

Technical Report

CRWR 253

**Porous Medium Advection-Dispersion Modeling
in a Geographic Information System**

by

**John D. Tauxe, B.A., M.S.
Environmental Restoration / Waste Management Fellow**

May 1994

CENTER FOR RESEARCH IN WATER RESOURCES

**Bureau of Engineering Research • The University of Texas at Austin
J. J. Pickle Research Campus • Austin, TX 78712**

ACKNOWLEDGEMENTS

Part of this work was performed under appointment to the Environmental Restoration and Waste Management Fellowship program administered by Oak Ridge Institute for Science and Education for the United States Department of Energy.

ABSTRACT

Solutions to fundamental groundwater flow and transport equations are incorporated into a geographic information system (GIS) as map algebra functions which operate on spatially distributed hydrogeologic data. These functions include a discrete form of Darcy's law to generate flow field maps and to assure conservation of mass, two particle tracking procedures to calculate advection along streamlines, and two gaussian dispersion functions to determine the distribution of a solute in the porous medium from both instantaneous and continuous sources. The modular design of the functions allows for calculation of advection and dispersion of any source which can be modeled as a collection of one or more point sources. The functions are applied in the two-dimensional block-centered finite difference raster GIS environment, using maps of aquifer saturated thickness, porosity, isotropic transmissivity, and head elevation. Additional values are supplied for location and strength of the sources, first-order decay coefficient of the solute, longitudinal and transverse dispersivities, retardation in the porous medium, and time horizon. From these data are calculated a flow field, advection path, and map of concentration of the dispersed constituent. Complex simulations involving transient head fields and multiple transient sources are performed by superimposing results of single-source solutions. All of these calculations take place within the native GIS environment.

TABLE OF CONTENTS

| | |
|---|-----|
| Acknowledgements | ii |
| Abstract | iii |
| Table of Contents | iv |
| List of Figures | vii |
| Chapter 1 | |
| Introduction | 1 |
| 1.1 Contemporary Groundwater Modeling | 2 |
| 1.2 Groundwater Modeling and GIS | 4 |
| 1.3 The Arc/Info GIS and Grid | 5 |
| 1.4 Map Algebra | 6 |
| 1.5 Research Objectives | 8 |
| Chapter 2 | |
| A Review of Environmental Modeling in GIS | 12 |
| 2.1 Geographic Information Systems | 13 |
| 2.2 GIS Data Types | 15 |
| 2.3 Environmental Modeling in GIS | 16 |
| 2.4 Spatial Analysis with GIS | 17 |
| 2.5 Modeling with GIS | 18 |
| 2.5.1 GIS as a Data Base | 19 |
| 2.5.2 GIS as a Pre- and Post-Processor | 20 |
| 2.5.3 GIS as an Integrated Modeling Tool | 23 |
| 2.5.4 GIS as a Decision Support System | 24 |
| 2.5.5 GIS as a Modeling Environment | 25 |

| | |
|--|-----|
| Chapter 3 | |
| Model Development | 28 |
| 3.1 Overview of the Groundwater Modeling Functions | 28 |
| 3.2 General Assumptions | 30 |
| 3.2.1 Simplifying Assumptions | 32 |
| 3.2.2 Limitations Imposed by Grid | 32 |
| 3.3 Porous Medium Flow and Transport | 33 |
| 3.3.1 Continuity | 36 |
| 3.3.2 Darcy's Law | 37 |
| 3.3.3 The DarcyFlow Function | 42 |
| 3.4 Advection Through the Flow Field | 44 |
| 3.4.1 Numerical Approximation of Streamlines | 45 |
| 3.4.2 Analytical Approximation of Streamlines | 49 |
| 3.4.3 Streamlines on a Grid | 57 |
| 3.5 Advection-Dispersion in Porous Media | 60 |
| 3.5.1 The Advection-Dispersion Equation | 73 |
| 3.5.2 Solution for an Instantaneous Point Source | 76 |
| 3.5.3 Solution for a Continuous Point Source | 86 |
| Chapter 4 | |
| Programming for the GIS | 90 |
| 4.1 Implementing the Functions In Grid | 90 |
| 4.2 Constructing the Grid Integration Programs | 91 |
| 4.3 Linking the New Version of Grid | 97 |
| Chapter 5 | |
| Model Applications | 100 |
| 5.1 Groundwater Analysis in Grid | 100 |
| 5.2 Model Verification | 103 |
| 5.2.1 The Capture Well Simulation | 104 |
| 5.2.2 The Well Dipole Simulation | 114 |
| 5.2.3 The Stepped Puff Simulation | 124 |
| 5.2.4 The Convolved Plume Simulation | 131 |
| 5.3 Model Demonstration | 136 |
| 5.3.1 GIS Model of the Culebra Dolomite | 136 |
| 5.3.2 Travel Time Calculation | 146 |
| 5.3.3 Transport Modeling Using PorousPuff | 155 |
| 5.3.4 Transport Modeling Using PorousPlume | 164 |

| | |
|---|-----|
| Chapter 6 | |
| Conclusions | 167 |
| 6.1 Research Objectives Revisited | 167 |
| 6.2 Discussion | 170 |
| 6.3 Refining the Technique | 171 |
| 6.4 Enhancing Grid | 172 |
| 6.5 The Future of Environmental Modeling in GIS | 174 |
| | |
| List of Symbols | 177 |
| | |
| Bibliography | 179 |
| | |
| Appendix | 202 |

LIST OF FIGURES

| | | |
|-----------|--|----|
| Figure 1 | Darcy's law is used to calculate fluid fluxes into adjacent cells. . | 39 |
| Figure 2 | Darcy's Law applied to one side of a grid cell. | 39 |
| Figure 3 | Input and output grids for DarcyFlow. | 43 |
| Figure 4 | Particle tracking by Euler's method. | 47 |
| Figure 5 | Particle tracking by Heun's method. | 47 |
| Figure 6 | Interpolated bilinear surface over a cell. | 50 |
| Figure 7 | Cells are redefined for the DarcyTrack function. | 58 |
| Figure 8 | Example of streamline intersections with cell boundaries for the DarcyTrack function. | 59 |
| Figure 9 | Flow chart for the DarcyTrack program. | 62 |
| Figure 10 | Diffusion modeled as a random process produces a normal (gaussian) distribution. | 63 |
| Figure 11 | Field dispersivity data show a rough correlation of α_L to L. (After Gelhar et al., 1992) | 66 |
| Figure 12 | Field dispersivity data fit of $\log(\alpha_L)$ to $\log(\log(L))$ | 68 |
| Figure 13 | Field dispersivity data with a linear regression of α_L vs L. | 69 |
| Figure 14 | Transverse vs longitudinal dispersivity. (Data from Gelhar, et al., 1992.) | 71 |
| Figure 15 | Advection and dispersion of a point source. | 76 |
| Figure 16 | Perspective illustration of the gaussian dispersion of a puff. | 79 |
| Figure 17 | A sequence of three puffs along a path line is shown superimposed on the underlying grid. | 79 |
| Figure 18 | Integration of the concentration function in (X_L, X_T) coordinates. | 80 |
| Figure 19 | Gauss-Legendre quadrature integration is used to estimate the average concentration over each cell. | 83 |
| Figure 20 | The order of integration is defined by the ratio of standard deviation of the distribution to the grid cell size. | 84 |

