is a node-by-node pressure specification, perhaps obtained from a previous simulation.

For the heat-transport equation, the initial-enthalpy or initial-temperature field needs to be specified. Again, this can be done as a function of position, or linearly interpolated along the z-coordinate direction. Interpolation of temperature in the vertical direction can be used to create an initial geothermal profile. Interpolation is not available in horizontal directions with HYDROTHERM. Extrapolation from an initial temperature distribution at the top of the region, using a specified temperature gradient is also possible. Finally, the initial temperature distribution can be calculated to be the boiling-point temperature as a function of depth using a hydrostatic pressure distribution.

Mathematically, the initial conditions are

$$p = p^0(\mathbf{x}), \text{ at } t = 0 \text{ in } V, \tag{2.5.1a}$$

$$h = h^0(\mathbf{x}), \text{ at } t = 0 \text{ in } V, \text{or} \tag{2.5.1b}$$

$$T = T^0(\mathbf{x}), \text{ at } t = 0 \text{ in } V, \tag{2.5.1c}$$

where

p^0 is the initial fluid pressure distribution (Pa),
h^0 is the initial specific enthalpy distribution (J/kg),
T^0 is the initial temperature distribution (°C), and
V is the simulation region.

Pressure, enthalpy, and porosity fields calculated by one simulation can define the initial conditions for a subsequent simulation using two output files designed for this purpose, as described in section 4.6. This often is the easiest way to establish a steady-state flow field before heat transport is simulated. Of course, one needs to determine whether or not an initial steady-state flow field is reasonable for the physical situation being simulated. It should be noted that, with a hydrostatic or other approximation of initial pressure conditions, it could take a substantial amount of simulation time to establish the steady-state flow field to a given numerical tolerance.

Specifying initial conditions in two-phase, one-component zones can be problematic because, for a given pressure, small differences in enthalpy can cause large differences in liquid saturation. It is difficult to specify initial conditions that will give stable flow fields. Thus, it is generally preferable to generate the initial conditions containing two-phase zones by using a previous simulation that starts from a single-phase zone.

Special considerations must be taken when setting up initial conditions in nearly dry porous media. Flow rates in nearly dry media are very slow and much simulation time can be required to compute equilibrium (steady-state) flow conditions. When flow in the partially saturated zone is not important, simulations can be facilitated by specifying an initial pressure distribution that creates a greater water saturation in this zone than is physically realistic. Greater saturation allows for larger flow rates in the partially saturated zone enabling an equilibrium water-table configuration to be approached more rapidly than for the actual system. Simulating a draining system is usually easier than simulating a wetting system.

In simulations with a steady precipitation flux, a useful initial pressure distribution in the partially saturated zone is a uniform value corresponding to the saturation required to accommodate the specified precipitation flux. This approach gives a stable computation when the steady-state flow solution is of primary interest rather than the solution for transient infiltration into a nearly dry porous medium.

2.6 DIMENSIONLESS NUMBERS

Several dimensionless numbers can be used to characterize flow and thermal transport in a porous medium. Two have been selected for computation by the HYDROTHERM simulator; the Nusselt number and the thermal Peclet number.

2.6.1 Nusselt Number

The Nusselt number may be defined as the ratio of total (advective plus conductive) heat flux to conductive heat flux in a thermal transport system (Nield and Bejan, 1992, p. 148). When defined for local conditions, this number is a function of space and time for a given simulation. The original Nusselt number, defined for one-dimensional regions, was extended to three-dimensional regions by using the magnitudes of the various vector constituents. Thus a defining equation may be written as

$$Nu = \frac{|\rho \mathbf{q}|\, h + K_a |\nabla T|}{K_a |\nabla T|}, \tag{2.6.1}$$

where

Nu is the Nusselt number (dimensionless),
$|\rho \mathbf{q}|$ is the magnitude of the fluid mass flux (kg/s-m^2), and
$|\nabla T|$ is the magnitude of the temperature gradient (°C/m).

The Nusselt number ranges from 1 for pure conduction to ∞ for pure advection and indicates the relative importance of these thermal transport mechanisms in different parts of the simulation region.

2.6.2 Thermal Peclet Number

The thermal Peclet number is the ratio of advective heat flux to conductive heat flux in a thermal transport system. It is the analog to the Peclet number commonly used for solute transport simulation (de Marsily, 1986, p. 237). When defined for local conditions, this number is a function of space and time for a given simulation. A defining equation for a thermal Peclet number may be written as

$$Pe = \frac{L|\mathbf{q}|}{\kappa_a} \tag{2.6.2a}$$

with

$$\kappa_a = \frac{K_a}{(\rho c)_a}, \tag{2.6.2b}$$

where

Pe is the thermal Peclet number (dimensionless),
L is a characteristic length scale parameter (m),
$|\mathbf{q}|$ is the magnitude of the Darcy velocity (m/s),
κ_a is the effective thermal diffusivity of the bulk porous medium (combined liquid, gas, and solid phases) (m^2/s),
$(\rho c)_a$ is the effective heat capacity per unit volume of the bulk porous medium (combined liquid, gas, and solid phases) (J/m^3-°C), and
K_a is the effective thermal conductivity of the bulk porous medium (combined liquid, gas, and solid phases) (fluid and porous matrix) (W/m-°C).

The thermal Peclet number can be extended from its original definition for one-dimensional regions to three-dimensional regions by defining it as the magnitude of the of the vector sum of the Peclet numbers in each of the three coordinate directions. The thermal Peclet number for each coordinate direction is computed

using the length scale and Darcy velocity in that direction. Thus,

$$Pe_x = \frac{L_x q_x (\rho c)_a}{K_a} \tag{2.6.3a}$$

$$Pe_y = \frac{L_y q_y (\rho c)_a}{K_a} \tag{2.6.3b}$$

$$Pe_z = \frac{L_z q_z (\rho c)_a}{K_a} \tag{2.6.3c}$$

so

$$Pe = (Pe_x^2 + Pe_y^2 + Pe_z^2)^{\frac{1}{2}}, \tag{2.6.3d}$$

where

Pe_d is the thermal Peclet number in coordinate direction d (dimensionless),
L_d is the characteristic length in coordinate direction d (m),
q_d is the Darcy velocity of the fluid in coordinate direction d (m/s), and
d is the coordinate direction; x, y, or z.

Like the Nusselt number, the thermal Peclet number also indicates the relative importance of advection and conduction as thermal transport mechanisms. It ranges from 0 for pure conduction to ∞ for pure advection. Presently in HYDROTHERM, the thermal Peclet number is computed only for locations in the single-phase liquid-water zone. These dimensionless numbers are applied to a discretized mesh in section 3.8 on numerical implementation.

3. NUMERICAL IMPLEMENTATION

This chapter covers the additions and modifications pertaining to the numerical calculations incorporated in the HYDROTHERM Version 3 simulator. New features and algorithms concerning equation discretization and boundary conditions are described. In order to make this program documentation complete, summaries of the numerical methods used in Versions 1 and 2 of HYDROTHERM also are included.

Calculating numerical solutions for a simulation involves several tasks. First, the partial-differential equations of flow and thermal-energy transport must be discretized in space and time. Then, various algorithms are used to compute parameters and to implement boundary conditions. Next, the nonlinear, discretized, governing equations are solved simultaneously for each time step using the Newton-Raphson algorithm (section 3.6.1). The Newton-Raphson algorithm solves non-linear equations with a sequence of iterations or steps. Each Newton step requires the solution of a set of linear equations computed using one of two iterative linear-equation solvers. This chapter contains descriptions of the numerical implementation for each of these tasks.

3.1 EQUATION DISCRETIZATION

The classical method of finite differences (Anderson and others, 1984) is used to discretize the partial-differential equations (2.1.1 and 2.1.4), boundary conditions (eq. 2.4.1–2.4.3), and initial conditions (eq. 2.5.1) in space and time. First, the spatial discretization will be described, then the temporal discretization. Finally, the finite-difference approximations to the flow and heat-transport equations will be given.

3.1.1 Spatial Discretization

The simulation region is a bounded volume of three-dimensional space, often irregularly shaped, over which the flow and heat transport equations are solved. A right-handed, three-dimensional Cartesian coordinate system is applied to the simulation region with the positive z-axis pointing in the opposite direction to the gravitational force. A cylindrical coordinate system may be used for flow systems with cylindrical symmetry. Spatial discretization is done by constructing a mesh of cells that covers the simulation region. This mesh of cells is formed by specifying the cell dimensions in each of the three coordinate directions for the Cartesian-coordinate case and in each of the two coordinate directions for the cylindrical-coordinate case. The sequence of cell dimensions can be variable in each coordinate direction. These dimensions determine the cell sizes and mesh layout of the cells; then node points are located in the center of each cell. This type of mesh is called a cell-centered (block-centered) grid. Two-dimensional simulations can be done by defining a mesh that is only one cell thick in the third dimension, usually the y- or z-direction.

Note that care must be taken when specifying the thickness in the third dimension of the domain for a two-dimensional simulation. A very thin simulation domain may necessitate substantial experimentation with the settings of the time-step, Newton-Raphson, and iterative-solver control parameters in order to obtain a stable simulation.

For cylindrical-coordinates, the cell-boundary faces in the radial direction usually are relocated at the logarithmic mean of the square of the distance between each pair of nodes. Automatic generation of cell boundary and node locations in the radial direction is possible in HYDROTHERM. These locations are

determined by a geometric distribution of a specified number of cells and nodes between the inner and outer radius of the region.

A cell is a volume of simulation region that is a rectangular prism in Cartesian coordinates and an annulus with rectangular cross section in cylindrical coordinates. The mesh of all cells occupies a rectangular prism of space or a cylindrical volume, referred to as the global mesh region. The global mesh region completely contains the simulation region. A set of cell-boundary faces in the mesh is selected to approximate the shape of the simulation region and the cells belonging to these faces are referred to as boundary cells. They contain the corresponding boundary nodes. Boundary cells and cells internal to the simulation region form the set of active cells. The active cells comprise the active mesh region. The remaining cells of the mesh are inactive cells. In Cartesian coordinates, boundaries of the active mesh region are sets of cell faces, which form planes that must be normal to one of the coordinate directions. In cylindrical coordinates, boundaries of the active mesh region are sets of cell faces which form annular rings normal to the z-coordinate direction and cylinders coaxial with the z-axis. Simulation results are computed only for nodes in active cells.

A simple Cartesian-coordinate, cell-centered mesh for a simulation region is shown in figure 3.1. Some cells are excluded in order to approximate an irregular shape of the simulation region. Each cell contains a node at its center. Note that simulation region boundaries that are not parallel to a coordinate axis must be approximated by a staircase-like pattern of cell boundary faces. No provision is made to define boundary faces for the active mesh region that are obliquely oriented to the coordinate axes. The nodal dimensions of the global mesh are the maximum number of cells along each of the three coordinate axes, N_x, N_y, and N_z. The total number of nodes in the global mesh is N ($= N_x \times N_y \times N_z$).

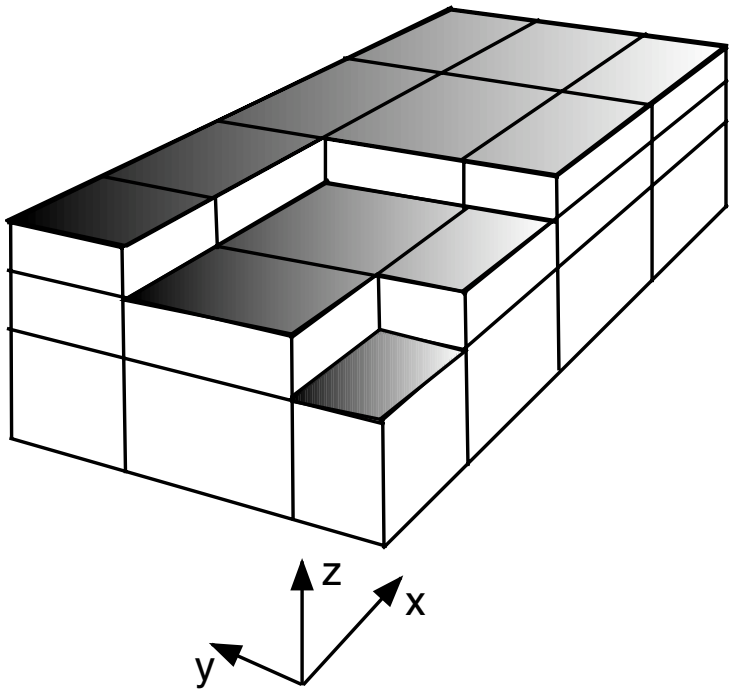


Figure 3.1. Cell-centered mesh in three dimensions with excluded cells.

The porous-medium flow and thermal-transport properties are discretized on a cell basis, with a set of cells of uniform properties forming a zone (fig. 3.2).

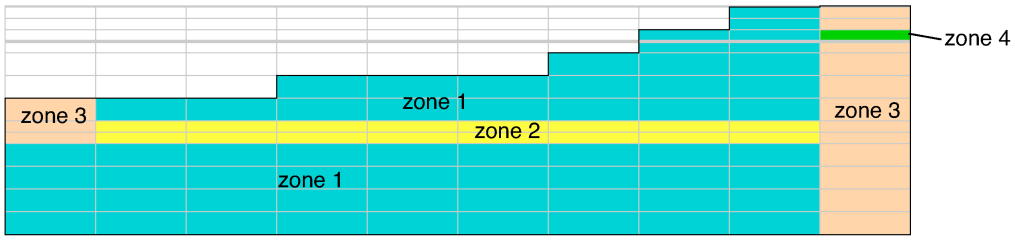


Figure 3.2. Zones in a two-dimensional mesh.

A zone is a contiguous set of cells with the same porous-medium properties, which is denoted as a rock type in the data file described in chapter 5 and a rock unit in the preprocessor window described in chapter 7. The dependent variables that are properties of the fluid are discretized on a cell-by-cell basis, as are the boundary-condition fluxes and source flow rates.

Some flow and transport properties are evaluated at cell faces. Transmissivities (eq. 3.1.3e) and thermal conductance coefficients (eq. 3.1.5b) between adjacent nodes are calculated as harmonic means. Fluid densities and viscosities are calculated as arithmetic means. Upstream weighting can be used for relative permeabilities (eq. 3.1.3f) and fluid enthalpies (eq. 3.1.5d) to suppress numerical oscillations. There is an option to change from upstream weighting to an arithmetic average weighting for relative permeability and enthalpy when the driving force for fluid flow is less than a specified threshold. Below the threshold, the upstream weighting factor is gradually converted to an arithmetic average using a logarithmic function. This option facilitates smooth spatial transitions in flow direction. Further details are given in section 3.2. Alternatively, cell-face weighting can be used, which corresponds to interpolation of relative permeability and enthalpy to the cell-face location. For a uniform mesh, cell-face weighting becomes centered-in-space weighting.

For flow and thermal-energy transport in the partially saturated zone, full upstream weighting always is used for the relative permeability, velocity, and enthalpy in the advective terms of the transport equation. Although upstream weighting suppresses numerical oscillations, it introduces numerical dispersion.

3.1.2 Temporal Discretization

To approximate the time derivative terms of equations 2.1.1 and 2.1.4, a backward-in-time (fully implicit) differencing method is used. The spatial finite-difference terms and source terms are evaluated at the end of the time step. The general form for the temporal discretization of the flow or heat-transport equation is:

$$\frac{(a_m u_m)^{(n+1)} - (a_m u_m)^{(n)}}{t^{(n+1)} - t^{(n)}} = F^{(n+1)} \tag{3.1.1}$$

where

a_m is the fluid or thermal capacitance coefficient for cell m (first term of eq. 3.1.2 or 3.1.4) [(m-s) or (kg/s)],
u_m is the pressure or enthalpy for cell m [(Pa) or (J/kg)], and
F is the discretized spatial derivative terms and source terms from equations 3.1.2 or 3.1.4 [(kg/s) or (W)].

Backward-in-time differencing does introduce numerical dispersion (see section 3.1.4) that must be kept under control by limiting the size of the time step. In HYDROTHERM, the parameters of the automatic time-step algorithm are used to control the time-step length for this and other purposes.

3.1.3 Finite-Difference Flow and Heat-Transport Equations

The finite-difference approximation to the ground-water flow equation (2.1.1), for interior cell m, is

$$\frac{V_m}{\Delta t}\Delta_t\Big[\phi(\rho_w S_w + \rho_s S_s)\Big] - \Delta\Big[\tilde{T}_w\Big(\Delta p^{(n+1)} + \rho_w g\hat{\mathbf{e}}_z\Big)\Big] - \Delta\Big[\tilde{T}_s\Big(\Delta p_g^{(n+1)} + \rho_s g\hat{\mathbf{e}}_z\Big)\Big] - V_m(q_w + q_s) = 0 \quad (3.1.2)$$

where

$$V_m = \Delta x_i \Delta y_j \Delta z_k, \quad (3.1.3a)$$

$$\Delta_t u = u^{(n+1)} - u^{(n)}, \quad (3.1.3b)$$

$$\Delta\tilde{T}_p\Delta u = \sum_{d=1}^{3}\Delta_d\tilde{T}_{pd}\Delta_d u, \quad (3.1.3c)$$

$$\Delta_d\tilde{T}_{pd}\Delta_d u = \tilde{T}_{pd}|_{i+\frac{1}{2}}(u_{i+1,j,k} - u_{i,j,k}) - \tilde{T}_{pd}|_{i-\frac{1}{2}}(u_{i,j,k} - u_{i-1,j,k}), \quad (3.1.3d)$$

$$\tilde{T}_{px_{i\pm\frac{1}{2}}} = \frac{k_{xx}k_{rp}\rho_p}{\mu_p}\bigg|_{i\pm\frac{1}{2}}^{(n)}\frac{\Delta y_j \Delta z_k}{\Delta x_{i\pm\frac{1}{2}}}, \quad (3.1.3e)$$

$$k_{rp}|_{i\pm\frac{1}{2}} = \sigma k_{rp,ups} + (1-\sigma)k_{rp,dns}, \quad (3.1.3f)$$

$$\Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}, \text{ and} \quad (3.1.3g)$$

$$\Delta x_{i+\frac{1}{2}} = x_{i+1} - x_i, \quad (3.1.3h)$$

and where

$\tilde{T}_{pd}$ is the fluid transmissivity for phase p in coordinate direction d; p=w(water) or s(steam) (m-s),
u is the pressure or enthalpy [(Pa) or (J/kg)],
$k_{rp,ups}$ is the relative permeability to phase p of the upstream cell (dimensionless),
$k_{rp,dns}$ is the relative permeability to phase p of the downstream cell (dimensionless), and
σ is the spatial weighting factor (dimensionless).

The spatial weighting factor is 0.5 for centered-in-space differencing and 1.0 for upstream-in-space differencing.

The finite-difference approximation to the heat-transport equation (2.1.4), for interior cell m, is

$$\frac{V_m}{\Delta t}\Delta_t\Big[\phi(\rho_w S_w h_w + \rho_s S_s h_s) + (1-\phi)\rho_r h_r\Big] - \Delta\tilde{T}_c\Delta T^{(n+1)}$$

$$- \Delta\left[\tilde{T}_w h_w\left(\Delta p^{(n+1)} + \rho_w g\hat{\mathbf{e}}_z\right)\right] - \Delta\left[\tilde{T}_s h_s\left(\Delta p_g^{(n+1)} + \rho_s g\hat{\mathbf{e}}_z\right)\right] - V_m q_h = 0, \quad (3.1.4)$$

where

$$\Delta\tilde{T}_p h_p \Delta u = \sum_{d=1}^{3}\Delta_d\tilde{T}_{pd} h_p \Delta_d u, \quad (3.1.5a)$$

$$\tilde{T}_{cx_{i\pm\frac{1}{2}}} = K_a\big|_{i\pm\frac{1}{2}}\frac{\Delta y_j \Delta z_k}{\Delta x_{i\pm\frac{1}{2}}}, \quad (3.1.5b)$$

$$\Delta_d\tilde{T}_{pd} h_p \Delta_d u = \tilde{T}_{pd} h_p|_{i+\frac{1}{2}}(u_{i+1,j,k} - u_{i,j,k}) - \tilde{T}_{pd} h_p|_{i-\frac{1}{2}}(u_{i,j,k} - u_{i-1,j,k}), \text{ and} \quad (3.1.5c)$$

$$h_p|_{i\pm\frac{1}{2}} = \sigma h_{p,ups} + (1-\sigma)h_{p,dns}. \quad (3.1.5d)$$

and where

$h_{p,ups}$ is the enthalpy of phase p in the upstream cell (J/kg),
$h_{p,dns}$ is the enthalpy of phase p in the downstream cell (J/kg), and
$\tilde{T}_{cd}$ is the thermal conductance in coordinate direction d (W/°C).

The coefficients $\tilde{T}_{pd}$ are evaluated at time level n. The permeability tensor in the flow equation is a diagonal matrix in the finite difference equations because the coordinate system is chosen to be along the principal directions of this tensor. These directions are assumed to be uniform throughout the simulation region.

3.1.4 Numerical Oscillation and Dispersion Criteria

As described in sections 3.1.1 and 3.1.3, variable weighting for spatial discretization of the advective term in the thermal-energy transport equation is available. Either upstream-in-space differencing or centered-in-space differencing may be selected. Here the term centered-in-space will be used to mean cell-face weighting so as to include cases where the cell face is not midway between the nodes on either side. Centered-in-space differencing can cause oscillations in the enthalpy solutions. This can be particularly problematic for pressure and enthalpy values near phase boundaries in the thermodynamic tables. Analysis of one-dimensional flow and transport with no sources, constant coefficients, and a uniform mesh (Fletcher, 1991, p. 286–293) shows that numerical oscillation does not occur if

$$Pe_c = \frac{q\Delta x}{\kappa_a} \le 2, \quad (3.1.6)$$

where

Pe_c is the cell thermal Peclet number (dimensionless),
q is the Darcy velocity (m/s), and
Δx is the cell size (m).

Upstream-in-space differencing introduces artificial numerical dispersion into the thermal transport solutions. Again, analysis of one-dimensional flow and transport with no sources, constant coefficients, and a uniform mesh (Fletcher, 1991, p. 302–303) yields a numerical dispersion coefficient given by

$$D_{ns} = \frac{q\Delta x}{2}, \tag{3.1.7}$$

where

D_{ns} is the numerical dispersion coefficient from spatial discretization (m^2/s).

Backward-in-time differencing also introduces artificial numerical dispersion into the thermal transport solutions. Again, analysis of one-dimensional flow and transport with no sources, constant coefficients, and a uniform mesh yields the numerical dispersion coefficient from temporal discretization given by

$$D_{nt} = \frac{q^2\Delta t}{2}, \tag{3.1.8}$$

where

D_{nt} is the numerical dispersion coefficient from temporal discretization (m^2/s).

These terms result from the truncation error of the finite-difference approximation.

For numerical accuracy, the total numerical dispersion should be much less than the physical thermal diffusivity. Thus, for upstream-in-space and backward-in-time differencing,

$$q\Delta x + q^2\Delta t \ll 2\kappa_a \tag{3.1.9a}$$

or

$$Pe(1 + Cr) \ll 2, \tag{3.1.9b}$$

where

Cr is the cell Courant number, $Cr = q\Delta t/\Delta x$ (dimensionless).

If only upstream-in-space or only backward-in-time differencing is used, then only the Δx term or only the Δt term of equation 3.1.9a is used, respectively. Although these equations are from a simplified, restricted one-dimensional analysis, they provide guidance for discretization of more complex three-dimensional simulations. Note that checking for excessive numerical dispersion is left to the HYDROTHERM user as no dispersion criteria calculations are done by the simulator.

3.1.5 Automatic Time-Step Algorithm

Two-phase flow simulation, with strong non-linearities in the equations and with transient source and boundary-condition terms, requires automatic time-step selection that can adapt to changing conditions of the numerical computation. In general, the more rapidly the conditions change, the smaller the time steps will need to be for a stable computation or for an accurate solution. Small time steps will be needed at the beginning of a simulation if the boundary conditions are greatly different from the initial conditions.

Therefore, the HYDROTHERM simulator has an automatic time-step selector that uses an empirical algorithm adapted from Aziz and Settari (1979, p. 403). The user specifies the maximum values considered acceptable for changes in pressure, enthalpy, and liquid saturation, as well as the minimum time step allowed. These maximum values also are considered to be target values for changes in the dependent variables. Then, at the beginning of each time step, the following adjustments are made, depending on the conditions:

$$\text{if } |\Delta u_{\max}| > 1.1\Delta u^s_{\max} \text{ or if } |\Delta u_{\max}| < 0.9\Delta u^s_{\max}, \text{ then } \Delta t_n = \Delta t_{n-1}\,\frac{\Delta u^s_{\max}}{|\Delta u_{\max}|}, \tag{3.1.10a}$$

with the constraint that

$$\Delta t_{\min} \le \Delta t_n \le f\Delta t_{n-1}, \tag{3.1.10b}$$

where

u	is the pressure, enthalpy, or liquid saturation [(Pa) or (J/kg) or (dimensionless)],
$\lvert\Delta u_{\max}\rvert$	is the magnitude of the maximum change calculated in u over the previous time step [(Pa) or (J/kg) or (dimensionless)],
$\Delta u^s_{\max}$	is the specified maximum change in u [(Pa) or (J/kg) or (dimensionless)],
Δt_n	is the new time step (s),
Δt_{n-1}	is the previous time step (s), and
f	is the factor for maximum increase of the time step (dimensionless).

For confined flow, the maximum changes allowed are specified as a percent change for pressure and enthalpy and an absolute change for saturation. In contrast, for unconfined flow, the maximum change allowed for pressure is specified as an absolute change, because pressures can range from positive to negative. Thus, a percent change in pressure would give too much influence to cells with low pressure found near the water table. However, pressure changes in the partially saturated zone, where pressure is less than 0.5 atm, are not considered for time step control. Because actual changes in the dependent variables are not directly proportional to time-step length, the specified maximum changes can be considerably exceeded with an adjusted time step based on a linear predictor. Therefore, the growth of a time step is limited to the factor, f, specified by the user, usually in the range of 1.1 to 2.

The new time step is selected to be the minimum of the three values calculated on the basis of changes in pressure, enthalpy, and saturation. This algorithm tends to increase the time step such that the maximum acceptable change in either pressure, enthalpy, or saturation is achieved with each time step as the simulation progresses.

There are several other controls on the length of the time step. Any of the following events may reduce the time step, usually by a factor of 0.5: (1) if the linear equation solver fails to converge in the maximum number of iterations allowed, (2) if the Newton-Raphson algorithm fails to converge in the maximum number of steps allowed, (3) if more than a specified number of nodes change phase during a Newton step, or (4) if more than a specified number of nodes change phase during a time step.

If changes in the dependent variables or convergence failures or phase changes cause the time step to be reduced to below the minimum time step specified, or if the time step has been reduced more than the maximum number of times allowed, the simulator stops. The remedy to this problem is to increase the

maximum values acceptable or reduce the minimum time step allowed, depending on which approach will yield a sufficiently accurate simulation.

3.1.6 Discretization Guidelines

No set of rules exists that will guarantee an accurate numerical solution with a minimum number of nodes and time steps, even for the simple case of constant coefficients. However, the following empirical guidelines should be observed:

- For upstream-in-space or backward-in-time differencing, estimate the truncation error (numerical dispersion coefficient), using parameter values at their maximum values from a short trial simulation. That is, verify that the grid-spacing and time-step control parameters do not introduce excessive numerical dispersion.
- Check on spatial-discretization error by refining the mesh. However, this may be impractical for large regions. A check on temporal-discretization error is relatively easy to make by refining the time-step length for a short trial simulation.
- To adequately represent a sharp enthalpy or temperature front, span it with at least four or five nodes. A large number of nodes may be required if a sharp front moves through much of the region during the simulation. Compromises often will have to be made.
- Well flows that highly stress the aquifer require a small time step, after a change in flow rate, to control errors from explicit flow-rate allocation.
- The global-balance summary table (section 6.1) may indicate that the time step is too large by exhibiting large imbalance residuals, particularly if the saturation, density, and viscosity variations are large.

3.2 PROPERTY FUNCTIONS AND TRANSPORT COEFFICIENTS

Fluid properties for the water component including density, viscosity, and temperature as functions of enthalpy and pressure are contained within HYDROTHERM in tabular form. Values are extracted using bicubic interpolation (Press and others, 1989, p. 98). Values of the partial derivatives of density, viscosity, and temperature with respect to pressure and enthalpy, used for the Newton-Raphson algorithm, also are obtained from the tables using bicubic interpolation. Values in two-phase, one-component (water-steam) zones are interpolated between the values for saturated liquid water and saturated steam using the liquid saturation values and cubic splines (Press and others, 1989, p. 86). Bicubic interpolation preserves continuity of values and derivatives across the tables and at boundaries between different tables which cover the several regions of the phase diagram for water. Further details appear in Hayba and Ingebritsen (1994).

For some purposes, temperature is used instead of enthalpy as a primary variable. Values of enthalpy and density of water and steam as a function of pressure and temperature are obtained from the functions published with the steam tables of the National Bureau of Standards (Haar and others, 1984). These functions are too extensive to reproduce in this documentation.

In the partially saturated zone, pressures below the limits of the tables are encountered often during a simulation. A count is kept of these occurrences and reported at the end of the simulation in the summary table of `Calc_log.`*`xxx`* (chapter 6).

Relative permeability to liquid water and water vapor as a function of liquid phase and steam saturation

is obtained by evaluating the functions from one of the sets described in chapter 2; linear, Corey, or fracture flow.

In the partially saturated, air-water zone, the water saturation is evaluated from the capillary pressure using one of three functions available. The appropriate function (linear, Corey, or Cooley) depends on which relative permeability function set is selected. The Cooley water saturation function is associated with the Corey relative permeability function. No water saturation function is associated with the fracture-flow relative permeability function because this function is not usually used for unconfined flow conditions.

Simulation of water infiltration into nearly dry porous media is a very difficult numerical calculation. The moisture saturation-capillary pressure function is extremely nonlinear, requiring very small time steps for convergence of the Newton-Raphson nonlinear equation algorithm. These time steps can be orders of magnitude smaller than the time step size required for small time-discretization errors. Also, sharp moisture fronts can occur which require fine discretization in the vertical direction for both numerical accuracy of the solution and stability of the Newton-Raphson algorithm. Rates of change of the flow variables in the partially saturated zone are much slower than in the saturated zone. The Newton-Raphson algorithm may have difficulties converging when the length scale of transition from full saturation to residual saturation is much smaller than the cell thickness. Steep pressure gradients occur under these conditions. For typical porous media, the capillary pressure at residual water saturation ranges from about 5×10^3 Pa to 2×10^4 Pa. This corresponds to a liquid water phase pressure in the partially saturated zone above the water table of 8×10^4 to 9.5×10^4 Pa. At water pressures less than this range, essentially no flow takes place. If flow in the partially saturated zone is relatively unimportant for a simulation, reasonable time-step lengths can be used if an artificial relation of saturation to capillary pressure is adopted that provides higher saturation values than actually occur. For example, using the linear saturation function (section 2.2.8) and distorting the curve by increasing the capillary pressure at residual water saturation to a very high value, such as 5×10^6 Pa, causes any precipitation flux into a dry porous medium to quickly bring the saturation to well above residual. Thus, precipitation can move easily down to the water table. The requirements of long infiltration times using small time steps and thin cell dimensions in the vertical direction are mitigated. However, an artificially large amount of water will be held in storage above the water table.

The relative permeability terms and the advective terms of the finite-difference equations are weighted in space (sections 3.1.1 and 3.1.3). Either a default weighting or an adaptive algorithm can be selected. The default weighting provides spatial interpolation to the location of the cell face between two nodes (cell-face weighting). The adaptive algorithm provides spatial interpolation that responds to the Darcy flux across a given cell face. The driving force of the hydraulic pressure is used to characterize the magnitude and direction of the Darcy flux. The hydraulic pressure is the sum of the fluid pressure plus the gravitational pressure at a point. Thus, at a given node,

$$\Phi_m = p_m + \rho_m g z_m \tag{3.2.1}$$

where

Φ_m is the hydraulic pressure at node m (Pa),

p_m is the fluid pressure at node m (Pa),
ρ_m is the fluid density for cell m (kg/m^3), and
z is the elevation coordinate (m).

However, in variable-density ground-water flow, the driving force on a parcel of water can not be obtained from the gradient of the hydraulic pressure (Hubbert, 1956). Thus, it is not a true potential field.

The weighting factor varies from fully upstream-in-space weighting to centered-in-space (cell-face) weighting. For a substantial Darcy flux, indicated by a ratio of hydraulic pressures of adjacent cells that is greater than a user-specified threshold, full upstream weighting is used. For a ratio of hydraulic pressures less than the threshold, the weighting factor is adjusted toward centered-in-space weighting using a logarithmic function. If the hydraulic pressures are equal, centered-in-space weighting is used.

When two adjacent cells have different water saturations, the simulator adjusts the weighting factor away from upstream when one phase becomes nearly immobile in one of the cells. In laterally adjacent cells, as the relative permeability of a phase approaches zero (less than 0.01), the weighting factor of that phase is gradually adjusted, during the Newton-Raphson iterations, toward the weighting factor of the other phase. The same technique is used for vertically adjacent cells when both phases are flowing in the same direction. When there is counter-current flow (water down and steam up), the weighting factor for water is gradually adjusted to fully upstream, relative to the steam flow direction, and the weighting factor for steam is gradually adjusted to fully upstream, relative to the water flow direction, as the respective relative permeabilities approach zero.

The purpose of this adaptive algorithm for spatial weighting is to minimize numerical oscillation in the solution while introducing as little numerical dispersion as possible where necessary. For unconfined flow systems, fully upstream-in-space weighting is used for the relative permeability and thermal advective terms in the air-water zone, which is necessary to prevent non-physical oscillations in the solution (Forsyth, and others, 1995). It is important to note that small changes in the spatial weighting control parameters can substantially affect the numerical solution. Because the derivatives of the weighting factors with respect to changes in pressure are not included in the Newton-Raphson equations, no further adjustment of these factors is made after the third Newton step unless there has been a phase change in the given cell.

All the property functions for the porous media can be defined for each different porous medium (rock type). Up to nine different rock types can be defined for a given simulation region. The porosity of the porous medium as a function of pressure is obtained by evaluating the linear function of equation 2.2.13. Some properties also can be defined as functions of temperature. They include porosity, permeability, thermal conductivity, heat capacity, solid matrix density, and matrix compressibility. The available functions include piecewise constant, linear, and logarithmic (eqs. 2.2.14–2.2.16). Permeability also can be defined as a linear function of time (eq. 2.2.4).

Porosity and permeability also can be defined as functions of depth. The available functions include piecewise uniform, linear, exponential, and logarithmic (eqs. 2.2.1–2.2.3) with a different function selected for each porous medium (rock type). Recall that the porous-medium flow properties are discretized on a cell basis; thus, any properties defined as functions of depth result in the interpolated values at each nodal elevation being applied to each cell. Spatial variation of a property with depth is not represented within a cell. In the air-water zone, all of the porous matrix properties are constant and uniform.

3.3 SOURCE TERMS

With spatial discretization, line- and point-source terms become distributed to columns of cells and individual cells, respectively. Note that a well source and a point (node) source cannot coexist in the same column of cells.

3.3.1 Well Model

The well bore is a line source penetrating a column of cells in the discretized simulation region. Discretization of the well bore gives a set of cells, each containing a segment of well bore or well screen. In this discussion, the terms open well-bore interval and screened interval are synonymous. The well bore may be cased through some cells and thus not communicate with the aquifer there. The well-bore segment in a given cell must be represented as either screened or cased off over the entire cell thickness. Partially screened cells can not be represented. Because the well bore is often screened over the more permeable zones of the region formations, the screened intervals can be specified by sets of cells.

In the HYDROTHERM simulator, only one well can exist in a particular column of cells of the mesh. Multiple wells in a cell column must be represented by an equivalent single well, or the spatial grid must be refined to separate them. This restriction also applies to separate wells that are located in the same column of cells but are screened in different vertical intervals.

In the case of an unconfined aquifer with a well screened through the free surface, the screen length is *not* adjusted as the saturated thickness varies in time. This simplification may cause substantial errors if the drawdown at the wellbore becomes a large fraction of the screened interval.

Allocation of the flow rate of a well to each of the cells in its screened intervals is obtained by discretizing equations 2.3.5a,b to give:

$$Q_{wsi} = \frac{Q_w \left[\dfrac{k_w k_{rs}}{\mu_s} \right]_i L_i}{\sum_{k=k_L}^{k_U} \left[\left(\dfrac{k_w k_{rw}}{\mu_w} \right)_k + \left(\dfrac{k_w k_{rs}}{\mu_s} \right)_k \right] L_k} \tag{3.3.1a}$$

and

$$Q_{wwi} = \frac{Q_w \left[\dfrac{k_w k_{rw}}{\mu_w} \right]_i L_i}{\sum_{k=k_L}^{k_U} \left[\left(\dfrac{k_w k_{rw}}{\mu_w} \right)_k + \left(\dfrac{k_w k_{rs}}{\mu_s} \right)_k \right] L_k} \tag{3.3.1b}$$

where

Q_{wpi} is the volumetric well flow rate of phase p for the cell at layer index i; p=w(water) or s(steam) (m^3/s),

k_L is the layer index of the bottom segment of the well screen,
k_U is the layer index of the top segment of the well screen, and
L_k is the length of well bore at layer index k (m).

If the screened interval is not continuous from k_L to k_U, the well-bore length element, L_k, is set to zero for the cells penetrated by cased-off subintervals of well bore. This method of flow-rate allocation is known as allocation by mobility, because the terms in brackets in equations 3.3.1a,b represent fluid mobility factors.

Although the well source allocation is presented in terms of volumetric flow rates, to simplify data input in HYDROTHERM, well sources are specified as mass flow rates. These flow rates are exact for injection wells because the density of the injected fluid is known. They are approximate for production wells because the density of the produced fluid is a function of pressure and enthalpy conditions in all the cells contributing to the given well.

3.3.2 Point Source

A point source occupies one cell in the discretized simulation region and thus becomes a node source for identification in the data-input file. A node source is equivalent to a well source with a screened interval that is entirely contained within a single cell. Thus, a node source is handled similarly to a well source. Flow-rate allocation is entirely to the single cell containing the source. However, multiple node sources can exist in the same column of cells. Point source flow rates in HYDROTHERM also are specified as mass flow rates for simplification of data input.

3.4 BOUNDARY CONDITIONS

All boundary conditions are specified on a cell-by-cell basis over the discretized simulation region. That is, the minimum unit of spatial discretization is the volume of one cell for a specified-value boundary condition or the area of one cell face for a specified-flux boundary condition or a seepage-boundary condition. The default-boundary condition is that of no advective or conductive flux through the cell faces that form the external boundaries of the simulation region. Only one type of boundary condition can be assigned to a given cell. Therefore, if a cell has two or three regional-boundary faces, a choice must be made as to which type of boundary condition that cell will have. Such cells exist along edges and at corners of the discretized region.

