

**Mathematica notebooks for steady groundwater
flow in leaky multi-aquifer systems using
Hemker's solutions**

Xiaoyong Zhan
Kansas Geological Survey

KGS Open-file Report 2003-17

February 2003

**Mathematica notebooks for steady groundwater
flow in leaky multi-aquifer systems using
Hemker's solutions**

Xiaoyong Zhan
Kansas Geological Survey

KGS Open-file Report 2003-17

February 2003



Disclaimer

This report was prepared as an account of work sponsored by Kansas Geological Survey. Neither the Kansas Geological Survey nor any of employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the Kansas Geological Survey. The views and opinions of authors expressed herein do not necessarily state or reflect those of Kansas Geological Survey.

Mathematica notebooks for steady groundwater flow in leaky multi-aquifer systems using Hemker's solutions

Xiaoyong Zhan
Kansas Geological Survey

Abstract

The use of the Mathematica language for groundwater modeling and in particular to solve the well hydraulic problems is explored. It is shown that Mathematica's symbolic derivation, numerical calculation, and graphical presentation features can be exploited to write programs that are not only significantly more concise than their counterparts written in Fortran, C, or C++, but also performing efficient. To illustrate this, a Mathematica notebooks for steady groundwater flow in leakage multi-aquifer systems have been written. The intention is to provide a flexible and easy-to-use framework for solving groundwater problems. Users do not have to learn Mathematica programming. All users need is to provide data and to interpret the results.

1. Introduction

Stream-aquifer interaction is critical to both groundwater and surface resources. Multi-layer aquifers interact with river and stream in various formats. To consider effect of recharge or discharge of aquifers, Hemker (1984) developed applicable models for steady flow in leaky multiple-aquifer system. However, a handy working tool for the solutions could be desirable to groundwater practitioners. In this report, Mathematica notebooks are prepared to provide an all-in-one and user-friendly tool. The Mathematica notebooks include (1) radial flow to wells completely penetrating one or more aquifers; (2) parallel flow to canals or drains completely penetrating one or more aquifers; and (3) parallel flow to canals or rivers partially penetrating the leaky top-layer aquifer. They are based on the theory introduced by Hemker (1984) and the results here are simply duplicated results presented in Hemker's paper (1984). This report mainly presents the notebooks and the application examples so the reader can use them to study their own problems by following these notebooks and examples. They can be used to easily investigate the aquifer and to avoid the complicated programming and computing procedure before getting a solution for a practical problem.

2. Radial flow to wells completely penetrating one or more aquifers

The first application example here is a four-layered semi-confined aquifers with pumping at the second aquifer from the top. The transmissivities are assumed to be 2000, 1500, 500, 2000 m²/day, from top to bottom, respectively. Estimates of hydraulic resistance for the top, intermediate, and bottom layers are 1000, 1500, 1000, 4000, and 20000 days. The well is screened in the second aquifer with pumping rate 10000m³/d. Using the Mathematica notebook Hemker84-1.nb, the responses in each individual aquifer are easily calculated and the results are the same as the Figure 5 of Hemker's paper (1984). Hemker84-1.nb is listed in the following.

Hemker84-1.nb: Radial flow to wells completely penetrating one or more aquifers

Needs

This section makes graphics add-ons available; plots big enough; current directory the working directory, name and date of last modification showing up.

```
In[23]:= Off[General::"spell1"];
Needs["Graphics`Master`"]
SetOptions[{Plot, ListPlot, LogLogListPlot}, Frame -> True,
  ImageSize -> 400, DefaultFont -> {"Times", 12}, AxesLabel -> True,
  PlotRange -> All];
nbInf = NotebookInformation[EvaluationNotebook[]];
thisFile = StringDrop[ToFileName[{"FileName" /. nbInf}][2]],
  -1];
thisDir = ToFileName[{"FileName" /. nbInf}][1]];
SetDirectory[thisDir];
modDate = FileDate[thisFile];
Print["Notebook: ", thisFile, "\nLast modified: ",
  modDate[[2]], "/ ", modDate[[3]], " /", modDate[[1]]];
```

Notebook: Hemker1.nb

Last modified: 3/ 25 /2003

Some global variables and procedures

dtMin:get time increment;muMb:

get memory usage;

numdig: significant digits
rst:format output;

```
In[32]:= dtMin := Module[{tnew, tdel}, tnew = N[ $\frac{\text{SessionTime}[]}{60}$ ];  
        tdel = tnew - toldmin; toldmin = tnew; tdel];  
  
muMb := N[ $\frac{\text{MemoryInUse}[]}{10^6}$ ];
```

Time and Memory Info at Start

```
In[34]:= Print[  
        {"Time used in min: ", NumberForm[dtMin, {3, 2}]},  
        {"Memory Used in Mb: ", NumberForm[muMb, {3, 2}]} //  
        TableForm];
```

Time used in min: 20.1

Memory Used in Mb: 1.76

Define Tests

```
In[35]:= (* data input *)  
n = 4; (*Qi discharge rate from aquifer i*)  
Q = {0, 10000, 0, 0};  
(*Ti transmissivity of aquifer i [L2/T-1], =KiDi*)  
T = {2000, 1500, 500, 2000};  
(*DdK vertical hydraulic resistance, =Di'/Ki' *)  
DdK = {1000, 1500, 1000, 4000, 20000};  
(* r distant from well *)r =  
{10, 50, 100, 500, 1000, 2000, 3000, 4000, 5000, 6000,  
7000, 8000, 9000, 10000, 15000, 20000};
```

```
In[36]:= (* algorithm *)  
a = Table[1 / (T[[i]] * DdK[[i]]), {i, n}];  
b = Table[1 / (T[[i]] * DdK[[i+1]]), {i, n}];  
A = DiagonalMatrix[a+b];  
Do[A[[i+1, i]] = -a[[i+1]], {i, n-1}];  
Do[A[[i, i+1]] = -b[[i]], {i, n-1}];  
A // MatrixForm  
A = N[A];  
w = Eigenvalues[A]; V = Eigenvectors[A];  
VSTAR = Table[V[[i]] / Sqrt[V^2 . T][[i]], {i, n}];  
VSTAR = Transpose[VSTAR];
```