| | | |
|-----------|--|-----|
| Figure 21 | Integration error as a function of standard deviation vs cell size for Gaussian-Legendre quadrature integration. | 85 |
| Figure 22 | Surface representation of the plume model. | 88 |
| Figure 23 | Dispersion plume calculated by equation (76). | 89 |
| Figure 24 | Head field of the capture well problem. Vertical extent 0 to 25 m. | 105 |
| Figure 25 | Flow direction vectors for the capture well, generated by DarcyFlow. | 107 |
| Figure 26 | Flow magnitude for the capture well, generated by DarcyFlow. | 108 |
| Figure 27 | Streamlines for the capture well, generated by ParticleTrack. | 110 |
| Figure 28 | Streamlines for the capture well, generated by DarcyTrack. | 111 |
| Figure 29 | Deviation of ParticleTrack streamlines for the capture well. | 112 |
| Figure 30 | Deviation of DarcyTrack streamlines for the capture well. | 112 |
| Figure 31 | Comparison of capture well travel times by ParticleTrack, DarcyTrack, and the analytical solution. | 113 |
| Figure 32 | Head field of the well dipole problem. Contour interval 100 m. | 115 |
| Figure 33 | Flow direction vectors for the well dipole, generated by DarcyFlow. | 116 |
| Figure 34 | Flow magnitude for the well dipole, generated by DarcyFlow. | 117 |
| Figure 35 | Flow streamlines and corresponding fractional breakthrough for the well dipole. | 119 |
| Figure 36 | Streamlines for the well dipole, generated by ParticleTrack. | 122 |
| Figure 37 | Streamlines for the well dipole, generated by DarcyTrack. | 123 |
| Figure 38 | Comparison of well dipole travel times by ParticleTrack, DarcyTrack, and the analytical solution. | 124 |
| Figure 39 | Basic application of PorousPuff to distribute a point source. | 125 |
| Figure 40 | Repeated application of PorousPuff to distribute several sources. | 125 |
| Figure 41 | The concentration distribution resulting from the five-step redistributed puff. | 127 |

| | | |
|-----------|--|-----|
| Figure 42 | Difference in concentration distributions from the five-step redistributed puff and the single-step puff. | 128 |
| Figure 43 | Plume generated by PorousPlume. | 132 |
| Figure 44 | Convolved plume generated by repeated application of PorousPuff. | 133 |
| Figure 45 | Difference between the plume convolved by PorousPuff and that calculated by PorousPlume. | 135 |
| Figure 46 | Arc/Info coverage of rivers in New Mexico. | 137 |
| Figure 47 | Arc/Info coverage of features in southeastern New Mexico, identifying the location of the WIPP study area. | 138 |
| Figure 48 | Base grid for analysis of the Culebra Dolomite in Grid, with an outline of the WIPP Site boundary. | 139 |
| Figure 49 | WIPP regional steady-state freshwater head elevations (after LaVenue and RamaRao, 1992). | 141 |
| Figure 50 | WIPP regional log transmissivities (after LaVenue and RamaRao, 1992). | 142 |
| Figure 51 | WIPP regional flow field generated by DarcyFlow. | 144 |
| Figure 52 | WIPP regional volume balance residual generated by DarcyFlow. | 145 |
| Figure 53 | Long range travel paths from the WIPP. | 147 |
| Figure 54 | Short range travel paths from the WIPP. | 148 |
| Figure 55 | WIPP Site steady-state head elevations. Compare to Figure 49. | 150 |
| Figure 56 | WIPP Site log transmissivities. Compare to Figure 50. | 151 |
| Figure 57 | WIPP Site volume balance residual grid. | 152 |
| Figure 58 | WIPP Site flow directions generated by DarcyFlow. | 153 |
| Figure 59 | Travel paths from the WIPP Site, generated by ParticleTrack and DarcyTrack. | 154 |
| Figure 60 | Initial puff released from the WIPP Site. | 156 |
| Figure 61 | Step 2 in the evolving puff released from the WIPP Site. | 157 |
| Figure 62 | Step 3 in the evolving puff released from the WIPP Site. | 158 |
| Figure 63 | Step 4 in the evolving puff released from the WIPP Site. | 159 |
| Figure 64 | Step 5 in the evolving puff released from the WIPP Site. | 160 |

| | | |
|-----------|--|-----|
| Figure 65 | Step 7 in the evolving puff released from the WIPP Site. . . . | 161 |
| Figure 66 | Step 9 in the evolving puff released from the WIPP Site. . . . | 162 |
| Figure 67 | Step 11 in the evolving puff released from the WIPP Site. . . | 163 |
| Figure 68 | Plume of a conservative, nonreactive tracer from the WIPP Site, generated by PorousPlume. | 165 |
| Figure 69 | Plume of ^{239}Pu from the WIPP Site, generated by PorousPlume. | 166 |

POROUS MEDIUM ADVECTION-DISPERSION MODELING IN A GEOGRAPHIC INFORMATION SYSTEM

Chapter 1

Introduction

Our need to understand the complex world of environmental processes easily surpasses our ability. A particularly elusive domain of environmental science lies in the subsurface: groundwater. As living beings we have a vital interest in hydrogeology, since our biosphere is intimately connected with our groundwater. We use it both as a vital resource and as a waste stream, and must understand its hydraulics as a matter of public and environmental health. Computers have become indispensable in cataloging and analyzing environmental data, and in modeling physical behavior in natural systems. Geographic information systems (GIS) were developed for storage and analysis of spatially distributed data, and are approaching a level of sophistication worthy of performing environmental analysis. This work is an effort to extend GIS abilities in the field of groundwater transport modeling.

This research develops new functions designed to aid groundwater modelers in simulating the transport of solutes through porous media by using semi-analytical solutions to Darcy's law and the advection-dispersion equation. This is a modular approach to modeling, where fundamental elements of the

analysis have been broken down into operations executed from the command-line interface or macro language of the GIS. Each command executes an elemental physical process, such as a dispersive puff from an instantaneous point source. The linearity of the solutions allows the modeler to superimpose the dispersion of multiple sources, a process greatly simplified by the raster GIS, so that spatially complex sources can be modeled as a grid of point sources.