3.4.1 Specified Pressure, Enthalpy, and Temperature Boundary Conditions

Specified pressure and enthalpy are the primary specified-value boundary conditions. Specified temperature is converted to specified enthalpy before application as a boundary condition. Specified-value boundary conditions are incorporated by replacing the discretized flow and transport equations for those nodes, by equations of the form of equations 2.4.1a,b defining the specified values, if the generalized-minimum-residual equation solver (section 3.6.2.2) is selected. If the slice-successive-over-relaxation equation solver (section 3.6.2.1) is selected, equations for these nodes are removed from the set of simultaneous linear equations to be solved, and the known boundary values are incorporated into the remaining equations. These boundary conditions are piecewise constant in time; thus, if they change, they change discontinuously. The value at time $t^{(n)}$ is taken to be the limit of the value at $t^{(n)} - \delta t$, as $\delta t \to 0$; that is, the jump in the boundary-condition value takes place *after* the time of change. Under backward-in-time differencing, this means that the

effective value of a boundary condition over a time step when a change occurs is the value at the end of the step. Note that a specified-value boundary condition applies to an entire cell, and, thus, does not realistically represent a value at the physical boundary of the region if the cell is thick in the dimension normal to the boundary surface.

Specified-value boundary conditions can be defined as spatial distributions with depth by specifying the value at the top of the regional mesh as a function of location and specifying the gradient with depth. Boundary-condition values are first defined together with the initial conditions, using the same data block (chapter 5), but they may be modified for subsequent time periods of a simulation. If a specified-pressure boundary condition is defined for a node and no specified or associated enthalpy (temperature) value is provided, the enthalpy (temperature) provided for the initial conditions will be used.

It should be noted that in a real flow system, a hydrostatic-pressure boundary condition will not be maintained under conditions of variable-density flow. Specification of a hydrostatic-pressure boundary condition using a uniform or other specified density distribution can cause disconcertingly large vertical flows to occur, when the boundary density changes due to fluid compressibility or changes in the boundary temperature. Also, boundary-pressure values need to be specified to four or five substantial digits to avoid small vertical flows caused by roundoff error.

3.4.2 Specified-Flux Boundary Conditions

Discretization of the flow and heat-transport equations causes the specified-flux boundary conditions to be incorporated into the finite-difference equations as source terms. Fluid fluxes, discretized on a cell-face basis, are input as volumetric flow rates; heat fluxes are input as energy flow rates. A positive flow is added to a boundary cell while a negative flow is withdrawn from a cell.

3.4.2.1 Precipitation-Recharge Boundary Condition. Precipitation recharge is specified as an areal distribution over the top surface of the global rectangular mesh region. Then the recharge flux is applied to the uppermost active cells whose external-boundary faces represent the land surface configuration, as described in section 3.1.1. At the present time only a net recharge flux may be specified. No evapotranspiration mechanism is available. An associated temperature must also be specified, which is used to calculate the enthalpy and density of the precipitation recharge.

The discretization of equations 2.4.2a,b gives, for land-surface boundary cell m,

$$Q_{Pm} = q_{Pm} A_{Pm} \tag{3.4.1a}$$

and

$$Q_{HPm} = h_{Pm} \rho_{Pm} q_{Pm} A_{Pm}, \tag{3.4.1b}$$

where

Q_{Pm} is the volumetric flow rate at a precipitation boundary for cell m (m^3/s),
q_{Pm} is the volumetric flux across the precipitation-recharge boundary for cell m ($m^3/s\text{-}m^2$),
A_{Pm} is the surface area of the precipitation boundary (land surface) for cell m (m^2),

Q_{HPm} is the heat flow rate at a precipitation boundary for cell m (W),
h_{Pm} is the specific enthalpy of the precipitation for cell m (J/kg), and
ρ_{Pm} is the density of the precipitation for cell m (kg/m^3).

If the recharge rate is greater than the flux permitted by the permeability value of the porous medium, a warning message is written and the simulation terminates.

3.4.2.2 Basal Heat-Flux Boundary Condition. The basal heat-flux boundary condition may be applied only to the bottom boundary of the mesh region, but may have an areal distribution. The limitation is that the bottom boundary of the active mesh region must be a single horizontal plane at the bottom of the global mesh. A regional bottom boundary elevation that varies in space cannot be used when this boundary condition is employed. The discretization of equation 2.4.3 yields, for boundary cell m:

$$Q_{HCm} = q_{HCm} A_{HCm} \tag{3.4.2}$$

where

Q_{HCm} is the basal heat flow rate for cell m (W),
q_{HCm} is the specified basal conductive heat flux for cell m (W/m^2),
A_{HCm} is the surface area of the basal heat flux boundary for cell m (m^2).

3.4.3 Seepage-Surface Boundary Condition

The numerical implementation of a seepage surface in HYDROTHERM involves a boundary condition that can switch between specified pressure and specified fluid flux of zero as the simulation progresses. The former characterizes a seeping cell and the latter a nonseeping cell. In addition, a seeping cell can only discharge fluid from the active mesh region. For a seeping cell, the boundary condition for the thermal energy transport equation is that of advective heat transport leaving the region at the enthalpy (temperature) of the boundary cell. For a nonseeping boundary cell, the boundary condition is that of no heat flux. The test for switching between a seeping and nonseeping boundary condition is as follows: a seeping cell is switched to nonseeping if the seepage rate becomes inflow to the region, a nonseeping cell is switched to a seeping cell if the pressure rises above atmospheric pressure. This test is applied after the establishment of initial conditions and at convergence of the Newton-Raphson steps for the current time step. If any cells switch condition, the next time step is taken with the new state at these cells. The advective heat flow for a seeping cell is lagged one Newton-Raphson step in the finite-difference equations to simplify the calculation. The seepage rate is calculated as the negative of the sum of the internal boundary flows. There is no change in storage because the pressure is fixed over the time step. There is also no source allowed in a seepage-boundary cell. Small time steps are needed to control potential instabilities resulting from application of the boundary condition switching only at the end of a sequence of Newton-Raphson steps.

This algorithm for a seepage-surface boundary condition is similar to that used in the VS2D simulator described by Lappala and others (1987). The boundary condition is nonlinear because the position of the junction between the seeping cells and the nonseeping cells on the boundary is a transient unknown and is determined as the simulation progresses. A major algorithm difference is that the seeping length of the surface of seepage is adjusted after every Newton-Raphson step in VS2D instead of at the end of every time step.

In the cell-centered grid, the boundary condition for a seeping cell (specified atmospheric pressure) is applied to the entire boundary cell. For the case of a seepage surface along a vertical boundary of the simulation region, applying atmospheric pressure to that vertical column of boundary cells will cause a strong downward vertical driving force for ground-water flow through this column. In the actual physical system, this driving force would be only at the vertical boundary of the region and would cause vertical runoff down this boundary. The nonphysical flows caused by the cell-centered boundary condition approximation can be minimized by making the cell width of the column of seepage-boundary cells narrow relative to the width of the adjacent cells. Nonseeping cells are assembled with a no-flow boundary at the seepage surface. Variation in relative permeability may cause steady-state flow solutions derived from a draining problem and a rewetting problem to differ.

A given boundary cell can have either a precipitation-recharge flux boundary condition or a seepage-boundary condition but not both. Furthermore, a seepage-boundary cell can not receive precipitation recharge, nor can a precipitation recharge boundary cell become a seepage-boundary cell during a time step. Along a sloping land surface, with a boundary condition of a seepage surface in combination with precipitation recharge, the contact point between the two boundary conditions is transient and is determined as the simulation progresses. Some simulators (Forster and Smith, 1988) iteratively determine the location of the contact point.

To avoid having to determine and iteratively relocate the contact point, the following approach can be used in HYDROTHERM. The region is first discretized giving a stair-step approximation to the sloping land surface. Then extra columns of thin facing cells are defined at the vertical boundary faces. These cells are specified as seepage-boundary condition cells. The boundary cells along the stair-step boundary with land-surface boundary faces that are only horizontal are specified as precipitation-recharge boundary cells. This approach removes the need to iteratively determine the contact point between the recharge part and the seepage-surface part of the boundary, and has the restriction that a cell not change type of boundary condition during the simulation. Of course, each seepage cell may or may not be seeping at any given time. A limitation of this approach is that precipitation recharge continues to be applied to a cell that lies between two seepage-boundary cells that are seeping. The physical situation would indicate that this middle cell also should be a seeping boundary cell, at least for a homogeneous medium. Continuing to apply recharge in this situation creates an unrealistically large seepage outflow from the adjacent seepage cell downgradient along the land surface.

Alternatively, a boundary can be specified as having a combination precipitation-recharge and seepage-surface boundary condition. At the beginning of the simulation, those cells with pressure greater than atmospheric are assigned as seeping-boundary condition cells and those cells with pressure less than atmospheric are assigned as precipitation boundary condition cells. Atmospheric pressure is applied to the seeping cells and precipitation flux is applied to the precipitation-recharge cells. At the end of each time step, a re-evaluation is done and cells are switched to the appropriate type of boundary condition. No provision for switching during the Newton-Raphson iteration steps is available. This approach is more realistic than locking in the type of boundary condition for each boundary cell in the combination boundary-condition region, but it may require small time steps because of the explicit switching of boundary-condition type.

A minor computational limitation is that no sources of fluid or heat can be defined in seepage-boundary condition cells. Furthermore, if the discretized region has a stack of cells with a vertical land-surface boundary, only the top cell can have the combination seepage/precipitation boundary condition. The underlying cells can have only a seepage-boundary condition. In other words, a seepage-boundary condition cell can have precipitation recharge only if it has a land-surface boundary face whose normal points in the z-direction.

Another limitation of the seepage-boundary condition algorithm is that the initial-condition temperature of a seeping cell may change slightly (by a few tenths of a degree Celsius) at the beginning of a simulation. This happens, for example, when a hydrostatic pressure distribution and an enthalpy field are specified as the initial condition. The pressures will be greater at lower elevations along the seepage surface, and the initial-condition temperature field is calculated from these initial pressure and enthalpy values. In this case, a seeping cell is assigned atmospheric pressure which yields a slightly higher temperature recalculated from pressure and enthalpy during the first time step of the simulation. This is a case of the initial conditions being somewhat incompatible with the boundary conditions. However, the effect of this incompatibility diminishes with simulation time.

3.5 INITIAL CONDITIONS

The numerical implementation of the initial conditions is straightforward. Values of pressure and enthalpy are set to the initial value distributions for each node in the simulation region, that is:

$$u_m = u_m^0 \quad \text{at } t = 0, \tag{3.5.1}$$

where

u_m^0 is the initial value of pressure or enthalpy for cell m [(Pa) or (J/kg)].

The specified distributions can vary on a node-by-node basis or be zones of uniform conditions such as rows or columns. The initial pressure distribution also can be selected to be hydrostatic equilibrium, calculated from a specified pressure at the top of the active region. Calculated initial pressure distributions are based on the initial enthalpy (temperature) distributions. An initial enthalpy distribution can be calculated from an initial temperature distribution.

The option for a hydrostatic-equilibrium pressure distribution takes fluid compressibility and current temperature profile into account. The calculation proceeds from the top node of the region downward using a predictor-corrector method with iteration to handle the density dependence on pressure. The pressure along the top of the region, which may be the land surface, is specified for the uppermost active cells only.

For an unconfined flow simulation with an unsaturated zone, it is common to specify an initial pressure distribution that is linear through the saturated zone and constant through the unsaturated zone. The piecewise linear pressure distribution option can be used to setup this initial condition.

When specifying the initial pressure field on a node-by-node basis, one value must be included for each node in the global mesh region. Pressure values specified for nodes that are outside the active simulation region are simply neglected.

Spatial distributions with depth of initial conditions for enthalpy or temperature can be defined by

specifying the value as a function of location at the top of the global mesh and specifying the gradient with depth. Specified-value boundary condition data for the first simulation period are defined simultaneously with the initial conditions in the data-input file. However they may be changed for any simulation period including the first.

3.6 EQUATION SOLUTION

The governing fluid-flow and thermal-energy-transport equations are strongly coupled and highly nonlinear, because many of the coefficients are functions of the dependent variables. Relative permeability, liquid saturation, fluid density, and viscosity vary strongly with pressure and enthalpy. Thus, a robust nonlinear algorithm is necessary to solve the finite-difference approximations to equations 2.1.1 and 2.1.4. These two equations are in residual-rate form because the units, in HYDROTHERM, of the fluid-flow equation are grams per second and the units of the energy-transport equation are ergs per second and all of the terms in both equations sum to zero. Thus, any errors in the numerical solutions to the finite-difference approximations to the governing equations produce non-zero residual rates. The solution technique should find the numerical solutions for fluid pressure and enthalpy which reduce the residual rates to zero within a user-specified tolerance.

3.6.1 Newton-Raphson Algorithm

The Newton-Raphson algorithm (Stoer and Bulirsch, 1993, p. 262) is used to solve the system of nonlinear difference equations for each time step. The discretized flow equation and the thermal-energy transport equation are solved simultaneously for fluid pressure and enthalpy. These nonlinear difference equations are written for flow and heat transport in sequence for each active node. The equations, written in residual-rate form for N_a active nodes, are

$$\mathbf{f}(\mathbf{u}) = 0, \tag{3.6.1}$$

where

- $\mathbf{f}$ is the vector of residual rates of the nonlinear finite-difference equations from equations 2.1.1 and 2.1.4 [(kg/s) or (W)],
- $\mathbf{u}$ is the 2×1 block vector of unknowns (pressure, enthalpy) of length N_a [element units: (Pa), (J/kg)], and
- N_a is the number of active nodes in the global mesh.

The vectors $\mathbf{f}$ and $\mathbf{u}$ are of length $2N_a$ which can be partitioned into sets of two components, because there are two equations with two unknowns for each active node point. For simplicity, assume that all nodes are active, so that $N_a = N$. The Newton-Raphson method, generalized to a set of equations, involves solving,

$$\mathbf{J}(\mathbf{u}^{(\nu)})(\mathbf{u}^{(\nu+1)} - \mathbf{u}^{(\nu)}) + \mathbf{f}(\mathbf{u}^{(\nu)}) = 0, \tag{3.6.2a}$$

with

$$\mathbf{J}(\mathbf{u}^{(\nu)}) = \left[\begin{array}{ccc} \dfrac{\partial \mathbf{f}_1}{\partial \mathbf{u}_1} & \cdots & \dfrac{\partial \mathbf{f}_1}{\partial \mathbf{u}_N} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \dfrac{\partial \mathbf{f}_N}{\partial \mathbf{u}_1} & \cdots & \dfrac{\partial \mathbf{f}_N}{\partial \mathbf{u}_N} \end{array} \right]_{\mathbf{u}=\mathbf{u}^{(\nu)}}, \tag{3.6.2b}$$

and

$$\mathbf{u}^{(\nu+1)} - \mathbf{u}^{(\nu)} = \begin{bmatrix} \mathbf{u}_1^{(\nu+1)} - \mathbf{u}_1^{(\nu)} \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{u}_N^{(\nu+1)} - \mathbf{u}_N^{(\nu)} \end{bmatrix}, \tag{3.6.2c}$$

where

J is the Jacobian 2×2 block matrix of size $N_a \times N_a$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)], and

$\mathbf{u}^{(\nu)}$ is the vector of pressure and enthalpy values at each active node at iteration ν [element units: (Pa) or (J/kg)].

The Jacobian of equation 3.6.2b is a sparse matrix because each node is only connected to its six nearest-neighbor nodes in a three-dimensional, Cartesian-coordinate mesh or its four nearest-neighbor nodes in a cylindrical-coordinate mesh. Actually, the Jacobian is a block matrix comprising 2×2 element sub-matrices derived from the two difference equations written for each active node. For a simulation region comprising a single two-dimensional slice, the Jacobian becomes a 2×2 element block penta-diagonal matrix. The Jacobian matrix consists of partial derivatives of the residual functions with respect to the dependent variables, pressure and enthalpy, at each node. In the partially saturated zone of an unconfined aquifer, the terms involving the partial derivatives of saturation with respect to pressure are incorporated into **J**. However, the partial derivatives of saturation with respect to enthalpy, of enthalpy with respect to pressure, and of porous matrix heat capacity with respect to pressure and temperature are assumed to be zero. Furthermore, the partial derivatives of liquid density with respect to pressure, of temperature with respect to pressure, and of liquid viscosity with respect to pressure are neglected. The partial derivative of temperature with respect to enthalpy is evaluated at the low pressure limit (0.5 atm) of the equation-of-state table. Finally, the boundary-condition fluxes are treated as explicit in the Newton equations, thus, any variation in flux with pressure or enthalpy changes over the Newton step is neglected.

Equation 3.6.2a is solved iteratively to convergence. Each iteration is referred to as a Newton step, because each solution, shown as equation 3.6.2c, is expressed as a step from the previous solution. Three tests are applied to determine convergence of the Newton-Raphson algorithm. The primary convergence test is that the infinity norms of the fluid-mass residual rates and the energy-residual rates for all the active cells are less than user-specified limits. The infinity norm of a vector is the absolute value of the largest element found in the vector (Stoer and Bulirsch, 1993, p. 184). The residual rates for fluid mass and thermal energy are actually

converted into rates of mass and energy per unit volume of cell, expressed as grams per second per cubic centimeter and ergs per second per cubic centimeter, respectively. This means that small and large cells have equal weight in the residual-norm calculation. Separate limits for both fluid mass and energy norms are specified for Newton-Raphson convergence for historical reasons. The previous versions of HYDROTHERM used unscaled Newton-Raphson equations where the coefficients of the energy equations were on the order of 10^{10} larger than those of the fluid flow equations. In HYDROTHERM Version 3, the Newton-Raphson equations are scaled, as described in section 3.6.2, before they are solved.

Boundary cells with specified pressure and enthalpy and seepage-surface cells that are seeping are not included in the convergence test because their residuals do not become small as the solution is approached. Actually, their residuals determine the flow rates through the regional boundary faces of these cells.

The secondary convergence test, invoked when a phase change has occurred for one or more cells, is that the infinity norms of the percent changes in pressure and enthalpy over a Newton step are less than user-specified limits, while the infinity norms of the residuals are less than the limits for the primary convergence test amplified by a factor of 100. This test is not applied until three Newton steps have been taken. For unconfined ground-water flow systems, the absolute change in pressure is used instead of the percent change in pressure because pressures can range from positive to negative. Thus, using the percent change in pressure would give too much influence to cells with pressure near zero.

The tertiary convergence test is that the infinity norm of the change in pressure is less than 1 dyne/cm^2 and the infinity norm of the change in enthalpy is less than 100 erg/g. This test is not applied until after four Newton steps have been taken, and detects that the Newton-Raphson algorithm has become stationary even though the residual rates are greater than the tolerance limits.

The three convergence tests are applied in order after four Newton steps as an algorithm to find a solution of acceptable accuracy for a given time step under a wide variety of fluid-pressure, temperature, and phase conditions. The parameter values that specify the convergence criteria for Newton-Raphson iterations for each time step (see section 5.2.2) must be chosen carefully, as solutions of poor accuracy can result if the values for residuals and dependent-variable changes are set too large.

3.6.2 Linear Equation Solution

The set of linearized Newton finite-difference equations (eq. 3.6.2a) are solved by one of two solution algorithms for linear, sparse-matrix equations, available in HYDROTHERM: a slice-successive-over-relaxation (SSOR) method, or a generalized-minimum-residual (GMRES) method. Both are iterative methods for three-dimensional meshes, while the SSOR solver becomes a direct method for two-dimensional meshes comprising one vertical slice.

The Newton-Raphson coefficients for the thermal transport equations are about 10 orders of magnitude larger than the coefficients for the ground-water flow equations. Therefore, to reduce the round-off error for the successive overrrelaxation solver and to improve the effectiveness of the preconditioning for the minimum residual solver, the Newton-Raphson difference equations (eq. 3.6.2a) are scaled before they are solved. Scaling also enables one convergence tolerance parameter to be used for both the flow and heat-transport equations. Both row and column scalings (Jennings, 1977, p. 115) are done using infinity norms and Euclidian

norms (Stoer and Bulirsch, 1993, p. 184) as scale factors for the SSOR solver and the GMRES solver, respectively.

3.6.2.1 The Slice-Successive-Overrelaxation Solver. The slice-successive-overrelaxation technique is one of the class of block-iterative methods described by Varga (1962, p. 199). For this method, each x-z plane of nodes or slice (forming a block) is solved by direct elimination. One iteration cycle consists of solving for the nodal values for every plane of nodes using a direct, sparse-equation solver with band-matrix storage. Overrelaxation is used to speed convergence, with the optimum overrelaxation factor provided by the user.

The set of linearized Newton equations (eq. 3.6.2a) for a mesh with N_a active nodes forms a 2×2 block matrix equation,

$$\mathbf{A}\,\mathbf{u} = \mathbf{b}, \tag{3.6.3}$$

where

- **A** is a 2×2 block matrix of size $N_a \times N_a$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)],
- **u** is a 2×1 block vector of unknowns (pressure, enthalpy) of length N_a [element units: (Pa), (J/kg)], and
- **b** is a right-hand-side 2×1 block vector of the difference equations of length N_a [element units: (kg/s), (W)].

Each element of **A** is a 2×2 submatrix, and each element of **u** and **b** has 2 components, because there are two equations and two unknowns for each node point. A generic stationary iterative method can be written as:

$$\mathbf{u}^{(\nu+1)} = \mathbf{u}^{(\nu)} - \mathbf{B}^{-1}[\mathbf{A}\mathbf{u}^{(\nu)} - \mathbf{b}], \tag{3.6.4}$$

where

- **B** is a 2×2 block matrix of size $N_a \times N_a$ that approximates **A** [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)], and
- ν is the iteration counter.

B is chosen to be easy to invert.

For simplicity, assume that all nodes are active, so $N_a = N$. Then, in equation 3.6.3, matrix **A** is partitioned into $N_y \times N_y$ blocks each containing equations for one x-z nodal plane, where $N_x \times N_z \times N_y = N$:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \cdots & \mathbf{A}_{1N_y} \\ . & & . \\ . & & . \\ . & & . \\ \mathbf{A}_{N_y1} & \cdots & \mathbf{A}_{N_yN_y} \end{bmatrix} \tag{3.6.5a}$$

$$= \begin{bmatrix} \mathbf{D}_1 & \mathbf{U}_{12} & & \mathbf{0} \\ \mathbf{L}_{21} & . & & \\ & . & . & \\ & & . & \mathbf{U}_{N_{y-1},N_y} \\ \mathbf{0} & & \mathbf{L}_{N_y,N_{y-1}} & \mathbf{D}_{N_y} \end{bmatrix} \tag{3.6.5b}$$

$$= \mathbf{D} - \mathbf{L} - \mathbf{U}, \tag{3.6.5c}$$

where

$\mathbf{A}_{ij}$ is a 2×2 block matrix of size $N_x \times N_z$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)],
$\mathbf{D}_i$ is a 2×2 block matrix of size $N_x \times N_z$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)],
$\mathbf{L}_{ij}$ is a diagonal 2×2 block matrix of size $N_x \times N_z$ (appropriate units),
$\mathbf{U}_{ij}$ is a diagonal 2×2 block matrix of size $N_x \times N_z$ (appropriate units),
$\mathbf{D}$ is a 2×2 block matrix of size $N \times N$ [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)],
$\mathbf{L}$ is a 2×2 block lower triangular matrix of size $N \times N$ (appropriate units), and
$\mathbf{U}$ is a 2×2 block upper triangular matrix of size $N \times N$ (appropriate units).

The matrix splitting is given by equation 3.6.5c where the sparse submatrices $\mathbf{D}_i$ are penta-diagonal and the submatrices $\mathbf{L}_{ij}$ and $\mathbf{U}_{ij}$ are diagonal. These matrices are of size $N_x \times N_z$, where the nodes have been numbered with the x-direction index increasing most rapidly, the z-direction index increasing next most rapidly, and the y-direction index increasing least rapidly. Choosing

$$\mathbf{B} = \frac{1}{\omega} [\mathbf{D} - \omega \mathbf{L}] \tag{3.6.6}$$

and

$$\mathbf{u}^{(\nu+1)} = \omega \mathbf{u}_{GS}^{(\nu+1)} + (1 - \omega)\mathbf{u}^{(\nu)}, \tag{3.6.7}$$

where

$\mathbf{u}_{GS}$ is the solution vector using the Gauss-Seidel algorithm [element units: (Pa), (J/kg)],

the matrix equation for SSOR is (Stoer and Bulirsch, 1993, p. 594),

$$\mathbf{u}^{(\nu+1)} = \mathbf{u}^{(\nu)} - \omega[\mathbf{D} - \omega \mathbf{L}]^{-1}[\mathbf{A}\mathbf{u}^{(\nu)} - \mathbf{b}] \tag{3.6.8a}$$

or

$$[\mathbf{D} - \omega \mathbf{L}]\mathbf{u}^{(\nu+1)} = [(1 - \omega)\mathbf{D} + \omega \mathbf{U}]\mathbf{u}^{(\nu)} + \omega \mathbf{b}, \tag{3.6.8b}$$

where

ω is the over-relaxation factor.

Equation 3.6.8b is a sparse matrix equation stored in band format and is solved, for each iteration, by direct Gaussian elimination without pivoting.

The iterations are terminated when a relative norm of the change in the solution vector is less than a specified tolerance (Jennings, 1977, p. 184). That is, when the relative norm of the change in $\mathbf{u}$,

$$\frac{||\mathbf{u}^{(\nu+1)} - \mathbf{u}^{(\nu)}||}{||\mathbf{u}^{(\nu+1)}||} \leq \tau_{SOR} \tag{3.6.9}$$

where τ_{SOR} is the specified tolerance. In HYDROTHERM, $\mathbf{u}$ is chosen to be only the vector of change in pressure at each node over a time step. The vector of change in enthalpy at each node is not used for the termination test. Under this convergence criterion, the SSOR solver iterations may terminate prematurely when very slow convergence rates are encountered. To use the SSOR equation solver, the user must provide values for the over-relaxation factor, the convergence tolerance, and the maximum number of iterations allowed. Default values are provided. The slice-successive-over-relaxation algorithm reduces to a direct-elimination band solver of equation 3.6.3 for two-dimensional meshes. In this case, no iterations are necessary.

3.6.2.2 The Generalized-Minimum-Residual Solver. An alternative iterative solver for the linear, sparse-matrix equations has been added to HYDROTHERM based on a preconditioned generalized minimum residual method (Kelley, 1995; Greenbaum, 1997; Barrett and others, 1994; Saad, 2003). The GMRES method is a Krylov subspace method which, in contrast to SSOR, does not have a stationary iteration matrix, $\mathbf{B}^{-1}$ in equation 3.6.4.

For simplicity, assume all nodes are active so that $N_a = N$. Now, the linear matrix equation (eq. 3.6.3) has a residual vector of

$$\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{u}. \tag{3.6.10}$$

The approximate solution vector, $\mathbf{u}$, is found which minimizes the Euclidian norm of the residual at each iteration over the initial solution vector plus the current Krylov subspace. A set of Krylov subspaces is defined by

$$\mathrm{K}_\nu = \mathrm{span}\left\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \cdots, \mathbf{A}^{\nu-1}\mathbf{r}_0\right\}, \tag{3.6.11}$$

where

K_ν is the Krylov subspace at iteration ν.

The Krylov subspaces are based on the matrix $\mathbf{A}$ and the sequence of residual vectors $\mathbf{r}_\nu$ from all the previous iterations. Starting with an initial guess of the solution vector $\mathbf{u}_0$, the GMRES method obtains the next estimate of the solution $\mathbf{u}$ by

$$\mathbf{u}_\nu = \mathbf{u}_0 + \mathbf{Q}_\nu \mathbf{y}_\nu, \tag{3.6.12}$$

where

$\mathbf{Q}_\nu$ is an $N \times \nu$ matrix of orthonormal basis vectors for the Krylov subspace, K_ν, and

$\mathbf{y}_\nu$ is a vector chosen to minimize the Euclidean norm of the current residual vector [element units: (kg/s), (W)].

The $\mathbf{y}_\nu$ vector is from the solution of a least-squares minimization problem with ν typically a small number. The full GMRES method incurs a workload and storage per iteration that increases linearly with the iteration count. Therefore, a modified algorithm, GMRES(m), is used by simply restarting GMRES every m iterations, using the latest solution vector as the initial guess for the next GMRES cycle (Barrett and others, 1994, p. 21).

The restart interval, m, is specified by the user; there is no algorithm for choosing m. This keeps the cost of computing **Q** and **y** reasonable, but, unfortunately, the restarted algorithm may have a slower convergence rate than the full method.

Compressed-sparse-row storage (Barrett and others, 1994, p. 64) is used for the sparse linear-equation system to minimize computer-memory requirements. The application of specified-pressure or specified-enthalpy boundary conditions and accommodating inactive nodes in the global mesh for the GMRES solver is done by inserting two trivial equations into equation 3.6.2a for each of the specified-value nodes and the inactive nodes. The numerical implementation of GMRES for HYDROTHERM was adapted from software originally developed by Saad (1990) using algorithms described in Saad (2003).

GMRES(m) is a competitive solver only when some type of preconditioning is used to accelerate the convergence rate, even when using a long restart interval. The preconditioning implemented in HYDROTHERM is based on incomplete lower-upper (ILU) triangular factorization as described by Meijerink and van der Vorst (1977) and Saad (2003).

3.6.2.3 Equation Preconditioning. The preconditioned linear system, for right-side preconditioning, is

$$\mathbf{A}\mathbf{M}^{-1}\mathbf{M}\mathbf{u} = \mathbf{b}, \tag{3.6.13}$$

where

M is a 2×2 block matrix of size $N \times N$ that is a preconditioning matrix for **A** [element units: (m-s), (kg^2/J-s), (W/Pa), (kg/s)].

M is constructed to be an approximation of matrix **A** such that **Mu** = **w** is easy to solve, where **w** is an intermediate vector of unknowns [element units: (Pa), (J/kg)]. Note that the sequence of residuals, $\mathbf{r}_\nu$, is not modified by right-side preconditioning. Equation 3.6.12 for the solution vector becomes

$$\mathbf{u}_\nu = \mathbf{u}_0 + \mathbf{M}^{-1}\mathbf{Q}_\nu\mathbf{y}_\nu. \tag{3.6.14}$$

ILU factorization is a powerful preconditioning technique for use with iterative methods applied to sparse linear systems of equations. For ILU preconditioning,

$$\mathbf{M} = \mathbf{L}\mathbf{U}, \tag{3.6.15}$$

where **L** and **U** are approximate lower and upper triangular factors of matrix **A**. In HYDROTHERM, two types of ILU factorization can be done depending on the criteria used to form the approximate **L** and **U** factors.

The first type is ILU factorization with a threshold drop strategy; ILU(τ_{ILU}), uses a drop tolerance τ_{ILU} as an absolute or relative drop criterion. New fill-ins are accepted only if they are greater in magnitude than τ_{ILU} for the absolute criterion, or if they are greater than τ_{ILU} times the L_1-norm of a_{ij} in row i divided by the number of nonzero elements in row i for the relative criterion. The relative criterion works better for badly scaled matrices and is used in HYDROTHERM. Note that it is impossible to predict the storage requirement for this ILU(τ_{ILU}) factorization. Saad (1994) achieved a predictable storage by limiting the number of nonzero elements to be retained in each row of the ILU factors. The preconditioner ILU(τ_{ILU}, k) uses the dual threshold strategy of a fixed drop tolerance and a fixed number, k, of fill-ins per row keeping at most the k elements of largest magnitude. This preconditioner has been found to be quite effective on a wide range of matrices with suitable values of τ_{ILU} and k. Using a drop tolerance of zero and a maximum fill equal to the order of **A** yields

an exact LU factorization with all the work and storage of a direct solution of the matrix equation. This limiting case will usually be prohibitively expensive in computation time, so it is never used. The ILU(τ_{ILU}, k) preconditioner will be referred to as ILUT.

The second type is ILU factorization with a given level of fill, ILU(k). A level-of-fill value is attributed to each matrix entry that occurs during the incomplete factorization process. Each potential fill-in element is caused by the summation of products of pairs of entries in the factored matrix up to the current step. Fill-in values are dropped based on their level-of-fill value. The initial level-of-fill value of a matrix element is defined as

$$F_{ij} = \begin{cases} 0, & \text{if } a_{ij} \neq 0 \text{ or } i = j \\ \infty, & \text{otherwise,} \end{cases} \tag{3.6.16a}$$

where

F_{ij} is the level of fill (integer) of matrix element a_{ij} during factorization, and
a_{ij} is an element of matrix **A** [(m-s) or (kg^2/J-s) or (W/Pa) or (kg/s)].

Each time this element is modified by the ILU process, its level of fill is updated by

$$F_{ij} = \min\left(F_{ij},\, F_{ik} + F_{kj} + 1\right). \tag{3.6.16b}$$

For ILU(k), all fill-in elements whose level is greater than k are dropped, with k specified by the user as a nonnegative integer. Level zero, ILU(0), means that, during the ILU factorization, no fill-in is allowed of any locations with a null-element in the original **A** matrix. Level one, ILU(1), means that fill-in of null element locations is allowed only when it results from products containing at least one original **A** matrix element. Level two, ILU(2), means that fill-in of null element locations is allowed only when it results from products containing at least one level-one fill element. Higher levels are defined similarly. This type of factorization was motivated by the fact that, for diagonally dominant matrices, the higher the level of fill of an element, the smaller the magnitude of the element tends to be. In many cases, an ILU(1) preconditioner gives much improvement over an ILU(0) preconditioner (Benzi, 2002). Once a level of fill, k, is specified, the storage requirements for the ILU(k) factorization can be calculated before factorization is performed. The ILU(k) preconditioner will be referred to as ILUK.

The numerical implementation of ILU preconditioning for HYDROTHERM also was adapted from software originally developed by Saad (1990) using algorithms described in Saad (2003). To employ the GMRES equation solver, the user must choose the type of ILU preconditioning method, provide either the level of ILU factorization or the threshold tolerance and the maximum number of fill elements per row of the L and U matrices, and provide values for the convergence tolerance and the maximum number of iterations allowed. Default values are provided in the simulator.

3.6.3 Choosing the Linear Equation Solver

The choice of equation solver depends on consideration of the computer memory storage requirement and the computational workload for a given simulation. Both factors are related to the mesh size (number of nodes) of the problem and to the rates of convergence of the iterative methods. The importance of each factor

depends on the computer system employed. Thus, it is difficult to give a definitive formula for the selection of the solver. As mesh size increases, the SSOR solver will become more costly than the GMRES solver because of the direct band solver used on the vertical slices of the mesh. The available computer memory and computational speed will determine when it is necessary to change from the SSOR to the GMRES solver. In general, all one-dimensional problems and small, two-dimensional problems with a few hundred nodes are most efficiently solved by the direct band solver of SSOR. The GMRES iterative solver usually is more efficient for problems with several thousand nodes or more. All three-dimensional problems are required to be solved with GMRES in HYDROTHERM. An indication of how large a problem is feasible with the GMRES solver is given by Sanford (2005), who reported that a field-scale simulation of several thousand years duration with about 10,000 nodes was run in several hours. Of course, the computation time is highly dependent on the computer processor speed, memory, compiler options, and libraries employed. The HYDROTHERM simulator writes out the storage requirements for the solver selected, as well as iteration counts per Newton step, if applicable.

There is also no quantitative guideline for choosing the most effective type of ILU preconditioning for the GMRES solver. Experience shows that the simpler, level-of-fill ILU method (ILUK preconditioning) may not give a strong preconditioner for matrices which are far from being diagonally dominant. In this case, a stronger preconditioner may be obtained using the more complex, drop-tolerance ILU method (ILUT preconditioning) because only the fill-in elements that contribute substantially to the quality of the preconditioner are employed. Limited experience shows a greater sensitivity of the GMRES convergence rate to the parameters of each preconditioning type rather than to the type chosen.

3.6.4 Choosing the Parameters for an Iterative Equation Solver

Both iterative solvers need several user-specified parameters. The exception is the SSOR solver when applied to one- or two-dimensional, vertical slice problems, because it then becomes a direct band solver. These parameters include a convergence tolerance and the maximum number of iterations allowed. Other parameters are specific to the solver chosen.

A limit on the iteration count prevents runaway conditions when the convergence rate becomes very slow. For some simulations, the user may need to double or triple the default limit of 100. However, more than a few hundred iterations indicates that adjustments probably need to be made in the spatial or temporal discretization or other parameters of problem definition.

3.6.4.1 SSOR Equation Solver. The test for convergence of the SSOR iterative solver is based on the change in the pressure solution from one iteration to the next approaching zero. The change in the solution is measured by the maximum norm of the vector of change in the solution scaled by the maximum norm of the current solution vector. Convergence is achieved when this scaled vector of change is less than the tolerance value. There is no simple, effective algorithm for specifying τ_{SOR} for a given problem. The user should test that the convergence tolerance is sufficiently small by rerunning the simulation with the tolerance reduced by a couple of orders of magnitude and verifying that the results do not change substantially. A short simulation period can be used to do this. Suitable values for τ_{SOR} fall in the range 10^{-4} to 10^{-10}. The default value is 10^{-5}.

The over-relaxation parameter must be between 1 and 2, with values between 1.3 and 1.7 usually near

optimal. The convergence rate of the SSOR solver can be quite sensitive to this parameter, and the user should experiment to determine a suitable value. Algorithms are available to estimate optimum values for this parameter, but none have been incorporated into HYDROTHERM. The default value is 1.3 with 1.0 used for the first iteration.

For one- and two-dimensional problems in a vertical plane, the SSOR iterative solver reduces to a direct band solver with no iterations needed. In this case, no solver parameters are required.

3.6.4.2 GMRES Equation Solver. Depending on the type of ILU preconditioning selected (ILUT or ILUK), one or two parameters, described in section 3.6.2.3, must be specified. For ILUT preconditioning, the drop tolerance (τ_{ILU}) and the maximum number of fill-in elements per row (k) are required. An acceptable value for the drop tolerance must be determined by trial-and-error, because it is strongly problem dependent and a sharp optimal value may not exist. Limited experience has shown that τ_{ILU} should be in the range of 10^{-2} to 10^{-4} and that k should be in the range of 5 to 15. Limited testing shows that the value of k should be changed by about 5 to affect the convergence rate substantially. The default values for τ_{ILU} and k are 10^{-3} and 5, respectively.

For ILUK preconditioning, the level-of-fill (k) is required. The workload on the preconditioner increases rapidly with k, and it rarely is worth using a high value. A suitable value is usually within the range of 0 to 10. The default value for level-of-fill for ILUK is 5.