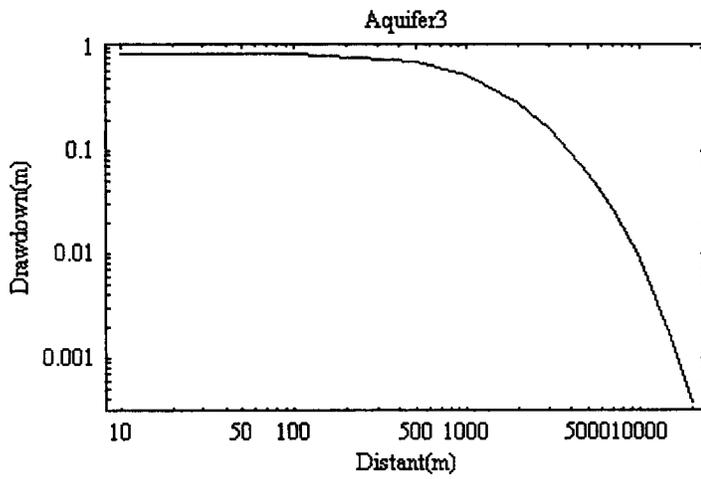
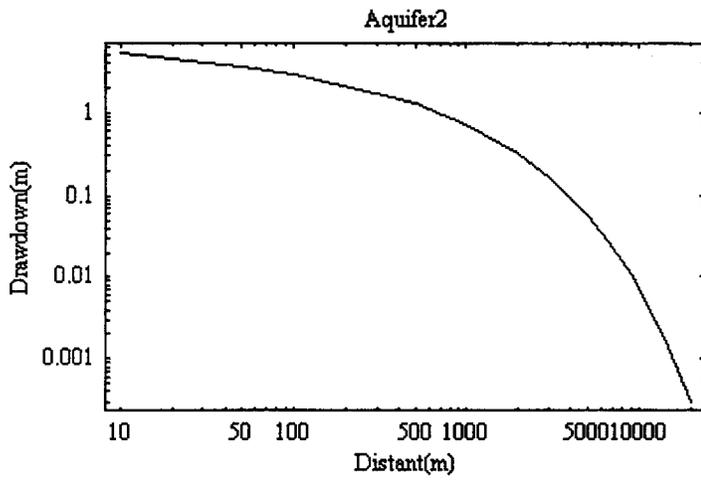
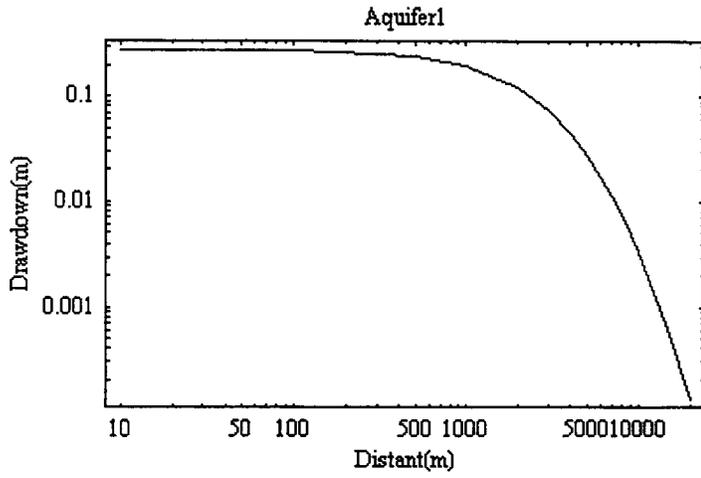
$$\begin{pmatrix} \frac{1}{1200000} & -\frac{1}{3000000} & 0 & 0 \\ -\frac{1}{2250000} & \frac{1}{900000} & -\frac{1}{1500000} & 0 \\ 0 & -\frac{1}{500000} & \frac{1}{400000} & -\frac{1}{2000000} \\ 0 & 0 & -\frac{1}{8000000} & \frac{3}{20000000} \end{pmatrix}$$

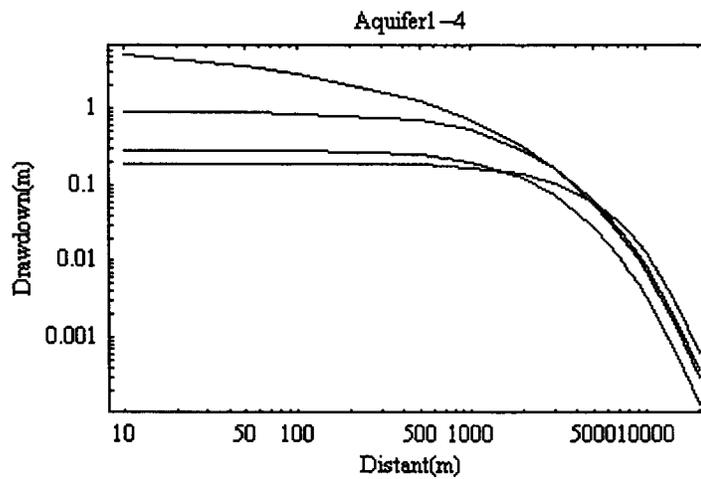
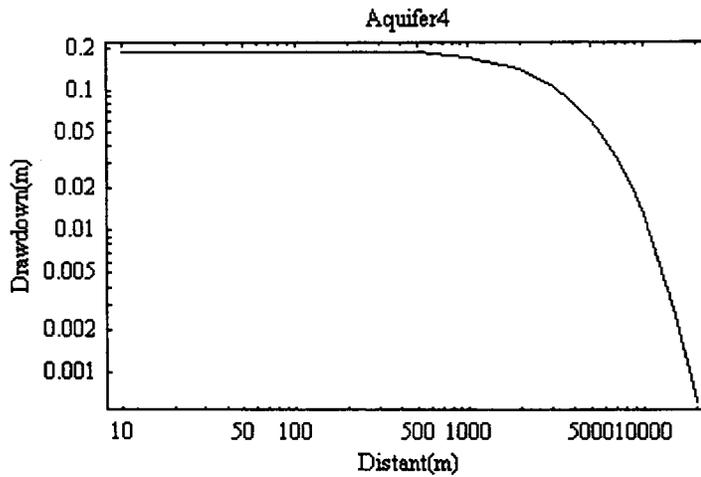
```

In[41]= (* display results *)
Do[
  (* Equation (24) *)
  s =
    Sum[
      N[
        
$$\frac{Q[[k]]}{2 \pi} \sum_{i=1}^n N[VSTAR[[j, i]] VSTAR[[k, i]] \text{BesselK}[0, r \text{Sqrt}[w[[i]]]]]$$

      ],
      {k, 1, n}
    ]
  data = Transpose[{r, s}];
  g[j] = LogLogListPlot[data,
    FrameLabel -> {"Distant (m)", "Drawdown (m)"},
    PlotLabel -> StringJoin["Aquifer", ToString[j]],
    PlotRange -> All, GridLines -> {None, None},
    PlotStyle -> {PointSize[0.01]}, PlotJoined -> True];
  , {j, 1, n}]
Show[Table[g[j], {j, 1, n}],
  AxesLabel -> {"Distant (m)", "Drawdown (m)"},
  PlotLabel -> StringJoin["Aquifer", ToString[1], "-",
    ToString[n]], PlotRange -> All];
Clear[g];