The fundamental equations which are solved by these new GIS functions are the theoretical starting point for most groundwater studies. While the functions produce simple results when used for a cursory screening, application of the same functions in creative combination can achieve sophisticated dispersive models. The functions are basic modeling tools, intended to be used for building more complex analyses in the context of environmental modeling in GIS.

The purpose of these functions is to provide tools for screening analysis of groundwater datasets with rapidly executing commands used directly from the GIS environment. They are also well suited for examining contamination scenarios and for general exploration of the dynamics implied by hydrogeological data.

1.1 Contemporary Groundwater Modeling

Today's standard groundwater models are generally stand-alone computer programs with unique data input and output formats. They have been developed by individuals or small groups of workers, usually to address a particular class of modeling problems. This lack of standardization often presents a problem for modelers, who are faced with either using a patchwork of programs to perform various functions or creating their own suite of codes. Even when using existing

software, it is necessary to write data conversion programs to bridge the different data formats.

For example, to interpolate a field of hydraulic conductivities, one might use a spreadsheet to convert point values to a log scale, a surface-generation program to make a field of values. Heads might then be generated by an iterative finite element or finite difference program. A visual examination of all of these fields requires some sort of graphics display program (with its own data formats) to draw a colorized map or wire mesh surface. Flow field generation and transport calculations may also require specialized programs. With this potpourri of programs, data conversion utilities, and intermediate data files, the organization of a suite of simulations becomes a task in itself. It is rare to find all of these functions available in one place, with one data model. The GIS environment is maturing to fulfill this need.

The standard groundwater modeling programs and utilities may not share a common data model or solution technique, but they often have a common basis in physics. For example, most use some form of Darcy's law to determine fluid fluxes. Most make use of the constraints of continuity in solving for a head surface. The various effects of diffusion and dispersion are often lumped together into a model of random normal redistribution. These physical models are typically executed in sequence or iteration to arrive at the desired solution. They are tools in the modeler's toolbox. The functions developed in this research are such tools, designed to perform discrete elemental functions to be used in combination to make a variety of models. As more tools are developed, the models will become more sophisticated.

In the implementation proposed here, analytical solutions of the advection-dispersion equation are built into the GIS, providing the analyst with a rapid

deterministic method for estimation of solute transport in porous media. Two scenarios are presented: the gaussian dispersion of an instantaneous point source and that of a continuous point source. Solutions to these problems exist for an ideal medium and are readily calculated, yet their implementation into a GIS can add to these simple functions a wealth of data from the world of geographical analysis. By extension of these two models by superposition, several other simulations can also be performed. For example, the cumulative effect of spatially distributed sources of a particular solute from different locations and times can be calculated by successive applications of the same fundamental advection-dispersion function.

1.2 Groundwater Modeling and GIS

The purpose of bringing together groundwater modeling with GIS is to provide the modeler with a productive working environment that integrates data management, presentable output, and the tools necessary to perform modeling analyses. The last of these is the focus of this work. This is an effort to incorporate transport modeling capabilities into a GIS in the form of user functions. For example, from information consisting of grids of hydrogeologic data (perhaps derived from point coverages) the modeler can easily generate grids for groundwater flow vectors and solute concentrations. These grids can then be added to maps containing other geographic information such as population density, land use, surface vegetation, or locations of tanks, wells, and pipelines in order to assist in exposure assessments.

Geographic information systems are designed to operate on two classes of data: vector and raster. A vector GIS recognizes three fundamental data types:

points, lines, and polygons. Most geographic features can be described using these data types, and they lend themselves well to storage in a structured query language and database. However, some geographically significant information is better described by a raster dataset or image, such as an aerial photograph or a continuously varying field of data. It is the latter which concerns us here, since groundwater modeling operates on fields of data. By combining raster and vector GIS, we can give geographical significance to our groundwater models by for example overlaying a map of calculated concentrations with land use polygons, pipelines, and point locations of wells.

This research uses a raster model GIS, which operates on discrete spatially distributed (gridded) data by using spatial analysis tools collectively termed map algebra. Map algebra performs calculations on individual grid cells or regions of cells based on the values of other cells in the same and other grids (Tomlin, 1990). In this way, the raster GIS provides a set of functions and operators for spatial modelers, including groundwater specialists.

1.3 The Arc/Info GIS and Grid

With the Arc/Info GIS, developed by Environmental Systems Research Institute (ESRI), all of the tools for data modification, storage, and display are built into its use of map algebra, innovative data storage techniques, and advanced display capabilities. With the addition of recently developed interfaces to standard modeling programs and the tools presented here, groundwater contaminant transport modeling can be done with a series of commands or by use of a command macro file. The Arc Macro Language (AML) can be used to construct modeling programs using the basic elements developed here.

Arc/Info is a vector GIS, but has been recently complemented by Grid, a closely integrated raster GIS which works with Arc/Info. Geographic information may be exchanged between Grid and Arc/Info by converting point, line, and polygon data into raster data, and vice versa. Grid uses a data model which allows for datasets of arbitrary size and extent, freeing the modeler from the often arduous task of data management and from the random access memory limitations of the computer (ESRI, 1991a). It also provides access to the growing range of data available in the form of Arc/Info coverages and datasets. This choice of platform has also introduced limitations, however. As an environment for performing modeling calculations, certain tasks are not readily executed, and computer performance is sluggish due mainly to Grid's dependence on the disk for data storage, rather than random access memory. A further limitation is Grid's data model: a regular two-dimensional grid of square cells of uniform size. This does not allow for local refinements of the grid mesh, nor does it permit three-dimensional modeling.

1.4 Map Algebra

GIS, traditionally used for storage and rudimentary spatial analysis of geographic data, has piqued the interest of the modeling community with the implementation of map algebra. Map algebra is an analysis language for rasterized datasets. It consists of a set of mathematical operators and functions which operate on the cells of a gridded field of data. Each field, or grid, contains one type of data, with varying values for the cells. Map algebra allows for operations on one or more grids, and may produce new grids as output.

Map algebra functions and operators may be classified by the domain of data used in the calculation. A function which operates on a single cell location, using that same location on all its input and output grids, is said to be a local function. Focal functions use information from a region of cells surrounding the central cell. Both of these types of functions can be applied to floating-point grids, such as continuous surfaces or material properties. Zonal functions use data from all cells of a particular type, and is restricted to integer cell data concerning geographic data types, rather than continuous values.