The GMRES iterative solver is used to solve equation 3.6.10 by finding a numerical solution vector, **u**, such that the residual vector, **r**, is sufficiently small. The test for convergence of the GMRES solver is based on achieving a specified reduction of the Euclidean norm of the residual vector from its value at the start of iteration (Barrett and others, 1994, p. 54). The tolerance, τ_{GMRES}, is the maximum acceptable value of the ratio of the Euclidean norm of the residual vector to the Euclidean norm of the initial residual vector, expressed mathematically as,

$$\frac{||\mathbf{r}_v||_2}{||\mathbf{b}||_2} \leq \tau_{GMRES}, \tag{3.6.17}$$

where

$||\mathbf{r}_v||_2$ is the Euclidean norm of the residual vector [element units: (kg/s), (W)],
$||\mathbf{b}||_2$ is the Euclidean norm of the initial residual vector [element units: (kg/s), (W)], and
τ_{GMRES} is the tolerance for convergence of GMRES linear equation solver (dimensionless).

From equation 3.6.10, the initial residual vector, **b**, is the residual when the initial solution vector, **u**, is chosen to be zero. It is also the right-hand-side vector of the original linear matrix equation to be solved in equation 3.6.3.

It would be desirable to set τ_{GMRES} to achieve a specified limit on the error of the solution. The criterion of equation 3.6.17 yields the forward error bound on the solution of

$$||\mathbf{e}_v||_2 \leq \tau_{GMRES} \left|\left|\mathbf{A}^{-1}\right|\right| \cdot ||\mathbf{b}||_2, \tag{3.6.18}$$

where

$\mathbf{e}_\nu$ is the error vector of the solution at iteration ν [element units: (Pa), (J/kg)].

Thus, the norm of the error in the solution is related to the norm of the right-hand-side vector and the norm of the inverse matrix. Unfortunately, it is usually not feasible to compute the norm of the inverse matrix. Therefore, no algorithm is presently available to set this tolerance based on the problem specifications. The user should test that the convergence tolerance is sufficiently small by rerunning the simulation with the tolerance reduced by a couple of orders of magnitude and verifying that the results do not change substantially. A short simulation period can be used to do this. Experience with test problems has shown that a tolerance of 10^{-6} to 10^{-12} is necessary to obtain solutions that have converged to 3 or 4 significant digits. The default value for τ_{GMRES} is 10^{-10}.

The GMRES solver does not seem to be very sensitive to the number of iterations between restarts. The default value of 5 seems to work satisfactorily for most problems. If solver convergence requires more than about 100 iterations, the restart interval should be increased to 10 or 15.

3.7 GLOBAL-BALANCE CALCULATIONS

The discretized ground-water flow equation and thermal-energy transport equation represent fluid and energy balances over each active cell. Summing over the active cells and integrating over time yields global-balance equations that relate the total change of mass of ground-water and heat to the net boundary inflow and the net injection by wells and point sources.

Global-balance information is calculated for each time step and for the cumulative duration of the simulation. The information produced by the balance calculation includes change in fluid mass and heat in the system, and the mass and heat inflows and outflows. Changes in fluid mass and heat due to net inflow (inflow minus outflow) are aggregated by each type of boundary condition.

The temporal integration is done over each time step and over the cumulative duration of the simulation. The cumulative balance quantities are simply the sums of the corresponding time-step amounts. The fluid or heat global-balance residual is defined as the imbalance between the change in mass of fluid or heat in the simulation region and the net inflow of fluid or heat. A positive residual means that there is an excess of fluid or heat present in the active simulation region over what would be expected based on net transport across boundaries and net input from sources. A percent residual is defined as the ratio of the residual to the inflow, outflow, or magnitude of accumulation, whichever is larger. The utility of the fractional residuals is not great. It is more informative to look at the residuals relative to the various flows and accumulations in the region.

The fluid-flow and heat-flow rates at specified-value boundary cells are obtained by evaluating the residual of the finite-difference flow and heat-transport equations. These residuals are the fluid masses and heat amounts that are necessary to satisfy the fluid-balance and heat-balance equations. Thus, the fluid mass-balance equation and the heat-balance equations are satisfied exactly for each of the specified-pressure and specified enthalpy (temperature) boundary cells, respectively.

The primary use of the global-balance calculations is to aid the interpretation of the magnitudes of fluid and heat flows that are occurring in the simulated system, and their distribution among the various types of boundary conditions and sources. The global-balance rate equations are integrated over the current time step to obtain the incremental changes. The total amounts of liquid water, water vapor (steam), and super-critical

fluid in the region also are tabulated. Super-critical fluid is at pressures above the critical pressure for pure water.

A mass and energy balance with a small residual is a necessary but not a sufficient condition for an accurate numerical simulation. The finite-difference system equations are a balance for each cell, the flows at specified-value boundary condition cells are calculated to give zero residual, and the fluxes between the cells are conservative. Thus, errors in the global-balance equations may result from the following causes: (1) the approximate solution from the Newton-Raphson iterative algorithm for the nonlinear difference equations, (2) the approximate solution from the iterative linear equation solver, (3) errors of interpolation in the tables for the equations of state for water and steam, and (4) roundoff error in special cases, such as wide variation in parameter magnitudes. Errors caused by discretization in time or space will not be revealed by these global-balance calculations. However, the inaccuracies resulting from a time step that is too long under conditions of substantially nonlinear parameters will be evident. Substantial nonlinearity can be caused by large variations in saturation or by phase changes, for example.

3.8 DIMENSIONLESS NUMBERS

The dimensionless numbers for each active cell are computed using the defining equations 2.6.1 and 2.6.3a–2.6.3d discretized on a cell basis. Only the active, fully liquid-phase, interior cells have dimensionless numbers computed.

3.8.1 Cell Nusselt Number

The cell Nusselt number is obtained by computing the advective heat flux and the conductive heat flux for each interior cell of the region. The equation for a cell, m, is

$$Nu_m = \frac{|(\rho \mathbf{q})_m| h_m + K_{am} |\nabla T|_m}{K_{am} |\nabla T|_m}, \tag{3.6.19}$$

where

Nu_m is the Nusselt number for cell m (dimensionless),
$|(\rho \mathbf{q})_m|$ is the magnitude of the fluid mass flux for cell m (kg/s-m^2),
h_m is the specific enthalpy of the fluid phase for cell m (J/kg),
K_{am} is the effective thermal conductivity of the bulk porous medium for cell m (W/m-°C), and
$|\nabla T|_m$ is the magnitude of the temperature gradient for cell m (°C/m).

The advective heat flux is based on the magnitude of the mass fluxes for water and steam, interpolated to the node, and on the cell enthalpy value. A three-point difference formula is used to approximate the component of the temperature gradient in each of the three coordinate directions. These component values are combined to calculate the magnitude of the temperature gradient. Nusselt numbers are not computed at boundary cells because there is no convenient method to calculate the temperature gradients at the regional boundary faces.

3.8.2 Cell Thermal Peclet Number

The cell thermal Peclet number is computed as the root-mean-square of the thermal Peclet numbers for each of the three coordinate directions. Each coordinate component of the cell thermal Peclet number is obtained using the cell dimension in that coordinate direction as the length scale in equations 2.6.3a,b,c. The

Darcy flux is taken to be the value in each direction interpolated to the node. Thus,

$$Pe_{xm} = \frac{\Delta x_m q_{xm} (\rho c)_{am}}{K_{am}} \tag{3.6.20a}$$

$$Pe_{ym} = \frac{\Delta y_m q_{ym} (\rho c)_{am}}{K_{am}} \tag{3.6.20b}$$

$$Pe_{zm} = \frac{\Delta z_m q_{zm} (\rho c)_{am}}{K_{am}} \tag{3.6.20c}$$

so

$$Pe = (Pe_{xm}^2 + Pe_{ym}^2 + Pe_{zm}^2)^{\frac{1}{2}}, \tag{3.6.20d}$$

where

Pe_{dm}	is the thermal Peclet number in coordinate direction d for cell m (dimensionless),
Δd_m	is the cell dimension in coordinate direction d for cell m (m),
q_{dm}	is the Darcy velocity of the fluid in coordinate direction d for cell m (m/s), and
d	is the coordinate direction; x, y, or z.

Only cells with a single phase fluid, liquid water, have thermal Peclet numbers computed.

4. COMPUTER CODE DESCRIPTION

Version 3 of the HYDROTHERM computer code is written in Fortran 90. The code conforms to American National Standards Institute (ANSI) standards (American National Standards Institute, 1992) as closely as possible to provide maximum portability. Version 3.1 consists of a main program and 144 subprograms and 24 modules. The modules contain 31 of the subprograms. The program length is approximately 238,000 lines of code including comment lines and data statements. The thermodynamic data tables are included in the source code accounting for about 90 percent of the total lines of code. Versions 1 and 2 of HYDROTHERM were written in FORTRAN 77, and much of Version 3 retains the programming style of that version of the language. HYDROTHERM has been tested on personal computers and desktop workstations running Windows or Linux/Unix operating systems. The source code for HYDROTHERM is freely available (appendix 1) and is expected to be portable to other computer platforms provided a Fortran 90 compiler is installed. In addition, the library of Basic Linear Algebra Subroutines (BLAS-1) is required.

The computer code for the optional graphical user interface for HYDROTHERM is written in C and Java. The code conforms to American National Standards Institute (ANSI) standards for the C language (American National Standards Institute, 1989) and the Java language specification (Gosling and others, 2005) as closely as possible to provide maximum portability. The graphical-user-interface routines wrap around the simulator routines to form HYDROTHERM INTERACTIVE (HTI). When the HYDROTHERM simulator is executed from within HTI, the main HYDROTHERM routine is superseded by the pre- and post-processing routines of the HTI program. HTI has been tested on personal computers and desktop workstations running Windows or Linux/Unix operating systems. The source code for HTI is freely available (appendix 1) and is expected to be portable to other computer platforms provided a Fortran 90 compiler, a C compiler, a Java Development Kit and the BLAS-1 library are installed.

No description of the routines and their functions and no calling tree is given in this documentation. For such descriptions, see the header comments in each routine. Programming utility tools can be used to generate calling trees to illustrate the structure of the program.

4.1 CODE ORGANIZATION

The HYDROTHERM code is organized to perform transient simulations. Simulation of steady-state conditions in one computation step is not possible. The basic steps of a simulation are:

1. Openfiles, read, error check, initialize, and write output for problem specifications and perform memory-space allocation.
2. Read,error check, initialize, and write output for static and transient information.
3. Starta time-step by calculating flow- and transport-equation coefficients, applying the boundary conditions, and calculating the source terms.
4. Assembleand solve the finite-difference, Newton-Raphson equations for the flow and heat-transport equations, iterating to convergence on the residuals.
5. Repeatstep 4 with a smaller time step if convergence fails.
6. Calculatethe summary rates and cumulative amounts at the end of the time step.

7. Writethe output information for the time-step completed.
8. Returnto step 3 to proceed with the next time step with an adjustment in time-step length if necessary. March through time (steps 3–7) until the time for a change in boundary conditions or source terms occurs.
9. Returnto step 2 for a new set of transient data until the simulation is finished.
10. Write final summary information, close files, and terminate program execution.

If errors occur, information to that point in the simulation is written out and the simulation terminates.

4.2 MEMORY ALLOCATION AND SUBPROGRAM COMMUNICATION

Fortran 90 supports dynamic memory allocation. Thus, all arrays are dimensioned during program execution to the sizes necessary for a given simulation. There are groups of arrays whose size depends on the number of nodes in the global mesh, or the number of cells with a given type of boundary condition, or the number of wells, or the number of point sources, or the type of equation solver selected. The arrays in these groups are dimensioned to the lengths required as problem definition data are read from the data-input file and processed. Temporary arrays are allocated as needed and then deallocated. If an allocation fails, the simulator prints a message to the screen and to the calculation log file and then terminates.

The following principles were used in the design of subprogram communication for HYDROTHERM Version 3. Most variables and arrays are communicated to subprograms through "use association" of modules (Redwine, 1995, p. 185). A module is a Fortran 90 program unit which can contain collections of data specifications and enable sharing data between separate subprogram routines. Subprogram communication with arguments also is used for arrays whose identity changes for different calls to the same routine. Low-level utility programs are called with argument lists. Each subprogram has access to only the variables that it needs, as nearly as practical.

4.3 FILE USAGE

The HYDROTHERM program uses many sequential-access files stored on the computer hard-disc. Fortran unit numbers for these files, defined in the `f_units` module in the `modules.f90` file, can be modified by the user. The following lists give the file names and a brief description of the contents of each file. Within each list, the file names are alphabetical by primary name. All files are opened from within the program during initialization. File names are fixed or provided by the user or built from a suffix, *xxx*, provided by the user. Files are closed at the end of the simulation and unused and temporary files are deleted.

The following font conventions are used in the descriptions of the file names:

`Constant Width` denotes file names and file name elements that must be used as given;

`Constant Width Bold` denotes pathnames, program names, and menu names that must be entered literally by the user;

`Constant Width Italic` denotes file name elements for which a context-specific substitution must be made.

4.3.1 Input Files

The following input files are required for a normal simulation:

input_filename	Input file with comments (provided by the user)
Dat.strip	Input file without comments (temporary file written by the simulator)

In addition, the following input files are used for continuing a previous simulation:

IC_pressporo.*xxx*	Pressure and porosity fields from a previous simulation
Out_restartdump.*xxx*	Dump file of porosity, pressure, enthalpy, and temperature fields from a previous simulation

Restarting a simulation is described in section 4.6.

4.3.2 Output Files

A set of up to 26 output files can be produced from running a HYDROTHERM simulation. The following list briefly describes the contents of each output file. For more complete descriptions, see section 6.1. The files are listed alphabetically.

Calc_log.*xxx*	Calculation log with information for each time step
IC_pressporo.*xxx*	Ending pressure and porosity fields to be used as initial conditions for a future simulation
Out_Probdefine.*xxx*	Simulation specification information defining the problem
Out_balance.*xxx*	Fluid and energy global-balance tables for the simulation region
Out_bcflow.*xxx*	Boundary-cell flow rates in tabular column format
Out_bcflow2.*xxx*	Boundary-cell flow rates in mesh array format
Out_density.*xxx*	Density fields
Out_dimensionless.*xxx*	Dimensionless Nusselt and thermal Peclet number fields
Out_enthalpy.*xxx*	Enthalpy fields
Out_permeability.*xxx*	Permeability and relative permeability fields
Out_pmthermalprop.*xxx*	Porous-media thermal property distributions
Out_porosity.*xxx*	Porosity fields
Out_potential.*xxx*	Hydraulic pressure and potentiometric head fields
Out_pressure.*xxx*	Pressure fields
Out_residual.*xxx*	Residual fields for the finite-difference equations of flow and energy transport
Out_restartdump.*xxx*	Restart dump of porosity, pressure, enthalpy, and temperature fields
Out_saturation.*xxx*	Saturation fields
Out_source.*xxx*	Flow rate tables for well- and point-sources
Out_temperature.*xxx*	Temperature fields
Out_velocity.*xxx*	Interstitial velocity and mass flux fields
Out_viscosity.*xxx*	Viscosity fields
Out_watertable.*xxx*	Water-table elevation field (unconfined flow only)
Plot_scalar.*xxx*	Pressure, enthalpy, temperature, and saturation fields for postprocess

	visualization
`Plot_scalar2.`*xxx*	Water-table elevation field (unconfined flow only) for postprocess visualization
`Plot_vector.`*xxx*	Velocity vector components for postprocess visualization
`Plot_timeseries.`*xxx*	Time series data of pressure, enthalpy, temperature, and saturation for postprocess visualization

Files that contain multiple fields can result from output at specified time intervals or time-step counts and (or) output of water- and steam-phase data for a given variable. All files are opened at the beginning of the simulation. Because output to most of these files is optional, any unused files are deleted at the end of the simulation. The output files that are always retained after a simulation are `Calc_log.`*xxx* and `Out_Probdefine.`*xxx*.

4.4 INITIALIZATION OF VARIABLES

Certain input parameters are set to default values if no data or data values of zero are input. These values are indicated by { } in the input-record descriptions of table 5.1. Four tables are initialized for the density, viscosity, and temperature of pure water as a function of enthalpy and pressure, and for the derivatives of each of these parameters with respect to enthalpy and pressure needed for the Newton-Raphson equations. These four tables cover different regions of the pressure-enthalpy phase diagram for pure water. Two tables are initialized for the spline functions used for definition of the two-phase region and for fluid properties along regional phase boundaries. These six tables are initialized in the modules `prptabl1`, `prptabl2`, `prptabl3`, `prptabl4`, `splnT0`, and `splnws`. Additional data for calculation of enthalpy and density of water and steam as a function of pressure and temperature are initialized in the module `steam_table_funct`. These files initialize the data tables with DATA statements using one numerical value per line. Thus they are the largest files in the HYDROTHERM program comprising 214,000 lines. This manner of initialization makes the program portable between big-endian format and little-endian format computer systems (Overton, 2001, p. 24) without needing special compilation or running flags.

4.5 PROGRAM EXECUTION

There are several options for running a HYDROTHERM simulation. The first option is using command-line execution to run the stand-alone simulator. It consists of preparing a data-input file in ASCII text format, then invoking the HYDROTHERM simulator with a command line and, finally, examining the output files and plotting the results of interest. The second option is a completely interactive operation. It consists of using HYDROTHERM INTERACTIVE (HTI) for defining the simulation problem, invoking the simulator, observing graphical output during the simulation, and saving the output files for postsimulation replaying. Under this option, it is also possible to interrupt a simulation, modify the problem definition, and restart the simulation, all interactively. All of the output files from the stand-alone simulator can be produced using HTI. The third option is stand-alone interactive postprocessor simulation. It consists of using HTI postprocessor for invoking the simulator, observing graphical output during the simulation, and saving the output files for postsimulation replaying. Using this option, it is not possible to modify the problem definition. The fourth option is stand-alone interactive postprocessor replaying. It consists of using HYDROTHERM

INTERACTIVE postprocessor to replay the output files from a previous simulation. This option is good for reviewing previous simulation results.

4.5.1 Execution of the Stand-Alone Simulator

To run HYDROTHERM using the stand-alone simulator option, the data-input file must be prepared first. Information for preparation of this file is contained in chapter 5. A template-file of comments is provided in the software distribution to simplify the creation of the data-input file. Using command-line execution of the program, the HYDROTHERM simulator is launched by typing the pathname to the executable file on the command line, for example, using a standard installation on a Linux platform, **`../../bin/ht`**. The name of the data-input file and a suffix to identify the simulation output files are entered at the keyboard following the prompts. During the simulation, basic time-step information is written to the monitor screen. Error messages also are written to the screen as they occur. A detailed step-by-step procedure for running HYDROTHERM with a demonstration example is given in appendix 2.

4.5.2 Execution of the Interactive Simulator

With HTI, a graphical user interface is used to define a simulation problem, run a HYDROTHERM simulation, and visualize the output during execution and afterwards, all interactively. This alternative is available for simulations defined for *two-dimensional regions only.* Information for using HTI appears in chapter 7, whereas the explanations in chapter 5 of the various types of input data required, remain applicable. HTI is invoked by either entering the pathname to the executable script file (**`../../bin/hti`**) on the command line (Linux) or by clicking on the **`Start`** button, going to the **`Programs`** menu and selecting **`HTI`** (Windows). Using interactive graphical-user-interface execution of the simulator, the default name of the data-input file is `ht.in` and there is no suffix on the output file names. A detailed step-by-step procedure for running HTI with a demonstration example is given in appendix 2.

4.6 RESTART OPTION

Although the HYDROTHERM simulator is not able to restart a simulation from an intermediate time level, the program does have the option to continue a previous simulation from its ending time. This is useful if, for example, boundary conditions or sources need to be changed based on examination of the results of the simulation, or if a steady-state flow field needs to be established before a heat source is introduced. Note that this restart option is not available using interactive execution of HTI.

To enable a simulation to be continued, the user must specify, for the previous simulation, that pressure and enthalpy fields are to be written to the restart file, `Out_restartdump.`*`xxx`*. This file also contains the ending time value of the previous simulation which must be entered for *`tmstr`* in the data-input file for the continuation. If the porous matrix is compressible, the file of initial porosity and pressure fields, `IC_pressporo.`*`xxx`*, as well as the final porosity field, from `Out_restartdump.`*`xxx`*, are needed from the previous simulation. These files provide the initial conditions for continuation of the simulation.

Finally, a nearly complete data-input file is needed. Only the initial conditions for pressure, enthalpy, and, possibly, porosity are obtained from files written during a previous simulation. File names for these data are entered into the data-input file at the locations indicated in chapter 5.

5. THE DATA-INPUT FILE

This chapter describes the structure and content of the data-input file for HYDROTHERM, Version 3. Legacy data-input files for Versions 1 and 2 cannot be run by Version 3 directly. However, they can be edited to conform to Version 3 input requirements and formats using the information in this chapter. The data-input file is a set of ASCII character text lines (records). Each input record may have up to 256 characters. If any input records exceed this limit, an error message is written to the terminal screen and to the calculation log file and the simulator stops. The data-input file has two general characteristics: (1) it is free format for ease of preparation at a computer terminal, and (2) it may be freely commented for rapid identification of the data items. Free-format is supported by Fortran 90 and is generally referred to as list-directed input.

5.1 LIST-DIRECTED INPUT

The data values may be located in any position of a data record, provided they correspond in number and type with the input list of the corresponding Fortran read statement. Data items are separated by a comma or by one or more spaces or both. Character strings must be enclosed in quotation marks or apostrophes. A slash, /, terminates a record and any remaining items in the input list are left unchanged from their previous values. Some Fortran compilers require a space before the slash. A data item within an input list may be left unchanged by separating the preceding item from the subsequent item with two commas; in other words, making no entry for that item. At the start of a simulation, any unchanged data items will retain their initial or default values as set by the simulator.

The list-directed input of a record continues until the end-of-record slash is encountered, or the input list is satisfied. If the input list is not satisfied at the end of a data line, the program will continue to read additional data lines, until the list is satisfied, the end-of-record slash is encountered, or the end of the file is reached. Having an insufficient number of data items in the input file is a common error. It usually results from a misinterpretation of the amount or type of data required by a given data record or block.

List-directed input also allows for a repeat count. Data in the form

$$n*d; \tag{5.1}$$

where

n is the repeat count integer, and

d is the data item,

causes n consecutive values of d to be input.

5.2 PREPARING THE DATA-INPUT FILE

To simplify the preparation of the data-input file, a file is available containing only comment lines, shown in table 5.1. Comment lines begin with the "#" character, which may be preceded by spaces. These comment lines identify the groups of input data, show the syntax of the input records using descriptive or program-variable names, indicate the variable type (integer, logical, real), show the conditions for the optional items, and give the default values, where available. The user actually can enter the data between the lines of a copy of this file, guided by section 5.2.2. Each name shown in the list for an input record is either the variable name used in the HYDROTHERM program or a descriptive mnemonic. All names are defined in section 5.2.2.

Optional input records are indicated by (O) followed by the logical variables that must be true, or the numerical conditions that must be met for inclusion. The names and keywords in brackets identify the record where the conditional variable is read. For variables that have default values defined, entering a zero selects the default value.

The data-input file has two types of formats for historical reasons. The first part of the file is in a line-by-line sequential format. Each line must be included in the order listed in table 5.1 with the exceptions for optional lines described in section 5.2.2. The middle part is in a keyword-group format. The keyword groups may be included in any order, with the few exceptions also described in section 5.2.2. However, the total number of groups is fixed once the type of flow (confined or unconfined) and the coordinate system (Cartesian or cylindrical) have been specified. Within each keyword group, secondary keywords and parameters determine the format of the data block used for that group. The order of the records within each keyword group must be followed exactly as described for each of the various format options. The last part of the file is also in a line-by-line sequential format with a specified number of keyword groups.

The following section contains a line-by-line presentation of the data-input file including the syntax of the keyword groups of data with explanations of all the variables and options. An example data-input file appears in section 8.3 providing an illustration of many of the input formats.

Table 5.1. Template file of comment lines from the data-input form

```
# HYDROTHERM Data-Input Form
# Version 3.1
#      Notes:
#      Syntax of input line is described by # ..
#      Comment line begins with #
#      A leading letter indicates an exclusive record choice must be made,
#             i.e. Choose A or B or C
#      (O) - Optional data with conditions for requirement
#      [a|b] - Indicates that either A or B must be entered
#      {nnn} - Indicates that the default number, nnn, is used if a zero
#              is entered for that variable
#      (nnn) - No default but suggested value
#      [T/F] - Indicates a logical variable
#      [I] - Indicates an integer variable
#-------------------------------------------------------------------------
#-------------------------------------------------------------------------
#   Start of the data file
# TITLE
# ..  TITLE: title line 1
# ..  VER3:  title line 2
# DIMENSIONS
# ..  nx[I],ny[I],nz[I],tmstr,iyear[I]
# TIME STEP
# ..A  tstepmx[I],tmincr{1.3},%PchgTS{10.},%HchgTS{5.},Wsatn{0.03},relax{1.},
# ..       deltmin,n_tmcut{10}; (O) - confined flow
# ..B  tstepmx[I],tmincr{1.3},PchgTS{1.e5},%HchgTS{5.},Wsatn{0.03},relax{1.},
# ..       deltmin,n_tmcut{10}; (O) - unconfined flow
```

Table 5.1. Template file of comment lines of the data-input form—Continued

```
# NEWTON-RAPHSON ITERATION
# ..A  itermx{10},resmas{1.e-12},reseng{1.e-2},nphasechg{10},%PchgNR{0.1},%HchgNR{0.1};
# ..       (O) - confined flow
# ..B  itermx{10},resmas{1.e-12},reseng{1.e-2},nphasechg{10},PchgNR{1.e5},%HchgNR{0.1};
# ..       (O) - unconfined flow
# LINEAR EQUATION SOLVER
# .. slmeth[I]
#    SSOR SOLVER CONTROL
# ..A  ssormx[I]{10},tol{0.05},ssorw0{1.0}; (O) - slmeth = 1
#    GMRES SOLVER CONTROL
# ..B  m_save_dir{5},stop_tol_gmres{1.e-10},maxit_gmres{100},ilu_method{2},lev_fill{5},
# ..       drop_tol{0.001}; (O) - slmeth = 2
# WEIGHTING AND AVERAGING
# ..  ioptupst[T/F],potdif{0.0002},theta{1.0}
# RELATIVE PERMEABILITY
# ..  kodrp[I],wsatn(0.3),ssatn(0.0)
# ROCK PROPERTIES
# ..  heatcpty{1.e7},rxden{2.5},rxcmprss{0.},grav(981.),initphi[T/F]
# ..  initphi_filename{IC_pressporo.xxx}; (O) - initphi = T
# CYLINDRICAL COORDINATES
# ..  irad[T/F],wellrad(cm),radiusmax(cm)
# PRINT/PLOT CONTROL
# ..  wide[I],long[I],vert_avg[I]
# PRINT 1
# ..  pres_pr_intrv,enth_pr_intrv,temp_pr_intrv,satn_pr_intrv,iprsat[I]
# PRINT 2
# ..  dens_pr_intrv,iprden[I],vis_pr_intrv,iprvis[I],potential_pr_intrv,iprpot[I]
# PRINT 3
# ..  velocity_pr_intrv,iprvel[I],bcflow_pr_intrv,iprbcflow[I],source_pr_intrv,iprsource[I]
# PRINT 4
# ..  pm_properties_pr_intrv,iprmprop[I],poros_pr_intrv,permeability_pr_intrv
# PRINT 5
# ..  balance_pr_intrv,dimno_pr_intrv,iprdimno[I],residuals_pr_intrv,dump_pr_intrv
# PRINT 6
# ..  plotscalar_pr_intrv,plotvector_pr_intrv,plotfile_type[I],time_series_pr_intrv
# ..  num_locations,(i(n),j(n),k(n),n=1 to num_locations)(all [I]);
# ..  (O) - time_series_pr_intrv > 0
# SLICE number
# ..  start_location: [TOP|BOTTOM]
# ..  index_of_rock_type(i,k),i=1 to nx, k=1 to nz (or k=nz to 1)[I]
# UNCONFINED
# ..  unconfined[T/F],patm
# SATURATION FUNCTION
#     LINEAR
# ..A pwb,pwr; (O) -  kodrp = 0 and unconfined
#     COREY
# ..B pwb,lambda; (O) - kodrp = 1 and unconfined
#     COOLEY
# ..C pwb,bb,cc; (O) - kodrp = 3 and unconfined
# ----------------------------------
```

Table 5.1. Template file of comment lines of the data-input form—Continued

```
# Start of Keyword Data Block section
# PARAMETER
# Sample format for data block
# ..  array_identifier,(units)
# ..  format_specifier
# ..  data record set
# ........
# End of Keyword Data Block section
# ---------------------------------
# Transient data
# TIME PERIOD
# .. tchg,delt,nsrce[I],lchgpr[T/F],nparms[I]
# ---------------------------------
# Start of Source Keyword Data Block section
# SOURCE/SINK number; (O) - nsrce > 0
# ..  WELL,(unit_flow),(unit_heat)
# ..  i[I],j[I],well_flow,[well_enth|well_temp],n-openint[I]
# ..  NODE,(unit_flow),(unit_heat)
# ..  i[I],j[I],k[I],node_flow,[node_enth|node_temp]
# End of Source Keyword Data Block section
# ---------------------------------
# ..  PRINT/PLOT data records, PRINT n, as above; (O) - lchgpr = true
# ..  PARAMETER keyword data block records, as above; (O) - nparms > 0
# ---------------------------------
# Insert additional groups of transient data as necessary
# ---------------------------------
# TIME PERIOD: End of simulation record
# ..  tchg
-1 /
# ..  End of the data file
#-------------------------------------------------------------------------------
#-------------------------------------------------------------------------------
```

5.2.1 General Information

A leading letter (A, B, or C) associated with a record list in table 5.1 indicates that one of the lettered records must be selected exclusively. Keywords separated by a vertical bar indicates that one of them must be chosen. All variables are real numbers unless otherwise indicated. The notation [T/F] indicates a logical variable. Variables that are indices or the number of something are integer variables. Real variables may be entered as real numbers or integer numbers, while integer variables must be entered as integer numbers, and logical variables must be entered as t or f (case insensitive).

The HYDROTHERM simulator uses the centimeter-gram-second (cgs) units of measure internally. However, other units of measure are available for input and output of the various parameters and variables as listed in table 5.2 and in section 5.2.2. The time unit may be selected to be seconds or years. The output units will match the input units for the following parameters and variables that are input by keyword-data block: pressure, enthalpy, permeability, thermal conductivity, rock specific heat, rock density, rock compressibility, heat flux, precipitation flux, well flow rate, and node-source flow rate. The remaining parameters and variables that are not input by keyword data block must be input using cgs units. All pressure values are absolute.

Table 5.2. Units of measure available for the HYDROTHERM simulator

[**Bold** indicates the units used internally by the simulator]

Quantity (generic units)	Metric (SI) units	Alternate metric units
Mass (M)	kilogram (kg)	**gram** (g)
Length (L)	meter (m)	**centimeter** (cm), kilometer (km)
Time (t)	**second** (s)	year (yr)
Pressure (F/L^2)	pascal (Pa)	**dyne per square centimeter** ($dyne/cm^2$), bar (bar)
Thermal energy (E)	joule (J) or watt-second (W-s)	**erg** (erg)
Temperature (T)	**degree Celsius** (°C)	kelvin (K)
Density (M/L^3)	kilogram per cubic meter (kg/m^3)	**gram per cubic centimeter** (g/cm^3)
Viscosity (M/L-t)	pascal-second (Pa-s)	**poise** (P)
Permeability (L^2)	square meter (m^2)	**square centimeter** (cm^2)
Diffusivity (L^2/t)	square meter per second (m^2/s)	**square centimeter per second** (cm^2/s)
Thermal conductivity (E/L-t-T) or (F/t-T)	watt per meter-degree Celsius (W/m-°C)	**erg per second-centimeter-degree Celsius** (erg/s-cm-°C)
Specific heat capacity (E/M-T) or (FL/M-T)	joule per kilogram-degree Celsius (J/kg-°C)	**erg per gram-degree Celsius** (erg/g-°C)
Velocity (L/t)	meter per second (m/s)	**centimeter per second** (cm/s)
Volumetric flow rate (L^3/t)	cubic meter per second (m^3/s)	**cubic centimeter per second** (cm^3/s)
Volumetric flux ($L^3/t\text{-}L^2$)	cubic meter per second-square meter ($m^3/s\text{-}m^2$)	**cubic centimeter per per second-square centimeter** ($cm^3/s\text{-}cm^2$)
Mass flux ($M/t\text{-}L^2$)	kilogram per second-square meter ($kg/s\text{-}m^2$)	**gram per second-square centimeter** ($g/s\text{-}cm^2$)
Heat flux ($E/t\text{-}L^2$)	watt per square meter (W/m^2)	**erg per second-square centimeter** ($erg/s\text{-}cm^2$)

5.2.2 Input-Record Descriptions

The following font conventions are used in the descriptions of the input records:

`Constant Width`	denotes file names and other output from the simulator,
`Constant Width Bold`	denotes keywords and text that must be entered literally by the user, but are case insensitive,
`Constant Width Italic`	denotes variables for which a context-specific substitution must be made, and
[*a*|*b*]	means either *a* or *b* ***must*** be chosen.

For the parts of the data-input file input as sequential records, each input record or record group has a keyword identifier, in the form # KEYWORD, that identifies a particular record in the data-input form listed in

table 5.1. The lines that follow give the syntax of the data records, with parameter definitions, and default values. Some input records in the listing below occupy more than one line. With list-directed input, the number of lines in the data file used for a given record may be determined by the user. However, at least one data item should be included on each line. Where applicable, the default value for a data item is chosen by entering a zero or a blank separated by commas.

I. Problem Specification Data

TITLE

TITLE: *Title line 1.*

A character string of up to 80 characters. The keyword **TITLE** *must* appear at the beginning of the string.

VER3: *Title line 2.*

A second character string of up to 80 characters. The keyword **VER3** *must* appear at the beginning of the string to identify this version of input format.

DIMENSIONS

nx, ny, nz, tmstr, iyear

nx	Number of nodes in the x-direction for Cartesian coordinates, or the r-direction for cylindrical coordinates.
ny	Number of nodes in the y-direction. Unused for cylindrical coordinates, but a space in the input record must be included.
nz	Number of nodes in the z-direction.
tmstr	Time value at the beginning of the simulation (s), (yr). Usually zero for new simulations. For continuation simulations, the value read from the file `IC_pressporo.xxx` will override this value.
iyear	Index for selection of time units for simulation: **1** selects seconds, **2** selects years. All temporal data must be entered in the time unit selected, except for data where units of measure must be specified by the user.

TIME STEP

tstepmax, tm_incrmx, [*%PchgTS*|*PchgTS*], *%HchgTS, Wsatn, relax, mindelt, n_timcut*

tstepmax	Maximum number of time steps allowed for this simulation.
tm_incrmx	Factor for maximum increase of the time step. Default set at 1. 3.
%PchgTS	Maximum percent change in pressure allowed for setting the time step for confined flow (percent). Default set at 10 percent.
PchgTS	Maximum absolute change in pressure allowed for setting the time step for

unconfined flow (dyne/cm^2). Default set at 1×10^5 dyne/cm^2.

%HchgTS Maximum percent change in enthalpy allowed for setting the time step (percent). Default set at 5 percent.

Wsatn Maximum change in water saturation allowed for setting the time step (dimensionless). Default set at 0.03.

relax Maximum number of blocks through which a fluid particle can flow during a given time step (presently not used). Default set at 1.

mindelt Minimum time step required (s),(yr). Simulation aborts if time step is cut to less than this value, except at the first time step.

n_timcut Maximum number of times a time step can be cut and the solution recalculated. Simulation aborts if this value is exceeded. Default set at 10.

Automatic time-step control is provided by these parameters. If any of these parameters are exceeded for any time step, the time step is reduced and the solution is recalculated for the step.

NEWTON-RAPHSON ITERATION

itermx, resmas, reseng, nphasechg, [*%PchgNR*|*PchgNR*], *%HchgNR*

itermx Maximum number of Newton-Raphson iterations allowed for a given time step. If none of the convergence criteria are satisfied after this number of iterations, the time step is cut in half and repeated. Default set at 10.

resmas Maximum residual mass rate of change per unit volume for active cells (g/s-cm^3). Primary mass-balance convergence criterion for Newton-Raphson iteration steps. Default set at 1×10^{-12}.

reseng Maximum residual enthalpy rate of change per unit volume for active cells (erg/s-cm^3). Primary energy-balance convergence criterion for Newton-Raphson iteration steps. This value should generally be about 10 orders of magnitude greater than *resmas*, because the enthalpy of liquid water at 225°C is approximately 1×10^{10} erg/g. Default set at 1×10^{-2}.

nphasechg Maximum number of cells that are allowed to change phase to/from the 2-phase region or to/from the air-water region during one Newton-Raphson iteration. Default set at 10.

%PchgNR Maximum percentage change in pressure over successive Newton-Raphson iterations. Secondary Newton-Raphson mass-balance convergence criterion. Used for confined flow. Default set at 0.1 percent.

PchgNR Maximum absolute change in pressure over successive Newton-Raphson iterations (dyne/cm^2). Secondary Newton-Raphson mass-balance convergence criterion. Used for unconfined flow. Default set at 1×10^5 dyne/cm^2.

%HchgNR Maximum percentage change in enthalpy over successive Newton-Raphson iterations. Secondary Newton-Raphson energy-balance convergence criterion. Default set at 0.1 percent.

These parameters specify the convergence criteria for Newton-Raphson iterations for each time step (see section 3.6.1). Absolute change in pressure is used for unconfined flow to remove the dominant effect that low pressure cells would have on the determination of convergence. These values for convergence criteria must be chosen carefully because solutions of poor accuracy can result if the values for *resmas* and *reseng* and *%PchgNR* or *PchgNR* and *%HchgNR* are too large.

LINEAR EQUATION SOLVER

slmeth

slmeth — Index for selection of the solution method for the linear Newton-Raphson equations:

- **1** selects the slice successive over-relaxation (SSOR) solver (a direct, triangular factorization for two-dimensional meshes). Not available for three-dimensional meshes.
- **2** selects the generalized minimum residual (GMRES) iterative solver. Not available for one-dimensional meshes.

SSOR SOLVER CONTROL

ssormx, tol, ssorw0; Optional, required only if the SSOR solver is selected.

ssormx — Maximum number of iterations allowed for the solution of the linear Newton-Raphson equations. Default set at 10.

tol — Convergence tolerance on the change in dependent variables for the SSOR iterative solution of the linear Newton-Raphson equations. Default set at 5×10^{-2}.

ssorw0 — Over-relaxation parameter value. Default set at 1.3.