```





Time and Memory Info at End

In[44]=

```
Print[
  {"Time used in min: ", NumberForm[dtMin, {3, 2}]},
  {"Memory Used in Mb: ", NumberForm[muMb, {3, 2}]} //
  TableForm];
```

Time used in min: 0.

Memory Used in Mb: 1.81

The Mathematica notebook Hemker84-2.nb is the same as Hemker84-1.nb except the way of programming or using different Mathematica functions. Hemker84-2.nb is listed in the following.

Hemker84-2.nb: Radial flow to wells completely penetrating one or more aquifers

Needs

This section makes graphics add-ons available; plots big enough; current directory the working directory, name and date of last modification showing up.

```
In[1]:= Off[General::"spell1"];
Needs["Graphics`Master`"]
SetOptions[{Plot, ListPlot, LogLogListPlot}, Frame -> True, ImageSize -> 400,
  DefaultFont -> {"Times", 12}, PlotRange -> All];
nbInf = NotebookInformation[EvaluationNotebook[]];
thisFile = StringDrop[ToFileName[("FileName" /. nbInf)[2]], -1];
thisDir = ToFileName[("FileName" /. nbInf)[1]];
SetDirectory[thisDir];
modDate = FileDate[thisFile];
Print["Notebook: ", thisFile, "\nLast modified: ", modDate[2], "/" , modDate[3],
  " /", modDate[1]];
```

Notebook: Hemker2.nb

Last modified: 3 / 25 /2003

Some global variables and procedures

dtMin: get time increment; muMb:

get memory usage;

numdig: significant digits

rst: format output;

```
In[10]:= dtMin := Module[{tnew, tdel}, tnew = N[ $\frac{\text{SessionTime}[]}{60}$ ]; tdel = tnew - toldmin;
  toldmin = tnew; tdel];
muMb := N[ $\frac{\text{MemoryInUse}[]}{10^6}$ ];
```

Time and Memory Info at Start

```
In[12]:= Print[
  {"Time used in min: ", NumberForm[dtMin, {3, 2}]},
  {"Memory Used in Mb: ", NumberForm[muMb, {3, 2}]} // TableForm];
```

Time used in min: 0.

Memory Used in Mb: 1.63

Define Tests

```

In[13]:= (* data input *)
n = 4; (*Qi discharge rate from aquifer i*) Q = {0, 10000, 0, 0};
(*Ti transmissivity of aquifer i [L2/T-1], =KiDi*) T = {2000, 1500, 500, 2000};
(*DdK vertical hydraulic resistance, =Di'/Ki' *)
DdK = {1000, 1500, 1000, 4000, 20000}; q = Q / (2 π T); (* r distant from well *)
r1 = {10, 50, 100, 500, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000,
      10000, 15000, 20000};

In[14]:= (* algorithm *)
a = Table [1 / (T[[i]] * DdK[[i]]), {i, n}]; b = Table [1 / (T[[i]] * DdK[[i+1]]), {i, n}];
A = DiagonalMatrix[a+b]; Do[A[[i+1, i]] = -a[[i+1]], {i, n-1}];
Do[A[[i, i+1]] = -b[[i]], {i, n-1}];
A // MatrixForm
A = N[A];
w = Eigenvalues[A];
K = DiagonalMatrix[BesselK[0, r Sqrt[w]]];
V = Eigenvectors[A]; V = Transpose[V];
(*Inverse[V] // MatrixForm;*)

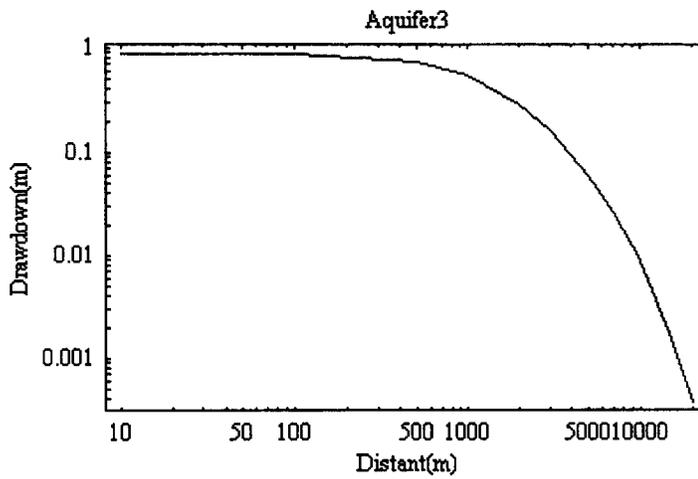
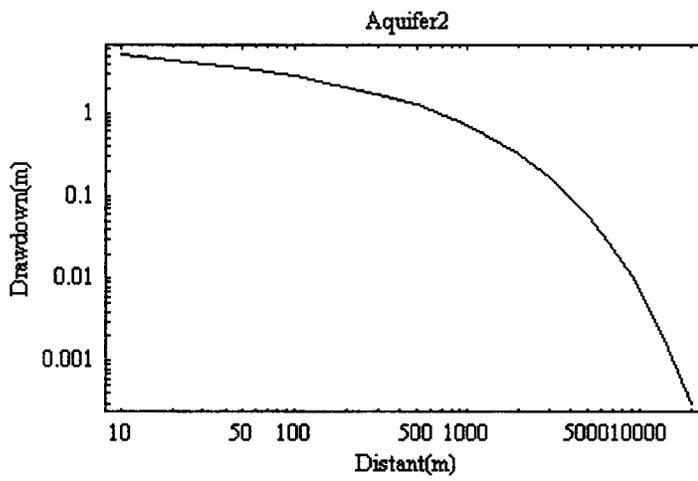
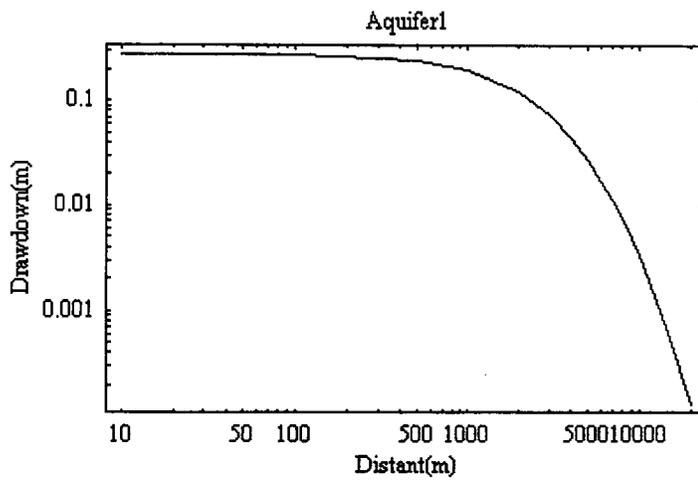
```