A simple example of a local function is the calculation of a grid of transmissivity values given grids of thickness b and hydraulic conductivity K of a geologic formation. We know that these parameters are related by a common calculation of transmissivity: $T = K \cdot b$. For a particular geographical region, we have defined grids for b and K , with values for each cell. To generate a grid of transmissivities, we simply execute the map algebra expression

$$T_grid = K_grid * b_grid$$

which uses the $*$ multiplication operator. To generate a grid of log transmissivities, we would use

$$\log T_grid = \log_{10}(K_grid * b_grid)$$

or to find K from $\log(T)$ and b ,

$$K_grid = \exp_{10}(\log T_grid) / b_grid$$

In fact, these are the exact forms of the expressions as implemented in the Grid program (ESRI, 1991a). This very simple way of generating new grids (and new data) becomes slightly more complex with more advanced functions.

Another example from Grid is the focal function that calculates the standard deviation of a variable over a region of cells surrounding each cell in the grid (ESRI, 1991b). This function takes the form

$$out_grid = FOCALSTD(in_grid, CIRCLE, radius)$$

where the area sampled is a circle of radius `radius` surrounding each cell of the input grid. The output grid contains the standard deviations for each cell. The groundwater modeling functions in this dissertation are intended to operate on floating-point data, and are of this same basic form:

```
out_grid = FUNCTION( in_grids, parameters, additional_out_grids ).
```

Grid has other advanced spatial analytical functions such as spatial averaging, kriging, network analysis, and watershed delineation. These are of obvious interest to practitioners in water resources and environmental analysis. The program also has the ability to handle grid datasets of any size, with either integer or floating point values. The only limitation to the dimensions of a grid is the amount of disk storage on the computer. By breaking the grid into subparts, Grid accesses only part of the grid at a time, yet can perform functions which require information from the entire grid so that Grid can analyze datasets with many millions of cells. This ability alone makes it attractive for handling large datasets. Furthermore, Grid smoothly interfaces with other ESRI programs such as Arc, ArcPlot and ArcView, so that overlaying concentration profile maps with topography, water well coverages or demographic information is straightforward.

1.5 Research Objectives

This research is exciting to me for several reasons. It is new, it is important, and it is challenging. To my knowledge, no one has implemented the solutions to differential equations into a GIS before, and in that sense this is ground breaking work. The potential for building new functionality specific to modeling with map algebra is great, and will become more significant as others

add to the work as I hope they will. The challenge comes in the effective programming and demonstration of such a work, complicated by working in a largely undocumented application programming interface (API) which is familiar to few and understood by still fewer.

Finally, this effort is a continuation of my lifelong ambitions as a craftsman. Ever since my high school days spent building musical instruments I have taken pride in my woodworking and other creative pursuits. In the same spirit, my graduate studies have been centered around designing computational tools for the modeler. The modeling engineer needs well-crafted tools to create good models. I see myself as both practitioner and craftsman, making and using fine tools to create an elegant model, and using that model to solve problems in applied hydrogeology.

My principal goals in embarking on this research are as follows:

- Determine the most appropriate form of the solutions to the advection-dispersion equation applied to the conditions of instantaneous and continuous point source input of constituents into an aquifer.
- Achieve the implementation of new functions into Grid. Once this has been demonstrated, the addition of other perhaps more sophisticated functions should be straightforward.
- Integrate the new Grid model with existing geological Arc/Info coverages to create a smoothly working modeling environment. This may involve the creation of Arc Macro Language (AML) scripts and dialogs to facilitate repetitive tasks.

- Assess the effectiveness of this approach to groundwater modeling. This is obviously critical to determining if this new technology will be useful to the modeling community.
- Provide through this work a blazed trail for others interested in implementing new functions into Grid. If others see the utility of modeling in the GIS, I invite them to add more functions of general interest to the Grid program.
- Test the hypothesis that a geographic information system can be used as an effective platform for performing groundwater modeling analysis.

The simulation of the fate and transport of solutes in groundwater is an exceedingly complex problem. Limitations in data and constraints of the model often impose such simplifying assumptions that the accuracy of the results is often compromised, but such is the state of the art. Those who have an appreciation for the complexity of geologic media are suspicious of the mathematical approach, and use its results only as a rough guide. It is for this purpose that these models are most useful, and the tools developed in this research are intended for this purpose.

Like most computer models, the models developed in this research are mathematically and conceptually ideal and may not accurately simulate natural conditions. This caveat must always be kept in mind. The equations assume that many properties are constant or uniform, as in a highly idealized medium. Results from these models are simply solutions to the equations -- they are not intended to predict actual concentration distributions except in the coarsest sense.

They show roughly in which direction transport may be expected and at what rates, but the user should not expect to find these concentrations in the field.

The organization of this dissertation is as follows: Chapter 2 is the literature review, to bring the reader up to date on research activities and directions in the emerging field of environmental modeling in geographic information systems. Chapter 3 discusses the conceptual, theoretical, mathematical, and algorithmic development of the new functions which are the core of this research. The implementation of the research into computer programs integrated into the GIS is the focus of Chapter 4. Applications of the finished product for the purposes of model verification and demonstration are described in Chapter 5. The final chapter discusses the results and proposes future research directions. The Appendix contains source code listings and Arc Macro Language files for all of the computer programs.

Chapter 2

A Review of Environmental Modeling in GIS

This research advances the state of the art in environmental modeling in geographic information systems. As GIS grows beyond its traditional tasks of storing, processing, and presenting static geographic data, there is an intense interest from researchers in a wide range of professional fields, focussed on the potential of environmental modeling in GIS to enhance our understanding of the natural world, and to help us manage our activities within it. This need is expressed concisely by Louis Steyaert (1993):

“Computer-based, mathematical models that realistically simulate spatially distributed, time-dependent environmental processes in nature are increasingly recognized as fundamental requirements for the reliable, quantitative assessment of complex environmental issues of local, regional, and global concern. These environmental simulation models provide diagnostic and predictive outputs that can be combined with socioeconomic data for assessing local and regional environmental risk or natural resource management issues.

More recently, the importance of scientific models for the assessment of potential global environmental problems, including regional response to global change, has been illustrated by the National Research Council, Earth System Sciences Committee, International Council of Scientific Unions, International Geosphere-Biosphere Programme, and Committee on Earth Sciences.”

The bulk of literature in the field of GIS and environmental modeling is not yet published in journals, but is rather to be found in conference proceedings. Fedra (1993) illustrates this point by sharing the results of an informal literature

search. Using Geographic Information Systems or GIS as a key, a search of Water Resources Abstracts or Enviroline using DIALOG would yield 100 to 200 entries, and the keyword Environment would produce several thousand. Combining the keywords generates only a few articles.

Yet conferences abound with discussion on GIS-based modeling, as Fedra also points out: “In a hefty volume on Computerized Decision Support systems for Water Managers [from 1989], a conference proceedings of close to 1000 pages, GIS is not mentioned once [in] the subject index. In contrast, and three years later, at a session of the 1991 General Assembly of the European Geophysical Society, dedicated to Decision Support Systems in Hydrology and Water resources Management, more than half the papers discuss GIS as a component of the research method.”