These variables control the slice-successive over-relaxation solution procedure used for three-dimensional mesh regions. However, this record *must* be provided also for one- or two-dimensional mesh regions when SSOR is selected.

GMRES SOLVER CONTROL

m_save_dir, stop_tol_gmres, maxit_gmres, ilu_method, lev_fill, drop_tol; Optional, required only if the GMRES solver is selected.

m_save_dir — Number of iterations between restarts of the GMRES solver. Default set at 5.

stop_tol_gmres — Convergence tolerance on the relative residual for the GMRES iterative solution of the linear Newton-Raphson equations. Default set at 1×10^{-10}.

maxit_gmres — Maximum number of iterations allowed for the solution of the linear Newton-Raphson equations. Default set at 100.

ilu_method — Index for selection of the method for incomplete factorization (ILU) used as a preconditioner.

1 selects ILUT with a drop tolerance and maximum fill. Fill-in elements with magnitude less than *drop_tol* times the L_1-norm of a_{ij} in row i divided by the number of non-zero elements in row i will be neglected and only the largest *lev_fill* elements will be included in L and in U. This is the default method.

2 selects ILUK with level of fill, k. Fill-in elements up through level k=*lev_fill* are included.

lev_fill specifies the maximum number of elements per row of L and of U if using ILUT or specifies the maximum level of fill-in elements if using ILUK. Default set at 5.

drop_tol specifies the relative drop tolerance for *ilu_method* of **1**. Fill-in elements with magnitude less than *drop_tol* times the L_1-norm of a_{ij} in row i divided by the number of non-zero elements in row i will be neglected. Not used for *ilu_method* of **2** but then a zero or **,,** must be entered. Default set at 0.001.

These variables control the GMRES iterative solution procedure used for two- or three-dimensional simulation meshes. There can be a substantial variation in the number of iterations needed to achieve convergence depending on the ILU preconditioning method and parameters selected.

WEIGHTING and AVERAGING

ioptupst[T/F], *potdif*, *theta*

ioptupst Logical switch for spatial weighting of the advective enthalpy-transport terms and relative permeability terms:

T selects upstream-in-space weighting.

F selects centered-in-space (cell-face) weighting.

potdif Threshold ratio of potential between two cells above which upstream weighting is used (dimensionless); below this value, a logarithmic weighted average of upstream and downstream values is used. This feature is intended to facilitate smooth changes in flow direction and to minimize spatial oscillation of pressure and enthalpy. Not used if *ioptupst* is false (cell-face weighting). Default set at 0.0002.

theta Factor for temporal-discretization method:

0.5 selects centered-in-time (Crank-Nicolson) differencing (presently unavailable).

1.0 selects backward-in-time (fully-implicit) differencing.

Default set at 1.0.

RELATIVE PERMEABILITY

kodrp, *wsatn*, *ssatn*

kodrp Index for selection of the relative permeability functions (section 2.2.7) and saturation functions (section 2.2.8) for unconfined flow (*unconfined*, UNCONFINED AQUIFER):

- **0** selects the linear relative permeability functions;
- **1** selects the Corey relative permeability functions (Corey, 1957)
- **2** selects the fracture-flow relative permeability functions (Sorey and others, 1980)
- **3** selects the Corey relative permeability functions to use with the Cooley (Cooley, 1983) function for capillary pressure and saturation.

wsatn Residual saturation of liquid water, S_{wr}.

ssatn Residual saturation of water vapor (steam), S_{sr}.

Rock (porous media) properties may be specified as spatial distributions or be defined by index of rock type. For these options, use the appropriate PARAMETER keyword data block record groups described below.

ROCK PROPERTIES

heatcpty, rxden, rxcmprss, grav, initphi[T/F]

heatcpty Heat capacity (specific heat) of porous matrix solid phase (rock) (erg/g-°C) or (erg/g-K). Default set at 1×10^7. Enter **−1** to specify using a PARAMETER keyword data block record group.

rxden Rock density of solid phase of porous medium (not bulk density) (g/cm^3). Default set at 2.5. Enter **−1** to specify using a PARAMETER keyword data block record group.

rxcmprss Porous medium bulk compressibility (cm^2/dyne). Default set at 0. Enter **−1** to specify using a PARAMETER keyword data block record group. NOTE: if *rxcmprss* is not zero, an output file containing the initial pressure and porosity values will be created and named `IC_pressporo.xxx`. This file is needed for restarting a simulation run (see initphi below).

grav Gravitational constant (cm/sec^2). Suggested value of 981. Enter **0.** for two-dimensional areal simulation regions.

initphi Logical switch for input of initial-condition pressure and porosity values; the name of the file containing these values must be given on the following line.

- **T** selects input of initial-condition values.
- **F** selects no input of initial-condition values.

NOTE: this option is used only for continuation runs in which *rxcmprss* is greater than zero. Then, the initial pressure and porosity values are required in order to calculate updated storativity and porosity values.

initphi_filename; Optional, required only if *initphi* is true.

initphi_filename Name of the file containing initial pressure and porosity field. Necessary for restarting a simulation involving compressible porous media.

CYLINDRICAL COORDINATES

irad[T/F], *wellrad*, *radiusmax*

irad Logical switch for selection of coordinate system:

- **T** selects cylindrical coordinates with no angular dependence.
- **F** selects Cartesian coordinates.

wellrad Well bore radius (cm). Also the radial coordinate location of the innermost cell boundary.

radiusmax Exterior radius of cylindrical-coordinate system (cm). Also the radial coordinate location of the outermost cell boundary.

The two radii values are used only for cylindrical coordinates, but a record terminator (/), data values, or comma-separated spaces must be entered.

PRINT/PLOT CONTROL

wide, *long*, *vert_avg*

wide Index for selection of page width in output files used for tables of cell values:

- **0** selects 80-column output.
- **1** selects 132-column output.
- **2** selects 159-column output.

long Index for selection of the relative amount of output to be written:

- **0** selects a limited amount of output written to files `Out_Probdefine.`*xxx* and `Calc_log.`*xxx* and no output written to monitor screen.
- **1** selects a moderate amount of output written to file `Out_Probdefine.`*xxx* and `Calc_log.`*xxx* and a limited amount of output written to monitor screen.
- **2** selects full amount of output written to file `Out_Probdefine.`*xxx* and `Calc_log.`*xxx* and a limited amount of output written to monitor screen.

vert_avg Index for selection of output of vertically averaged values of pressure, enthalpy, temperature, and water saturation.

- **0** selects no vertically averaged values will be written.
- **1** selects vertically averaged values of variables selected in PRINT 1 will be written to files.

The following six records (for PRINT 1 through PRINT 6) contain switches and print interval parameters for printing specific arrays. For each of the print interval parameters, shown as

xxx_pr_intrv, one of the following values must be selected.

xxx_pr_intrv	Printout interval for output to print and plot files (s), (yr).	
	0	selects no printout to be produced.
	t	selects printout to occur at every *t* simulation time units and at the end of each simulation period (*timchg*, TIME PERIOD). *t* is a real number.
	−n	selects printout to occur at every *n*th time step and at the end of each simulation period (*timchg*, TIME PERIOD). *n* is an integer number.

PRINT 1

pres_pr_intrv, enth_pr_intrv, temp_pr_intrv, satn_pr_intrv, iprsat

pres_pr_intrv	Printout interval for pressure field.	
enth_pr_intrv	Printout interval for enthalpy field.	
temp_pr_intrv	Printout interval for temperature field.	
satn_pr_intrv	Printout interval for saturation field.	
iprsat	Index for selecting the choice of saturation fields.	
	0	selects no printout to be produced.
	1	selects printout of the volumetric saturation.
	2	selects printout of the mass fraction of water.
	3	selects printout of both saturation and mass fraction.

The above record controls output of primary dependent variable data to files `Out_pressure.`*xxx*, `Out_enthalpy.`*xxx*, `Out_temperature.`*xxx*, and `Out_saturation.`*xxx*.

PRINT 2

dens_pr_intrv, iprden, vis_pr_intrv, iprvis, poten_pr_intrv, iprpot

dens_pr_intrv	Printout interval for water and steam density fields.	
iprden	Two-digit index for selecting the choice of density fields.	
	00	selects no printout to be produced.
	10	selects printout of the water density.
	01	selects printout of the steam density.
	11	selects printout of both water and steam density.
vis_pr_intrv	Printout interval for water and steam viscosity fields.	
iprvis	Two-digit index for selecting the choice of viscosity fields.	
	00	selects no printout to be produced.
	10	selects printout of the water viscosity.
	01	selects printout of the steam viscosity.
	11	selects printout of both water and steam viscosity.
poten_pr_intrv	Printout interval for water and steam hydraulic pressure and head fields.	
iprpot	Two-digit index for selecting the choice of hydraulic pressure and head fields.	

00 selects no printout to be produced.
10 selects printout of the water hydraulic pressure.
01 selects printout of the steam hydraulic pressure.
11 selects printout of both water and steam hydraulic pressure.
20 selects printout of the water potentiometric head.
02 selects printout of the steam potentiometric head.
22 selects printout of both water and steam potentiometric head.
30 selects printout of both the hydraulic pressure and potentiometric head for water.
03 selects printout of both the hydraulic pressure and potentiometric head for steam.
33 selects printout of all water and steam pressures and heads.

The above record controls output of water and steam data to files `Out_density.`*xxx*, `Out_viscosity.`*xxx*, and `Out_potential.`*xxx*.

PRINT 3

velocity_pr_intrv, iprvel, bcflow_pr_intrv, iprbcflow, source_pr_intrv, iprsource

velocity_pr_intrv Printout interval for water and steam velocity fields.
iprvel Two-digit index for selecting the choice of velocity fields.

00 selects no printout to be produced.
10 selects printout of the water interstitial velocity.
01 selects printout of the steam interstitial velocity.
11 selects printout of both water and steam interstitial velocity.
20 selects printout of the water mass flux and Darcy flux.
02 selects printout of the steam mass flux and Darcy flux.
22 selects printout of the water and steam mass flux and Darcy flux.
30 selects printout of both the interstitial velocity, mass flux, and Darcy flux for water.
03 selects printout of both the interstitial velocity, mass flux, and Darcy flux for steam.
33 selects printout of all water and steam velocities and fluxes.

bcflow_pr_intrv Printout interval for water and steam boundary flow rate fields.
iprbcflow Presently not used. Enter **0** or **,,**.
source_pr_intrv Printout interval for source/sink flow rate tables.
iprsource Presently not used. Enter **0** or **,,**.

The above record controls output of data to files `Out_velocity.`*xxx*, `Out_bcflow.`*xxx*, and `Out_source.`*xxx*.

PRINT 4

pm_prop_pr_intrv, iprmprop, poros_pr_intrv, perm_pr_intrv

pm_prop_pr_intrv Printout interval for flow and thermal porous media property fields.

iprmprop Two-digit index for selecting the choice of porous media properties.

- **00** selects no printout will be produced.
- **10** selects printout of the porous media flow property fields: bulk compressibility, rock density.
- **01** selects printout of the porous media thermal property fields: thermal conductivity, rock heat capacity.
- **11** selects printout of both flow and thermal property fields.

poros_pr_intrv Printout interval for porosity field.

perm_pr_intrv Printout interval for permeability field.

The above record controls output of data to files `Out_pmthermalprop.`*xxx*, `Out_porosity.`*xxx*, and `Out_permeability.`*xxx*.

PRINT 5

balance_pr_intrv, dimno_pr_intrv, iprdimno, resid_pr_intrv, dump_pr_intrv

balance_pr_intrv Printout interval for flow and thermal energy balance tables.

dimno_pr_intrv Printout interval for dimensionless numbers, Peclet and Nusselt.

iprdimno Two-digit index for selecting the choice of dimensionless numbers.

- **00** selects no printout will be produced.
- **10** selects printout of the cell thermal Peclet number field.
- **01** selects printout of the cell Nusselt number field.
- **11** selects printout of both thermal Peclet and Nusselt number fields.

resid_pr_intrv Printout interval for residuals of the difference equations.

dump_pr_intrv Printout interval for restart data file.

The above record controls output of data to files `Out_balance.`*xxx*, `Out_dimensionless.`*xxx*, `Out_residual.`*xxx*, and `Out_restartdump.`*xxx*.

PRINT 6

pltscal_pr_intrv, pltvect_pr_intrv, plotfile_type, time_ser_pr_intrv

pltscal_pr_intrv Printout interval for fields of scalar dependent variables: pressure, enthalpy, temperature, saturation.

pltvect_pr_intrv Printout interval for fields of vector dependent variables: interstitial velocity, mass flux.

plotfile_type Index for selection of the visualization software to be used (Explorer, HTgnuplot, IDL, Basin2) and the format of output (ASCII text or binary data).

0 selects no plotfile will be written.
1 selects Explorer plotfile to be written in ASCII text format.
2 selects HTgnuplot plotfile to be written in binary format.
3 selects HTgnuplot plotfile to be written in ASCII text format.
4 selects IDL plotfile to be written in binary format.
5 selects IDL plotfile to be written in ASCII text format.
6 selects tab-separated columnar plotfile to be written in ASCII text format.
8 selects Basin2 plotfile to be written in ASCII text format.
9 selects Basin2 plotfile to be written in binary format.

time_ser_pr_intrv Printout interval for time series of dependent variables at selected locations.

num_locations, (*i*(n), *j*(n), *k*(n), n=1 to num_locations); Optional, required if *time_ser_pr_intrv* is non-zero.

num_locations Number of selected node locations at which time series data are to be collected.

i, j, k Sets of indices identifying the node locations for time series data.

The above records control output of data to files `Plot_scalar.`*xxx*, `Plot_scalar2.`*xxx*, `Plot_vector.`*xxx*, and `Plot_timeseries.`*xxx*. If multiple time periods are being simulated with several sets of output intervals to the plotfiles, the initial PRINT 6 record must have the selection of the visualization software and the format, even if no output to plotfiles is specified for the first time period. No changes to this selection may be made subsequently. Note that the print interval for each type of plotfile written during the simulation must not be zero during the last simulation period or that plotfile will be deleted.

SLICE *number*

start_location [**TOP** | **BOTTOM**]

start_location Secondary keyword to select the order of input records containing indices of rock type.

TOP selects values to be input from the top node layer down (k=N_z to 1).
BOTTOM selects values to be input from the bottom node layer up (k=1 to N_z).

index_rock_type(i,k), i=1 to N_x, k=1 to N_z (or N_z to 1)

index_rock_type Index values defining the rock type for each cell in the current SLICE number. The ordering is row by row sweeping in the positive x direction.

n Index value of rock type for a cell in the simulation region. Range is from 1 to 9.
−n specifies a cell with a constant value of pressure and enthalpy and index value of rock type *n*.
-1*n* specifies a cell with a specified value of pressure and associated enthalpy and temperature for incoming fluid and index value of rock

type *n*.

-3*n* specifies a cell with a seepage face and index value of rock type *n*.

-5*n* specifies a cell with a seepage face or precipitation flux and index value of rock type *n*.

0 specifies a cell that is excluded from the simulation region.

UNCONFINED AQUIFER

unconfined[T/F], *patm*

unconfined Logical switch for selection of unconfined flow system:

T selects that the aquifer is unconfined.

F selects that the aquifer is confined.

patm Atmospheric pressure (dyne/cm^2).

SATURATION FUNCTION

Presently, there are three choices for the function relating capillary pressure to saturation for unconfined flow systems: LINEAR, COREY, and COOLEY. They are linked to the choice made for the function relating relative permeability to saturation.

LINEAR

pwb, pwr; Optional, required only if *kodrp* is 0 and *unconfined* is true (*kodrp*, RELATIVE PERMEABILITY; *unconfined*, UNCONFINED AQUIFER).

pwb Bubble-point pressure for the capillary pressure as a linear function of saturation (dyne/cm^2).

pwr Pressure at residual water saturation for linear function (dyne/cm^2).

COREY

pwb, lambda; Optional, required only if *kodrp* is 1 and *unconfined* is true (*kodrp*, RELATIVE PERMEABILITY; *unconfined*, UNCONFINED AQUIFER).

pwb Bubble-point pressure for the Corey function of capillary pressure and saturation (dyne/cm^2).

lambda Pore size index for the Corey function (dimensionless). Presently not used. Default set at 2.

COOLEY

pwb, bb, cc; Optional, required only if *kodrp* is 3 and *unconfined* is true (*kodrp*, RELATIVE PERMEABILITY; *unconfined*, UNCONFINED AQUIFER).

pwb Bubble-point pressure for the Cooley function of capillary pressure and saturation (dyne/cm^2).

bb Parameter for the Cooley function (dimensionless).

cc Exponent parameter for the Cooley function (dimensionless).

II. Parameter Data by Keyword-Data Block

Data blocks or groups of parameter data, identified by a keyword, are read in order to: (1) define the cell dimensions that form the finite-difference mesh, (2) assign values to the porous medium properties, and (3)

apply initial and boundary conditions to the dependent variables. Ten to 16 keyword groups of data are required depending on whether or not a cylindrical coordinate system is being used (*irad*, CYLINDRICAL COORDINATES), whether or not the rock properties (*heatcpty*, *rxden*, and *rxcmprss*) have been specified above (ROCK PROPERTIES), and whether or not an unconfined aquifer is being simulated (*unconfined*, UNCONFINED AQUIFER). A keyword data block group consists of three or more records, as follows:

KEYWORD_group_identifier, (unit)
FORMAT_SPECIFIER
data_set_records

The first record is the *KEYWORD* group identifier. The second record is the secondary keyword specifying the input format. The remaining records are the data set that completes the data block. The records needed depend on the particular group identifier and format specifier selected.

The *KEYWORD* group identifier is selected from the following list. The dimensional units of measure are selected from the options shown. Each of the 10 mandatory keyword data blocks must appear in the data-input file plus the appropriate selection of optional data blocks. The three keyword groups defining the cell dimensions (**XSPACING**, **YSPACING**, **ZSPACING**) *must* be input first, followed by the remaining parameter keyword groups in any order. Templates for each type of keyword data block group are listed below with syntax and options described. In the following *KEYWORD* group identifier list, the three keywords for cell dimensions are presented first, followed by the others in alphabetical order.

XSPACING List of the cell dimensions in the x-coordinate direction if Cartesian coordinates. List of the radial dimension of each cylindrical ring if cylindrical coordinates **(cm)**,**(m)**,**(km)**. The cell dimensions must be listed in ascending order along the coordinate direction.

YSPACING List of the cell dimensions in the y-coordinate direction if Cartesian coordinates **(cm)**,**(m)**,**(km)**. Optional, required only if Cartesian coordinates (*irad*=F, CYLINDRICAL COORDINATES). The cell dimensions must be listed in ascending order along the coordinate direction.

ZSPACING List of the cell dimensions in the z-coordinate direction **(cm)**,**(m)**,**(km)**.

COMPRESSIBILITY Bulk vertical compressibility of the porous media **(cm^2/dyne)**, **(Pa^-1)**, **(MPa^-1)**, **(bar^-1)**. Optional, required only if not entered previously (ROCK PROPERTIES).

CONDUCTION Thermal-conduction heat flux along the bottom boundary of the region **(erg/s-cm^2)**, **(mW/m^2)**, **(W/m^2)**. Required although it may be a value of zero.

ENTHALPY Initial-condition specific enthalpy distribution for the region **(erg/g),(J/g),(J/kg),(kJ/kg)**. These values will be the boundary condition values at any specified pressure and enthalpy boundary nodes. If used, do not also specify a temperature distribution.

POROSITY Porosity of the media **(-),(%)**.

PRECIPITATION Precipitation-recharge flux (volumetric) along the top boundary of the active region **(cm^3/s-cm^2)**, **(m^3/s-m^2)**, **(mm/yr)**, **(m/yr)**. Optional, required only if an unconfined aquifer is being simulated (*unconfined*, UNCONFINED AQUIFER),

although it may be a value of zero.

PRESSURE Initial-condition pressure distribution for the region **(dyne/cm^2)**, **(Pa)**, **(bar)**. These values will be the boundary condition values at any specified pressure boundary nodes.

RDENSITY Density of the porous media solid phase (not the bulk density) **(g/cm^3)**, **(kg/m^3)**. Optional, required only if not entered previously (ROCK PROPERTIES).

SPECIFIC_HEAT Specific heat or heat capacity of the porous matrix solid phase **(erg/g-K)**, **(J/g-K)**, **(J/kg-K)**, **(kJ/kg-K)**, **(erg/g-C)**, **(J/g-C)**, **(J/kg-C)**, **(kJ/kg-C)**. Optional, required only if not entered previously (ROCK PROPERTIES).

TEMPERATURE Initial-condition temperature distribution for the region **(C)**, **(Deg.C)**. These values will be the boundary condition values at any specified pressure and enthalpy boundary nodes. If used, do not also specify an enthalpy distribution. Specific enthalpy will be calculated from the temperature and pressure distributions.

TFLUX Associated temperature of the precipitation flux **(C)**, **(Deg.C)**. Optional, required only if an unconfined aquifer is being simulated (*unconfined*, UNCONFINED AQUIFER), although it may be a value of zero when there is no precipitation flux.

THERMAL_COND Thermal conductivity of the porous media and fluid phases **(erg/s-cm-K)**, **(W/m-K)**, **(J/s-m-K)**, **(erg/s-cm-C)**, **(W/m-C)**, **(J/s-m-C)**. This is an effective value for the aggregate of solid, liquid, and gas phases.

XPERMEABILITY Permeability of the porous media in the x-coordinate direction **(cm^2)**, **(m^2)**, **(darcy)**.

YPERMEABILITY Permeability of the porous media in the y-coordinate direction **(cm^2)**, **(m^2)**, **(darcy)**. Optional, required only if Cartesian coordinates (*irad*=F, CYLINDRICAL COORDINATES).

ZPERMEABILITY Permeability of the porous media in the z-coordinate direction **(cm^2)**, **(m^2)**, **(darcy)**.

In the # PARAMETER data section, each *KEYWORD* group identifier with its associated data block may appear *only once.* Multiple data blocks with the same *KEYWORD* group identifier will cause an input error and the simulator will terminate.

The *FORMAT_SPECIFIER* keyword must be selected from the following list. The format specifiers are listed in approximate order of increasing complexity of the associated data block and decreasing generality of application.

FREE_FORMAT specifies free-format (list-directed) input. One data value must be input for each node in the global region and repeat factors may be used.

CONSTANT specifies a uniform value for the entire mesh. Free-format input is used for the single value.

COLCONSTANT specifies a uniform value for each column of the mesh. The data must be in ascending order in the x-direction. Free-format input is used for the single value for each column.

ROWCONSTANT specifies a uniform value for each row of the mesh. Free-format input is used for the

	single value for each row.
NODE	specifies that values for selected cells will be entered. It is used to change values at a few cells during the simulation. It is not useful for specifying initial-condition distributions.
CALCULATE	specifies that values for this parameter are to be calculated by the simulator. The calculation performed depends on the *group_identifier* keyword.
ROCK_TYPE	specifies that property values are indexed to each rock type defined for the simulation region and are constant values.
RTFDEPTH	specifies that property values are indexed to each rock type defined for the simulation region and are a function of depth within the region.
RTFTEMP	specifies that property values are indexed to each rock type defined for the simulation region and are a function of temperature.
RTFTIME	specifies that property values are indexed to each rock type defined for the simulation region and are a function of time.

For each keyword data block needed in the data-input file, a suitable format specifier is selected. Not every combination of a keyword and a format specifier is available for input by keyword-data block. The valid combinations are indicated by numbers in table 5.3. The numbers are indices to the data-block templates listed below and designate which *KEYWORD* data block template should be used to input the data for the *KEYWORD/FORMAT_SPECIFIER* pair selected.

Templates of the syntax for each type of *KEYWORD/FORMAT_SPECIFIER* are presented in the following list. Specific keywords and options are used for restricted cases, whereas generic keywords and variable names are used for general cases. Options and variable names for the data sets are explained after each template of a data-set group. Although some of the *KEYWORD* group identifiers appear several times in the following templates, each *KEYWORD* group identifier with its associated data block may appear *only once* in the # PARAMETER section of the data-input file. Specific examples of keyword data-block input appear in the tutorial example data-input file of section 8.3.

Table 5.3. Index numbers of templates for data input by keyword data block for each format option available

[-- indicates format option not available]

	FORMAT_SPECIFIER									
KEYWORD	FREE_FORMAT	CONSTANT	COLCONSTANT	ROWCONSTANT	NODE	CALCULATE	ROCK_TYPE	RTFDEPTH	RTFTEMP	RTFTIME
XSPACING	1	4	--	--	--	8	--	--	--	--
YSPACING	1	4	--	--	--	--	--	--	--	--
ZSPACING	2	4	--	--	--	--	--	--	--	--
COMPRESSIBILITY	2	4	5	6	7	--	24	--	27	--
CONDUCTION	3	4	--	--	7	9	--	--	--	--
ENTHALPY	2	4	5	6	7	--	--	--	--	--
POROSITY	2	4	5	6	7	--	24	25	27	--
PRECIPITATION	3	4	--	--	7	10	--	--	--	--
PRESSURE	2	4	5	6	7	11–14	--	--	--	--
RDENSITY	2	4	5	6	7	--	24	--	27	--
SPECIFIC_HEAT	2	4	5	6	7	--	24	--	27	--
TEMPERATURE	2	4	5	6	7	15–21	--	--	--	--
TFLUX	3	4	--	--	7	22	--	--	--	--
THERMAL_COND	2	4	5	6	7	--	24	--	27	--
XPERMEABILITY	2	4	5	6	7	--	24	26	27	28
YPERMEABILITY	2	4	5	6	7	23	24	--	--	--
ZPERMEABILITY	2	4	5	6	7	23	24	--	--	--

PARAMETER
Template 1

group_identifier, (unit)
FREE_FORMAT
data_values(n), n=1 to $N_x \times N_z$

group_identifier	A keyword that must be selected from the above list.
(unit)	Dimensional units of measure that must be selected from the options shown with the *group_identifier* keyword. Parentheses are required but the units are case insensitive.
data_values	Records of data values for the *group_identifier* specified. The order is slice by slice of x-z planes of nodes starting at the slice for j=1. Within each slice, the order is row by row, i=1 to N_x, then k=1 to N_z. Each node of the slice from 1 to $N_x \times N_z$ requires a data value even if it is excluded from the active simulation region.

Template 2

group_identifier, (unit)
FREE_FORMAT
start_location
data_values(n), n=1 to $N_x \times N_z$

group_identifier	A keyword that must be selected from the above list.
(unit)	Dimensional units of measure that must be selected from the options shown with the *group_identifier* keyword. Parentheses are required but the units are case insensitive.
start_location	Keyword to select the order of data-set records. **TOP** selects values to be input from the top node layer down (k=N_z to 1). **BOTTOM** selects values to be input from the bottom node layer up (k=1 to N_z).
data_values	Records of data values for the *group_identifier* specified. The order is slice by slice of x-z planes of nodes starting at the slice for j=1. Within each slice, the order is row by row, i=1 to N_x, then k=1 to N_z or k=N_z to 1 depending on whether **TOP** or **BOTTOM** was specified. Each node of the slice from 1 to $N_x \times N_z$ requires a data value even if it is excluded from the active simulation region.

Template 3

group_identifier, (unit)
FREE_FORMAT
data_values(n), n=1 to $N_x \times N_y$

group_identifier	A keyword that must be selected from the above list.
(unit)	Dimensional units of measure that must be selected from the options shown with the *group_identifier* keyword. Parentheses are required but the units are case insensitive.
data_values	Records of data values for the *group_identifier* specified. The order is row by row, i=1 to N_x, then j=1 to N_y. Each node of the plane from 1 to $N_x \times N_y$ requires a data value even if it is excluded from the active simulation region.

Template 4

group_identifier, (unit)
CONSTANT
data_value

group_identifier	A keyword that must be selected from the above list.
(unit)	Dimensional units of measure that must be selected from the options shown with the *group_identifier* keyword. Parentheses are required but the units are case insensitive.
data_value	The single data value for the *group_identifier* specified.

Template 5

group_identifier, (unit)
COLCONSTANT
data_values(i), i=1 to N_x

group_identifier	A keyword that must be selected from the above list.
(unit)	Dimensional units of measure that must be selected from the options shown with the *group_identifier* keyword. Parentheses are required but the units are case insensitive.
data_values	Records of data values for the *group_identifier* specified. The order is slice by slice of x-z planes of nodes starting at the slice for j=1. Within each slice, the order is a single value for each column, with i=1 to N_x. Each column of nodes requires a data value even if it is excluded from the active simulation region.

Template 6

group_identifier, (unit)
ROWCONSTANT
start_location
data_values(k), k=1 to N_z or k=N_z to 1

group_identifier	A keyword that must be selected from the above list.
(unit)	Dimensional units of measure that must be selected from the options shown with the *group_identifier* keyword. Parentheses are

required but the units are case insensitive.

start_location Keyword to select the order of input records.

TOP Means data are input from the top node layer down (k=N_z to 1).

BOTTOM Means data are input from the bottom node layer up (k=1 to N_z).

data_values Records of data values for the *group_identifier* specified. The order is slice by slice of x-z planes of nodes starting at the slice for j=1. Within each slice, the order is a single value for each row, with k=1 to N_z or k=N_z to 1 depending on whether **TOP** or **BOTTOM** was specified. Each row of nodes requires a data value even if it is excluded from the simulation region.

Template 7

group_identifier, (unit)

NODE

i(m), *j*(m), *k*(m), *data_value*(m), m=1 to num_dat_nodes

0,0,0,0

group_identifier A keyword that must be selected from the above list.

(unit) Dimensional units of measure that must be selected from the options shown with the *group_identifier* keyword. Parentheses are required but the units are case insensitive.

i, j, k Indices in the x, y, and z directions for node m.

data_value The data value for node m for the *group_identifier* selected. There is one record for each node at which a *data_value* is specified for a total of num_dat_nodes records.

0, 0, 0, 0. Denotes the end of this data set.

Data for nodes not specified in this set remain unchanged.

Template 8

XSPACING, *(unit)*

CALCULATE

inner_radius

(unit) Dimensional units of measure that must be selected from the options shown with the **XSPACING** keyword. Parentheses are required but the units are case insensitive.

inner_radius Radial coordinate of the innermost *node* of the cylindrical mesh *(unit)*. The radial spacing of the nodes will be calculated using a geometric spacing. Optional, required only if cylindrical coordinates (*irad*=T, CYLINDRICAL COORDINATES) are selected.

Template 9

CONDUCTION, *(unit)*
CALCULATE
1, *heat_flux_1, heat_flux_nx*

(unit)	Dimensional units of measure that must be selected from the options shown with the **CONDUCTION** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **1** to calculate a linear distribution of heat flux along the bottom of the region.
heat_flux_1	Value of heat flux at i=1 *(unit)*.
heat_flux_nx	Value of heat flux at i=N_x *(unit)*.

Template 10

PRECIPITATION, *(unit)*
CALCULATE
1, *precipitation_flux_1, precipitation_flux_nx*

(unit)	Dimensional units of measure that must be selected from the options shown with the **PRECIPITATION** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **1** to calculate a linear distribution of precipitation-recharge flux along the top of the region (land surface).
precip_flux_1	Value of precipitation flux at i=1 *(unit)*.
precip_flux_nx	Value of precipitation flux at i=N_x *(unit)*.

Template 11

PRESSURE, *(unit)*
CALCULATE
1, *top_pressure*

(unit)	Dimensional units of measure that must be selected from the options shown with the **PRESSURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **1** to calculate a hydrostatic pressure distribution based upon a uniform pressure at the top nodes of the active mesh region and the specified temperature (enthalpy) distribution with depth.
top_pressure	Pressure at the uppermost active nodes of the region *(unit)*. Default set at 1.013×10^6 dyne/cm^2.

Template 12

PRESSURE, *(unit)*
CALCULATE
11, 0., 0.
num_zppts, (*z_depth*(k), *pressure*(k), k=1 to num_zppts)

(unit)	Dimensional units of measure that must be selected from the options shown with the **PRESSURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **11** to calculate a piecewise linear distribution of pressure from top to bottom of the global mesh region. Only active cells are assigned a pressure value. The pressure is uniform in each horizontal plane.
num_zppts	Number of points defining the piecewise linear pressure profile. Up to six points are allowed defining five profile segments.
z_depth(k)	Depth values from the *top* of the global mesh region of the kth point of the piecewise linear function (units of z-coordinate). The end points may be outside the z-coordinate range of the mesh. If the end points are within the z-coordinate range of the mesh, the first or last linear segment will be extrapolated, as appropriate, to compute pressure values at nodes outside the range of defined linear segments.
pressure(k)	Pressure at the corresponding depth value of the kth point of the piecewise linear function *(unit)*.

Template 13

PRESSURE, *(unit)*
CALCULATE
21, 0.
top_press(ij), ij=1 to $N_x \times N_y$

(unit)	Dimensional units of measure that must be selected from the options shown with the **PRESSURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **21** to calculate a hydrostatic pressure distribution based on the pressure at the top nodes of the active mesh region and the specified temperature (enthalpy) distribution with depth allowing for a different top node pressure for each column of cells.
top_press	Set to **0.** as a place holder.
top_press(ij)	Pressureat the top node of the active region for each column of cells *(unit)*. Each node at the top of the global mesh region requires a data value even if it is at the location of an excluded column of cells.

Template 14

PRESSURE, *(unit)*
CALCULATE
22, *top_pressure*
i(m), *j*(m), *top_pressure*(m), m=1 to num_dat_cols
0, 0, 0.

(unit)	Dimensional units of measure that must be selected from the options shown with the **PRESSURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **22** to calculate a hydrostatic pressure distribution based on the pressure at the top nodes of the active mesh region and the specified temperature (enthalpy) distribution with depth allowing for a default top node pressure for most cell columns and different top node pressures for a selected subset of cell columns.
top_pressure	Pressure at the uppermost active nodes of the mesh region (default) *(unit)*.
i(m), *j*(m)	Indices in the x and y directions of the mth column of cells.
top_pressure(m)	Pressure at the uppermost active node of the mesh region for the mth column of cells *(unit)*. There is one record for each column at which a *top_pressure* is specified to be different from the default for a total of num_dat_cols records.
0, 0, 0.	Denotes the end of this data set.

Template 15

TEMPERATURE, *(unit)*
CALCULATE
1, *top_temp, bottom_temp*

(unit)	Dimensional units of measure that must be selected from the options shown with the **TEMPERATURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **1** to calculate a linear temperature distribution based upon a specified temperature at the top nodes of the active mesh region and a specified temperature at the bottom of the mesh region.
top_temp	Temperature at the top node elevation of the global mesh region (°C).
bottom_temp	Temperature at the bottom node elevation of the global mesh region (°C).

Template 16

TEMPERATURE, *(unit)*
CALCULATE
11, 0., 0.
num_ztpts, (*z_depth*(k), *temp*(k),k=1 to num_ztpts)

(unit)	Dimensional units of measure that must be selected from the options shown with the **TEMPERATURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **11** to calculate a piecewise linear distribution of temperature from top to bottom of the global mesh region. Only active cells are assigned a temperature value. The temperature is uniform in each horizontal plane.
num_ztpts	Number of points defining the piecewise linear temperature profile. Up to six points are allowed defining five profile segments.
z_depth(k)	Depth value from the *top* of the global mesh region of the kth point of the piecewise linear function (units of z-coordinate). The end points may be outside the z-coordinate range of the mesh. If the end points are within the z-coordinate range of the mesh, the first or last linear segment will be extrapolated, as appropriate, to compute pressure values at nodes outside the range of defined linear segments.
temp(k)	Temperature at the corresponding depth value of the kth point of the piecewise linear function (°C).

Template 17

TEMPERATURE, *(unit)*
CALCULATE
2, *top_temp, temp_grad*

(unit)	Dimensional units of measure that must be selected from the options shown with the **TEMPERATURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **2** to calculate a temperature distribution based on the temperature at the top node of the active mesh region and the temperature gradient in the vertical direction.
top_temp	Temperature at the top node of the active mesh region (°C).
temp_grad	Temperature gradient in the vertical direction (°C/km).

Template 18

TEMPERATURE, *(unit)*
CALCULATE
21, 0., *temp_grad*
top_temp(ij), ij=1 to $N_x \times N_y$

(unit)	Dimensional units of measure that must be selected from the options shown with the **TEMPERATURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **21** to calculate a temperature distribution based on the temperature at the top nodes of the active mesh region and the temperature gradient in the vertical direction allowing for a different top node temperature for each column of cells.
top_temp	Set to **0.** as a place holder.
temp_grad	Temperature gradient in the vertical direction (°C/km).
top_temp(ij)	Temperature at the top node of the active mesh region for each column of cells (°C). Each node at the top of the mesh region requires a data value.

Template 19

TEMPERATURE, *(unit)*
CALCULATE
22, *top_temp, temp_grad*
i(m), *j*(m), *top_temp*(m), *temp_grad*(m), m=1 to num_dat_cols
0, 0, 0., 0.

(unit)	Dimensional units of measure that must be selected from the options shown with the **TEMPERATURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **22** to calculate a temperature distribution based on the temperature at the top nodes of the active mesh region and the specified temperature gradient in the vertical direction allowing for a default top node temperature for most cell columns and different top node temperatures and temperature gradients for a selected subset of cell columns.
top_temp	Temperature (default) at the uppermost active nodes of the mesh region (°C).
temp_grad	Temperature gradient (default) in the vertical direction (°C/km).
i(m), *j*(m)	Indices in the x and y directions of the mth column of cells.
top_temp(m)	Temperature at the uppermost active node of the region for the mth column of cells (°C).
temp_grad(m)	Temperature gradient in the vertical direction for the mth column of cells (°C/km). There is one record for each column at which a *top_temp* and *temp_grad* is specified to be different than the default for a total of num_dat_cols records. In the case of only a default temperature gradient, this record set is empty.
0, 0, 0., 0.	Denotes the end of this data set.

Template 20

TEMPERATURE, *(unit)*
CALCULATE
3, 0, 0

(unit)	Dimensional units of measure that must be selected from the options shown with the **TEMPERATURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **3** to calculate a temperature distribution based on the boiling point temperature as a function of depth. Requires that the **CALCULATE** option also be used for **PRESSURE** so that an initial hydrostatic pressure distribution is calculated.

Template 21

TEMPERATURE, *(unit)*
CALCULATE
33, *top_pressure*, **0**
i(m), *j*(m), *top_pressure*(m), m=1 to num_dat_cols
0, 0, 0.