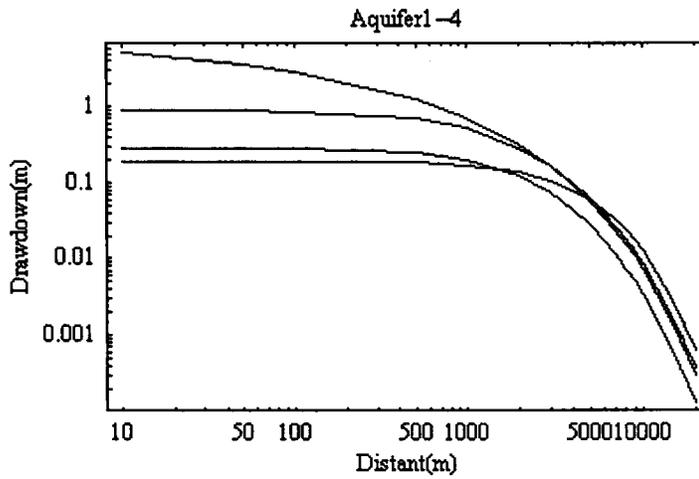
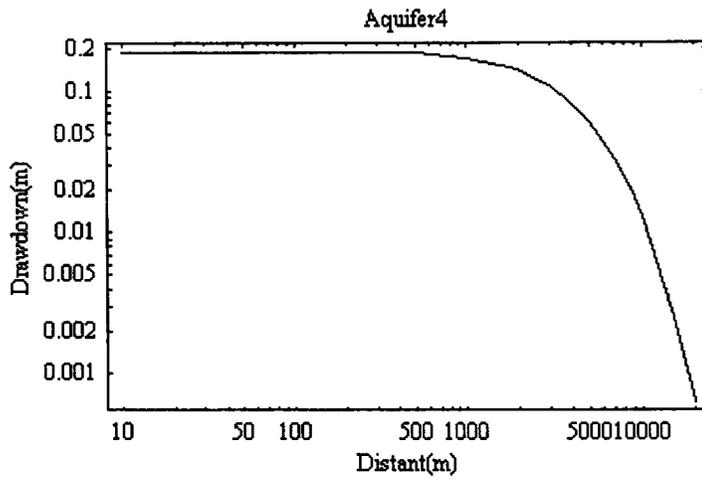
$$\begin{pmatrix}
 \frac{1}{1200000} & -\frac{1}{3000000} & 0 & 0 \\
 -\frac{1}{2250000} & \frac{1}{900000} & -\frac{1}{1500000} & 0 \\
 0 & -\frac{1}{500000} & \frac{1}{400000} & -\frac{1}{2000000} \\
 0 & 0 & -\frac{1}{800000} & \frac{3}{2000000}
 \end{pmatrix}$$

```

In[21]:= (* display results *)
tabres = {};
Do[
  r = r1[[i]];
  drawdown = N[V . K . Inverse[V] . q];
  AppendTo[tabres, {r, drawdown}];
  (* Print["r = ", r, " ", drawdown]; *)
  , {i, 1, n*n}];
Do[
  data = Table[{tabres[[i, 1]], tabres[[i, 2, j]]}, {i, 1, Length[tabres]}];
  g[j] = LogLogListPlot[data, FrameLabel -> {"Distant (m)", "Drawdown (m)"},
    PlotLabel -> StringJoin["Aquifer", ToString[j]], PlotRange -> All,
    GridLines -> {None, None}, PlotStyle -> {PointSize[0.01]}, PlotJoined -> True];
  , {j, 1, n}];
Show[Table[g[j], {j, 1, n}], AxesLabel -> {"Distant (m)", "Drawdown (m)"},
  PlotLabel -> StringJoin["Aquifer", ToString[1], "-", ToString[n]], PlotRange -> All];
Clear[g];

```





Time and Memory Info at End

```

In[20]:= Print[
  {"Time used in min: ", NumberForm[dtMin, {3, 2}]},
  {"Memory Used in Mb: ", NumberForm[muMb, {3, 2}]} // TableForm];

```

Time used in min: 0.

Memory Used in Mb: 1.75

3. Parallel flow to canals or drains completely penetrating one or more aquifers

The second example is a leaky four-aquifer system with drain or gallery on the top aquifer. The transmissivities and the hydraulic resistance are the same as the first example. The discharge per unit width from the drained aquifer (both sides), i.e. top aquifer is $3.15\text{m}^2/\text{day}$ per meter in decreasing the water level. The responses for each individual aquifer are easily computed from the mathematica notebook Hemker84-3.nb and the results are the same as the Figure 6 of Hemker's paper. Hemker84-3.nb is listed in the following.