2.1 Geographic Information Systems

The term “geographic information system” requires definition. In its broadest sense, it can mean an entire complex of organizations, personnel, databases, hardware, and software used to process geographic information (Maguire, et al., 1991). Burrough (1986) uses the term to indicate a set of hardware and software tools for storing, retrieving, analyzing, and displaying spatial data. More recently, GIS is used to identify the software alone, and it is in this sense used in this dissertation. Obviously, the software cannot be used without the hardware, but it has become rather hardware independent, running on several platforms.

The specific abilities defining a modern GIS are further clarified by Goodchild (1993): A GIS should have the ability to preprocess data from large

stores into a form suitable for analysis, including such operations as reformatting, change of projection, resampling, and generalization. It should have direct support for analysis and modeling, such that forms of analysis, calibration of models, forecasting, and prediction are all handled through instructions to the GIS. And finally, a GIS should handle postprocessing of results, including such operations as reformatting, tabulation, report generation, and mapping. This research addresses needs in the area of analysis and modeling, and implements groundwater modeling analysis as GIS functions.

Geographic information systems are used in several professional disciplines, including cartography, surveying, remote sensing, image processing, geodesy, demography, and more recently hydrology, geoscience and environmental modeling. Each of these disciplines deals with georeferenced data.

Many modeling codes have been linked to a GIS for data management purposes, but the tools necessary for groundwater analysis within the GIS environment are not well developed. The reasons for this are various: The modeling community has traditionally been unaware of GIS and its capabilities, the geographically-oriented GIS community has not understood the modeling of dynamic processes, the software developers who produce the GIS lack the technical understanding to implement modeling functionality into the GIS program, and most GIS have great demands on the computing hardware, leaving few resources for numerically intensive modeling.

As multidisciplinary cooperation and understanding increase, only the last of these problems still inhibits serious development of modeling in GIS. We hope that the trend of increasing affordability of sophisticated computer hardware will promote further development of numerically intensive modeling within the GIS.

2.2 GIS Data Types

All GIS software share the common functions of a spatial database and analysis system. To different degrees, they are able to handle a wide variety of data by indexing to an attribute common to all spatial data: location. How these data are stored and analyzed differs between systems, with a primary differentiation being vector vs raster data. Goodchild (1993) classifies GIS data into six types, or models: 1) irregular point sampling, such as wells or weather stations, 2) regular point sampling, such as digital elevation models (DEM), 3) contours, such as topographic contour lines and flow streamlines, 4) polygons, such as soil maps and municipal zoning maps, 5) triangular irregular networks (TIN), such as TIN elevation models, and 6) cell grids, such as photogrammetric maps and finite difference grids. (This research operates mostly in the realm of cell grids, although particle tracks are saved as ordered sets of points, and may be classified as contours.) The regular points and cell grids are considered raster data, and the others vector data, since they do not conform to a regular indexing by row and column. GIS excels in the ability to convert between these disparate data types, and a primary function of the GIS is their integration.

Even with the variety of data types supported, GIS has not overcome the discretization problem. Most current systems support only the basic data types (which includes those discussed above) of points, lines, polygons, pixels, or voxels (Robinove, 1986), and these are fundamentally inadequate ways for storing continuous or stochastic data (Burrough, 1992). Goodchild (1993) and Maidment (1993a) make the distinction between discrete data, meaning vector data which specify values only over a small part of space, and continuous data, such as grids and TINs which have values which completely cover the space.

While this distinction is valid, the nomenclature conflicts with another interpretation of “continuous” data: that of a smooth valued function defined over space. From this perspective, all GIS data are discretized, and there is no continuous data structure represented by GIS (Kemp, 1993). This limitation is especially relevant to this research, which deals with continuous functions in the form of differential equations, and is compromised by having to use discretized GIS data structures. This discretization is an ever-present source of error. Purely analytical models, such as EPA's Hydrocarbon Spill Screening Model (HSSM, Weaver, et al., 1994), do not suffer from this limitation, and are able to define values continuously in space. Someday, perhaps GIS will be able to handle truly continuous data.

2.3 Environmental Modeling in GIS

With a growing public perception of imminent catastrophic changes to the global environment, a great deal of attention has been paid to environmental modeling and information analysis. Global climate studies have drawn together experts from such diverse disciplines as earth and environmental sciences, remote sensing, computer science, civil engineering, and public policy. As each learns more about the others' practice and techniques, new ideas and applications are born. With this interaction, the role of GIS in environmental modeling is that of a bridge between the numerically-oriented scientists and engineers and the socially-oriented policymakers.

In GIS, the unifying concept is one of location, extended to include spatial distribution and the relationships between spatial entities. In environmental modeling, the fundamental entity is one of state, modified by processes of

interaction and dynamics. Environmental modeling in a GIS requires an integration of these representations of reality. This integration takes several forms, from simple spatial analysis like determining buffer zones around GIS objects to more sophisticated representation of natural processes like flow and transport in porous media.

2.4 Spatial Analysis with GIS

Given its rich set of spatial functions, GIS is particularly well suited for analysis of spatial data. For example, Arc/Info and Grid support a suite of spatial interpolation functions for distance weighting, Thiessen tessellation, Delauney triangulation, moving least squares, and spatial statistical functions for splines, trends, and kriging. Other spatial analysis functions include volume modeling, slope and aspect determination, hill shading (rendering a surface lit from a given angle), and visibility analysis, which identifies all areas visible from a certain location and elevation, such as the top of a clock tower.

Through the use of AML scripts, Grid performs a variety of useful spatial analyses. For example, in locating a site for a dam, Grid automates calculation of reservoir volume and areal extent, showing affected areas. These analyses may include the cost analysis of inundated lands, and effects on demography and habitat. Most recently, a suite of hydrological functions has been developed to assist in delineation of watersheds, stream ordering, rainfall-runoff analysis, and calculation of the elusive time-area diagram, which was heretofore a far too laborious computation to be practical, but is a task well-suited to GIS (Maidment, 1993a). Network modeling is also enhanced by GIS, providing, for example, an automated solution of Dijkstra's algorithm (Dijkstra, 1959), used to determine the

shortest route connecting a series of locations. With the interfaces to global positioning system (GPS) data, analysis of land subsidence in the Sacramento Valley, California, has been done just by examining the GPS data (Blodgett, et al., 1990). With the advent of portable GIS, the registering of field data such as wellheads has become cheap and easy.