(unit)	Dimensional units of measure that must be selected from the options shown with the **TEMPERATURE** keyword. Parentheses are required but the units are case insensitive.
option_no	Set to **33** to calculate a temperature distribution based on the boiling point temperature as a function of depth allowing for a default top node pressure for most cell columns and different top node pressures for a selected sub-set of cell columns.
top_pressure	Default pressure at the uppermost active nodes of the region for all columns (dyne/cm^2).
i(m), *j*(m)	Indices in the x and y directions of the mth column of cells.
top_pressure(m)	Pressure at the uppermost active node of the region for the mth column of cells (dyne/cm^2). There is one record for each column at which a *top_pressure* is specified to be different from the default for a total of num_dat_cols records.
0, 0, 0.	Denotes the end of this data set.

Template 22

TFLUX, *(unit)*
CALCULATE
1, *assoc_temp_1, assoc_temp_nx*

(unit)	Dimensional units of measure that must be selected from the options shown with the **TFLUX** keyword. Parentheses are required but the units

are case insensitive.

option_no Set to **1** to calculate a linear distribution of associated temperature of precipitation flux along the top of the region (land surface).

assoc_temp_1 Value of associated temperature of precipitation flux at i=1 *(unit)*.

assoc_temp_nx Value of associated temperature of precipitation flux at i=N_x *(unit)*.

Template 23

[**Y**|**Z**]**PERMEABILITY**, *(unit)*
CALCULATE
perm_factor

(unit) Dimensional units of measure that must be selected from the options shown with the [**Y**|**Z**]**PERMEABILITY** keyword. Parentheses are required but the units are case insensitive.

perm_factor Multiplication factor for computing the y- or z-direction permeability from the x-direction permeability.

Template 24

group_identifier, *(unit)*
ROCK_TYPE
{*index_rock_type*(n), *data_value*(n)}, n=1 to num_rock_type

group_identifier A keyword that must be selected from the above list.

(unit) Dimensional units of measure that must be selected from the options shown with the *group_identifier* keyword. Parentheses are required but the units are case insensitive.

index_rock_type Index of rock type selected from the set defined under # SLICE.

data_value Value of the parameter denoted by *group_identifier* selected *(unit)*.

A record as shown between braces, { }, must be included for each index of rock type defined for the simulation region (# SLICE *number*) for a total of num_rock_type records.

Template 25

POROSITY
RTFDEPTH
depth_scale_opt, *z_datum*
{*index_rock_type*(n), *function_no*(n)
[*poro*(n) | [*z1*|*depth1*](n), *poro1*(n), [*z2*|*depth2*](n), *poro2*(n)]},n=1 to num_rock_type

depth_scale_opt Index for selection of the option for entry of depth-location information.

- **1** selects location data to be entered as elevation along the z-axis. Origin is at the bottom of the global mesh region.
- **2** selects location data to be entered as depth from *z_datum*. Origin

	is at *z_datum*.
	3 selects location data to be entered as depth from the land surface. Origin is at the land surface at each (x,y) location in the active mesh region.
z_datum	Z-elevation of the datum level from which depth is measured *(z-axis unit)*. Only used if *depth_scale_opt* is 2, but a value must be entered in any case.
index_rock_type	Index of rock type, or zone, selected from the set defined under # SLICE.
function_no	Index for selection of the function used to calculate the porosity distribution with depth.
	1 selects a constant porosity from top to bottom of the zone of *index_rock_type*.
	2 selects a linear distribution of porosity from top to bottom of the zone of *index_rock_type*.
	3 selects an exponential (base 10) distribution of porosity from top to bottom of the zone of *index_rock_type*. This distribution is equivalent to the logarithm of porosity as a linear function of depth.
poro	Constant value of porosity for *function_no*=1.
[*z1*|*depth1*]	Z-elevation or depth from *z_datum* or from the land surface of the first specified porosity value *(z-axis unit)*.
poro1	Porosity at *z1* or *depth1*.
[*z2*|*depth2*]	Z-elevation or depth from *z_datum* or from the land surface of the second specified porosity value *(z-axis unit)*.
poro2	Porosity at *z2* or *depth2*.

The z or depth values are usually in the zone of *index_rock_type*. However, they do not need to match a node elevation. The land surface must be contained within or be at the upper boundary of the global mesh region, if *depth_scale_opt*=3. Pairs of data records as shown between braces, { }, must be included for each index of rock type defined for the simulation region (# SLICE *number*) for a total of num_rock_type record pairs. Porosity values are interpolated at the node elevations to calculate the porosity for each cell, which is uniform throughout the cell.

Template 26

XPERMEABILITY, *(unit)*
RTFDEPTH
depth_scale_opt, z_datum
{*index_rock_type*(n), *function_no*(n)
[*perm*(n) | [*z1*|*depth1*](n), *perm1*(n), [*z2*|*depth2*](n), *perm2*(n)]},n=1 to num_rock_type

depth_scale_opt	Index for selection of the option for entry of depth-location information.

	1 selects location data to be entered as elevation along the z-axis. Origin is at the bottom of the global mesh region.
	2 selects location data to be entered as depth from *z_datum*. Origin is at *z_datum*.
	3 selects location data to be entered as depth from the land surface. Origin is at the land surface at each (x,y) location in the active mesh region.
z_datum	Z-elevation of the datum level from which depth is measured *(z-axis unit)*. Only used if *depth_scale_opt* is 2, but a value must be entered in any case.
index_rock_type	Index of rock type, or zone, selected from the set defined under # SLICE.
function_no	Index for selection of the function used to calculate the permeability distribution with depth.
	1 selects a constant x-permeability from top to bottom of the zone of *index_rock_type*.
	2 selects a linear distribution of x-permeability from top to bottom of the zone of *index_rock_type* (eq. 2.2.1).
	3 selects a distribution of the logarithm of x-permeability as a linear function of the depth from top to bottom of the zone of *index_rock_type* (eq. 2.2.3).
	4 selects a distribution of the logarithm of x-permeability as a function of the logarithm of depth from top to bottom of the zone of *index_rock_type* (eq. 2.2.2).
perm	Constant value of x-permeability for *function_no*=1.
[*z1*|*depth1*]	Z-elevation or depth from *z_datum* or from the land surface of the first specified permeability value *(z-axis unit)*.
perm1	X-Permeability at *z1* or *depth1*.
[*z2*|*depth2*]	Z-elevation or depth from *z_datum* or from the land surface of the second specified permeability value *(z-axis unit)*.
perm2	X-Permeability at *z2* or *depth2*.

The z-elevation or depth values are usually in the zone of *index_rock_type*. The land surface must be contained within or be at the upper boundary of the global mesh region, if *depth_scale_opt*=3. Pairs of data records as shown between braces, { }, must be included for each index of rock type defined for the simulation region (# SLICE *number*) for a total of num_rock_type record pairs. Permeability values are interpolated at the node elevations to calculate the permeability for each cell, which is uniform throughout the cell.

Template 27

group_identifier, (unit)

RTFTEMP

{*index_rock_type*(n), *function_no*(n), *num_points*(n)
[*data_value* | (*data_value*(i), *temp*(i), i=1 to *num_points*)]},
n=1 to num_rock_type

group_identifier	A keyword that must be selected from the above list.
(unit)	Dimensional units of measure that must be selected from the options shown with the group_identifier keyword. Parentheses are required but the units are case insensitive.
index_rock_type	Index of rock type selected from the set defined under # SLICE.
function_no	Index for selection of the type of temperature-dependence interpolation function for the variable specified by *group_identifier*. **1** selects a constant temperature function. There is no variation of the property with temperature. **2** selects a piecewise constant temperature dependence function. **3** selects an interpolation function such that the logarithm of the property is a piecewise linear function of temperature. **4** selects an interpolation function such that the property is a piecewise linear function of temperature.
num_points	Number of data points defining the temperature dependence function (maximum of 4). Enter **1** if a constant temperature dependence function is selected.
data_value	Constant value of the property for a constant temperature dependence function *(unit)*.
data_value(i)	Value of the property for temperature between *temp*(i-1) and *temp*(i) if *function_no*=2, or value of the property at *temp*(i) if *function_no*=3 or 4 *(unit)*.
temp(i)	Temperature at which *data_value*(i) changes to *data_value*(i+1) if *function_no*=2, or temperature at which *data_value*(i) applies if *function_no*=3 or 4 (°C). For temperatures less than *temp*(1), *data_value*(1) is used. For temperatures greater than *temp(num_points)*, *data_value(num_points)* is used.

Pairs of data records as shown between braces, { }, must be included for each index of rock type defined for the simulation region (# SLICE *number*) for a total of num_rock_type record pairs.

Template 28

XPERMEABILITY, *(unit)*
RTFTIME
{index_rock_type(n), *function_no*(n), *num_points*(n)
[*data_value* | (*data_value*(i), *time*(i), i=1 to *num_points*)]},
n=1 to num_rock_type

(unit)	Dimensional units of measure that must be selected from the options shown with the **XPERMEABILITY** keyword. Parentheses are required but the units are case insensitive.
index_rock_type	Index of rock type selected from the set defined under # SLICE.
function_no	Index for selection of the type of time-dependence function for **XPERMEABILITY**. **1** selects a constant time dependence function. There is no variation of the property with time. **2** selects an interpolation function such that the property is a piecewise linear function of time.
num_points	Number of data points defining the time dependence function (maximum of 4). Enter **1** if a constant time dependence function is selected.
data_value	Constant value of the property for a constant time dependence function *(unit)*.
data_value(i)	value of the property at *time*(i) if *function_no*=2, *(unit)*.
time(i)	Time at which *data_value*(i) applies if *function_no*=2, (s), (yr). For times less than *time*(1), *data_value*(1) is used. For times greater than *time(num_points), data_value(num_points)* is used.

Pairs of data records as shown between braces, { }, must be included for each index of rock type defined for the simulation region (# SLICE *number*) for a total of num_rock_type record pairs.

III. Transient Data for each Simulation Period

A group of transient data is read at the beginning of the simulation. Other groups are read during the simulation as necessary, whenever sources, boundary conditions, calculation parameters, or output options are to be changed. Only the parameters that are to be changed need to be input. The remaining parameters will retain their previous values. The format of the transient data is sequential records in the order specified below with keyword groups for sources and parameters.

TIME PERIOD *number*

timchg, deltim, nsrce, lchgpr[T/F], *nparms*

timchg	Time value at which the current simulation period will end and new transient data will be read, (s) or (yr) depending on *iyear* (DIMENSIONS). Enter **-1** to end the simulation. *timchg* ***must*** be at least two times *deltim*.
deltim	Initial time step for this simulation period. Enter **-1** to continue automatic calculation of time step if this is after the first group of transient data.

deltim ***must*** be less than *timchg*. Otherwise, a message is printed to the screen and to the calculation log file and the simulator stops.

nsrce Index for selection of the types of # SOURCE/SINK data groups to follow with new transient data.

- **-1** selects no changes to source/sink data so no # SOURCE/SINK data groups will follow.
- **0** selects all sources/sinks to be shut off for this simulation period so no # SOURCE/SINK data groups will follow.
- **1** selects a **WELL** source/sink data group *or* a **NODE** source/sink data group will follow.
- **2** selects a **WELL** source/sink data group *and* a **NODE** source/sink data group will follow.

lchgpr Logical switch for changing the output-frequency control parameters:

- **T** selects that changes to the output-frequency control parameters are to be read.
- **F** selects that no changes to the output-frequency control parameters are to be read.

If *lchgpr* is true, another full set of PRINT/PLOT CONTROL data records (# PRINT 1 through # PRINT 6) is required, after this record. Note that if all printout to a given output file is turned off for the final simulation time period, that output file *will be deleted* at the end of the simulation.

nparms Number of groups of parameter data, identified by a *KEYWORD group_identifier*, for which values are to be changed. Keyword parameters available for changing include: **POROSITY**, **PERMEABILITY**, **THERMAL CONDUCTIVITY**, **CONDUCTION** heat flux, **TEMPERATURE**, **ENTHALPY**, **PRESSURE**, **PRECIPITATION** flux, and associated precipitation temperature (**TFLUX**). The normal usage for the latter six keyword parameters is to specify boundary-condition values which may change for each simulation period. A keyword data block group of data records is required for each parameter to be changed. Each group is prepared using the templates presented above in the # PARAMETER section.

In the # TIME PERIOD data section, each *KEYWORD* group identifier with its associated data block may appear *only once.* A total of *nparms* data blocks will be read after the set of PRINT/PLOT CONTROL data records are read, if necessary. Multiple data blocks with the same *KEYWORD* group identifier will cause an input error and the simulator will terminate.

The well and point source/sink data groups are described below. One or both or neither of these data groups may be used depending on the specification for *nsrce*.

SOURCE/SINK

WELL, *(unit_flow), (unit_heat)*
{*i, j, well_flow, well_enth, n_openint*
k(m), m=1 to *n_openint*}
0, 0, 0., 0., 0

WELL	Keyword specifying a well-source data group.
(unit_flow)	Dimensional units for mass flow rate of wells **(g/s)**, **(kg/s)**, **(g/yr)**, **(kg/yr)**.
(unit_heat)	Dimensional units for specific enthalpies or temperatures of the injection fluids **(erg/g)**, **(J/g)**, **(J/kg)**, **(kJ/kg)**, or **(C)**, **(Deg.C)**.
i	Index of well location along the x-coordinate. In cylindrical coordinates, this would be 1 for the single well at the axis of the coordinate system.
j	Index of well location along the y-coordinate. Not used in cylindrical coordinates but a zero or a blank space must be included in the input record.
well_flow	Mass flow rate of the well (*unit_flow*). Positive is injection (inflow), negative is production (outflow).
well_enth	Specific enthalpy or temperature of the injection fluid (*unit_heat*). The dimensional units specified determine whether the input value is specific enthalpy or temperature. Enter **0** for a production well. If temperature is specified, specific enthalpy, based on a single-phase fluid, will be calculated.
n_openint	Number of cells with an open or screened interval defined for this well.
k(m)	Index along the z-coordinate of a cell that has an open or screened interval. There must be *n_openint* values of these data.
	Denotes the end of this data set.

Pairs of data records as shown between braces, { }, must be included for each well to be defined for the simulation region or each well to have new transient data specified.

Optional, **WELL** data group required only if *nsrce* is two, may be used if *nsrce* is one. All wells with new data for this # TIME PERIOD must be included in one **WELL** data group. Note that two wells can not coexist in the same column of cells.

SOURCE/SINK

NODE, (unit_flow), (unit_heat)
{*i, j, k, node_flow, node_enth*}
0, 0, 0, 0., 0.

NODE	Keyword specifying a point-source data group.
(unit_flow)	Dimensional units for mass flow rate of point sources **(g/s)**, **(kg/s)**,

	(g/yr), **(kg/yr)**.
(unit_heat)	Dimensional units for specific enthalpies or temperatures of the source fluids **(erg/g), (J/g), (J/kg), (kJ/kg)**, or **(C),(Deg.C)**.
i	Index of point-source location along the x-coordinate.
j	Index of point-source location along the y-coordinate. Not used in cylindrical coordinates. Enter **0.** or **,,**.
k	Index of point-source location along the z-coordinate.
node_flow	Mass flow rate of the point source (*unit_flow*). Positive is a source, negative is a sink.
node_enth	Specific enthalpy or temperature of the source fluid (*unit_heat*). The dimensional units specified determine whether the input value is specific enthalpy or temperature. Enter **0** for a sink. If temperature is specified, specific enthalpy, based on a single-phase fluid, will be calculated.
0, 0, 0, 0., 0.	Denotes the end of this data set.

A data record as shown between braces, { }, must be included for each point source to be defined for the simulation region or each point source to have new transient data specified.
Optional, **NODE** data group required only if *nsrce* is two, may be used if *nsrce* is one. All nodes with new data for this # TIME PERIOD must be included in one **NODE** data group. The point (node) source is ultimately handled by HYDROTHERM like a well source. Note that a well and a node source cannot coexist in the same column of cells.

PRINT/PLOT CONTROL

See previous description of this data record group under I. Problem Specification Data, PRINT/PLOT CONTROL and PRINT 1-6; Optional, required only if *lchgpr*=**T**. If present, the entire group is required, even if only a few parameters are being changed.

PARAMETER

See previous descriptions of these keyword data groups under II. Parameter Data by Keyword-Data Block; Optional, required only if *nparms* is greater than zero.

TIME PERIOD end

timchg

timchg	Enter **-1** to end the simulation.

This ends the data-input file description.

6. OUTPUT DESCRIPTION

A set of up to 26 output files can be produced from running a HYDROTHERM simulation, and many of these files can be quite large. Most of the output files are in list or tabular form designed to be displayed or printed. Some files are intended only for postprocessing by visualization programs and these files can be written in either ASCII text or binary format. The file for a simulation restart is written in ASCII text format.

Data for most variables and parameters are output using the same units of measure employed for their respective input. Some data are output in centimeter-gram-second units as used internally by the simulator. The time unit for output is either seconds or years, as chosen by the user.

6.1 SUMMARY OF OUTPUT FILES

The following list contains the primary name, identification suffix, and a brief description of contents of each of the possible output files. The `Calc_log.`*xxx* and `Out_Probdefine.`*xxx* files are always produced. All other output files are optional. The user-provided identification suffix is represented by *xxx*. Note that this identification suffix is not available when running HYDROTHERM INTERACTIVE. Data are written to the optional output files at a user-specified frequency or time interval as described in section 6.2 and chapter 5 in the PRINT n data section. The files are listed in alphabetical order by primary name.

`Calc_log.`*xxx*	Calculation log with information on time step lengths, iteration counts, residuals, convergence, and maximum changes in dependent variables
`IC_pressporo.`*xxx*	Ending pressure and porosity fields to be used as initial conditions for a future simulation
`Out_Probdefine.`*xxx*	Simulation specification information including: problem title, simulator control, equation solver, mesh geometry, porous media properties, initial conditions, source terms, and simulation periods
`Out_balance.`*xxx*	Fluid and energy global-balance tables for the simulation region
`Out_bcflow.`*xxx*	Flow rate tables, in column format, for boundary cells at specified-value nodes, precipitation recharge nodes, and heat flux nodes
`Out_bcflow2.`*xxx*	Flow rate tables, in spatial-mesh format, for boundary cells at specified-value nodes, precipitation recharge nodes, and heat flux nodes
`Out_density.`*xxx*	Water and steam density fields
`Out_dimensionless.`*xxx*	Dimensionless number fields: Nusselt number, thermal Peclet number
`Out_enthalpy.`*xxx*	Enthalpy fields
`Out_permeability.`*xxx*	Permeability and relative permeability fields for water and steam
`Out_pmthermalprop.`*xxx*	Porous-media property distributions for thermal conductivity and heat capacity
`Out_porosity.`*xxx*	Porosity fields
`Out_potential.`*xxx*	Hydraulic-pressure and potentiometric-head fields for water and steam
`Out_pressure.`*xxx*	Pressure fields
`Out_residual.`*xxx*	Residual fields for the finite-difference equations of flow and energy

	transport
`Out_restartdump.`*xxx*	Restart file of porosity, pressure, enthalpy and temperature fields for use as initial conditions for a future simulation
`Out_saturation.`*xxx*	Saturation fields for water and steam
`Out_source.`*xxx*	Flow rate tables for wells and point sources
`Out_temperature.`*xxx*	Temperature fields
`Out_velocity.`*xxx*	Interstitial-velocity and mass-flux fields for water and steam
`Out_viscosity.`*xxx*	Viscosity fields for water and steam
`Out_watertable.`*xxx*	Water-table elevation fields (unconfined flow only)
`Plot_scalar.`*xxx*	Pressure, enthalpy, temperature, and saturation fields for post-process visualization plots
`Plot_scalar2.`*xxx*	Water-table elevation field (unconfined flow only) for post-process visualization plots
`Plot_vector.`*xxx*	Velocity vector components for post-process visualization plots
`Plot_timeseries.`*xxx*	Time series data of pressure, enthalpy, temperature, and saturation at selected locations in the simulation region for post-process visualization plots

Plot files, `Plot_.`*xxx*, can be output as either ASCII text or binary files. See section 6.3 for tables of contents of the plot files.

6.2 OUTPUT SELECTION

Nearly all of the output is optional and is invoked using print control variables. In the following discussion, the keyword identifiers of the records in the data-input form (table 5.1) containing relevant control variables are listed within parentheses. Every simulation produces a `Out_Probdefine.`*xxx* file containing program version, release number, and date of execution. The simulation title, mesh-dimensioning data, and mesh-geometry information are always printed. Most of the static parameter information for definition of the simulation problem is contained in this output file. Limited, moderate, or long amounts of output are available and are selected using the `long` parameter (PRINT/PLOT CONTROL). The printed static data may include porous-media properties (PRINT 4), fluid properties, initial-condition distributions (PRINT 1, PRINT 5), boundary-condition information (PRINT 3), solution-method information, and well and point source information (PRINT 3). The width of the printout of the dependent-variable and parameter fields can be chosen to be 80, 132, or 159 columns (`wide`, PRINTOUT).

Temporal print intervals for `Out_.`*xxx* and `Plot_.`*xxx* files are specified individually. The intervals can be defined either as the time interval between printouts or the number of time steps between printouts. Note that the automatic time-step algorithm may have its maximum time-step length limited by the minimum print interval specified. The printed transient information may include: solution-method information, primary dependent-variable fields (PRINT 1), density and viscosity fields (PRINT 2), velocity fields (PRINT 3), regional fluid-flow and heat-flow residuals (PRINT 5), boundary-condition flow rates (PRINT 3), well and point source flow rates (PRINT3), global flow- and heat-balance tables (PRINT 5), dimensionless number fields (PRINT 5), and equation residual fields (PRINT 5). The selection of which of the primary dependent

variables (pressure, enthalpy, temperature, or saturation) will be printed is made in PRINT 1. Vertically averaged values of the selected primary dependent variables will be printed if specified in PRINT/PLOT CONTROL. Secondary dependent variables (density, viscosity, velocity, hydraulic pressure, potentiometric head, and water-table elevation) are selected for output in PRINT2 and PRINT3. Up to 10 output files are produced for selected primary and secondary dependent variables. The printout of dependent-variable fields is arranged by vertical slices of the mesh.

Output frequency of the file for restart is selectable (PRINT 5), but output of initial conditions to restart or continue a simulation is handled automatically. That is, if the porous matrix is compressible, the initial pressure and porosity fields, required to restart a simulation, are written to `IC_pressporo.`*xxx*.

The `Calc_log.`*xxx* file is always written and contains the calculation log and summary information about the completed simulation. The quantity of information can be chosen as limited, moderate, or long using the `long` parameter (PRINT/PLOT CONTROL). Selecting limited output (`long` = **0**, PRINT/PLOT CONTROL) produces a time-step summary for the calculation log which includes: time step length, control factor for that step length, current time of simulation, number of Newton-Raphson iterations, number of reductions in the time step, and number of cells undergoing a phase change. Summary information includes: statistics on time-step length and factors that controlled that length. Selecting moderate output (`long` = **1**, PRINT/PLOT CONTROL) causes additional information to be written. After each Newton-Raphson iteration, the maximum mass and energy residual values and locations in the mesh are printed. The individual time-step summaries contain additional information. Selecting long output (`long` = **2**, PRINT/PLOT CONTROL) causes further information to be written. An expanded summary of each Newton-Raphson iteration is produced and global mass and energy balances are written for each time step. The three options for output amount produce `Calc_log.`*xxx* file sizes that are in the ratio of about 1:10:50. Thus, the long output option should be used only for debugging purposes.

6.3 OUTPUT OF DATA FOR POSTPROCESSING INTO PLOTS

Data for contour maps or iso-surface plots of pressure, enthalpy, temperature, saturation, and the secondary-dependent variables, and for vector maps of velocity fields can be written to `Plot_.`*xxx* files (PRINT 6) at user-specified intervals. These plot files can be written in either ASCII text or binary format as a sequence of arrays for each variable in turn at each output time. Data for temporal plots of selected variables—pressure, enthalpy, temperature, and saturation—at specified spatial locations, can be written to the file `Plot_timeseries.`*xxx* (PRINT 6). There is also an option to write plot-file data in tab-separated columnar format (PRINT 6). For unconfined flow systems, elevation data for the water-table surface also are written in the same format. Using these output files, plots can be produced by postprocessing graphical programs, such as HTgnuplot, IDL, or Surfer. HTgnuplot is provided in the release packages of HYDROTHERM Versions 1 and 2.

Tables of contents for each of the output files for plotting are listed below (see section 5.2.2 for definitions of variables). These lists are grouped into sets for different output formats and different plotting programs. When output is selected as a sequence of arrays for all nodes in the global mesh, the data are written in the following order:

I. `Plot_scalar.`*xxx* file
   1. Two title lines
   2. The *xxx* suffix for the simulation
   3. Mesh dimensioning parameters: N_x, N_y, N_z, $N_x \times N_y$, $N_x \times N_z$, $N_x \times N_y \times N_z$
   4. Nodal x-location array: X
   5. Nodal y-location array: Y
   6. Nodal z-location array: Z
   7. Boundary-condition index array: IBC
   8. Time-value header
   9. Pressure label
   10. Pressure array: P
   11. Enthalpy label
   12. Enthalpy array: H
   13. Temperature label
   14. Temperature array: T
   15. Water-saturation label
   16. Water-saturation array: S
   17. Water mass-fraction label
   18. Water mass-fraction array: W
   19. Repeated blocks of items 8–18 based on user-specified print interval

II. `Plot_scalar2.`*xxx* file
   1. Two title lines
   2. The *xxx* suffix for the simulation
   3. Mesh dimensioning parameters: N_x, N_y, N_z, $N_x \times N_y$, $N_x \times N_z$, $N_x \times N_y \times N_z$
   4. Nodal x-location array: X
   5. Nodal y-location array: Y
   6. Nodal z-location array: Z
   7. Boundary-condition index array: IBC
   8. Time-value header
   9. Free-surface elevation label
   10. Free-surface elevation array: FS
   11. Repeated blocks of items 8–10 based on user-specified print interval

III. `Plot_vector.`*xxx* file
   1. Two title lines
   2. The *xxx* suffix for the simulation
   3. Mesh dimensioning parameters: N_x, N_y, N_z, $N_x \times N_y$, $N_x \times N_z$, $N_x \times N_y \times N_z$
   4. Nodal x-location array: X

5. Nodaly-location array: Y
6. Nodalz-location array: Z
7. Boundary-condition index array: IBC
8. Time-value header
9. Water X-velocity label
10. Water velocity in x-direction array: XWVEL
11. Water Y-velocity label
12. Water velocity in y-direction array: YWVEL
13. Water Z-velocity label
14. Water velocity in z-direction array: ZWVEL
15. Water X-mass flux label
16. Water mass flux in x-direction array: XWMFLX
17. Water Y-mass flux label
18. Water mass flux in y-direction array: YWMFLX
19. Water Z-mass flux label
20. Water mass flux in z-direction array: ZWMFLX
21. Steam X-velocity label
22. Steam velocity in x-direction array: XSVEL
23. Steam Y-velocity label
24. Steam velocity in y-direction array: YSVEL
25. Steam Z-velocity label
26. Steam velocity in z-direction array: ZSVEL
27. Steam X-mass flux label
28. Steam mass flux in x-direction array: XSMFLX
29. Steam Y-mass flux label
30. Steam mass flux in y-direction array: YSMFLX
31. Steam Z-mass flux label
32. Steam mass flux in z-direction array: ZSMFLX
33. Repeated blocks of items 8–32 based on user-specified print interval

IV. `Plot_timeseries.`*xxx* file
1. Two header lines defining the columns
2. Transient data in 10 columns: Time step number, Time, i, j, k, Pressure, Enthalpy, Temperature, Saturation, Mass Fraction
3. Repeatedlines of item 2 based on user-specified print interval

A record is written for each cell in the mesh, where i, j, and k are the indices of the cell in the x-, y-, and z-coordinate directions, respectively. All arrays of dimension $N_x \times N_y$ and $N_x \times N_y \times N_z$ are written in natural node number order. This means the x-direction index is incremented first, the y-direction index is incremented second, and the z-direction index is incremented third.

The pressure, enthalpy, temperature, and saturation fields from the `Plot_scalar.`*xxx* file of one

simulation may be used to establish initial conditions for a subsequent simulation. Node-by-node input of initial conditions can be accomplished by copying the appropriate blocks of data from the `Plot_scalar.`*xxx* file to the new data-input file.

When output is selected as a set of tab-separated columns, the data are written in the following order:

I. `Plot_scalar.`*xxx* file
   1. Two title lines
   2. The *xxx* suffix for the simulation
   3. Two header lines defining the columns
   4. Transient data in nine columns: X, Y, Z, Time, Temperature, Pressure, Saturation, Phase Index, Nusselt No.
   5. Repeated lines of item 4 based on user-specified print interval

The phase index indicates the thermodynamic phase of the water-component in each cell.

- **0** indicates the air-water unsaturated region,
- **1** indicates the compressed water region,
- **2** indicates the two phase (water, steam) region,
- **3** indicates the superheated steam region, and
- **4** indicates the supercritical fluid region.

II. `Plot_scalar2.`*xxx* file
   1. Two title lines
   2. The *xxx* suffix for the simulation
   3. Two header lines defining the columns
   4. Transient data in four columns: X, Y, Time, Water-table Elevation
   5. Repeated lines of item 4 based on user-specified print interval

III. `Plot_vector.`*xxx* file
   1. Two title lines
   2. The *xxx* suffix for the simulation
   3. Two header lines defining the columns
   4. Transient data in 10 columns: X, Y, Z, Time, Water x-velocity, Water y-velocity, Water z-velocity, Steam x-velocity, Steam y-velocity, Steam z-velocity
   5. Repeated lines of item 4 based on user-specified print interval

When output is selected as a sequence of arrays for the HTgnuplot program, the data are written in the following order:

I. `Plot_scalar.`*xxx* file

1. Two title lines
2. Mesh dimensioning parameters: N_x, N_z, N_y
3. Nodal x-location array: X
4. Nodal z-location array: Z
5. Nodal y-location array: Y
6. Time-value header
7. Pressure array: P
8. Enthalpy array: H
9. Temperature array: T
10. Water-saturation array: S
11. Water mass-fraction array: W
12. Repeated blocks of items 6–11 based on user-specified print interval

II. `Plot_vector.`*xxx* file

1. Two title lines
2. Mesh dimensioning parameters: N_x, N_z, N_y
3. Nodal x-location array: X
4. Nodal z-location array: Z
5. Nodal y-location array: Y
6. Time-value header
7. Water velocity in x-direction array: XWVEL
8. Water velocity in z-direction array: ZWVEL
9. Water mass flux in x-direction array: XWMFLX
10. Water mass flux in z-direction array: ZWMFLX
11. Steam velocity in x-direction array: XSVEL
12. Steam velocity in z-direction array: ZSVEL
13. Steam mass flux in x-direction array: XSMFLX
14. Steam mass flux in z-direction array: ZSMFLX
15. Repeated blocks of items 6–14 based on user-specified print interval

All arrays of dimension $N_x \times N_y$ and $N_x \times N_y \times N_z$ are written by vertical slices. This means the x-direction index is incremented first, the z-direction index is incremented second, and the y-direction index is incremented third. Note that only the x- and z-coordinate vector components are written to the file, for historical reasons.

6.4 DIAGNOSTIC OUTPUT

The HYDROTHERM simulator provides diagnostic output to three files plus the monitor screen. The output is described below in order of likely usage for determining the cause of a simulation failure.

Screen output and `Calc_log.`*xxx* output includes information for each time step including: time step length, time-step adjustment reason, Newton-Raphson iteration count, maximum mass and energy residuals, convergence criterion met, and maximum changes in dependent variables. The amount of information written

is controlled by the `long` parameter (PRINT/PLOT CONTROL) (see section 6.2). The long output option needs to be selected to obtain iteration information for the linear equation solver, if applicable.

The file `Out_Probdefine.`*xxx* contains general information defining the simulation problem. This includes dimensioning information, parameter information for the fluid and porous medium, initial condition information, and boundary-condition information. The amount of information written is controlled by the `long` parameter (PRINT/PLOT CONTROL) (see section 6.2).

The first set of data in the `Out_pressure.`*xxx*, `Out_enthalpy.`*xxx*, `Out_temperature.`*xxx*, `Out_saturation.`*xxx*, `Out_density.`*xxx*, and `Out_viscosity.`*xxx* files contains the initial condition distributions of pressure, enthalpy, temperature, saturation, density, and viscosity, respectively. The file `Out_permeability.`*xxx* contains the saturated permeability fields and the relative permeability fields. The latter are transient as a function of changes in saturation.

7. GRAPHICAL USER INTERFACE

A graphical user interface has been developed for defining a simulation, running the HYDROTHERM simulator interactively, and displaying the results. The combination of the graphical user interface and the HYDROTHERM simulator forms the HYDROTHERM INTERACTIVE (HTI) program. HTI creates a visual-interactive environment and consists of three components: a preprocessor for setting up a model for simulation, a numerical simulation program, HYDROTHERM Version 3, for flow and heat transport, and a postprocessor for controlling the simulation and visualizing simulation results. The major limitation of HTI is that the simulation region must be two-dimensional.

7.1 OVERVIEW

The graphical user interface comprises a preprocessor program and a postprocessor program. The preprocessor is used to enter the input data and initiate a HYDROTHERM simulation by invoking the postprocessor. The postprocessor runs the simulator and displays the results as contour and vector plots, which can be viewed as animations. The postprocessor can replay the animation of simulation results and also can be invoked in stand-alone mode to view the results of earlier simulations.

In the HTI preprocessor and postprocessor, options are selected and data are entered using pull-down menus with buttons, check boxes, and text fields. Some of the menu items open a dialog window for data input. Input data that are not spatially dependent are entered into tables or forms. Data that are spatially dependent are entered and edited with a set of graphical tools. The preprocessor generates an input file for the HYDROTHERM simulator. After all required data have been entered, the user can run the simulator from within the graphical user interface. The postprocessor controls the simulation and displays results graphically during the transient evolution of the simulation. Simulation time value and error metrics for fluid and thermal energy balances also are displayed as the simulation progresses. The output files available from the simulator, described in chapter 6, also can be written when using the HTI program.

HTI is written in both Java and C programming languages and executes on personal computers and workstations running Windows or Linux/Unix operating systems. HTI is public-domain software (see appendix 1) except for the utility program Triangle: A Two-Dimensional Quality Mesh Generator and Delaunay Triangulator (`triangle.c`[1]). Although used by HTI, `triangle.c` is *not* considered to be part of the HTI source code.

This chapter provides a summary of the features and operation of the graphical user interface, HTI. Detailed information on the use of HTI can be found in the on-line help manual that is contained in the software distribution package. The on-line help manual is read using the JavaHelp System program provided in the HTI distribution. A separate on-line help manual is provided for the HTI postprocessor.

---

[1] `triangle.c, version 1.3`, copyright by Jonathan R. Shewchuk, available at URL;
`www.cs.cmu.edu/~quake/triangle.html`

7.2 PREPROCESSOR

The preprocessor consists of a main window with associated menus and toolbar buttons and a drawing area for entering graphical information (fig. 7.1). Options are selected from menus and numerical data are entered into tables and forms that appear in separate dialog windows. The HTI preprocessor creates an ASCII text file that is passed to the HYDROTHERM simulator. The content and format of this file conform to the descriptions given in chapter 5.

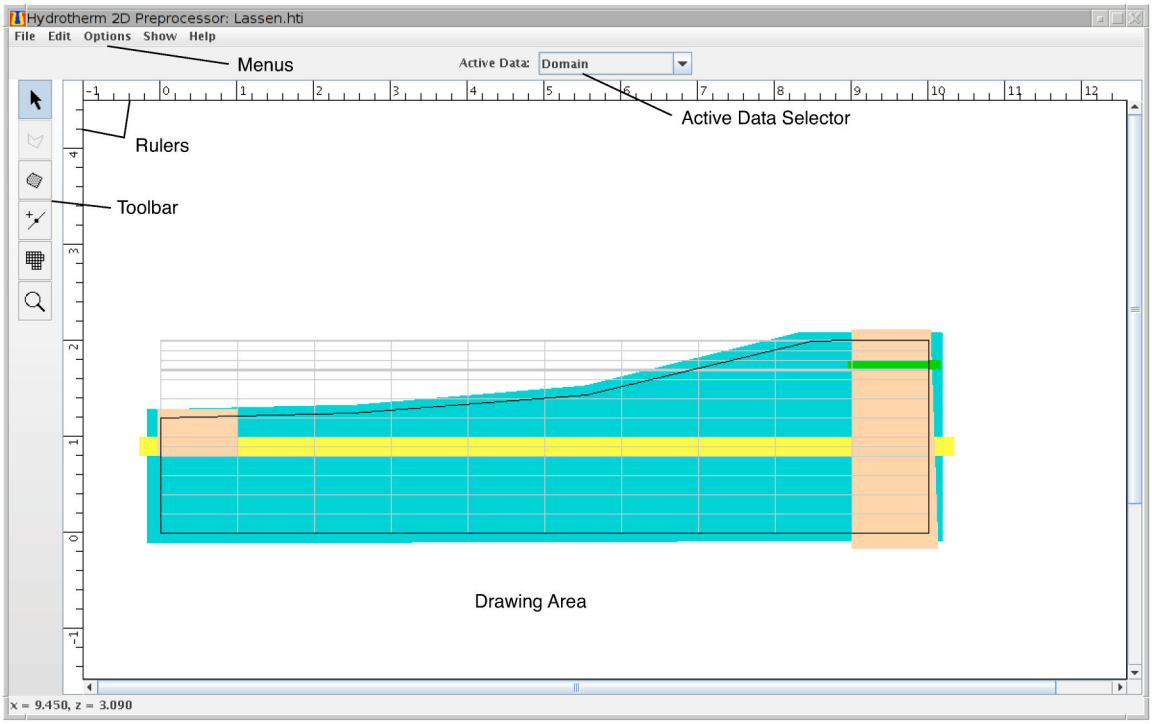


Figure 7.1. Main window of the HTI preprocessor with an example region showing rock units and grid.

The drawing area is where spatial data are viewed graphically. The `Active Data` selector is used to choose the type of spatial data to be entered or edited in the drawing area. Types of spatial data include the model domain, porous media (rock type) zones, initial conditions, boundary conditions, fluid sources, observation points, finite-difference mesh, and site map. Several of these data types are optional for a given simulation. The tool bar, on the left side, contains buttons for data entry and editing. Menus are used for file management, displaying and editing data, selecting options, and accessing the help file. Horizontal and vertical rulers show the range of the two-dimensional coordinate system that is currently visible in the drawing area. Coordinate rulers can display, in the drawing window, either model distance units of measure or inches for a printed page. The coordinates of the cursor appear in the lower left corner of the window. These graphical user interface elements are described more fully in the following sections.

7.2.1 Pull-Down Menus

Five pull-down menus are located across the top of the main window: **File, Edit, Options, Show,** and **Help** (fig. 7.1). The items in any of the pull-down menus that end with three dots, **...**, open a dialog window upon selection.