Hemker84-3.nb: Parallel flow to canals or drains completely penetrating one or more aquifers

Needs

This section makes graphics add-ons available; plots big enough; current directory the working directory, name and date of last modification showing up.

```
In[1]:= Off[General::"spell1"];
Needs["Graphics`Master`"]
SetOptions[{Plot, ListPlot, LogLogListPlot}, Frame -> True, ImageSize -> 400,
  DefaultFont -> {"Times", 12}, PlotRange -> All];
nbInf = NotebookInformation[EvaluationNotebook[]];
thisFile = StringDrop[ToFileName[("FileName" /. nbInf) [2]], -1];
thisDir = ToFileName[("FileName" /. nbInf) [1]];
SetDirectory[thisDir];
modDate = FileDate[thisFile];
Print["Notebook: ", thisFile, "\nLast modified: ", modDate[2], "/", modDate[3],
  " /", modDate[1]];
```

```
Notebook: Hemker3.nb
Last modified: 3/ 25 /2003
```

Some global variables and procedures

dtMin: get time increment; muMb:
get memory usage;
numdig: significant digits
rst: format output;

```
In[10]:= dtMin := Module[{tnew, tdel}, tnew = N[ $\frac{\text{SessionTime}[]}{60}$ ]; tdel = tnew - toldmin;
  toldmin = tnew; tdel];
muMb := N[ $\frac{\text{MemoryInUse}[]}{10^6}$ ];
```

Time and Memory Info at Start

```

In[12]:= Print[
  {"Time used in min: ", NumberForm[dtMin, {3, 2}]},
  {"Memory Used in Mb: ", NumberForm[muMb, {3, 2}]} // TableForm];

```

Time used in min: 0.

Memory Used in Mb: 1.63

Define Tests

```

In[13]:= (* data input *)
(*n number of aquifers*) n = 4; (*Qi discharge rate from aquifer i*)
Q = {3.15, 0, 0, 0}; (*Ti transmissivity of aquifer i [L2/T-1], =KiDi*)
T = {2000, 1500, 500, 2000}; (*DdK vertical hydraulic resistance, =Di'/Ki' *)
DdK = {1000, 1500, 1000, 4000, 20000}; q = Q / (2 T); (* r distant from well *)
ri = {10, 50, 100, 500, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000,
  10000, 15000, 20000};

In[14]:= (* algorithm *)
a = Table [1 / (T[[i]] + DdK[[i]]), {i, n}]; b = Table [1 / (T[[i]] + DdK[[i+1]]), {i, n}];
A = DiagonalMatrix[a+b]; Do[A[[i+1, i]] = -a[[i+1]], {i, n-1}];
Do[A[[i, i+1]] = -b[[i]], {i, n-1}];
A // MatrixForm
A = N[A];
w = Eigenvalues[A];
K = DiagonalMatrix[Exp[-r Sqrt[w]] / Sqrt[w]]
V = Eigenvectors[A]; V = Transpose[V];
(*Inverse[V] // MatrixForm;*)