2.5 Modeling with GIS

The following sections discuss various styles of modeling with geographic information systems. In its simplest application, the GIS is used as a data base. As a pre-processor, it is useful in the preparation of datasets for use by external models, coupled by an input data file. Output files from the external model may also be subject to post-processing by the GIS in order to produce maps and other rich presentations of data. With more effort on the part of software designers, the GIS may be integrated into a modeling system, with the exchange of data between the components hidden from the user. Such integrated modeling tools are useful not only for the modelers themselves, but are often developed expressly for use by planners and decision-makers. At this level, the suite of tools is called a decision support system (DSS), commonly used in resource planning and management applications. The GIS is at its highest level of sophistication when it can be used as a modeling environment or platform, with models of process dynamics embedded into the language of the GIS itself (Fedra, 1993, Goodchild, 1993). The following sections discuss current trends in each of these levels of modeling.

2.5.1 GIS as a Data Base

The most common use of GIS is as a database for spatial data. For example, data for locations of water supplies are maintained on the scale of municipalities, counties, states, and nations. To be able to share these data in a common format and to communicate about them in a common language is essential to creating successful water management strategies, such as that proposed by the United States Environmental Protection Agency (EPA)'s Region III Groundwater Protection Section (Kerzner, 1989). In the realm of groundwater alone, GIS databases of well locations and hydrogeological properties and locations of aquifers, have been painstakingly compiled for the Great Lakes Basin (Warner, et al., 1991), the Culpeper Basin of Virginia (Nelms and Richardson, 1990), Cape Cod (Steppacher, 1988), Long Island (Haefner, 1992), the Appalachian Mountains (Swain, et al., 1991), the Italian Alps (Allewijn, 1986), the Santa Ana River Basin (Weghorst, et al., 1991), Deerfield Township, Michigan (Grossa, et al., 1989), the City of Albuquerque (Earp, 1987), the Counties of St. Lucie, Florida (Tan, 1991), Gosper, Phelps, and Kearney in Nebraska (Stansbury, et al., 1991), Clarke in Virginia (Lee and Christoffel, 1990), and Kalamazoo in Michigan (Kittleson and Kruska, 1987), the States of Arizona (Totman, 1989), Illinois (Hlinka and Shafer, 1989), Kansas (Juracek, 1992), Michigan (Brotten, et al., 1987), Mississippi (Mallory, 1990), Montana (Nielsen, et al., 1990), Nevada (Battaglin, 1989, Kilroy, 1989), Ohio and Indiana (Bugliosi, 1990), Pennsylvania (Kerzner, 1989), Rhode Island (Baker and Panciera, 1990), Wisconsin (Osborne, et al., 1987), and the entire countries of Germany (Wendland, et al., 1993) and Thailand (Faist, 1993).

The dataset for Germany is particularly extensive, containing complete coverages for mean annual precipitation, depth to groundwater, amount of evaporation, potential recharge, soil class, soil quality, average soil moisture capacity, aquifer porous media types, aquifer conductivity, effective average porosity, regional groundwater flow velocity, direction, and gradient, groundwater residence time, and several others on a 3 km grid. It is clear by this growing wealth of information that GIS has become a major tool in the archiving of environmental data.

The investigation of hazardous waste sites has also benefitted from the use of GIS. Gupta et al. (1989) write: "At a fraction of the cost of field and laboratory investigations, computerized data processing using geographic information systems and custom made software, as initiated at the Stringfellow¹ site, maximizes efficient interpretation, use, and storage of the database." GIS has also been used in ranking the priority of hazardous waste cleanup under EPA's Hazard Ranking System (HRS) (Bissex, 1991, Fitzsimmons, et al., 1988, Warner, et al., 1991.)

2.5.2 GIS as a Pre- and Post-Processor

Recently, geographic information systems have become prominent tools in model preparation and evaluation of modeling results (van der Heijde, 1992). The most common method of groundwater modeling with a GIS has been to couple an existing groundwater code with a spatial dataset via external files.

¹The Stringfellow Hazardous Waste site near Riverside has been marked by EPA as California's topmost priority hazardous waste site for cleanup under the federal Superfund program (Gupta, et al., 1989.)

Typically, the GIS is used as a data pre-processor to obtain, organize and clean datasets of aquifer properties, head elevations, well locations, contaminant concentrations, and the like. Once these data have been prepared, the GIS exports a file to be used as input to the modeling program. After this external program has performed its calculations, its output is reformatted into a form which can be imported back into the GIS for further spatial analysis and display. GIS excel in producing professional maps and overlays, which aid immensely in determining, for example, the extent of a contaminant plume with respect to critical natural and political boundaries.

Extensive use of GIS has been involved in the data preparation and map production for the DRASTIC model developed by the National Water Well Association. This is an external model which estimates groundwater pollution potential based on several spatially distributed geographic and hydrogeologic criteria. Maps of DRASTIC parameters have been generated for the States of Georgia (Trent, 1993), Texas (Halliday and Wolfe, 1991), Ohio (Petty and Hallfrisch, 1989), and southeastern Delaware (Evans and Myers, 1990), and Harvey County, Kansas (Whittemore, et al., 1987). Groundwater pollution potential programs similar to DRASTIC have been developed by Barrocu and Biallo (1993), who have integrated georeferenced geological, hydrogeological, and soil use data bases with vector and raster data from digital maps and remote sensing. Sokol, et al. (1993) linked Arc/Info and ORACLE to external programs via AML scripts to perform pollution potential analysis based on the filtration capacity of soils and a recharge rate based on volume balance of groundwater.

Data pre-processing to generate input files for external programs has also been automated for the U.S. Geological Survey (USGS) groundwater flow model MODFLOW (MacDonald and Harbaugh, 1988) for the Amsterdam Water Supply

(Olsthoorn, 1993) and for a groundwater flow model of the Middle Patuxent River Basin in Maryland (Hinaman, et al., 1993) by use of the MODELGRID AML, which makes Arc/Info coverages of the MODFLOW cells. A groundwater model for the Netherlands has been developed by coupling Arc/Info to AQ-FEM (Lieste, et al., 1993), and an atmospheric model for Mexico City has been generalized into the Geographic Information System for Atmospheric Modeling (SIGMA) by Reyes, et al. (1993). Data preparation for the Storm Water Management Model (SWMM) was done for Jefferson Parish, Louisiana by Barbé (1993), and has been automated into the SWMM.AML by Curtis (1992). Wilson, et al., (1992) have done similar work for the Chemical Movement through Layered Soils (CMLS) model, and French and Reed (1993) have studied GIS analysis of oil and chemical spills into aquatic systems. An extensive collection of coupled models has been assembled by D'Agnese, et al. (1993), including Arc/Info, Intergraph GIS, CPS-3, Stratamodel, LYNX, and IVM into the Hydrogeologic Framework Model used for regional groundwater flow system modeling at Yucca Mt., Nevada.