7.2.1.1 File Menu. The **File** menu provides the selection of a new (**New**) or existing (**Open...**) file for use as HTI input. It is called a document file and contains all the data necessary to specify a HYDROTHERM simulation using the preprocessor. It is not intended to be read by the user, but, rather, to save the work done with the preprocessor between HTI work sessions. The current input document file can be saved (**Save**) or renamed (**Save As...**). Note that the HTI document input file is not the same as the HYDROTHERM data-input file described in chapter 5.

A data-input file is automatically created by HTI to transmit the simulation definition data to HYDROTHERM. This data-input file can be saved to a directory specified by the user with the **Export data...** menu item. An exported data-input file can be used to run the HYDROTHERM simulator outside of the graphical user interface, with the possibility of further editing. An exported data-input file also can be run by the postprocessor in stand-alone mode as described in section 7.3. However, existing HYDROTHERM input data files *cannot* be imported directly into the preprocessor. The contents of the drawing area can be printed using the **Print...** menu item or saved as a bitmap using the **Export bitmap...** menu item. The **Exit** menu item terminates the HTI preprocessor program.

7.2.1.2 Edit Menu. The **Edit** menu provides the selection of editing tools including **Undo**, **Redo**, **Select All**, **Delete**, **Bring to Front**, and **Send to Back**. Which menu items appear and are active depends on the selection for `Active Data`. These menu items are described in section 7.2.2 on spatial data.

7.2.1.3 Options Menu. The **Options** menu contains three items: **Model...**, **Drawing...**, and the **Buffered Graphics** check box. Selection of **Drawing...** opens the `Drawing Options` dialog window which is used to set the size and scales of the drawing area as well as the location of the origin and ruler units. There is also a provision for selecting the number of decimal places used to display coordinate distances in the preprocessor. Selection of **Model...** opens the `Model Options` dialog window containing seven tabbed data forms for selecting options and entering parameters. These seven data forms: **Basic**, **Units**, **Relative Permeability**, **Initial Conditions**, **Time Step**, **Newton-Raphson**, and **Solver**, are described below.

The **Basic** data form provides for entry of two title lines identifying the simulation, radio buttons for selection of coordinate system, selection of confined or unconfined flow (air-phase present or absent), selection of the spatial weighting for coefficients in the finite-difference equations, and selection of enthalpy or temperature for specification of heat content for sources. If required, the threshold potential value for central spatial weighting (defined under Weighting and Averaging, section 5.2) can be entered. If unconfined flow is being simulated, the value for atmospheric pressure can be entered.

The **Units** data form provides for selection of units of measure for the parameters and variables. Selections are made from lists of available units. If any units are changed after numerical data have been

entered, the user must reenter the data affected by the change, because HTI *will not* automatically convert numerical data to different units.

Note that the units chosen for the distance measurements in the domain may affect the stability of the simulation. This is because HYDROTHERM is a three-dimensional simulator; for Cartesian coordinates, the third dimension of the domain, which is perpendicular to the two dimensions accessible with the graphical user interface, is always one distance unit. Thus, a very thin simulation domain may result from using, for example, centimeter units to describe a field-scale domain. A very thin domain may necessitate considerable experimentation with the settings of the time-step, Newton-Raphson, and iterative-solver control parameters in order to obtain a stable simulation.

The **`Relative Permeability`** data form provides for selection of the relative-permeability function and entry of residual saturation values. Three functions for relative permeability are available: linear, Corey, and fracture. For unconfined flow, entry of the parameters for the corresponding water-saturation function also is done in this data form. Three functions for water saturation are available: linear, Corey, and Corey-Cooley. Information about these functions is given in sections 2.2.7 and 2.2.8 and in the HTI on-line help files.

The **`Initial Conditions`** data form provides options to specify initial conditions for the pressure and enthalpy fields. Temperature may be used instead of enthalpy. Both tabular and graphical methods are available for data input. There are two options for specifying initial conditions for the pressure field: a hydrostatic profile or graphical specification. A check box allows for the specified pressure at the top boundary of the domain to be applied to the nodes of the uppermost active cells of the mesh or grid. There are four options for specifying initial conditions for the temperature field; linear interpolation between temperatures at the top and bottom of the mesh, linear extrapolation from the temperature at the top of the mesh using a temperature gradient with depth, a temperature profile with depth based on the boiling point pressure curve, or graphical specification. A check box allows for the specified temperature at the top boundary of the domain to be applied to the nodes of the uppermost active cells of the grid. Initial conditions for the enthalpy field can be specified only graphically.

The **`Time Step`** data form provides for entry of the parameters that control the automatic time step algorithm. These parameters include: the maximum number of time steps allowed to complete the simulation, the factor for the maximum increase in time step, the maximum percentage change allowed in pressure per time step (confined flow), the maximum absolute change allowed in pressure per time step (unconfined flow), the maximum percentage change allowed in enthalpy per time step, the maximum change allowed in liquid water saturation per time step, the minimum time step allowed, and the maximum number of times a time step may be reduced before simulation failure is declared.

The **`Newton-Raphson`** data form provides for entry of the parameters that control the Newton-Raphson algorithm for solution of the non-linear finite-difference approximations to the flow and heat transport equations. These parameters include: the maximum number of Newton iterations allowed per time step, the convergence tolerance on the residual for the flow equation, the convergence tolerance on the residual for the heat-transport equation, the maximum number of cells allowed to change phase per Newton iteration, the convergence tolerance on the percentage change in pressure (confined flow) or the absolute change in pressure

(unconfined flow) for the flow equation, and the convergence tolerance on the percentage change in enthalpy for the heat transport equation. These convergence tolerances are specified as the maximum changes over the domain.

The **Solver** data form provides for entry of the parameters that control the choice and operation of the linear equation solver for solution of the Newton-Raphson equations. Solver choice is a direct banded matrix equation solver or an iterative equation solver based on the generalized minimum residual (GMRES) method. Parameters for GMRES include: selection of the number of solution vectors computed between restarts of the solver, the convergence tolerance on the relative residual, and the maximum number of iterations allowed. The choice of a preconditioning method for GMRES is: ILU preconditioning with a maximum level of fill-in elements (ILUK), or ILU preconditioning with a drop tolerance and maximum number of fill-in elements (ILUT). The level of fill-in must be specified for ILUK, whereas the relative drop tolerance and the maximum number of elements per row of L and of U must be specified for ILUT. Further explanations of these parameters appear in section 3.6.2.3 and 3.6.4.2 and in the solver section of the data file described in chapter 5.

7.2.1.4 Show Menu. The **Show** menu is used to select the type of spatial data to display in the drawing area. One or more of these data types can be displayed simultaneously. Selecting the **Rock Properties Window** menu item opens the `Rock Properties` window containing a table of all defined rock types and their property values. Selecting the **Simulation Period Window** menu item opens the `Simulation Period` window containing a table of all defined simulation periods with duration and first time step. There is also a table of print options showing which variables and parameters have been selected for printing to files. Selecting the **Postprocessor** menu item opens the `Postprocessor` window, which is discussed in section 7.3.

7.2.1.5 Help Menu. The **Help** menu has two items: **Contents...** , and **About...** . Selecting **Contents...** opens a JavaHelp window with the table of contents of the on-line help manual for the HTI preprocessor and postprocessor appearing in a sub-window on the left side. This manual contains detailed information on the use of HTI, including descriptions of all windows, menus, types of spatial data and parameter data, and a step-by-step tutorial on setting up a new simulation. The table of contents and page links can be used to jump to any topic of interest in the manual. The section on how to set up a simulation presents a logical sequence of steps to input the data for a new simulation. Selecting **About...** displays the version number of HTI, the disclaimer, the use of the `triangle.c` graphical tool under copyright, the authorship of HTI and HYDROTHERM, and the contact for inquires about HTI.

7.2.2 Entering and Editing Spatial Data

Nine types of spatial data are associated with a HYDROTHERM simulation. Five of these types are optional. To enter or edit a particular data type, it must be selected as the `Active Data` in the main window of the preprocessor (fig. 7.1). The following sections describe usage of the menu items available for `Active Data`.

7.2.2.1 Domain. The domain is the region in which fluid flow and heat transport are simulated. It corresponds to the active mesh region defined in chapter 3. The domain is defined by an exterior boundary. Internal subregions may be excluded by defining them with interior boundaries. Domain boundaries are

formed by polygons with any number of vertices. The six buttons on the toolbar, on the left side of the drawing area, are used to manipulate polygons. The top button is used to select a polygon for reshaping or moving. The second button down from the top is used to create a polygon as an exterior boundary. The third button down from the top is used to create a polygon as an interior boundary. The fourth button down from the top is used to add vertices to a polygon. The fifth button down from the top is used to show the discretized domain region, and the sixth button down from the top is used to magnify or reduce the viewing range of the drawing area. Items dealing with domain polygons in the **Edit** menu include: **Undo**, **Redo**, **Select All**, and **Delete**.

The result of drawing the domain in the two-dimensional coordinate space of the graphical user interface is a 3-dimensional domain for the HYDROTHERM simulator which is one unit thick in the third dimension, for the case of Cartesian coordinates. For cylindrical coordinates, the three-dimensional domain is a cylinder of radial-vertical cross section as specified in the drawing area and with angular symmetry. See the note under the units options in section 7.2.1.3 about possible consequences of defining a very thin simulation domain.

7.2.2.2 Rock Units. The rock units drawing shows the spatial distribution of porous-medium zones. Each zone is a polygon within which flow and heat transport properties of the porous medium are uniform. The properties for each rock unit are entered into the table in the `Rock Properties` window. The five buttons on the toolbar, on the left side of the drawing area, are used to manipulate zones. The top button is used to select a zone polygon for reshaping or moving. The second button down from the top is used to create a polygon zone. The third button down from the top is used to add vertices to a polygon zone. The fourth button down from the top is used to view the discretized zone map (available after the mesh is defined), and the fifth button down from the top is used to magnify or reduce the viewing range of the drawing area. Items dealing with rock unit zones in the **Edit** menu include: **Undo**, **Redo**, **Select All**, **Delete**, **Bring to Front**, and **Send to Back**. The last two items are used to handle overlapping zones. When a portion of the domain is covered by more than one rock unit, the unit that is currently visible in the drawing area determines the properties of that subdomain. Any portion of the domain that is not covered by a rock-unit zone is defined to be an inactive or excluded zone.

When the rock units map is selected as the `Active Data`, the `Rock Properties` window appears. Figure 7.2 shows the `Rock Properties` window for the example region of figure 7.1. This window may be displayed also from the **Show** menu by selecting the **Rock Properties Window** item. It contains a table of parameter values for each rock unit. The color codes in figure 7.2 correspond to the colors of each rock unit appearing in figure 7.1. The rock parameters are defined in the on-line help file and in chapter 5.

The user *cannot* enter data directly into the cells of this table. Instead, the buttons below the table are used to **Add**, **Edit**, or **Delete** a rock type. On first opening, the `Rock Properties` window contains a single rock type named `inactive`. This is the default rock type and represents impermeable zones of the domain. None of the parameters in the inactive rock type may possess a value. Before drawing a polygon defining the shape of a rock-type zone, at least the rock-type name must be entered into the table. Pressing the **Add** button causes the `Add Rock Properties` dialog window to appear. Here the display color, name, and property values are specified. Definitions of the properties are given in the on-line help file and in chapter 5.

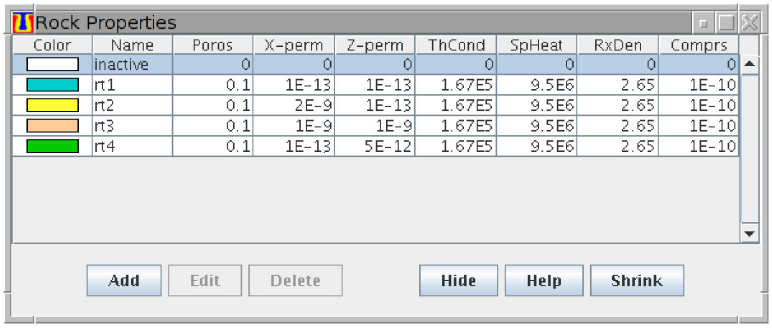


Figure 7.2. Rock properties window of the HTI preprocessor for the example region shown in figure 7.1.

Buttons in the **func(T)** set on the right hand side of the `Add Rock Properties` dialog window enable specification of properties as functions of temperature using a `Temperature dependence for...` dialog window. When a type of function is chosen, a set of boxes appears for entry of the appropriate function parameters. Different functions can be selected for each parameter for each rock type defined.

Buttons in the **func(d)** set on the right-hand side of the `Add Rock Properties` dialog window enable specification of properties as functions of depth using a `Depth dependence for...` dialog window. When a type of function is chosen, a set of boxes appears for entry of the appropriate function parameters. Different functions can be selected for each parameter for each rock type defined.

Buttons in the **func(t)** set on the right-hand side of the `Add Rock Properties` dialog window enable specification of properties as functions of time using a `Time dependence for...` dialog window. When a type of function is chosen, a set of boxes appears for entry of the appropriate function parameters. Different functions can be selected for each parameter for each rock type defined. Although only the x-permeability parameter has a function-of-time button, the other permeabilities will take on the same function if they are related to the x-permeability by a multiplicative factor.

The zones for each rock type can be drawn using the drawing tools from the toolbar after highlighting the corresponding row for that rock type in the table. Property values for existing rock types can be edited after highlighting the row of the desired rock type in the table and pressing the **Edit** button. The `Edit Rock Properties` dialog window opens and values can be changed. An entire definition of properties for a selected rock type can be removed using the **Delete** button. The zone for the corresponding rock type disappears from the drawing area.

7.2.2.3 Initial Pressure. Initial conditions for pressure must be specified for a simulation. If the `Specify graphically` option is chosen in the **Initial Conditions** data form of the **Model...** item in the **Options** menu, the user may draw contour lines of equal pressure across the simulation region. The six buttons on the toolbar allow selection of contour lines, addition of open or closed contours, addition of a vertex to a contour line, display of discretized initial conditions, zooming in or out, and viewing labels on the contour lines. Items dealing with initial pressure conditions in the **Edit** menu include: **Undo**, **Redo**, **Select All**, **Delete**, **Contour Value...**, **Interpolation Method...** and **Color Scale...**. Values on the contours can be edited using the **Contour Value...** item.

Interpolation Method... is used to select the method of interpolation for the contour display as continuous or stepwise, and **Color Scale...** is used to modify the color scale used for contour display.

7.2.2.4 Initial Temperature. Initial conditions for temperature or enthalpy must be specified for a simulation. If the Specify graphically option is chosen in the **Initial Conditions** data form of the **Model...** item in the **Options** menu, the user may draw contour lines of equal temperature across the simulation region. This method of temperature specification is analogous to that described in section 7.2.2.3 for pressure. The six buttons on the toolbar allow selection of contour lines, addition of open or closed contours, addition of a vertex to a contour line, display of discretized initial conditions, zooming in or out, and viewing labels on the contour lines. Items dealing with initial temperature conditions in the **Edit** menu include: **Undo**, **Redo**, **Select All**, **Delete**, **Contour Value...** , **Interpolation Method...** and **Color Scale...** . Values on the contours can be edited using the **Contour Value...** item. **Interpolation Method...** is used to select the method of interpolation for the contour display as continuous or stepwise, and **Color Scale...** is used to modify the color scale used for contour display.

7.2.2.5 Initial Enthalpy. Initial conditions for enthalpy or temperature must be specified for a simulation. Radio buttons in the **Initial Conditions** tabbed data form ensure that only one of these is chosen. Only graphical specification is available for the initial distribution of enthalpy. The user may draw contour lines of equal enthalpy across the simulation region. The six buttons on the toolbar allow selection of contour lines, addition of open or closed contours, addition of a vertex to a contour line, display of discretized initial conditions, zooming in or out, and viewing labels on the contour lines. Items dealing with initial enthalpy conditions in the **Edit** menu include: **Undo**, **Redo**, **Select All**, **Delete**, **Contour Value...** , **Interpolation Method...** and **Color Scale...** . Values on the contours can be edited using the **Contour Value...** item. **Interpolation Method...** is used to select the method of interpolation for the contour display as continuous or stepwise, and **Color Scale...** is used to modify the color scale used for contour display.

7.2.2.6 Boundary Conditions. Boundary conditions must be specified for each simulation period. A simulation period is a time span during which the boundary conditions and sources remain constant. Simulation periods must be defined before boundary conditions can be specified. Selecting **Boundary Conditions** as the Active Data causes the Simulation Period window to open. This window also can be opened from the **Show** menu. Each row of the table in this window corresponds to a different simulation period. To add a new simulation period, the row of the previous simulation period first must be highlighted. Then, pressing the **Add** button causes the Add Simulation Period dialog window to appear. Here the parameters for the next simulation period are specified. The Print Options: table in the Add Simulation Period dialog window provides control of output to the various files available. These files are described in chapter 6. Printout intervals can be set to an amount of simulation time, a number of time steps, or the end of the simulation period. The parameters for this dialog window are defined in the the on-line help file and in chapter 5.

The parameter values in the table can not be edited directly. Parameters for an existing simulation period can be edited after highlighting the row of the desired period and pressing the **Edit** button. The Edit Simulation Period dialog window opens and values defining the simulation period and printout

intervals can be changed. An entire simulation period can be removed using the **Delete** button. The parameters for this dialog window are the same as for the Add Simulation Period dialog window.

Boundary conditions must be specified over each exterior and interior boundary segment. A boundary segment is the straight line connecting two vertices of a boundary polygon. Specified values of the dependent variables (pressure, enthalpy, temperature) are uniform along each boundary segment. However, there is an option for pressure to vary linearly along a boundary segment. If it is desired that boundary values for enthalpy or temperature vary along a segment, that segment must be subdivided into smaller segments. Then, piecewise uniform values can be specified along each of the smaller segments. Specified values of the dependent variables (pressure, enthalpy, temperature) can be inherited optionally from the initial conditions. This enables values calculated by the HYDROTHERM simulator to be used as boundary-condition values; for example, hydrostatic distribution of pressures or linear distribution of temperatures. Specified flux boundary conditions (precipitation-recharge, basal heat) can be either uniform or vary linearly along a boundary segment. Note that approximating a continuous variation of a boundary-condition value by piecewise uniform values may introduce artificial fluxes. The default boundary condition is that of no flow and no conductive or advective heat transport across that boundary segment.

After the simulation periods have been defined, boundary-condition types and values can be specified. **Boundary Conditions** must be selected as the Active Data and the period of interest must be highlighted in the Simulation Period window. Boundary conditions are specified for one or more boundary segments. The arrow tool is used to select a boundary segment in the drawing area. Multiple segments are selected by holding down the left mouse button while dragging the mouse across them. The boundary segments selected are highlighted in gray. Clicking the **BC** button on the toolbar causes a dialog window to open entitled BC for Simulation Period n. Here the type of boundary condition can be selected and applied to the selected simulation period or to the selected and all subsequent simulation periods. Segments with specified types of boundary condition are colored according to the key of colored icons for boundary condition types shown on the toolbar. Other buttons on the toolbar toggle the display of boundary-condition values, view discretized boundary segments on the grid (available after the grid is defined), and magnify or reduce the range of viewing in the drawing area. The **Edit** menu contains two items pertaining to boundary conditions. **Copy All To...** enables all data for all boundary conditions to be copied from one simulation period to another, while **Corner Policy...** determines whether the boundary condition of the horizontal boundary segment or the vertical boundary segment is applied to corner cells where these segments meet.

7.2.2.7 Fluid Source. Fluid sources are represented in HYDROTHERM by well sources and point sources. Well sources communicate with one or more cells in a column of the grid, whereas point sources communicate with a single cell each. A sink is simply a source with a negative fluid flow rate. A well is applicable for a two-dimensional simulation domain when a cylindrical coordinate system is being used and there is a single well at the origin of the radial coordinate. A well is applicable also for a two-dimensional areal simulation domain, although it becomes identical to a point source. Wells are not applicable to a two-dimensional simulation domain consisting of a vertical slice in a Cartesian coordinate system. Fluid sources are an optional type of spatial data. Selecting **Fluid Source** as the Active Data causes a button for adding a point source to appear on the toolbar. This button will appear also by using the **Show** menu. Clicking

this button and selecting a location for the point source causes the Fluid Source window to open. It contains a table of the currently specified point source for all the simulation periods defined and includes a data form and an **Apply** button for specifying the flow rate and enthalpy or temperature. The choice of source enthalpy or temperature is made with a radio button in the **Basic** tabbed data form opened from the Model Options dialog window. To modify a source, select it with the arrow tool and click the right hand button on the mouse. Select the **Value...** item from the menu. A black dot will show which source has been selected and the Fluid Source dialog window will appear. Then values for flow rate, enthalpy, or temperature can be edited.

7.2.2.8 Observation Points. Observation points are locations in the domain where pressure, enthalpy, temperature, and liquid saturation are written to the output file Plot_timeseries.*xxx*, during the simulation at the time interval specified by the user. Observation points are an optional type of spatial data. Selecting **Observation Points** as the Active Data causes the button for adding an observation point to appear on the toolbar. This button will appear also using the **Show** menu. Clicking on this button and selecting a location for the observation point specifies it for transient data accumulation. Points can be moved using the arrow button on the toolbar. Items dealing with observation points in the **Edit** menu include: **Undo**, **Redo**, **Select All**, and **Delete**.

7.2.2.9 Grid. The HYDROTHERM simulator uses a finite-difference method to discretize the governing equations as described in section 3.1.1. Spatial discretization of the simulation region is accomplished using a mesh or grid of rectangular cells. Selecting **Grid** as the Active Data causes seven buttons for defining and editing the grid to appear on the toolbar. The grid may be edited either graphically in the drawing area or using the table in the Grid dialog window. This box appears by selecting the sixth button from the top on the toolbar.

A grid consisting of cells with uniform dimensions in each of the two coordinate directions is created by selecting the **Uniform Grid...** item from the **Edit** menu. Selecting **Uniform Grid...** opens the Uniform Grid dialog window where the number of cell columns and cell rows comprising the grid are specified. The rectangular boundary of the initial grid will just contain the polygon of the simulation domain. If the shape of the simulation domain is modified later, there is an option, using a radio button, of shrinking or expanding the boundary of the grid to just contain the active simulation domain. In any case, cells outside the simulation domain are flagged as inactive cells for the HYDROTHERM simulator. Buttons on the toolbar enable the user to move the entire grid, resize the grid, move individual grid lines, add vertical or horizontal grid lines, magnify or reduce the viewing range of the drawing area, show or hide the Grid dialog window, and display the grid, boundary conditions, sources, and observation points in the drawing area.

The Grid dialog window can be used to add vertical or horizontal grid lines and edit their locations. Items dealing with the grid in the **Edit** menu include: **Undo**, **Redo**, **Uniform Grid...**, **Subdivide...**, **Select All**, and **Delete**. Some items are activated using buttons on the toolbar. The on-line help manual explains the various procedures for editing the grid.

7.2.2.10 Site Map. A site map may be used as an overlay on the drawing area to guide the graphical specification of the simulation domain, boundary conditions, and rock-type zones. This map of spatial data is optional. The site map must be in the gif or jpeg graphics file format and can not be edited within the HTI

preprocessor. Selecting **`Site Map`** as the `Active Data` causes four buttons for dealing with the site map to appear on the toolbar. The fourth button from the top of the toolbar opens the `Load Site Map` dialog window, where the directory path of the site map file desired is selected or specified. The top button on the toolbar moves or resizes the site map. The second button is for stretching or compressing the map while keeping one point fixed. This feature can be used to align the map with the coordinate rulers of the drawing area. This procedure is explained in the on-line help manual. The third button magnifies or reduces the viewing range of the drawing area.

7.3 POSTPROCESSOR

The HTI postprocessor consists of a window to display simulation results including fluid and heat balance errors, buttons on a toolbar to control a simulation, and menu items to save simulation results in a file for later playback (fig. 7.3). The postprocessor is interfaced with the numerical simulator HYDROTHERM 3 in such a way that as the simulator runs, the calculated dependent variable fields may be displayed for each time step, thus creating a crude animation of model results. The dependent variables available for animated display include: pressure, enthalpy, temperature, phase region, water saturation, and fluid mass flux (water, steam, or super-critical fluid). The speed of the animation depends on the computation time taken for each time step on the computer platform being used. Because the time step length varies during the simulation, the resulting animation will be distorted along the simulation time axis. The quality of animation depends on the time-step length relative to the rates of change of the field being displayed, because no interpolation of dependent-variable plots within time steps is done.

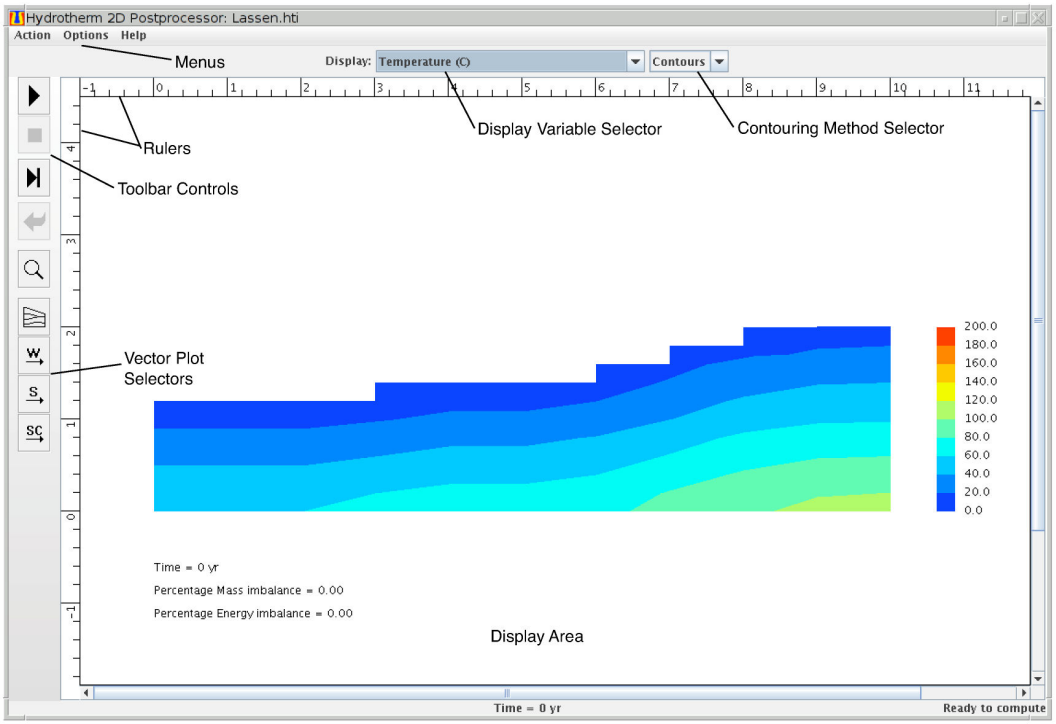


Figure 7.3. Main window of the HTI postprocessor showing an initial condition temperature field.

An example of the HTI `Postprocessor` window at the end of a simulation appears in figure 7.4. Here a contour plot of a temperature field is displayed along with a color scale. Below the plot is the simulation time value and information on fluid and heat balance errors. The nine buttons on the toolbar enable the user to run a simulation, stop a simulation, advance a simulation one time step, reset a simulation for replay (in playback mode), magnify or reduce the viewing range of the display window, and toggle the display of the rock-type distribution and mass-flux vector fields for water, steam, or super-critical fluid.

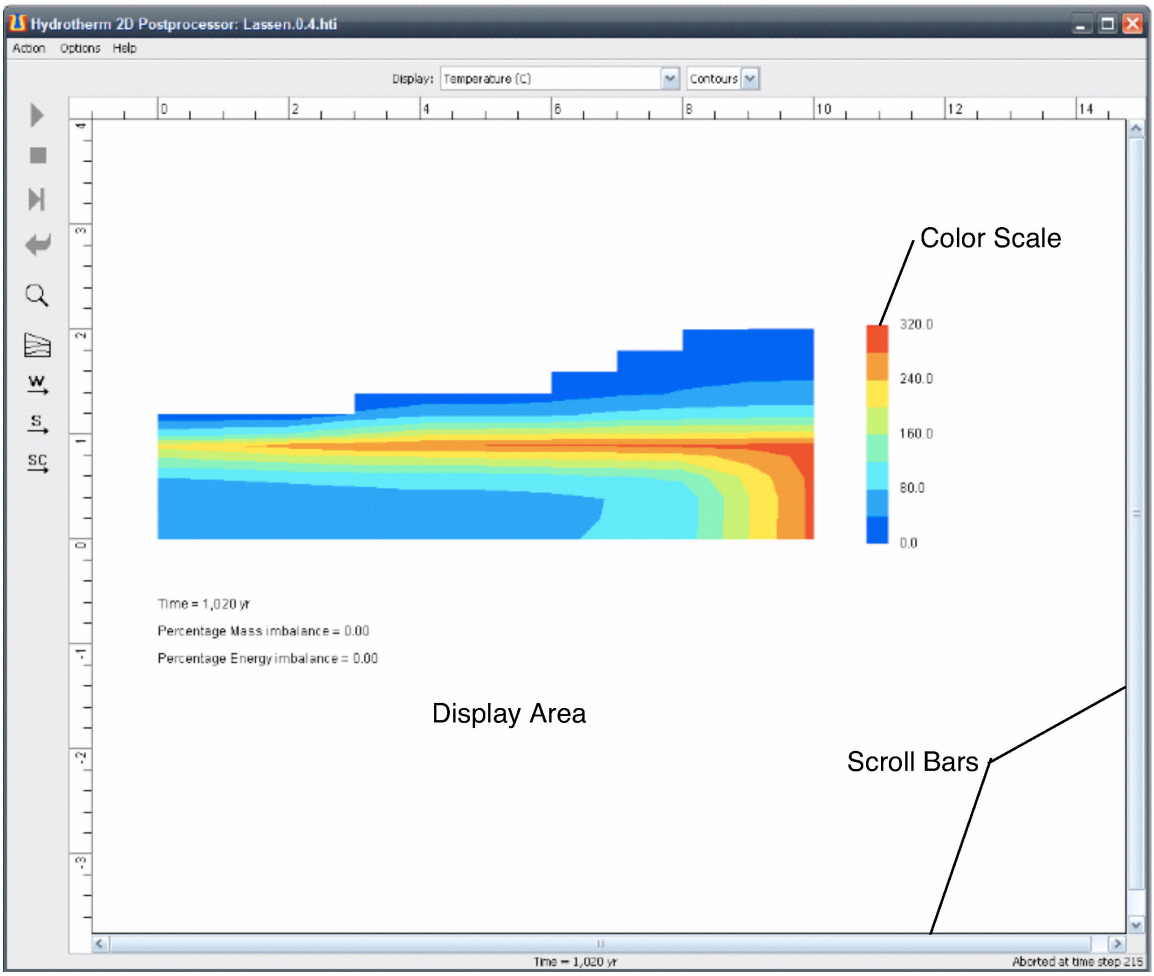


Figure 7.4. An example temperature field display of the HTI postprocessor.

The border above the postprocessor display area has two drop-down lists for `Display` options. The list on the left contains the menu of dependent variable fields available for display: **`Pressure`**, **`Enthalpy`**, **`Temperature`**, **`Phase Region`**, **`Liquid Water Saturation`**, **`Liquid Water Mass Flux`**, **`Steam Mass Flux`**, **`Super-Critical Fluid Mass Flux`**, **`Peclet Number`**, **`Nusselt Number`**, or **`None`**. The list on the right gives the choice of display as **`Cells`** or **`Contours`**. Selection of **`Cells`** causes display of the chosen variable with each cell having a uniform color. Selection of **`Contours`**

causes display of the chosen variable as a contour map with each band between contours having a uniform color. In both cases, a color scale appears to the right side of the simulation domain. The color scale can be modified using the dialog window described in section 7.3.1.2.

The fluid-mass and thermal-energy balance errors appear at the bottom of the display area. They indicate the accuracy of the numerical solutions to the flow and thermal-transport equations. However, a relatively small global-balance error (for example, less than 0.1 percent) is a necessary but not a sufficient condition for an accurate simulation as discussed in section 3.7.

7.3.1 Pull-Down Menus

Three pull-down menus are located along the top of the `Postprocessor` window: **`Action`**, **`Options`**, and **`Help`** (fig. 7.3). The menu items that end in three dots, **`...`**, open a dialog window upon selection.

7.3.1.1 Action Menu. The **`Action`** menu contains five items: **`Terminate Computation`**, **`Restart Computation`**, **`Print`**, **`Export Bitmap...`**, and **`Done`**. Selection of **`Terminate Computation`** stops the simulation at the completion of the current time step and puts the postprocessor into playback mode. Selection of **`Restart Computation`** resets the simulator to the initial conditions and restarts the postprocessor in computational mode. Selection of **`Print`** causes a printout of the `Postprocessor` window. Selection of **`Export Bitmap...`** causes output of the `Postprocessor` window to a file in bitmap format. Selection of **`Done`** terminates the HTI postprocessor.

7.3.1.2 Options Menu. The **`Options`** menu contains five items: **`Drawing...`**, **`Color Scale...`**, **`Vector...`**, **`Simulation...`**, and the **`Buffered Graphics`** check box. Selection of **`Drawing...`** opens the `Drawing Options` dialog window, which is used to set the size and scales of the drawing area as well as the location of the origin and ruler units. There is also provision for selection of the number of decimal places used to display coordinate distances in the postprocessor. Selection of **`Color Scale...`** opens the `Color Scale` dialog window. Here the lower and upper contour levels and the contour interval are specified. Note that the spectrum range of colors from blue to red can not be modified. Selection of **`Vector...`** opens the `Vector` dialog window. Here the scale and spacing of the vectors are specified. Selection of **`Simulation...`** opens the `Simulation` dialog window. The simulation results can be saved for later playback by entering a value for the `Target time interval between save`, clicking on the check box for the option `Save simulation`, and selecting the types of output to be saved. The options are: pressure, enthalpy, temperature, phase region, water saturation, and fluid mass flux (includes water, steam, and super-critical fluid). Simulation results will be saved when integral multiples of the target time interval have elapsed. Note that increasing the value for the target time interval will cause the animation speed to appear faster. However, the smoothness of the animation will decrease. Playback mode can display a more rapid animation than computational mode because the simulation work has already been done.

The checkbox for **`Buffered Graphics`** causes graphical objects to be saved in memory resulting in faster response when moving among different displays. It is a desirable option for computers with at least several hundred megabytes of memory available.

7.3.1.3 Help Menu. The **Help** menu has the item **Postprocessor Help...** . Selecting **Postprocessor Help...** opens a JavaHelp window with the table of contents of the on-line help manual for the HTI postprocessor appearing in a sub-window on the left side. This manual contains detailed information on the use of the HTI postprocessor, including descriptions of all windows, menus, and types of spatial data and parameter data. The table of contents and page links can be used to jump to any topic of interest in the manual.

7.3.2 How to Run a Simulation

After the setup of a simulation is completed, the postprocessor can be invoked to run the model by clicking the **Show** menu and selecting **Postprocessor**. The Postprocessor window will be displayed. If the data set is not complete for the simulation, the Postprocessor window will not appear, and an error message will be displayed. After entering the missing data, the Postprocessor can then be restarted.

While the Postprocessor window is displayed, the setup of the simulation *cannot* be modified from within the preprocessor. Control buttons on the left side allow the user to run the playback, stop the playback, and advance one time step. The Postprocessor window is closed by clicking the **Action** menu and selecting **Done**. This returns the user to the Preprocessor window.

7.3.3 How to Replay a Simulation

The HTI postprocessor may be invoked independently of the HTI preprocessor to replay the saved output of a previous simulation or to run a simulation using data exported from the preprocessor and possibly modified by the user. The former is the most common use and the latter is an advanced use. The existing input data file (ht.in in figure 7.5) may have been exported from the HTI preprocessor, or created directly from a template using the information in chapter 5, or it may be a legacy data file created for a previous version of HYDROTHERM. Note, however, that most older HYDROTHERM data files will need some editing to be successfully run with version 3.

By default, the simulation results are saved in the file ht.sim. This file name can be changed from the computer operating system as long as it retains the suffix .sim. The saved simulation results are loaded using the **File** menu and the **Load...** item which causes the Load dialog window to be displayed. Navigate, if necessary, to the directory containing the file with the saved simulation results and double click on the file name. Now the user can run the playback, stop the playback, advance one time step, and reset the playback file to the start of the simulation. The animation speed may be reduced, if it is too fast. The default animation speed is the fastest possible for the computer being used.

7.3.4 The Postprocessor in Stand-Alone Mode

The HTI Postprocessor is launched in stand-alone mode by invoking a shell script named **htpost** that resides in the **bin** directory or by clicking on an icon alias (if available). The postprocessor can perform two functions in the stand-alone mode:

1. Playback the animation of a previously saved simulation (main use), or
2. Runa simulation using data exported from the preprocessor, and possibly modified by the user (advanced use). In order to edit the data file, the user must be familiar with chapter 5 of this manual.

As described earlier, the animation of a simulation in progress is distorted along the simulation time axis. But the animation of a previously saved simulation is nearly undistorted along the simulation time axis because results are saved at nearly equal intervals of simulation time.

When in stand-alone mode, menus and buttons in the postprocessor window allow you to: display simulation results, set the drawing size and scale, control the playback or simulation, save a simulation, zoom in and out of the display, enable or disable graphics buffering, print the display, and export a frame of the display as a bitmap.

For more detailed information on running the postprocessor in stand-alone mode, see the postprocessor on-line help pages by launching the postprocessor in stand-alone mode and then selecting the **`Postprocessor help...`** item from the **`Help`** menu.

7.4 FILE MANAGEMENT

Four types of files are used in HYDROTHERM INTERACTIVE. They are document files, HYDROTHERM 3 simulator input files, HYDROTHERM 3 simulator output files, and saved simulation files. The relations between these files and the preprocessor, postprocessor, and HYDROTHERM 3 are shown in figure 7.5. All files for a given simulation are placed in a user-specified working directory. This is the directory in which the document file is saved.