```

$$\begin{pmatrix}
 \frac{1}{1200000} & -\frac{1}{3000000} & 0 & 0 \\
 -\frac{1}{2250000} & \frac{1}{900000} & -\frac{1}{1500000} & 0 \\
 0 & -\frac{1}{500000} & \frac{1}{400000} & -\frac{1}{2000000} \\
 0 & 0 & -\frac{1}{8000000} & \frac{3}{20000000}
 \end{pmatrix}$$

```

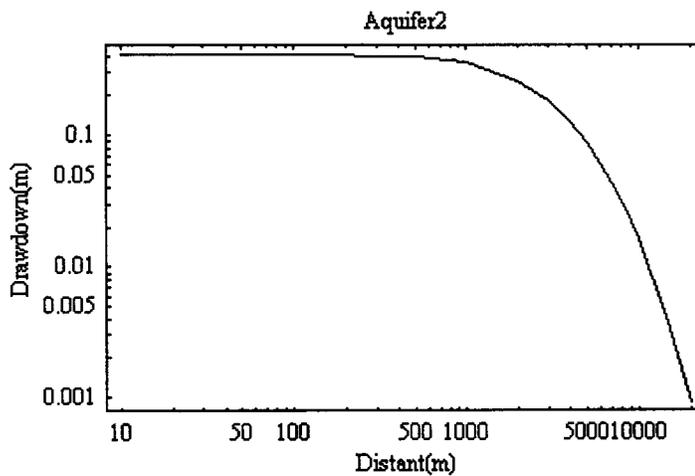
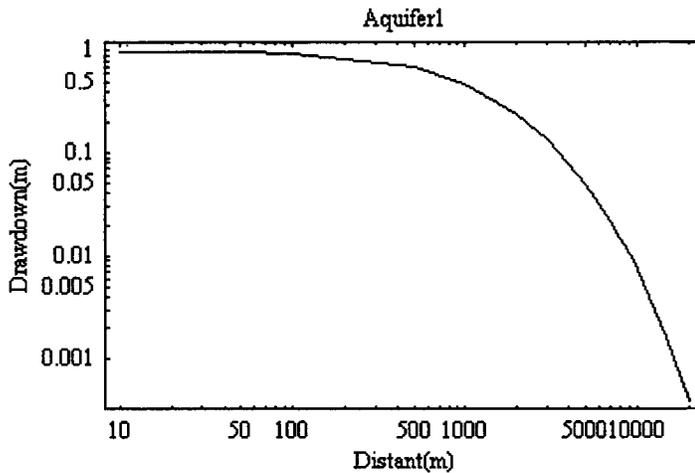
{{560.42 e-0.00178438 x, 0, 0, 0}, {0, 987.998 e-0.00101215 x, 0, 0},
{0, 0, 1794.3 e-0.000557322 x, 0}, {0, 0, 0, 3641.84 e-0.000274586 x}}

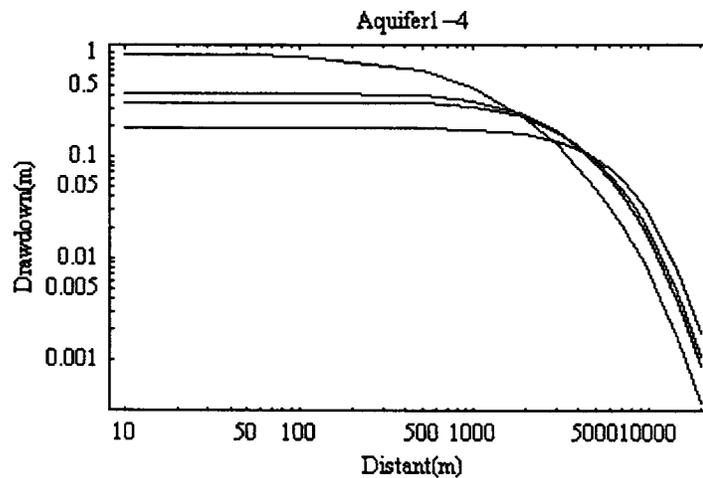
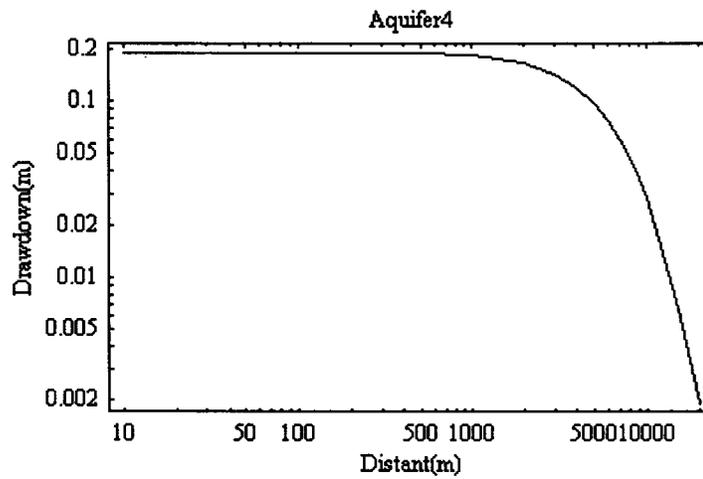
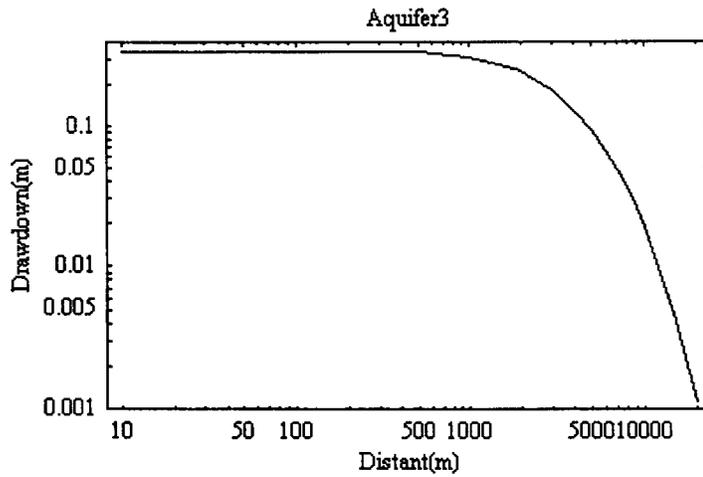
```

```

In[21]:= (* display results *)
tabres = {};
Do[
  r = ri[[i]];
  drawdown = N[V . K . Inverse[V] . q];
  AppendTo[tabres, {r, drawdown}];
  (* Print["r = ", r, " ", drawdown]; *)
  , {i, 1, n * n}];
Do[
  data = Table[{tabres[[i, 1]], tabres[[i, 2, j]]}, {i, 1, Length[tabres]}];
  g[j] = LogLogListPlot[data, FrameLabel -> {"Distant (m)", "Drawdown (m)"},
    PlotLabel -> StringJoin["Aquifer", ToString[j]], PlotRange -> All,
    GridLines -> {None, None}, PlotStyle -> {PointSize[0.01]}, PlotJoined -> True];
  , {j, 1, n}];
Show[Table[g[j], {j, 1, n}], AxesLabel -> {"Distant (m)", "Drawdown (m)"},
  PlotLabel -> StringJoin["Aquifer", ToString[1], "-", ToString[n]], PlotRange -> All];
Clear[g];

```





Time and Memory Info at End

```

In[25]= Print[
  {"Time used in min: ", NumberForm[dtMin, {3, 2}]},
  {"Memory Used in Mb: ", NumberForm[muMb, {3, 2}]} // TableForm];

```

Time used in min: 0.02

Memory Used in Mb: 1.75

Converted by *Mathematica* March 25, 2003

4. Parallel flow to canals or rivers partially penetrating the leaky top-layer aquifer

The last example presents parallel groundwater flow resulting from a recharge river or stream. The transmissivities and the hydraulic resistance are the same as last two examples. At the top layer, the river level is assumed 1m higher than the head in the top layer. The river-bed width and the resistance of the river are assumed 500m and 100 days, respectively. The water head in each individual aquifer can be computed by using Mathematica notebook Hemker84-4.nb and the results are the same as Figure 7 of Hemker's paper (1984). The rate of stream depletion (both sides) or the recharge rate can then be calculated. Hemker84-4.nb is listed in the following.