Pre-processing has also taken the form of generalized mesh generation for a variety of modeling programs. Kuniatsky and Lowther (1993), inspired by the Edwards Aquifer of central Texas, have developed an AML to generate a finite element mesh which accounts for placement of georeferenced objects relevant to the simulation, so that the mesh surrounding them has an appropriate geometry. Finite element mesh generation by a GIS was also done for a study of saltwater intrusion in Escambia County, Florida, by Roaza, et al. (1993). Similar work was done for a finite difference grid for MODFLOW by C.J. Richards, et al. (1993). By using Arc, polygons of any shape may be used, so it is useful in making

irregular grids for finite difference models — a feat still not possible with raster GIS systems.

2.5.3 GIS as an Integrated Modeling Tool

The coupled models described above may be effective at bringing several software tools together for geographic analysis, but they are generally still cumbersome to use, often requiring of the user an intimate knowledge of data file structure and the inevitable tricks of getting particular programs to run. Integrated models are the next step in sophistication, providing the user with one interface, which may be one of the programs involved in the system or a specially designed shell, from which other external models are run. The user is insulated completely from the intricacies of data file transfer and program quirks. The development of this integrated modeling system takes more time and effort than the simple coupled model, but the final product can save enough time and prevent enough aggravation that it is usually worth the effort.

MODFLOW, with its complex input file structure, is a good example. Links between the Arc/Info GIS and the USGS program MODFLOW have been developed by Orzol and McGrath (1992) and Watkins (1993). The links are written in Arc Macro Language (AML) and appear to be part of the GIS. From within the GIS, the user responds to graphical dialogs to organize information. The AML script generates a MODFLOW input file and runs the model as a spawned process. Since this is a separate process, one is free to return to the GIS to continue other work while MODFLOW executes in the background.

GIS has been similarly linked to other flow models (Kernodle and Philip, 1987, Van Metre, 1990) and to a flow path model (Brotten, et al., 1987). The

latter could now be entirely replaced by the ParticleTrack function in Arc/Info, developed as part of this research. Trent (1993) has developed an integrated tool for DRASTIC modeling for the State of Georgia, Mladenhoff, et al. (1993) have a forest dynamics model, and Craig (1993) has integrated the Intergraph GIS with the QUAL2E water quality model. The Agricultural Nonpoint Source (AGNPS) model has been linked to GIS by Yoon, et al. (1993) for study of water pollution and erosion in Minnesota and by Mitchell, et al. (1993) for runoff and sediment delivery in small watersheds in Illinois. Jankowski and Haddock (1993) have also integrated GIS and a nonpoint source pollution model. Srinivasan (1993) has linked the Geographic Resources Analysis Support System (GRASS, a public domain, open architecture GIS developed by the U.S. Army Construction Engineering Research Laboratory, 1991), to the Soil and Water Assessment Tool (SWAT). Buckley, et al. (1993) have integrated Arc/Info with an ecosystem management model. Fisher (1993) has developed a sophisticated GIS link to a three-dimensional groundwater code, and discusses the emergence of the geoscientific information system (GSIS).

2.5.4 GIS as a Decision Support System

If an integrated collection of models is designed for aiding in decision making, and especially if it has expert logic built in, it is called a decision support system (DSS). FEFLOW (Diersch, et al., 1992) is a highly polished graphical user interface (GUI) for finite element groundwater modeling which includes links to a GIS. GEO-WAMS (DePinto, et al., 1993) does the same for watershed analysis and modeling, with links to models like WASP4, and TABS (D.R. Richards, et al., 1993) is a GUI for finite element analysis of surface water flow.

DSS for water management of the Colorado River provides real time data on stage and irrigation gate status to farmers and bureaucrats alike (Reitsma, 1993). A spatial decision support system has been developed for spill concentration and travel times for the Ohio River Valley by linking Arc/Info NETWORK to EPA's WASP4 water quality model and the US Army Corps of Engineers' FLOWSED hydraulics model (Heath, et al., 1993). Djokic (1993) has constructed a generalized "smart" GIS by programming links between Arc/Info and the Nexpert Object expert system, using the HEC-1 rainfall-runoff model as an example for development. Burgin, (in progress) is designing a DSS for water allocation in Texas which integrates GIS, Nexpert Object, and a linear programming algorithm.

Additional work abounds, and the enthusiastic reader is referred to Cuddy, et al. (1993), Lam and Swayne (1993), Sen and Kelk (1993) Deckers (1993) and Nachtnebel, et al. (1993) for more examples.

2.5.5 GIS as a Modeling Environment

Within the GIS environment, models can be constructed from existing functions by writing macro scripts. These are collections of GIS commands which are written in a text file and executed by the GIS. Some GIS support a macro language, with conditionals and control statements similar to programming languages like BASIC. An example is Arc/Info's Arc Macro Language, which supports definition of variables, branching, conditionals, system calls, and even file input/output. AML has been enhanced to generate GUI tools like dialogs and menus, so that the programmer can create a user-friendly interface. This helps the user and the programmer as well, who can limit the user's access to

commands and file privileges. AML also supports Grid's map algebra and the DOCELL operator, which executes a specified block of commands for each cell in the grid. Using these tools, sophisticated models can be constructed completely within AML, and do not require a compiler or access to any of the GIS code development tools.

McKinney and Tsai (1993) have developed a multigrid solver for use in determining groundwater head fields using Grid and AML and have considered the effectiveness of using Grid to perform groundwater modeling. Hetrick, et al. (1993) produced SOLARFLUX, an insolation model which computes total direct solar radiation over a specified time interval. RHINEFLOW (Van Deursen and Kwadijk, 1993) is a water balance model for the Rhine River. AML is a useful if quirky language, and once mastered can produce fine results. But one is still limited to using those commands, functions, and operators which come with the GIS.

The "most elegant form of integration" (Fedra, 1993) involves embedding a model into the GIS, so that it is available as a native function which may be used at the command interface or from within a macro script. Batelaan, et al. (1993) have written a groundwater model with a deep level of integration to the GRASS GIS. Programmed in C, the groundwater code makes calls to GRASS library functions, and uses GRASS maps for data input and output. Output maps are generated for maps of groundwater levels, seepage rates, and flow directions and velocities, but are constrained by GRASS to integer data. Other embedded GRASS models are a complete stormwater runoff model by Vieux, et al. (1993) and a distributed hydrologic model by Saghafian (1993).

This research develops groundwater modeling functions for the Arc/Info Grid raster GIS, enabling it to derive velocity fields and perform advection-

dispersion modeling in porous media. These functions are true Grid functions, embedded into the Grid program by using the Arc/Info Software Development Libraries (SDL), available from ESRI. These libraries include all of the object code necessary to relink the executable code into a new version of Grid. The functions make calls to the Grid library and read and write Grid data structures. They can be used within AMLs to create complex groundwater models, with some examples given later in the dissertation. This research brings groundwater modeling in Arc/Info, which is the leading GIS worldwide, to an unprecedented level, and provides a springboard for further development of analytical functions for Grid by users outside of ESRI who wish to add to Grid their own specialized functions.