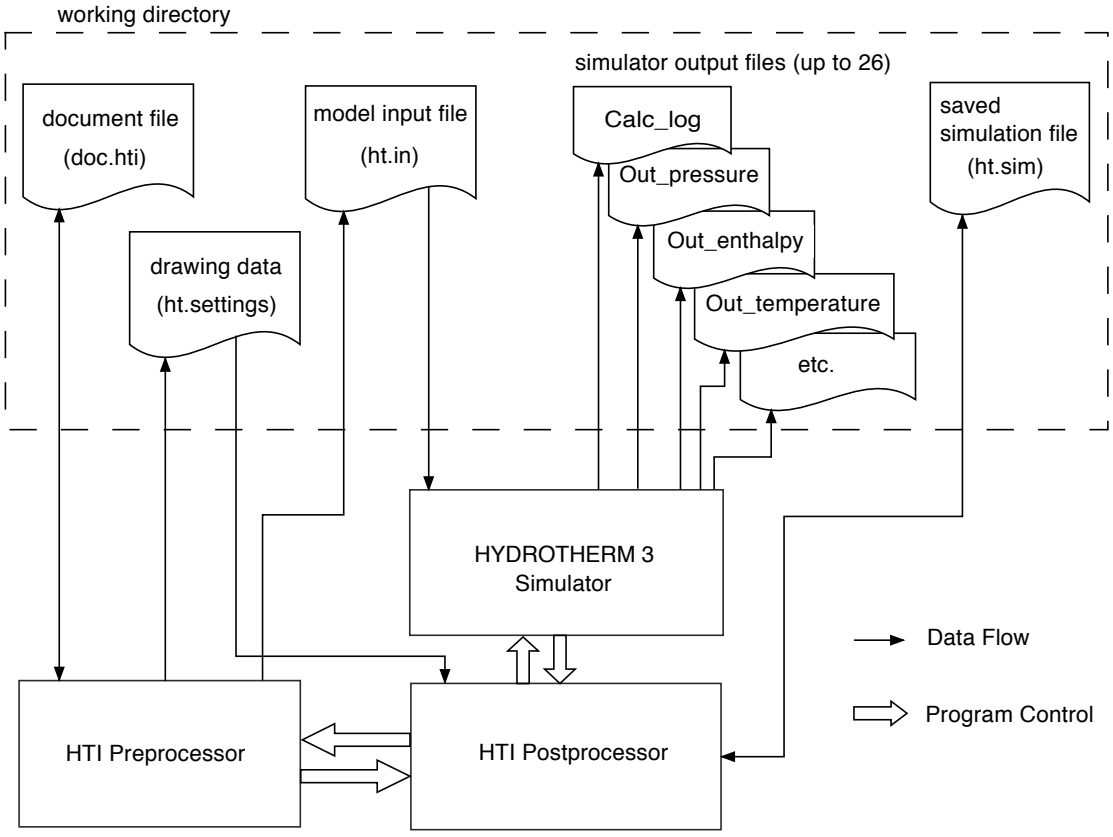


Figure 7.5. HTI files and program components.

For typical usage, the document file is the only file accessed by the user for initial setup of a simulation and subsequent modifications. On termination, the preprocessor writes a file in binary format containing the current state of all the input parameters, drawing area parameters, and postprocessor parameters that have been entered into the HTI preprocessor to define the model for simulation. The default name of this file is `doc.hti`, but it can be renamed using the **`Save As...`** option of the **`File`** menu (section 7.2.1.1). This file is used for continuation of input-data preparation in subsequent work sessions using HTI. A document file contains all of the simulation setup data that was entered into the HTI preprocessor. Document files are saved in binary format and can be opened and read only in the preprocessor. Each document file should be saved in a separate, unique directory or folder.

An additional output file from the preprocessor contains current data relating to the size and scale of the drawing area, color bars, and vector scaling. This file is automatically written by the HTI preprocessor upon invocation of the postprocessor. The HTI postprocessor will rewrite this file if the `Drawing Options` for the display window are changed. It is given the default file name of `ht.settings`. The primary use of this file is to provide drawing area data to the postprocessor when it is invoked in stand-alone mode.

The HTI preprocessor creates an input data file for HYDROTHERM as ASCII text. The content and format of this file are described in chapter 5. After all the data required to define a model have been entered into the preprocessor, the postprocessor is invoked causing the model input file (`ht.in`) to be generated in the working directory. This file is created or rewritten automatically each time the HTI postprocessor is invoked. The default name of this file is `ht.in`. In most cases, the simulator input file can be entirely transparent to the user because this file is automatically generated and requires no user handling.

Sometimes a user may wish to create a model input file using the preprocessor, but without invoking the postprocessor. This would enable the model input file to be modified by editing it directly outside of the preprocessor. Exporting a model data file can be performed in the preprocessor by clicking the **`File`** menu and selecting the **`Export data...`** item, navigating to the desired folder and entering the name of the file ending with the suffix `.in`, and clicking **`Save`**. Exporting enables knowledgeable users to obtain a copy of the model data file for direct editing. Such editing is necessary if the preprocessor is not able to create a model setup in the particular configuration desired. The modified model data file can be run by the postprocessor in stand-alone mode.

A set of up to 26 output files can be produced from running a HYDROTHERM simulation and many of these files can be quite large. The number of output files is determined by the user. These output files, which are written to the working directory, are described in section 6.1. During a simulation, previously generated output files in the working directory *will be overwritten.*

Most of the output files are in list or tabular form to be displayed or printed and are designed to be read by the user. Some files are intended only for post processing by visualization programs and these files can be written in either ASCII text or binary format. The file created to restart a simulation is written in ASCII text format. Data for most variables and parameters are output using the same units of measure as were employed for their respective input. Some data are output in centimeter-gram-second units as used internally by the simulator. The time unit for output is either seconds or years, as chosen by the user.

The simulation results that are saved by the postprocessor are written in binary format. By default, the

simulation results are saved in the file `ht.sim`, which is placed in the working directory. The user may rename this file, but should retain the suffix `.sim`. The saved simulation file can be played back at a later time by running the postprocessor in stand-alone mode.

8. COMPUTER-PROGRAM CODE VERIFICATION AND EXAMPLES

Verification of a computer program is the process of ensuring that the code performs the intended calculations correctly. This is in contrast to computer-model validation, which is the demonstration that a particular model with a particular set of parameters adequately describes a given physical situation. Program verification is accomplished by running various test problems for which an analytical solution is known, or for which numerical results from another verified program are available. Verification of a complex simulator, like HYDROTHERM, is a continuing process, because many combinations of program options need to be tested.

In contrast, an example problem has no independently derived solution with which to compare it to for verification. Three example problems are provided with the release package of HYDROTHERM, Version 2. Additional example problems were run to demonstrate the capabilities of HYDROTHERM, Version 3, and HTI with the graphical user interface. These examples can serve as tutorial problems for setting up various types of simulations.

8.1 SUMMARY OF VERIFICATION TEST PROBLEMS

All of the seven verification test problems described in the documentation for HYDROTHERM, Version 1, were executed correctly by Version 3. The documentation for Version 1 (Hayba and Ingebritsen, 1994) contains descriptions of these verification tests and will not be repeated here.

Four additional test problems have been used for verification of the new capabilities of HYDROTHERM, Version 3. The new capabilities include: unconfined flow with a partially-saturated zone, the precipitation-recharge boundary condition, and the seepage-surface boundary condition. These additional test problems are summarized in this chapter. The first problem involves simulation of transient, partially saturated infiltration in one dimension, and is from the VS2D manual, a variably saturated flow simulator of Lappala and others (1987). The second problem is transient, one-dimensional, horizontal infiltration presented by Huyakorn and others (1984). The results can be compared to the semi-analytical solution originally given by Philip (1955, 1969). The third problem is simulation of a steady-state drainage experiment performed by Duke (1973) involving two-dimensional flow with a seepage surface and also simulated by Lappala and others (1987). The fourth problem is an extension of the third with the addition of thermal-energy transport.

8.1.1 Transient Vertical Infiltration in One Dimension

This problem simulates one-dimensional, transient infiltration in the vertical direction and is based on example number 1 from Lappala and others (1987). This problem verifies the ability of the simulator to represent partially saturated fluid flow in the air-water zone under isothermal conditions. The region is a one-dimensional vertical column 0.6 m long. A source of fluid at constant pressure 1.0077×10^6 dyne/cm^2 is applied to the top boundary. From the original problem specification (Lappala and others, 1987) the potentiometric heads were converted to pressures and the hydraulic conductivity was converted to permeability for HYDROTHERM input. The initial condition was that of uniform pressure at 8.84489×10^5 dyne/cm^2. The Corey functions for saturation and relative permeability were used with the following parameters:

- Bubble point (air-entry) pressure: 1.0077×10^6 dyne/cm^2

- Corey-function pore-size index exponent: 2
- Residual water saturation: 0
- Porosity: 0. 52
- Permeability: 1.6565×10^{-8} cm^2.

Backward-in-time and centered-in-space differencing were used. The region was uniformly discretized with a cell size and consequent node spacing of 1 cm. A uniform time step of 360 s was used and the simulation was run for 10,800 s (3 h). Infiltration occurred from the top down creating a transient moisture profile.

Depth profiles of saturation were computed at four time planes, and time profiles of pressures and saturations were output for six locations down the column. Lappala and others (1987, p. 74) obtained their solution using the VS2D simulator and presented their results as tables of the pressure head and saturation fields. A direct comparison of results can not be made because Lappala and others (1987) used a Corey-function pore-size index exponent of 0.2, and HYDROTHERM is restricted to a fixed exponent of 2. Therefore the HYDROTHERM results were compared to results from Version 3.0 of VS2DH (Healy and Ronan, 1996) obtained using an exponent of 2.

There was very good agreement between the results of HYDROTHERM and VS2DH. At the end of the simulation, the saturation profiles agreed to within 2 significant digits and the transition zones of pressure across the wetting front matched closely in thickness. The HYDROTHERM cumulative global mass balance error was less than 0.0001 percent, and the final infiltration flow rates agreed to within 3 significant digits with the VS2DH results.

Minor differences between the results from the two simulators can be attributed to several causes. They include different sequences of adaptive time-step lengths, different spatial weighting algorithms for relative permeabilities, and different convergence tests for the Newton-Raphson iterations.

8.1.2 Transient Horizontal Infiltration in One Dimension

This problem is a second test involving one-dimensional transient, partially saturated flow under isothermal conditions. It was taken from Huyakorn and others (1984) and a semianalytical solution is available (Philip, 1955, 1969).

The region is a horizontal rectangular prism, forming a column 20 cm long with a cross section 1 cm wide by 4 cm high. The initial condition is uniform pressure within the column. A constant higher pressure is applied to the left end causing water to infiltrate into the column. The outlet at the right end is maintained at the initial-condition pressure.

Linear functions for saturation and relative permeability were used with the following parameters for the system:

- Bubble point pressure: 1.013×10^6 dyne/cm^2
- Residual saturation pressure: 9.1493×10^5 dyne/cm^2

- Residual water saturation: 0.333
- Porosity: 0.45
- Permeability: 1.84×10^{-10} cm^2.

The specified pressure at the right end of the column was 9.31278×10^5 dyne/cm^2 matching the initial-condition pressure. A specified pressure of 1.013×10^6 dyne/cm^2 was imposed at the left end of the column. The difference equations were formed using backwards-in-time and centered-in-space differencing. The column was nonuniformly discretized with a cell-size sequence of $0.1, 0.9, 18 \times 1, 0.9$, and 0.1 cm in the horizontal direction. The thin cells at the ends of the column were used to more closely approximate the finite-element boundary conditions used by Huyakorn and others (1984). Cell sizes were 4 cm in the vertical direction and 1 cm transverse to the column. An initial time step of 43.2 s was used with a maximum time step of 864 s set by the print frequency. The simulation was run for 12,960 s (0.15 d).

Results for the water-saturation profiles, pressure profiles, and inflow rate as a function of time were produced. These results were compared to the graphical results presented by Huyakorn and others (1984) obtained using a finite-element simulator of partially saturated flow. Tabular results were available for the net inflow rate as a function of time. The saturation and pressure profiles (figs. 8.1, 8.2) closely matched the graphical results of Huyakorn and others (1984). The net inflow rate was about 36 percent low at 0.01 d improving to about 2 percent low after 0.06 d. Differences between the results from the two simulators can be attributed to different sequences of adaptive time-step lengths, different spatial weighting algorithms for relative permeabilities, and different convergence tests for the Newton-Raphson iterations.

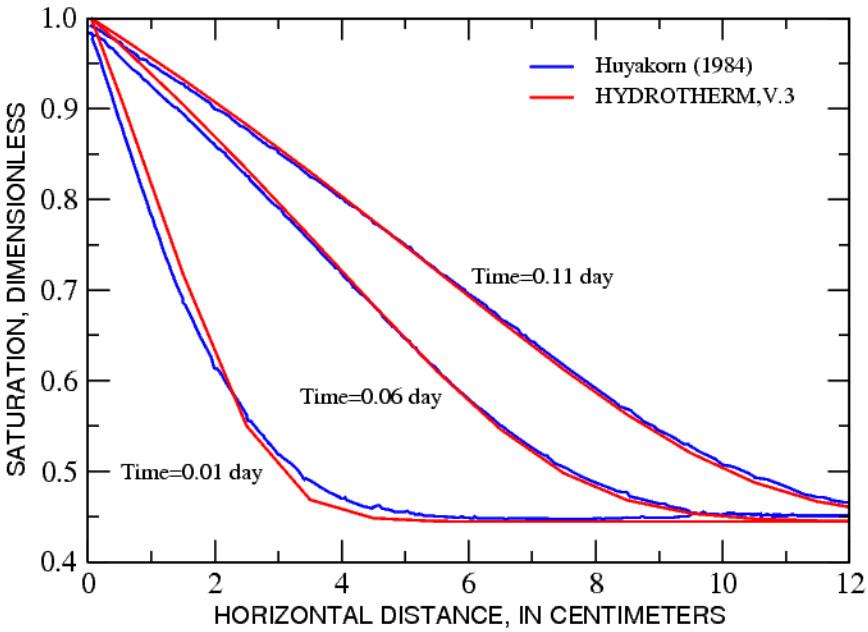


Figure 8.1. Comparison of water-saturation profiles for the horizontal infiltration problem.

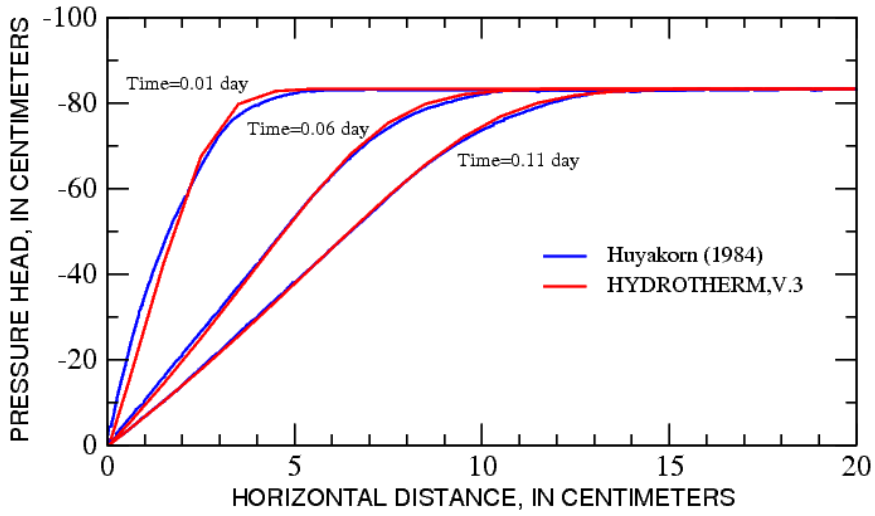


Figure 8.2. Comparison of pressure profiles for the horizontal infiltration problem.

8.1.3 Two-Dimensional, Steady-State Drainage

This test problem is based on a laboratory-flume experiment reported by Duke (1973) involving drainage through a seepage surface combined with recharge from precipitation. The comparison simulation is verification test problem number 5 from the VS2D manual (Lappala and others, 1987, p. 72). The HYDROTHERM seepage-surface boundary condition is verified with this problem.

The rectangular simulation region was one-half of the experimental flume with a length of 6.1 m and a depth of 1.22 m, as shown in figure 8.3. A constant, uniform flux of water was applied to the top boundary. The right side was an impermeable boundary due to symmetry of the experimental apparatus. The bottom boundary was impermeable and the left boundary was a seepage surface. The spatial discretization was uniform in each direction with a cell size of 0.1 m in the horizontal direction and 0.029 m in the vertical direction. Centered-in-space weighting was used for the advective terms in the heat transport equation. The time step was automatically adjusted from a starting value of 1 s and backward-in-time differencing was used.

The Poudre Sand used in the experiment was represented by Corey functions for saturation and relative permeability. The following parameters were chosen to match the values used by Lappala and others (1987) except for the Corey exponent:

- Bubble point pressure: 9.9437×10^5 dyne/cm^2
- Corey-function pore size index exponent: 2
- Residual water saturation: 0
- Porosity: 0.348
- Permeability: 1.0241×10^{-11} m^2.

The precipitation-recharge flux was 1.1979×10^{-6} m/s. Because the Corey exponent parameter in HYDROTHERM is fixed at 2, this test problem was rerun using Version 3.0 of VS2DH (Healy and Ronan, 1996) with a matching Corey exponent in order to generate comparable numerical results for verification.

The simulation duration was 1.296×10^6s (15 d) at which time the system had reached steady-state flow

conditions within 0.003 percent for the mass balance. The water table was taken to be the surface of atmospheric pressure and was located by interpolation of the pressure or pressure-head fields. Figure 8.3 shows the water-table configurations obtained from HYDROTHERM and VS2DH. The water table surfaces coincide. Pressure-head profiles with depth for the cells along the left boundary (surface of seepage) are shown in figure 8.4. The HYDROTHERM results match those of VS2DH to plotting accuracy: 3 digits. Differences elsewhere in the pressure field ranged from 0.3 to 7 percent, while the saturation fields were very similar with the HYDROTHERM results being slightly greater. The causes of the differences were not determined.

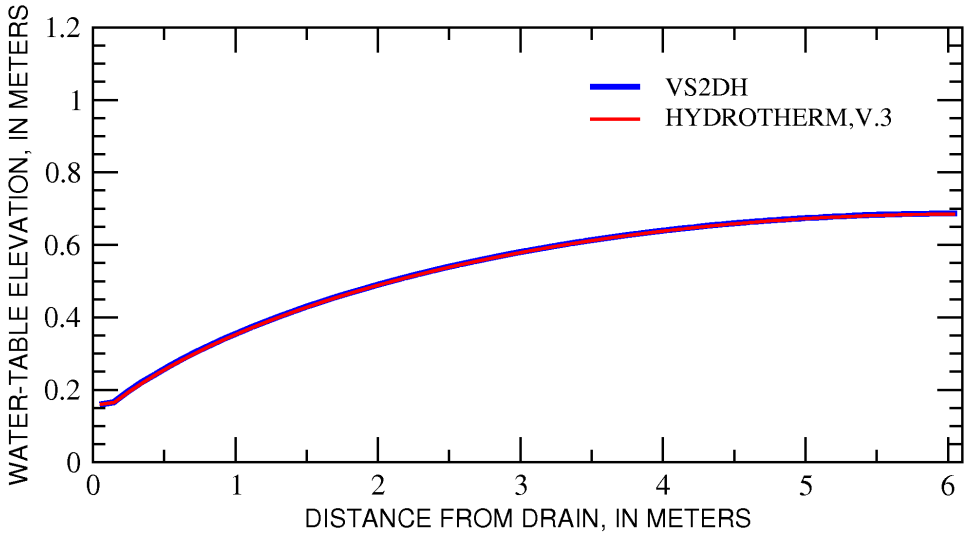


Figure 8.3. Comparison of water-table profiles for the two-dimensional drainage problem of Duke (1973).

In order to obtain these matching results, a temporary modification to HYDROTHERM had to be made. Vertical flow through the actively seeping cells had to be suppressed to make the HYDROTHERM simulator handle the seepage-surface boundary condition in the same fashion as the VS2DH simulator. Otherwise, allowing vertical flow in HYDROTHERM caused the length of the steady-state seeping face to be about 1/6 the length computed by VS2DH. Reducing the width of the seepage-boundary cells for HYDROTHERM to 5 percent of the uniform cell width caused the length of the seepage face to increase to about 5/7 of the length computed by VS2DH. These results illustrate the type of errors incurred using relatively thick cells to discretize a seepage-boundary condition that is actually mathematically specified along a set of cell faces.

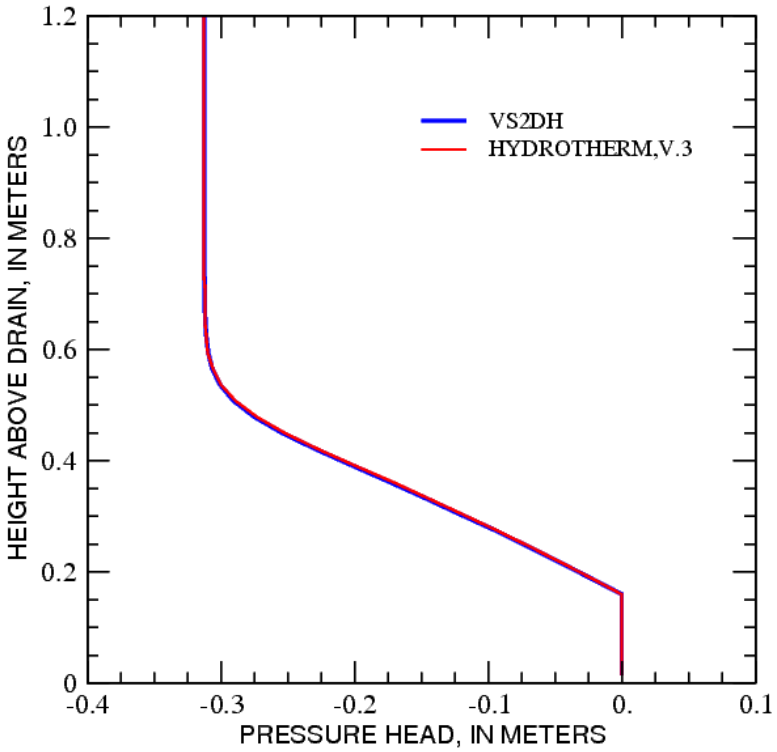


Figure 8.4. Comparison of pressure-head profiles along the seepage surface for the two-dimensional drainage problem of Duke (1973).

8.1.4 Two-Dimensional, Steady-state Drainage with Thermal-Energy Transport

The final test problem of this set is an extension of the Duke (1973) drainage problem presented above. Thermal-energy transport was added by specifying that the precipitation recharge enters at a temperature greater than the initial uniform temperature and also including a heat flux through the bottom boundary (fig. 8.5).

The initial condition temperature was uniform at 4°C. A constant, uniform flux of freshwater was applied to the top boundary of the region at a rate of 1.1979×10^{-4} cm/s with an associated temperature of 5°C. The heat flux along the bottom boundary was specified as 1.1954×10^{4} mW/m^2. Although this large flux was not realistic for a geothermal environment, it was chosen to create a substantial thermal profile at steady state. The left boundary was a seepage surface as in the previous problem. The thermal parameters used for the simulation are the following:

- Thermal conductivity of the porous medium and fluid: 0.1157 W/m-°C
- Heat capacity of the porous matrix: 890 J/kg-°C.

Backwards-in-time differencing and upstream-in-space differencing were used. Simulation duration was 2.592×10^{6}s (30 d) at which time the system had reached steady-state conditions within 0.003 percent for flow and within 0.2 percent for heat transport. Numerical results for verification came from a simulation using Version 3.0 of VS2DH, a simulator of variably saturated flow and heat transport (Healy and Ronan, 1996).

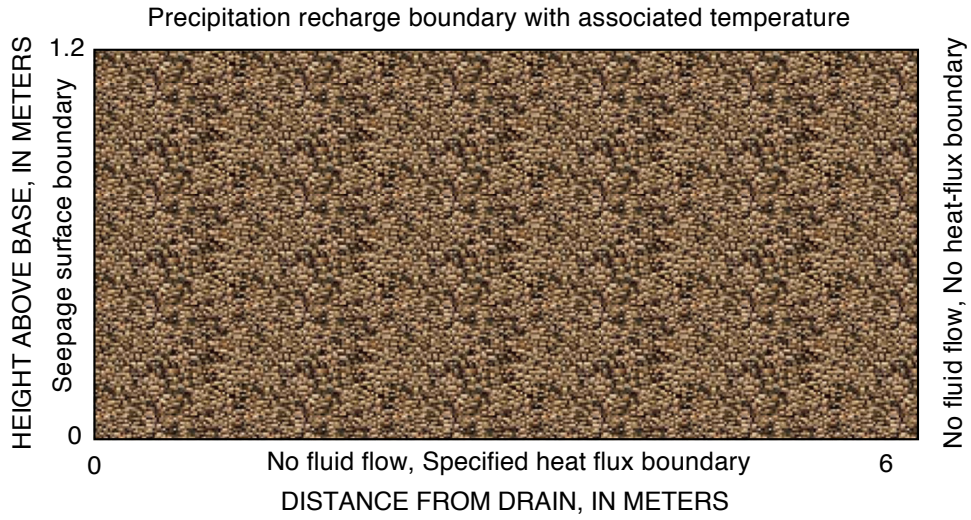


Figure 8.5. Region and boundary conditions for the Duke (1973) drainage problem, with addition of heat transport.

Figure 8.6 shows the temperature contours from HYDROTHERM and from VS2DH. The temperature contour results of HYDROTHERM are similar in pattern to those of VS2DH, but the contours exhibit some offset which increases as the bottom boundary is approached. The HYDROTHERM saturations near the top of the region are about 10 percent lower than those of VS2DH, and the VS2DH temperatures are up to 1.4°C greater along the bottom boundary. These differences can be attributed to the fact that VS2DH does not include buoyancy driving forces resulting from thermal gradients. Furthermore, the VS2DH simulator calculates the effective thermal conductivity of the porous matrix and water by accounting for the amount of saturation, while the HYDROTHERM simulator uses an effective value that is independent of saturation.

8.2 ADDITIONAL EXAMPLE PROBLEMS

Three additional example problems were provided in the distribution package for Version 2, but not for Version 1. HYDROTHERM, Version 3 produced results matching Version 2. These three problems are presented as examples rather than verification tests because no independent solutions are available for comparison. All of these examples have model regions that are two-dimensional cross-sections. These additional examples include:

- A geyser in a pre-eruption condition modeled by a high-permeability vertical conduit in a 100- by 200-m region,
- A free-convection problem of water near the critical point in a 10-m square region with specified pressure and temperature along the upper and lower boundaries, and
- A cooling pluton with a magma source in a 10- by 4-km region (Hayba and Ingebritsen, 1997).

Data files and selected output files for these examples are included in the HYDROTHERM program distribution package (appendix 1).

In addition to these three examples, seven more examples were used for testing the operation of the

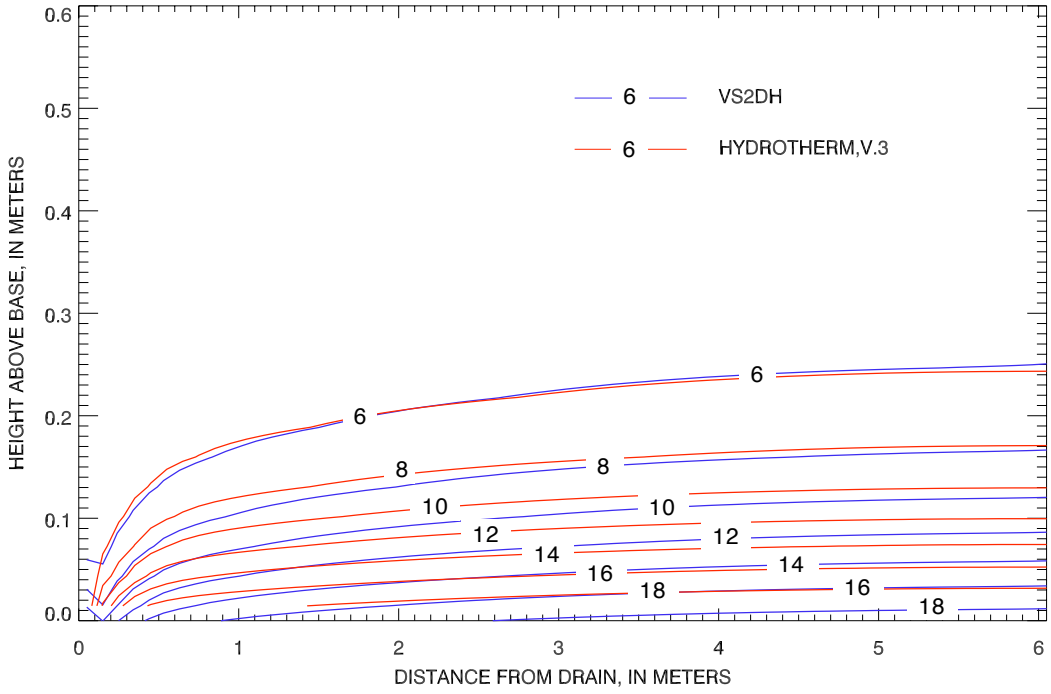


Figure 8.6. Contours of the temperature field, in degrees Celsius, computed by HYDROTHERM and VS2DH for the Duke (1973) drainage problem with addition of heat transport.

HYDROTHERM INTERACTIVE program. They include:

- Single-phase, convective heat transport in a two-dimensional (40- by 5-km) slice of a basin (Smith and Chapman, 1983, p. 598),
- Single-phase free convection induced by a perturbation within a field of uniform temperature gradient,
- Horizontal saturated flow with heat advection and conduction in one dimension with associated temperature or specified temperature boundary condition,
- A geyser in a pre-eruption condition represented by a one-dimensional model of only the conduit,
- A heat pipe model in one dimension,
- A two-dimensional model of the geothermal system at Lassen volcano in California, and
- Advection and free convection in a long basin with high permeability.

Data files and selected output files for these examples are included in the HYDROTHERM INTERACTIVE program distribution package (appendix 1).

8.3 A TUTORIAL EXAMPLE PROBLEM

An example problem for tutorial purposes is taken from the paper on water-table position within volcanic edifices by Hurwitz and others (2003) and identified there as run number C-2. This example is provided to acquaint new users of HYDROTHERM with the data file for a field-scale simulation. Discussion of the motivations for selection of the values used for the key parameters appears in Hurwitz and others (2003).

The model of the volcanic edifice has the following configuration with boundary and initial conditions. The simulation region is a radial cross section through a cylindrical volume of 8.3-km radius and 3.5-km thickness overlain by a cone of 3.0-km base radius and 1.5-km height (fig. 8.7). The axis of the cylinder and cone lies along the left side of the cross section. The cone has a surface slope of 27 degrees representing the edifice. The cylindrical region is divided into two stratigraphic units: an upper unit of 1.5-km thickness, and a lower unit of 2-km thickness. The permeability of the lower unit is two orders of magnitude smaller than that of the upper unit. A cylindrical coordinate system was used with the spatial domain discretized into 105 columns and 43 rows of variable-size cells. A linear relative-permeability function and a simplified, linear saturation function were chosen because of lack of data for fractured volcanic rocks at high temperature. An artificially high value for capillary pressure at the residual water saturation was used to facilitate precipitation recharge and accelerate the approach of the water table to steady state.

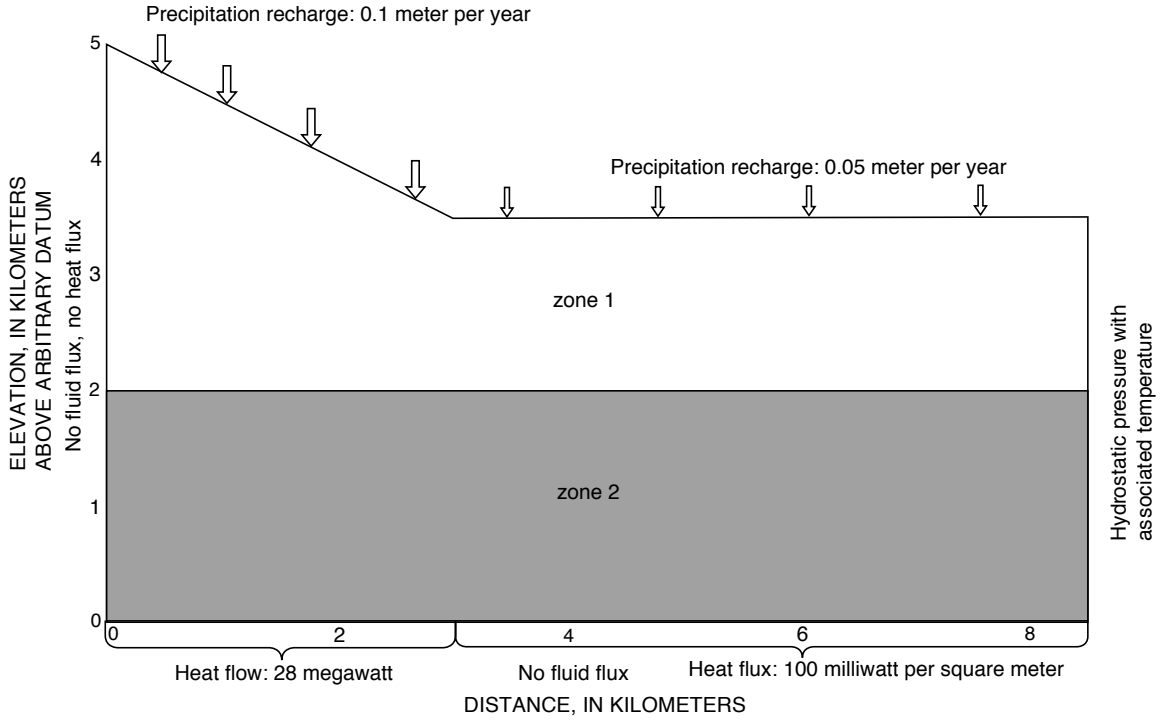


Figure 8.7. Simulation region and boundary conditions for the volcanic edifice example of Hurwitz and others (2003).

The left-side boundary was impermeable and insulating by symmetry. The horizontal land-surface boundary had a net precipitation flux of 0.05 m/yr, while the cone surface had a net precipitation flux of 0.1 m/yr. Precipitation recharge was taken to be at a uniform temperature of 5°C, neglecting any thermal lapse rate effect. Variation of atmospheric pressure with elevation also was neglected. The right-side boundary was maintained at the initial hydrostatic fluid pressure distribution and the initial associated temperature profile with depth. Any inflowing fluid across this boundary entered the region at the associated temperature. The bottom boundary was impermeable and had a specified constant heat flux. This heat flux was divided into two

zones, an inner zone, underneath the cone, of radius 2.1 km and an outer zone from radius 2.1 km to 8.3 km. A 28 MW heat source was applied to the inner zone, while the outer zone received a geothermal heat flux of 100 mW/m^2.

Initial conditions for pressure were 1 atm between the elevations of 3,500 m to 5,000 m, from the base to the top of the cone, and a hydrostatic pressure distribution from 0 m to 3,500 m, in the underlying cylindrical region. Initial temperature distribution for the entire domain was linear with depth below the land surface, starting at 5 °C and increasing at 50 °C/km. A quasi steady-state solution was obtained by simulating for a long period of 50,000 years.

Run C-2, from the paper of Hurwitz and others (2003), was done to evaluate the effects of doubling the heat input under the cone relative to the base case with isotropic permeability conditions. The results appear in table 3 and figure 9 of Hurwitz and others (2003). The data-input file for simulation run C-2 is presented in table 8.1 of this report. Reference should be made to chapter 5 to obtain the definitions of the various parameters. Annotations are included in bold within the data file shown in table 8.1 and follow the data descriptor lines but precede the actual data lines to which they refer. These annotations are *not* part of the input file, but they could be included as comments by preceding each line of them with the # character.