Hemker84-4.nb: Parallel flow to canals or rivers Partially penetrating the leaky top-layer aquifer

Needs

This section makes graphics add-ons available; plots big enough; current directory the working directory, name and date of last modification showing up.

```
In[1]:= Off[General::"spell1"];
Needs["Graphics`Master`"]
SetOptions[{Plot, ListPlot, LogLogListPlot}, Frame -> True, ImageSize -> 400,
  DefaultFont -> {"Times", 12}, PlotRange -> All];
nbInf = NotebookInformation[EvaluationNotebook[]];
thisFile = StringDrop[ToFileName[("FileName" /. nbInf)[2]], -1];
thisDir = ToFileName[("FileName" /. nbInf)[1]];
SetDirectory[thisDir];
modDate = FileDate[thisFile];
Print["Notebook: ", thisFile, "\nLast modified: ", modDate[2], "/" , modDate[3],
  " /", modDate[1]];
```

Notebook: Hemker4.nb

Last modified: 3/ 25 /2003

Some global variables and procedures

dtMin:get time increment;muMb:

get memory usage;

numdig: significant digits

rst:format output;

```
In[10]:= dtMin := Module[{tnew, tdel}, tnew = N[ $\frac{\text{SessionTime}[]}{60}$ ]; tdel = tnew - toldmin;
  toldmin = tnew; tdel];
muMb := N[ $\frac{\text{MemoryInUse}[]}{10^6}$ ];
```

Time and Memory Info at Start

```
In[12]= Print[
  {"Time used in min: ", NumberForm[dtMin, {3, 2}]},
  {"Memory Used in Mb: ", NumberForm[muMb, {3, 2}]} // TableForm];
```

Time used in min: 0.

Memory Used in Mb: 1.63

Define Tests

```
In[13]= (* data input *)
(*n number of aquifers*) n = 4; (*half width of river*) Lriver = 250;
(*Ti transmissivity of aquifer i [L2/T-1], =KiDi*) T = {2000, 1500, 500, 2000};
(*DdK vertical hydraulic resistance, =Di'/Ki' *)
DdK = {100, 1500, 1000, 4000, 20000}; DdK1 = {1000, 1500, 1000, 4000, 20000};
c = {-1/(T[[1]]*DdK[[1]]), 0, 0, 0}; (* r distant from well *)
ri = {0, 50, 100, 150, 200, 250, 500, 1000, 1250, 1500, 1750, 2000, 2250, 2500, 2750, 3000}
```

{0, 50, 100, 150, 200, 250, 500, 1000, 1250, 1500, 1750, 2000, 2250, 2500, 2750, 3000}

```
In[14]= (* algorithm *)
a = Table [1/(T[[i]]*DdK[[i]]), {i, n}]; b = Table [1/(T[[i]]*DdK[[i+1]]), {i, n}];
A = DiagonalMatrix[a+b]; Do[A[[i+1, i]] = -a[[i+1]], {i, n-1}];
Do[A[[i, i+1]] = -b[[i]], {i, n-1}];
A // MatrixForm
A = N[A];

a = Table [1/(T[[i]]*DdK1[[i]]), {i, n}];
b = Table [1/(T[[i]]*DdK1[[i+1]]), {i, n}];
B = N[Table[Switch[i-j, -1, -b[[i]], 0, a[[i]]+b[[i]], 1, -a[[j+1]], _, 0],
  {i, n}, {j, n}]];
B // MatrixForm

w = Eigenvalues[A];
V = Eigenvectors[A]; V = Transpose[V]; (*Inverse[V]// MatrixForm;*)
u = Eigenvalues[B]; Q = Eigenvectors[B]; (*Q// MatrixForm*)
Q = Transpose[Q]; (*Inverse[Q]// MatrixForm*)

M = DiagonalMatrix[Cosh[r Sqrt[w]]]; M1 = M /. r -> Lriver;
NN = DiagonalMatrix[Exp[-r Sqrt[u]]]; NN1 = NN /. r -> Lriver;
NSTAR = DiagonalMatrix[-Sqrt[u]];
MSTAR = DiagonalMatrix[Sqrt[w] Tanh[Lriver Sqrt[w]]];
ESTAR = DiagonalMatrix[Tanh[Lriver Sqrt[w]] / Sqrt[w]]];
```

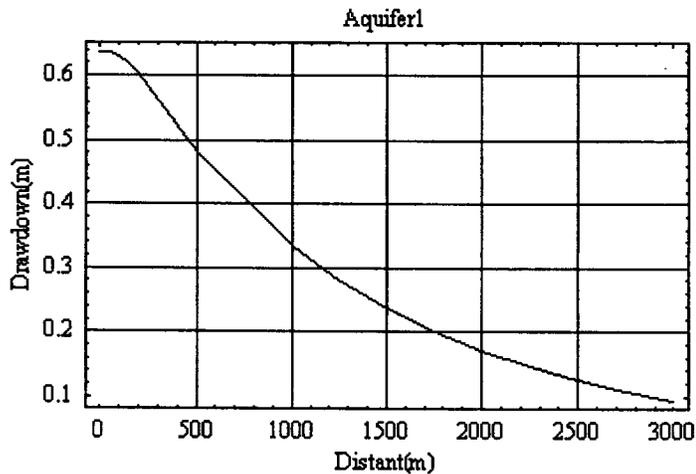
$$\begin{pmatrix} \frac{1}{187500} & -\frac{1}{3000000} & 0 & 0 \\ -\frac{1}{2250000} & \frac{1}{900000} & -\frac{1}{1500000} & 0 \\ 0 & -\frac{1}{500000} & \frac{1}{400000} & -\frac{1}{2000000} \\ 0 & 0 & -\frac{1}{8000000} & \frac{3}{20000000} \end{pmatrix}$$

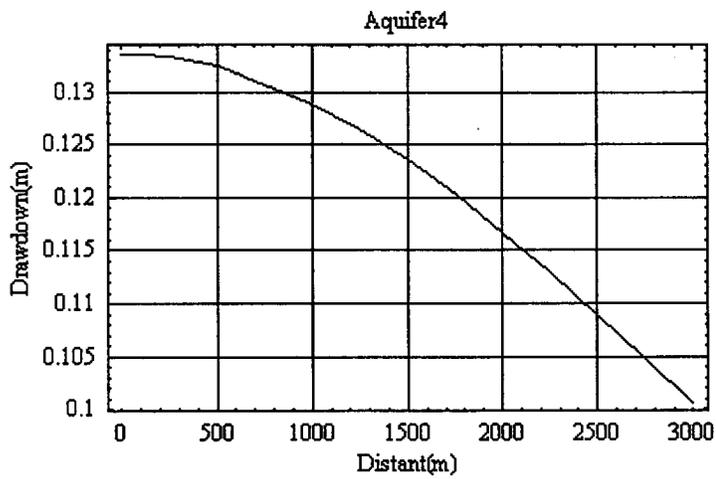
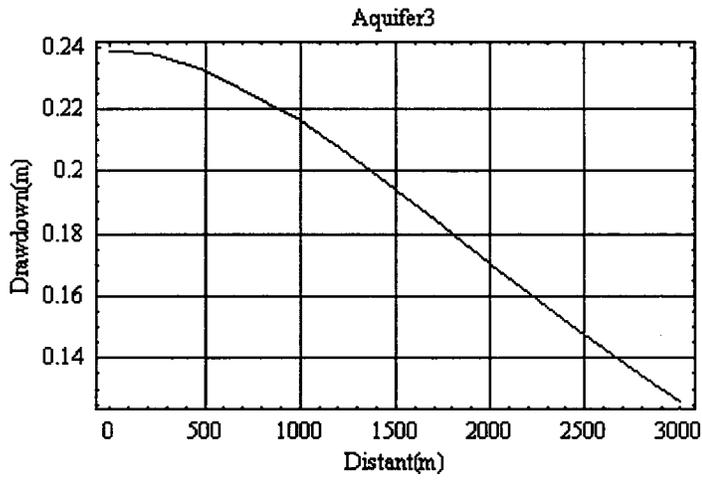
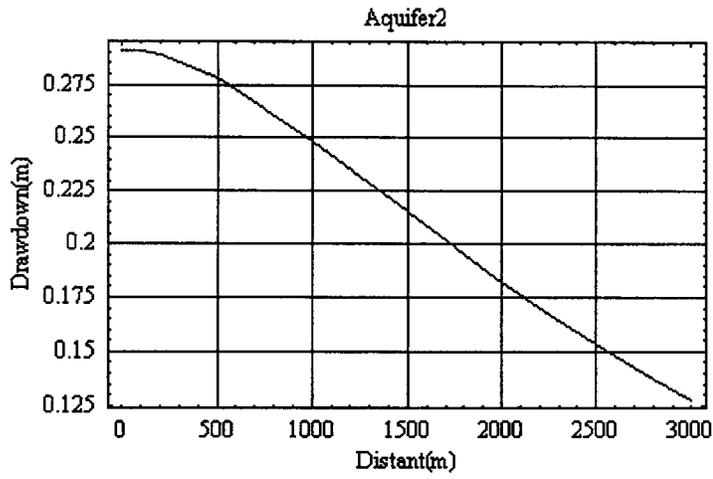
$$\begin{pmatrix} 8.33333 \times 10^{-7} & -3.33333 \times 10^{-7} & 0. & 0. \\ -4.44444 \times 10^{-7} & 1.11111 \times 10^{-6} & -6.66667 \times 10^{-7} & 0. \\ 0. & -2. \times 10^{-6} & 2.5 \times 10^{-6} & -5. \times 10^{-7} \\ 0. & 0. & -1.25 \times 10^{-7} & 1.5 \times 10^{-7} \end{pmatrix}$$

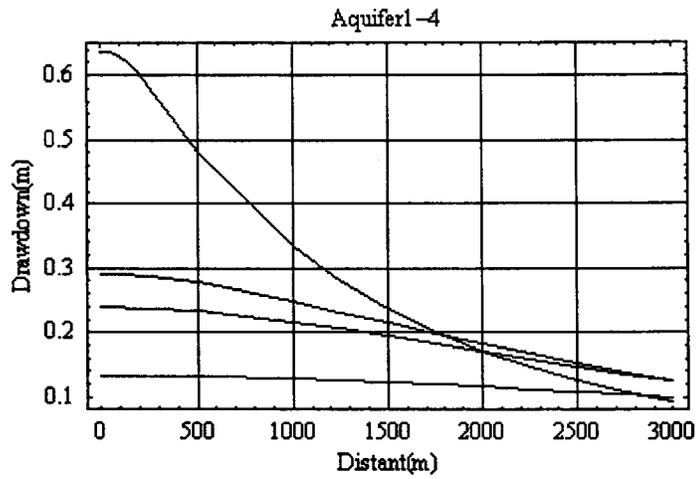
```

n[27]= (* display results *)
tabres = {};
Do[
  r = ri[[i]];
  hl = Inverse[Q . NSTAR . Inverse[Q] - V . MSTAR . Inverse[V]] . V . PSTAR . Inverse[V] . c;
  (* Eq. (30) *)
  drawdown =
    Which[r < Lriver,
      N[
        (V . M . Inverse[ML] . Inverse[V] .
          (hl + V . Inverse[ DiagonalMatrix[w]] . Inverse[V] . c) -
          V . Inverse[ DiagonalMatrix[w]] . Inverse[V] . c) ], r = Lriver, hl,
      r > Lriver, N[ (Q . NN . Inverse[NN1] . Inverse[Q] . hl)]];
  AppendTo[tabres, {r, drawdown}];
  , {i, 1, Length[ri]};
Do[
  data = Table[{tabres[[i, 1]], tabres[[i, 2, j]]}, {i, 1, Length[tabres]}];
  g[j] = ListPlot[data, PlotRange -> All, FrameLabel -> {"Distant(m)", "Drawdown(m)"},
    PlotLabel -> StringJoin["Aquifer", ToString[j]], GridLines -> {Automatic, Automatic},
    PlotJoined -> True];
  , {j, 1, n}];
Show[Table[g[j], {j, 1, n}], PlotRange -> All,
  PlotLabel -> StringJoin["Aquifer", ToString[1], "-", ToString[n]]];
Clear[g];

```







Time and Memory Info at End

```

In[32]:= Print[
  {"Time used in min: ", NumberForm[dtMin, {3, 2}]},
  {"Memory Used in Mb: ", NumberForm[muMb, {3, 2}]} // TableForm];

```

Time used in min: 0.02

Memory Used in Mb: 1.77

Converted by *Mathematica* March 25, 2003

5. Final Remark

Mathematica notebooks are prepared to facilitate the analysis of a multi-aquifer system. These notebooks provide all-in-one, including data-input, derivation of formula, matrix calculation, and graphical presentation, for groundwater practitioners. These notebooks may be used as educational tools as well.

Acknowledgments

Appreciation is expressed to Dr. Ming-Shu Tsou for his review of this report.

Reference

Hemker, C.J., 1984. Steady groundwater flow in leaky multi-aquifer systems. Journal of Hydrology, 72(1984) 355-374.