Table 8.1. Data file for example problem, run C-2, with annotations

```
# Hydrotherm Data-Input Form
# Version 3.1
#      Notes:
#      Syntax of input line is denoted by # ..
#      A suffix letter indicates an exclusive record choice must be made.
#             i.e. A or B
#      (O) - Optional data with conditions for requirement
#      [a|b] - Indicates that either item a or item b must be entered
#      {nnn} - Indicates that the default number, nnn, is used if a zero
#              is entered for that variable
#      [T/F] - Indicates a logical variable
#      [I] - Indicates an integer variable
#-------------------------------------------------------------------------
#-------------------------------------------------------------------------
#   Start of the data file
# TITLE
# ..  TITLE: title line 1
```

Extra comments can be inserted using lines beginning with #.

```
# ..  TITLE: VWT-C-2
TITLE    GRID FOR water table EXPERIMENTS Test a fine grid; cylindrical coordinates
# ..  VER3:  title line 2
VER3   Hydrostatic, conductive initial conditions
```

Extra title description lines can be added using lines beginning with #. They will not appear in the output files.

```
#linear saturation curve; specified pressure at valley floor
# DIMENSIONS
# ..  nx[I],ny[I],nz[I],tmstr,iyear[I]
```

A comma and/or one or more spaces may be used to separate data items.

```
105,     1,    43,       0.0      2
```

Table 8.1. Data file for example problem, run C-2, with annotations—Continued

```
# TIME STEP
# ..A  tstepmx[I],tmincr{1.3},%PchgTS{10.},%HchgTS{5.},Wsatn{0.03},relax{1.},
# ..       deltmin,n_tmcut{10}; confined flow
# ..B  tstepmx[I],tmincr{1.3},PchgTS{1.e5},%HchgTS{5.},Wsatn{0.03},relax{1.},
# ..       deltmin,n_tmcut{10}; unconfined flow
   100000,           1.5,          9.8e6,     5.,         0.1,   1.0,  1.0D-9,    15
# NEWTON-RAPHSON ITERATION
# ..A  itermx{10},resmas{1.e-12},reseng{1.e-2},nphasechg{10},%PchgNR{0.1},
# ..        %HchgNR{0.1}; confined flow
# ..B  itermx{10},resmas{1.e-12},reseng{1.e-2},nphasechg{10},PchgNR{1.e5},
# ..        %HchgNR{0.1}; unconfined flow
   20,           1.e-11,        1.e-2,           180            2.e5       0.5
# LINEAR EQUATION SOLVER
# .. slmeth[I]
 1
```

The SSOR solver becomes a direct band solver for a 2-d slice region. The following parameter record is required, but the data are not used.

```
#    SSOR SOLVER CONTROL
# ..A  ssormx[I]{10},tol{0.05},ssorw0{1.0}; (O) - slmeth = 1
    1,     .01,    1.0
#    GMRES SOLVER CONTROL
# ..B  m_save_dir{5},stop_tol_gmres{1.e-10},maxit_gmres{100},ilu_method{2},lev_fill{5},
# ..       drop_tol{0.001}; (O) - slmeth = 2
# WEIGHTING AND AVERAGING
# ..  ioptupst[T/F],potdif{0.0002},theta{1.0}
    T,           0.00002,    1
# RELATIVE PERMEABILITY
# ..  kodrp,wsatn{0.3},ssatn{0.0}
    0,           0.3,   0.0
# ROCK PROPERTIES
# ..  heatcpty{1.e7},rxden,rxcmprss{0.},grav{981.},initphi[T/F]
```

All units in this record are cgs.

```
    1.0e7,  2.50,    0.0,   981.0,   F
# ..  initphi_filename{IC_pressporo.xxx}; (O) - initphi = T
# RADIAL COORDINATES
# ..  irad[T/F],wellrad(cm),radiusmax(cm)
```

All units in this record are cgs.

```
    t,           0,      830000.
# PRINT/PLOT CONTROL
# ..  wide[I],long[I],v-avg[I]
  1   2   0
# PRINT 1
```

Print intervals are specified as an interval of simulation time (positive number) or a number of time steps (negative number).

```
# ..  pres_pr_intrv,enth_pr_intrv,temp_pr_intrv,satn_pr_intrv,iprsat[I]
  10000.   10000.   10000.   200.   1
# PRINT 2
# ..  dens_pr_intrv,iprden[I],vis_pr_intrv,iprvis[I],potential_pr_intrv,iprpot[I]
  0.       0      0.     0       0.         0
# PRINT 3
# ..  velocity_pr_intrv,iprvel[I],bcflow_pr_intrv,iprbcflow[I],source_pr_intrv,iprsource[I]
    20000.        01        0.       0          0.      0
```

Table 8.1. Data file for example problem, run C-2, with annotations—Continued

```
# PRINT 4
# ..  pm_properties_pr_intrv,iprmprop[I],poros_pr_intrv,permeability_pr_intrv
    0.           0          0.          0.
# PRINT 5
# ..  balance_pr_intrv,dimno_pr_intrv,iprdimno[I],residuals_pr_intrv,dump_pr_intrv
     -100.            -100.             11            -100.        0.
# PRINT 6
# ..  plotscalar_pr_intrv,plotvector_pr_intrv,plotfile_type[I],time_series_pr_intrv
```

The output intervals for contour-plot and vector-plot data are specified as 50000 years. Specifying a value greater than the time of simulation will produce output only at the end of the simulation.

```
    50000.     50000.        6          0.
# ..  num_locations,(i(n),j(n),k(n),n=1 to num_locations)(all [I]);
#          (O) - time_series_pr_intrv > 0
```

A slice number or other identification can be entered on the SLICE header line.

```
# SLICE number 1
# ..  start_location: [TOP|BOTTOM]
TOP
# ..  index_of_rock_type(i,k),i=1 to nx, k=1 to nz (or k=nz to 1)[I]
```

To define the mesh rock types and boundary condition types, one data record line is used for each row of nodes in the mesh with one data item for each node (NX items per record). Repeat factors may be used. A total of NZ records are used.

```
2*1,-31,102*0
5*1,-31,99*0
8*1,-31,96*0
11*1,-31,93*0
14*1,-31,90*0
17*1,-31,87*0
20*1,-31,84*0
23*1,-31,81*0
26*1,-31,78*0
29*1,-31,75*0
32*1,-31,72*0
35*1,-31,69*0
38*1,-31,66*0
41*1,-31,63*0
44*1,-31,60*0
47*1,-31,57*0
50*1,-31,54*0
53*1,-31,51*0
56*1,-31,48*0
59*1,-31,45*0
62*1,-31,42*0
65*1,-31,39*0
68*1,-31,36*0
71*1,-31,33*0
74*1,-31,30*0
77*1,-31,27*0
80*1,-31,24*0
83*1,-31,21*0
86*1,-31,18*0
89*1,-31,15*0
89*1,16*-11
```

Table 8.1. Data file for example problem, run C-2, with annotations—Continued

```
104*1,-11
104*1,-11
104*1,-11
104*1,-11
104*1,-11
104*1,-11
104*1,-11
104*1,-11
104*2,-12
104*2,-12
104*2,-12
104*2,-12
```

Two rock types or media types have been defined for this simulation by the SLICE data group.

```
# UNCONFINED
# ..  unconfined[T/F],patm
     t            1.013e6
# SATURATION FUNCTION
#     LINEAR
# ..A pwb,pwr; (O) -  kodrp = 0 and unconfined
```

Superseded data records may be kept for future use by commenting them out with a #.

```
#  1.013e6     -3.0e7
  1.0077e6    -5.0e7
#     COREY
# ..B pwb,lambda; (O) - kodrp = 1 and unconfined
#     COOLEY
# ..C pwb,bb,cc; (O) - kodrp = 3 and unconfined
# -----------------------------------
# Start of Keyword Data Block section
# PARAMETER
```

Information on format for each data block below can be found in chapter 5.

```
# Sample format for data block
# ..  array_identifier,(units)
# ..  format_specifier
# ..  data record set
# ........
```

Data blocks for grid spacing must appear first in the parameter keyword block section.

```
XSPACING (m)
FREE
2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,
2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,
2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,2*50,5,
4*100,3*250,8*500
ZSPACING (m)
FREE
TOP
30*50,1*10,1*90,4*100,2*250,5*500
POROSITY (-)
ROCK
```

Two rock types or media types have been defined for this simulation in the SLICE data group.

```
1,0.10
2,0.01
```

Table 8.1. Data file for example problem, run C-2, with annotations—Continued

```
XPERMEABILITY (m^2)
ROCK
1,1.0e-15
2,1.0e-17
ZPERMEABILITY (m^2)
ROCK
1,1.0e-15
2,1.0e-17
THERMAL CONDUCTIVITY (W/m K)
ROCK
1, 2.0
2, 2.5
CONDUCTION Heat Input along Base (mW/m^2)
FREE
```
The inner zone is the first group of 60 cells and the outer zone is the second group of 45 cells. The 28 MW heat source is converted into 2000 mW/m^2 heat flux and applied to the inner zone of cells.
```
60*2000.  45*100.
PRESSURE (Pa)
CALCULATE
```
This data block establishes the initial pressure distribution which is used also for each of the specified pressure boundary condition nodes.
```
11, 0., 0.
3,  0. 1.0130E5, 1505. 1.0130E5, 4750. 3.0965E7
TEMPERATURE (C)
```
An abbreviated form of the keyword can be used.
```
CALC
```
This data block establishes the initial temperature distribution which is used also for each of the associated temperature boundary condition nodes. Advected fluid that enters the simulation region has the associated temperature specified. Fluid that leaves the region has the ambient temperature of the respective boundary cell.
```
2,  5., 50.
PRECIP (m/yr)
FREE
```
Although the 89 cells along the edifice slope have precipitation flux specified, no precipitation flux will be applied to the seepage surface cells which have been specified at locations along this sloping land surface in the SLICE data block.
```
89*0.1 16*0.05
TFLUX  (C)
CONSTANT
5.
# End of Keyword Data Block section
# ---------------------------------
# Transient data
# TIME PERIOD
# .. tchg,delt,nsrce[I],lchgpr[T/F],nparms[I]
```
Units of tchg and delt are set to years by iyear in the DIMENSIONS data record.
```
    50000.,    1.e-5,    0,     f,     0
#    5.e-5             1.e-5,         0,          f,      0
# Start of Source Keyword Data Block section
# SOURCE/SINK number; (O) - nsrce > 0
# ..  WELL,(units_flow),(units_enth)
# ..  i[I],j[I],well_flow,well_enth,n-openint[I]
# ..  NODE,(units_flow),(units_enth)
# ..  i[I],j[I],k[I],node_flow,node_enth
# End of Source Keyword Data Block section
```

Table 8.1. Data file for example problem, run C-2, with annotations—Continued

```
# ---------------------------------
# ..  PRINT/PLOT data records, PRINT n, as above; (O) - lchgpr = true
# ..  PARAMETER keyword data block records, as above; (O) - nparms > 0
# ---------------------------------
# Insert additional groups of transient data as necessary
```
There is only one set of transient boundary condition and parameter data for this example because all boundary conditions and parameters are constant in time.
```
# ---------------------------------
# TIME PERIOD: End of simulation record
# ..  tchg
-1 /
#   ..End of the data file
#-------------------------------------------------------------------------------
#-------------------------------------------------------------------------------
```

Several alternative forms for some of the keyword data blocks are shown in chapter 5 under templates 1−28. They provide for more complex spatial distributions of porous-media properties and dependent variables that can be specified.

There are 11 output files produced by this tutorial example. Because of the large sizes of these files, they are not included in printed form in this manual. However, all output files from this example are included in the program distribution package available as described in appendix 1. Reviewing these input and output files and those of the verification and other example problems is a good way to gain familiarity with preparing input and running the HYDROTHERM simulator.

9. REFERENCES CITED

Abramov, O., and Kring, D.A., 2004, Numerical modeling of an impact-induced hydrothermal system at the Sudbury crater: Journal of Geophysical Research, v. 109. doi:10.1029/2003JE002213

Abramov, O., and Kring, D.A., 2005, Impact-induced hydrothermal activity on early Mars: Journal of Geophysical Research, v. 110. doi:10.1029/2005JE002453

Abramov, O., and Kring, D.A., 2007, Numerical modeling of impact-induced hydrothermal activity at the Chicxulub crater: Meteoritics & Planetary Science, v. 42, p. 93–122.

American National Standards Institute, 1989, American national standard programming language C (ANSI X3.159-1989): New York.

American National Standards Institute, 1992, American national standard programming language Fortran 90 (ANSI X3.198-1992): New York.

Anderson, D.A., Tannehill, J.C., and Pletcher, R.H., 1984, Computational fluid mechanics and heat transfer: Bristol, Penn., Tayor & Francis, 599 p.

Aziz, Khalid, and Settari, Antonin, 1979, Petroleum reservoir simulation: Barking, Essex, England, Applied Science, 476 p.

Barrett, Richard, Berry, Michael, Chan, T.F., Demmel, James, Donato, June, Dongarra, Jack, Eijkhout, Victor, Pozo, Roldan, Romine, Charles, and van der Vorst, Henk, 1994, Templates for the solution of linear systems—Building blocks for iterative methods: Philadelphia, Penn., Society for Industrial and Applied Mathematics, 112 p.

Bear, Jacob, 1972, Dynamics of fluids in porous media: New York, American Elsevier, 764 p.

Benzi, Michele, 2002, Preconditioning techniques for large linear systems—A survey: Journal of Computational Physics, v. 182, p. 418–477.

Brooks, R.H., and Corey, A.T., 1966, Properties of porous media affecting fluid flow: Proceedings of the American Society of Civil Engineers, Journal of the Irrigation and Drainage Division, v. IR 2, no. 4855, p. 61–88.

Cooley, R.L., 1983, Some new procedures for numerical solution of variably saturated flow problems: Water Resources Research, v. 19, no. 5, p. 1,271–1,285.

Corey, A.T., 1957, Measurement of water and air permeability in unsaturated soil: Soil Science Society Proceedings, v. 21, p. 7–10.

de Marsily, Ghislain, 1986, Quantitative hydrogeology: San Diego, Calif., Academic Press, 440 p.

Duke, H.R., 1973, Drainage design based upon aeration: Colorado State University Hydrology Paper 61, Fort Collins, Colorado State University, 59 p.

Faust, C.R., and Mercer, J.W., 1977, Finite-difference model of two-dimensional, single- and two-phase heat transport in a porous medium-version I: U.S. Geological Survey Open-File Report 77–234, 84 p.

Faust, C.R., and Mercer, J.W., 1979a, Geothermal reservoir simulation 1. Mathematical models for liquid- and vapor-dominated hydrothermal systems: Water Resources Research, v. 15, no. 1, p. 23–30.

Faust, C.R., and Mercer, J.W., 1979b, Geothermal reservoir simulation 2. Numerical solution techniques for liquid- and vapor-dominated hydrothermal systems: Water Resources Research, v. 15, no. 1, p. 31–46.

Faust, C.R., and Mercer, J.W., 1982, Finite-difference model of three-dimensional, single- and two-phase heat transport in a porous medium, Scepter documentation and user's manual: Reston, Va., Geotrans, Inc., 73 p.

Fletcher, C.A.J., 1991, Computational techniques for fluid dynamics, v.1, (2d ed.): Berlin, Springer-Verlag, 401 p.

Forster, C.B., and Smith, Leslie, 1988, Groundwater flow systems in mountainous terrain; 1, Numerical modeling technique: Water Resources Research, v. 24, no. 7, p. 999–1010.

Forsyth, P.A., Wu, Y.S., and Pruess, Karsten, 1995, Robust numerical methods for saturated-unsaturated flow with dry initial conditions in heterogeneous media: Advances in Water Resources, v. 18, p. 25–38.

Fujimitsu, Y., Ehara, S., and Oki, R., 2006, Geothermal fluid flow model in Shimabara Peninsula: Journal of the Geothermal Resources Society of Japan, v. 28, p. 373–382. [in Japanese]

Gosling, James, Joy, Bill, Steele, Guy, and Bracha, Gilad, 2005, The Java language specification, (3d ed.): New York, Prentice-Hall, 688 p.

Greenbaum, Anne, 1997, Iterative methods for solving linear systems: Philadelphia, Penn., Society for Industrial and Applied Mathematics, 220 p.

Haar, Lester, Gallagher, J.S., and Kell, G.S., 1984, NBS/NRC Steam tables—Thermodynamic and transport properties and computer programs for vapor and liquid states of water in SI units: Washington, D.C., Hemisphere Publishing Corporation, 320 p.

Harmako, Y., Fujimitsu, Y., and Ehara, S., 2007, Shallow ground temperature anomaly and thermal structure of Merapi volcano, central Java, Indonesia: Journal of the Geothermal Resources Society of Japan, v. 29, p. 25–37.

Harrison, K.P., and Grimm, R.E., 2002, Controls on Martian hydrothermal systems— Application to valley network and magnetic anomaly formation: Journal of Geophysical Research, v. 107, no. E5. doi:10.1029/2001JE001616

Hayba, D.O., and Ingebritsen, S.E., 1994, The computer model HYDROTHERM, A three-dimensional finite-difference model to simulate ground-water flow and heat transport in the temperature range of 0 to 1,200 °C: U.S. Geological Survey Water-Resources Investigations Report 94–4045, 85 p.

Hayba, D.O., and Ingebritsen, S.E., 1997, Multiphase groundwater flow near cooling plutons: Journal of Geophysical Research, v. 102, no. B6, p. 12,235–12,252.

Healy, R.W., and Ronan, A.D., 1996, Documentation of computer program VS2DH for simulation of energy transport in variably saturated porous media--Modification of the U.S. Geological Survey's computer program VS2DT: U.S. Geological Survey Water-Resources Investigations Report 96–4230, 36 p.

Hogeweg, N., Keith, T.E.C., Colvard, E.M., and Ingebritsen, S.E., 2005, Ongoing hydrothermal heat loss from the Valley of 10,000 Smokes, Alaska: Journal of Volcanology and Geothermal Research, v. 143, p. 279–291.

Hubbert, M.K., 1956, Darcy's law and the field equations of the flow of underground fluids: American Institute of Mining, Metalurgical, and Petroleum Engineers Petroleum Transactions, v. 207, p. 222–239.

Hurwitz, Shaul, and Ingebritsen, S.E., 2003, Good news or bad?—New study of temperature inversions in NSF deep geothermal well at Kilauea volcano: Geothermal Resources Council Bulletin, v. 32, p. 111–115.

Hurwitz, Shaul, Ingebritsen, S.E., and Sorey, M.L., 2002, Episodic thermal perturbations associated with groundwater flow— An example from Kilauea Volcano, Hawaii: Journal of Geophysical Research, v. 107, no. B11. doi:10.1029/2001JB001654

Hurwitz, Shaul, Kipp, K.L., Ingebritsen, S.E., and Reid, M.E., 2003, Groundwater flow, heat transport, and water-table position within volcanic edifices— Implications for volcanic processes in the Cascade Range: Journal of Geophysical Research, v. 108, no. B12. doi:10.1029/2003JB002565

Huyakorn, P.S., and Pinder, G.F., 1983, Computational methods in subsurface flow: New York, Academic Press, 473 p.

Huyakorn, P.S., Thomas, S.D., and Thompson, B.M., 1984, Techniques for making finite elements competitive in modeling flow in variably saturated porous media: Water Resources Research, v. 20, no. 8, p. 1099–1115.

Ingebritsen, S.E., and Hayba, D.O., 1994, Fluid flow and heat transport near the critical point of H20: Geophysical Research Letters, v. 21, p. 2,199–2,203.

Ingebritsen, S.E., and Rojstaczer, S.A., 1993, Controls on geyser periodicity: Science, v. 262, p. 889–892.

Ingebritsen, S.E., and Rojstaczer, S.A., 1996, Geyser periodicity and the response of geysers to deformation: Journal of Geophysical Research, v. 101, p. 21,891–21,905.

Ingebritsen, S.E., and Sanford, W.E., 1998, Groundwater in geologic processes: Cambridge, United Kingdom, Cambridge University Press, 341 p.

Ingebritsen, S.E., and Sorey, M.L., 1985, A quantitative analysis of the Lassen hydrothermal system, north-central California: Water Resources Research, v. 21, p. 853–868.

Ingebritsen, S.E., and Sorey, M.L., 1988, Vapor-dominated zones within hydrothermal systems: Evolution and natural state: Journal of Geophysical Research, v. 93, p. 13,635–13,655.

Jennings, Alan, 1977, Matrix computation for engineers and scientists: New York, John Wiley and sons, 330 p.

Jupp, T., and Schultz, A., 2000, A thermodynamic explanation for black smoker temperatures: Nature, v. 403, p. 880–883.

Kelley, C.T., 1995, Iterative methods for linear and nonlinear equations: Philadelphia, Penn., Society for Industrial and Applied Mathematics, 166 p.

Kubota, K., Nishijima, J., Fujimitsu, Y., and Ehara, S., 2006, Geothermal fluid flow derived from microseismic observation—A case study of Kuju volcanic field, central Kyushu, Japan: Butsuri-Tansa (Geophysical Exploration), v. 59, p. 181–192. [in Japanese]

Lappala, E.G., Healy, R.W., and Weeks, E.P., 1987, Documentation of computer program VS2D to solve the equations of fluid flow in variably saturated porous media: U.S. Geological Survey Water-Resources Investigations Report 83–4099, 184 p.

Manning, C.E., and Ingebritsen, S.E., 1999, Permeability of the continental crust: Implications of geothermal data and metamorphic systems: Reviews of Geophysics, v. 37, p. 127–150.

Meijerink, J.A., and van der Vorst, H.A., 1977, An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix: Mathematics of Computation, v. 31, no. 137, p. 148–162.

Nield, D.A., and Bejan, A., 1992, Convection in porous media: New York, Springer-Verlag, 408 p.

Okubo, A., Kanda, W., and Ishihara, K., 2006, Numerical simulation of volcanomagnetic effects due to hydrothermal activity: Kyoto, Japan, Kyoto University, Annuals of Disaster Prevention Research Institute, no. 49C, p. 211–217.

Overton, M.L., 2001, Numerical computing with IEEE floating point arithmetic: Philadelphia, Penn., Society for Industrial and Applied Mathematics, 104 p.

Philip, J.R., 1955, Numerical solution of equations of the diffusion type with diffusivity concentration-dependent: Transactions of the Faraday Society, v. 51, p. 885–892.

Philip, J.R., 1969, Theory of infiltration: Advances in Hydroscience, v. 5, p. 216–291.

Pinder, G.F., and Gray, W.G., 1977, Finite element simulation in surface and subsurface hydrology: New York, Academic Press, 295 p.

Polyansky, O.P., Riverdatto, V.V., Khomenko, A.V., and Kuznetsova, E.N., 2003, Modeling of fluid flow and heat transfer induced by basaltic near-surface magmatism in the Lena-Tunguska petroleum basin (Eastern Siberia, Russia): Journal of Geochemical Exploration, v. 78–79, p. 687–692.

Polyansky, O.P., Riverdatto, V.V., and Sverdlova, V.G., 2002, Convection of two-phase fluid in a layered porous medium driven by the heat of magmatic dikes and sills: Geochemistry International, v. 40, no. Suppl.1, p. S69–S81.

Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., 1989, Numerical recipes—The art of scientific computing (FORTRAN version): Cambridge, U.K., Cambridge University Press, 702 p.

Rathbun, J.A., and Squyres, S.W., 2002, Hydrothermal systems associated with Martian impact craters: Icarus, v. 157, p. 362–372.

Redwine, Cooper, 1995, Upgrading to Fortran 90: New York, Springer-Verlag, 501 p.

Reid, M.E., 2004, Massive collapse of volcano edifices triggered by hydrothermal pressurization: Geology, v. 32, p. 373–376.

Rowan, E.L., and Goldhaber, M.B., 1995, Duration of mineralization and fluid-flow history of the upper Mississippi Valley lead-zinc district: Geology, v. 23, p. 609–612.

Rowan, E.L., and Goldhaber, M.B., 1996, Fluid inclusions and biomarkers in the upper Mississippi Valley Zn-Pb district: Implications for the fluid flow and thermal history of the Illinois basin: U.S. Geological Survey Bulletin 2094–F, p. F1–F34.

Saad, Yousef, 1990, Sparskit—A basic tool kit for sparse matrix computations: Technical Report RIACS–90–20, Moffet Field, Calif., Research Institute for Advanced Computer Science, NASA Ames Research Center. Software available at:`http://www.ca.umn.edu/~saad/software`

Saad, Yousef, 1994, ILUT—A dual threshold incomplete LU factorization: Numerical Linear Algebra with Applications, v. 1, p. 387–402.

Saad, Yousef, 2003, Iterative methods for sparse linear systems, (2d ed.): Philadelphia, Penn., Society for Industrial and Applied Mathematics, 528 p.

Saibi, H., Ehara, S., Fujimitsu, Y., Nishijima, J., and Fukuoka, K., 2006, Hydrothermal numerical simulation model of Obama geothermal field: Geothermal and Volcanological Research Report of Kyushu University, no. 15, p. 49–57.

Sanford, W.E., 2005, Hydrothermal response to the Chesapeake Bay bolide impact: Geofluids, v. 5, p. 185–201.

Sengers, J.V., and Kamgar-Parsi, B., 1984, Representative equations for the viscosity of water substance: Journal of Physical and Chemical Reference Data, v. 13, p. 185–205.

Smith, Leslie, and Chapman, D.S., 1983, On the thermal effects of groundwater flow: Journal of Geophysical Research, v. 88, p. 593–608.

Sorey, M.L., Grant, M.A., and Bradford, E., 1980, Nonlinear effects in two-phase flow to wells in geothermal reservoirs: Water Resources Research, v. 16, no. 4, p. 767–777.

Stoer, J., and Bulirsch, R., 1993, Introduction to numerical analysis, (2d ed.): New York, Springer-Verlag, 660 p.

Varga, R.S., 1962, Matrix iterative analysis: Englewood Cliffs, New Jersey, Prentice-Hall, 322 p.

Watson, J.T.R., Basu, R.S., and Sengers, J.V., 1980, An improved representative equation for the dynamic viscosity of water substance: Journal of Physical and Chemical Reference Data, v. 9, p. 1255–1290.

Williams, C.F., and Narisimhan, T.N., 1989, Hydrogeologic constraints on heat flow along the San Andreas fault—A testing of hypotheses: Earth and Planetary Science Letters, v. 92, p. 131–143.

Winter, T.C., 1983, The interaction of lakes with variably saturated porous media: Water Resources Research, v. 19, no. 5, p. 1,203–1,218.

APPENDIX 1 — SOFTWARE FOR HYDROTHERM AND HYDROTHERM INTERACTIVE PROGRAMS

This appendix covers obtaining and installing the computer codes for Version 3 of the HYDROTHERM and HYDROTHERM INTERACTIVE programs. The distribution packages are provided as compressed files in a format that is compatible with the computer platform on which it will be used. Each release package of the HYDROTHERM and the HYDROTHERM INTERACTIVE program codes carries an identification number in the form: *v.r.p* where

v is the version number,
r is the release number, and
p is the patch number.

Distribution packages for HYDROTHERM, HYDROTHERM INTERACTIVE, and the present documentation are available as:

`ht-3.r.p.tar.gz` The HYDROTHERM simulator program package for stand-alone use on a computer platform running the Linux operating system. The distribution is a compressed `tar` file and contains source code, a makefile, a Linux executable file, and example data files with selected output files.

`hti-3.r.p.tar.gz` The HYDROTHERM INTERACTIVE program package with graphical user interface for use on a computer platform running the Linux operating system. The distribution is a compressed `tar` file and contains Linux executable files, Java archive libraries, shared object libraries, and example data files with selected output files.

`ht-3.r.p.msi` The HYDROTHERM simulator program package for stand-alone use on a computer platform running the Microsoft Windows operating system. The distribution is a Windows Installer package, `msi`, file and contains a Windows executable file and example data files with selected output files.

`hti-3.r.p.msi` The HYDROTHERM INTERACTIVE program package with graphical user interface for use on a computer platform running the Microsoft Windows operating system. The distribution is a Windows Installer package, `msi`, file and contains Windows executable files, dynamically linked libraries, Java archive libraries, and example data files with selected output files.

`HT_Doc-3.r.pdf` The Guide to the Revised Ground-Water Flow and Heat Transport Simulator: HYDROTHERM — Version 3 as a `pdf` file.

A1.1 OBTAINING THE SOFTWARE

The distribution packages and documentation are available electronically through the World Wide Web. If an Internet connection is not available, contact the code custodian at the address given in appendix 3. There are two repositories for the HYDROTHERM software. The USGS applications software repository is at the USGS Water Resources Information web page at URL:

`http://water.usgs.gov/software/ground_water.html`

The developer's public software repository is at the HYDROTHERM Web site at URL:

`http://wwwbrr.cr.usgs.gov/projects/GW_Solute/hydrotherm/index.shtml`

These URLs point to index pages containing links to the distribution packages available.

For HYDROTHERM INTERACTIVE, the Java Runtime Environment, version 1.5 or later, is a software prerequisite. This free software can be obtained from the World Wide Web at URL:

`http://www.java.com`

After obtaining a HYDROTHERM distribution and a Java Runtime Environment (if required), the packages will need to be uncompressed and files extracted. Then the distribution directory tree may be rearranged as desired.

A1.2 INSTALLING THE SOFTWARE

To install HYDROTHERM INTERACTIVE, first install the software for the Java Runtime Environment, if not already present. The installation of the HT and HTI software normally only involves uncompressing and unpacking the distribution. For Linux platforms, first create and change to the directory chosen for installing HYDROTHERM or HYDROTHERM INTERACTIVE. The minimum disc space required on a Linux platform to install HYDROTHERM is 20 Megabytes (MB) and to install HYDROTHERM INTERACTIVE is 40 MB. Note that the distribution packages will unpack into a top level directory named `HT` or `HTI`. For the `.tar.gz` package, use a command like:

`gunzip ht[i]-3.r.p.tar.gz | tar -xvf -` or

`tar -xzf ht[i]-3.r.p.tar.gz`

For Windows platforms, the distribution uses Windows Installer to install the HYDROTHERM or HYDROTHERM INTERACTIVE software. Administrator privileges on the target computer are necessary to execute the installation package, `ht[i]-3.r.p.msi`. The brackets,[], indicate that the `i` is included when working with HYDROTHERM INTERACTIVE. The minimum disc space required on a Windows platform to install HYDROTHERM is 20 MB and to install HYDROTHERM INTERACTIVE is 35 MB. Double clicking on the package name will invoke the unpacking and installation process.

If difficulties are encountered, contact the code custodian at the address in appendix 3.

A1.2.1 Distribution Layout for HYDROTHERM

Unpacking the distribution creates a top-level directory named `HT` and several subdirectories. Figure A1.1 illustrates the distribution layout for Linux and Windows platforms showing the directories and selected files. The names in italic font are file names while the names in Roman font are directory names. A name with a number range in square brackets denotes a range of directory or file names. An asterisk with a suffix denotes all the file names incorporating that suffix. Note that the Linux distribution includes the Fortran source code. Note that the executable program files are invoked by moving up two directory levels from the example file locations and over to the `bin` directory. The essential program for the stand-alone simulator is either the Linux executable **`ht`**, built using Lahey/Fujitsu Fortran 95 Compiler Release L6.20d, or the Windows executable **`ht.exe`**, built using Intel Visual Fortran Compiler v9.1.

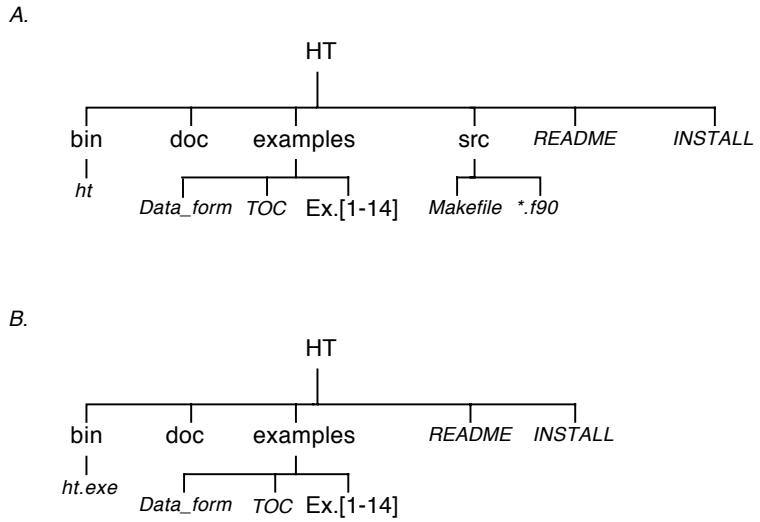


Figure A1.1. HT directory layout for (*A*) Linux and (*B*) Windows platforms.

A1.2.2 Distribution Layout for HYDROTHERM INTERACTIVE

Unpacking the distribution creates a top-level directory named `HTI` and several subdirectories. Figure A1.2 shows the distribution layout for Linux and Windows platforms showing the directories and selected files. The names in italic font are file names while the names in Roman font are directory names. A name with a number range in square brackets denotes a range of directory or file names. An asterisk with a suffix denotes all the file names incorporating that suffix. Neither distribution of HTI includes the Fortran, C, or Java source code. Note that the executable script files to launch the program are invoked by moving up two directory levels from the example file locations and over to the `bin` directory. The shared object libraries for the simulator component of the interactive versions are either the Linux shared object libraries, `*.so`, built using Lahey/Fujitsu Fortran 95 Compiler Release L6.20d and GNU C Compiler GCC 4.1.1, or the Windows dynamically loaded libraries, `*.dll`, built using Intel Visual Fortran Compiler v9.1. and Microsoft Visual Studio 2005 C compiler. The Java Development Kit used for building the Java archive libraries was jdk 1.5.0_09 for both the Linux and Windows platforms.

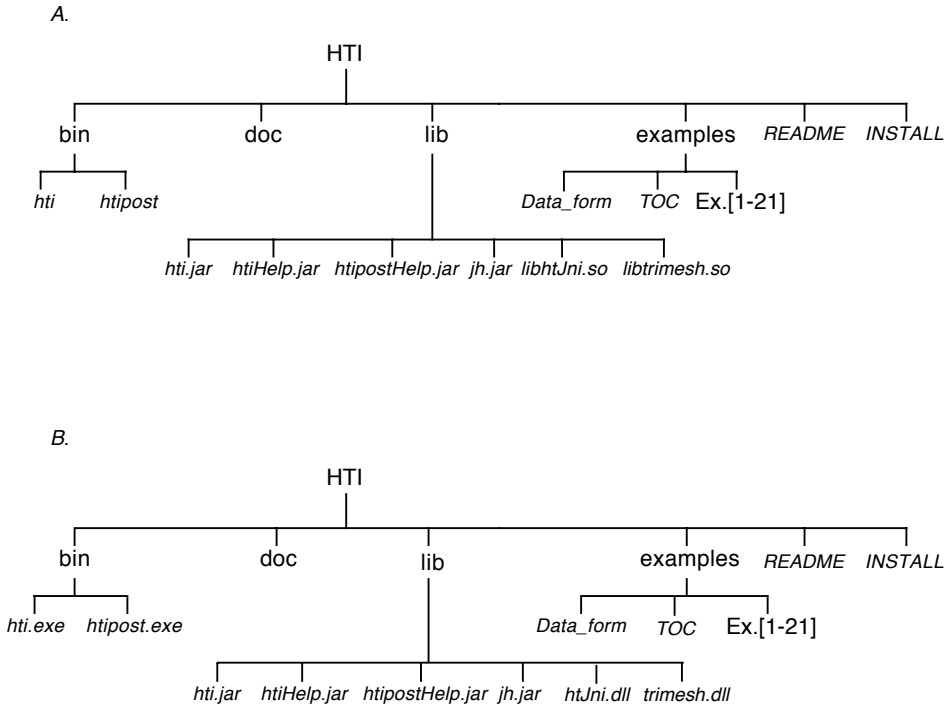


Figure A1.2. HTI directory layout for (*A*) Linux and (*B*) Windows platforms.

A1.3 BUILDING THE SIMULATOR EXECUTABLE

This section describes how to build and install the HYDROTHERM executable code for the stand-alone simulator only. Building the executable normally follows making changes to the source code. Prerequisites for building are a Fortran-90 compiler and a Basic Linear Algorithm Subroutines (BLAS-1) library. Please see the README and INSTALL files in the distribution package for the applicable operating system. They contain the most up-to-date information and instructions.

For Linux operating systems:

1. Changethe working directory to `HT/src`.
2. Edit the`Makefile` in the `src` directory to specify the local Fortran compiler and appropriate compiler flags.
3. Type**`make all`** to compile the executable file.
4. Type **`make install`** to install the executable files and shell scripts into their proper locations.

To build and install HYDROTHERM for a Windows operating system or HYDROTHERM INTERACTIVE for either a Linux or a Windows operating system, please contact the code custodian at the address in appendix 3 to obtain the source code.

APPENDIX 2 — RUNNING THE PROGRAMS

This appendix describes running the computer codes for the HYDROTHERM and HYDROTHERM INTERACTIVE programs. The instructions cover how to run one of the demonstration examples using an input file, document file, or saved-simulation file that is provided. Preparation of these files is described in chapters 5 and 7. The path names in the following instructions are valid for a standard installation from the distribution package. For convenience of future work using HYDROTHERM, users may wish to add the location of the executable file to their path environment variable. All of the example output files have the suffix `.orig` in their file names to prevent overwriting by the simulator.

A2.1 RUNNING HYDROTHERM

For Linux operating systems:

1. Changeto one of the `Ex.` directories under the `examples` directory.
2. Runthe example by invoking HYDROTHERM, **`../../bin/ht`**, and entering the name of the data file and the suffix for the output files when prompted.
3. Comparethe results to the output files provided.

For Windows operating systems:

1. Opena window to use as a command line interface.
2. Changeto one of the `Ex.` directories under the `examples` directory.
3. Runthe example by invoking HYDROTHERM, **`..\..\bin\ht.exe`**, and entering the name of the data file and the suffix for the output files when prompted.
4. Comparethe results to the output files provided.

A2.2 RUNNING HYDROTHERM INTERACTIVE

For Linux operating systems:

1. Changeto one of the `Ex.` directories under the `examples` directory.
2. Startup the HYDROTHERM preprocessor by typing **`../../bin/hti`**.
3. Goto the **`File`** menu and select an existing document file, `xxx.hti`, using **`Open...`**. This step bypasses the setting up of the example using the preprocessor.
4. Goto the **`Show`** menu and select the **`Postprocessor`** item to open the `Postprocessor` window.
5. Selectthe dependent variable to observe during the simulation from the `Display:` list.
6. Runthe simulation by clicking on the `Run` button at the left side of the `Postprocessor` window.
7. Observe the simulation and, after completion, compare the results to the output files provided.

For Windows operating systems:

1. Startup the HYDROTHERM preprocessor by clicking on the `Start` button, going to the **`Programs`** menu, and selecting **`HTI`**.
2. Goto the **`File`** menu, change to an example directory, and select an existing document file, `xxx.hti`, using **`Open...`**. This step bypasses the setting up of the example using the preprocessor.
3. Goto the **`Show`** menu and select the **`Postprocessor`** item to open the `Postprocessor` window.

4. Selectthe dependent variable to observe during the simulation from the `Display:` list.
5. Runthe simulation by clicking on the `Run` button at the left side of the `Postprocessor` window.
6. Observe the simulation and, after completion, compare the results to the output files provided.

A2.3 RUNNING HYDROTHERM INTERACTIVE POSTPROCESSOR IN STAND-ALONE MODE

To playback a previous simulation, for Linux operating systems:

1. Changeto one of the `Ex.` directories under the `examples` directory.
2. Startup the HYDROTHERM postprocessor by typing **`../../bin/htpost`**.
3. Goto the **`File`** menu and select an example directory and a saved simulation file, `ht.sim`, using **`Load...`**.
4. Selectthe dependent variable to observe during the playback from the `Display:` list.
5. Playbackthe simulation by clicking on the `Run` button at the left side of the `Postprocessor` window.
6. Observe the simulation, stop, advance one time step, continue, and replay as desired.

To playback a previous simulation, for Windows operating systems:

1. Startup the HYDROTHERM postprocessor by clicking on the `Start` button, going to the **`Programs`** menu and selecting **`HTpost`**.
2. Goto the **`File`** menu and select an example directory and a saved simulation file, `ht.sim`, using **`Load...`**.
3. Selectthe dependent variable to observe during the playback from the `Display:` list.
4. Runthe simulation by clicking on the `Run` button at the left side of the `Postprocessor` window.
5. Observe the simulation, stop, advance one time step, continue, and replay as desired.

To run a simulation using exported data, for Linux operating systems:

1. Changeto one of the `Ex.` directories under the `examples` directory.
2. Startup the HYDROTHERM postprocessor by typing **`../../bin/htpost`**.
3. Goto the **`File`** menu and select an existing input file, `ht.in`, using **`Load...`**. This step imports an example previously set up using the preprocessor.
4. Selectthe dependent variable to observe during the simulation from the `Display:` list.
5. Runthe simulation by clicking on the `Run` button at the left side of the `Postprocessor` window.
6. Observe the simulation and, after completion, compare the results to the output files provided.

To run a simulation using exported data, for Windows operating systems:

1. Startup the HYDROTHERM postprocessor by clicking on the `Start` button, going to the **`Programs`** menu and selecting **`HTpost`**.
2. Goto the **`File`** menu, change to an example directory, and select an existing input file, `ht.in`, using **`Load...`**. This step imports an example previously set up using the preprocessor.
3. Selectthe dependent variable to observe during the simulation from the `Display:` list.
4. Runthe simulation by clicking on the `Run` button at the left side of the `Postprocessor` window.
5. Observe the simulation and, after completion, compare the results to the output files provided.

If difficulties are encountered, contact the code custodian at the address in appendix 3 or at the address in the README file of the distribution.

APPENDIX 3 — USER REGISTRATION

To register as a user of HYDROTHERM, please return or fax this page to:

HYDROTHERM Code Custodian
U.S. Geological Survey
P.O. Box 25046, Mail Stop 413
Denver Federal Center
Denver, Colorado, USA 80225

fax: (303) 236-5034

or send e-mail to:

klkipp@usgs.gov

with your:

Name

Organization

Address

E-mail address

Date

Registered users will be notified by e-mail of new releases of the programs, and revisions to the documentation.