Chapter 3

Model Development

Model development is the process of turning ideas into functional entities such as computer programs. In thinking about phenomena and processes we imagine conceptual models – possible solutions or methods of investigation. In engineering, we then attempt to imitate the behavior of the conceptual model with mathematics. A mathematical model may be derived from theoretical first principles, it may be borrowed from analogous behavior in another area of study, or it may be found empirically or even by accident. Regardless of its source, it must be shown to accurately mimic the observed behavior. If a mathematical model is simple calculations can be done by hand, but a complex model requires a more sophisticated number cruncher such as a computer. Once a computer model of the phenomenon has been constructed, efficient analysis and exploration can begin. In this research, five new models are developed as functions for the Grid program, providing tools to assist the GIS modeler in groundwater transport analysis.

3.1 Overview of the Groundwater Modeling Functions

The Grid groundwater modeling functions developed here are intended to bring to GIS new tools for environmental analysis. The following needs are addressed: 1) the ability to generate a flow field from fields of head, transmissivity, porosity, and thickness, 2) methods for particle tracking through a flow field, and 3) calculation of concentration distributions resulting from advection-dispersion of point sources of constituents introduced into the flow field.

The first of these is accomplished by the DarcyFlow function, a function which determines the local flow velocity and direction for each cell in the grid. DarcyFlow applies Darcy's law to the cells adjacent to the center cell to calculate the fluid velocities through the cell walls. Doing this for each cell, a grid of velocities is generated: a flow field. A cell-based focal model is used here, since the input data fields (grids) are defined as cell-based discretizations of continuous fields. DarcyFlow differs from the existing Grid function FLOWDIRECTION, which determines which of the eight immediate cell neighbors receives flow from the center cell, effectively confining flow direction to one of eight values. DarcyFlow also calculates the volume balance residual for each cell, i.e. the difference between flow into and out of the cell.

The problem of particle tracking is addressed by either of two independent functions: ParticleTrack and DarcyTrack². Both of these generate a series of positions and travel times for a passively advected particle released from a given starting location in the flow field. Where the existing Grid function FLOWPATH generates a path from cell center to cell center, these functions determine a path which is independent of the cells. ParticleTrack approximates the flow path by interpolating the local velocity field (produced by DarcyFlow) from the nearest cell centers and advancing through the field by steps of a fixed length. DarcyTrack instead starts with the fields of head, porosity, thickness, and transmissivity and calculates analytically the streamline across each cell based on a bilinear interpolation of the head field, thus eliminating the intermediate step of flow field calculation. These functions cross the line between raster and vector

²DarcyTrack was developed principally as a verification of ParticleTrack, and has not been integrated into the Grid module of Arc/Info.

GIS, since they produce a series of line segments (vectors) from gridded (rasterized) data.

The dispersion of groundwater constituents is performed by the functions `PorousPuff` and `PorousPlume`³. These are each solutions to the advection-dispersion equation in uniform porous media, for an instantaneous and continuous point sources, respectively. Advection of the centroid of mass introduced at the source follows the flow path calculated by either `ParticleTrack` or `DarcyTrack`. Dispersion is modeled as a gaussian distribution around this center of mass, either as a dispersed puff (to borrow terminology from atmospheric transport) or as a plume. An analytical solution is employed for its simplicity and computational efficiency. The results are highly idealized and are suitable for rough estimates of the direction and character of solute transport. These functions both produce a grid of solute concentrations after advection and dispersion.

3.2 General Assumptions

Porous medium modeling assumes that for the range of scales which concern us we may model the medium as a continuum. The parameters used to describe the medium (porosity, permeability, etc.) are the results of spatial averaging over a representative elementary volume or REV (Bear, 1972). That is, even though we know that at the pore scale the medium is indescribably complex, at scales upwards of several centimeters the small variations are smoothed out and blended together to the extent that flow through the medium

³`PorousPlume` was developed after `DarcyFlow`, `ParticleTrack`, and `PorousPuff`, and will not be released coincident with them in Arc/Info version 7.0.

behaves as if the porous medium were instead a uniform medium of continuous permeability properties. This makes the problem analogous to that of heat conduction through a continuous material, such as a metal slab. In fact, these problems are solved by using the same mathematical models.

Unfortunately for the modeler, natural geologic materials are rarely homogeneous. Worse, the subsurface is inaccessible except through drilling holes or digging mines and excavations. The first two are expensive and provide only small windows of understanding. The third can give a complete description of the medium, but is destructive in doing so. So, like the paradox of Schrödinger's Cat⁴, we cannot know precisely the parameters of the problem without destroying the experiment. An understanding of process geology is essential in filling in the gaps between known outcrops and drilling sites, yet even the experienced geologist must make assumptions about what he doesn't see. One relies heavily on the observation that sedimentary formations are often roughly two-dimensional in geometry (or at least are created as such), so that lateral continuity dominates over vertical. Since most subsurface data sources are oriented vertically (such as well logs and seismic profiles), formation properties are interpolated between them. The interpolation may take many forms, from geostatistical analysis (used when there is little genetic information) to geological interpretation of the genesis and diagenesis of the materials.

⁴ In the field of quantum mechanics, the paradox of Schrödinger's Cat is often used as an illustration of Heisenberg's Uncertainty Principle: A cat is in a box with a bottle of instantly lethal poison, which would be overturned upon opening the box. It is impossible to determine if the cat in the box is currently alive or dead, since opening the box would kill it. The moral is: We cannot conduct an experiment without influencing the result, sometimes catastrophically.

3.2.1 Simplifying Assumptions

Even given detailed information about material properties for a given hydrogeological problem, several assumptions must be made in order to arrive at an analytical solution to the governing differential equations. Although these will be examined in more detail in subsequent chapters, it is useful to summarize them at this point.

Since the models developed in this research are intended to be a first attempt at groundwater modeling in GIS, these early developmental versions are conceptually simple. Depending on their success, more complex and sophisticated models may be developed in the future. In order to use the gaussian dispersion model for porous medium advection-dispersion, the flow and transmissivity fields are ideally steady-state, isotropic, and homogeneous. I will show later that these strict limitations may be relaxed under certain conditions without loss of integrity of the model, allowing greater flexibility in the range of problems addressed. For example, one may model transient flow fields by discretization into short time steps, and inhomogeneous transmissivity fields by appropriately refined spatial discretization. The requirement of isotropy does remain, however, since these models currently have no accounting for any type of anisotropy in the transmissivity field.

3.2.2 Limitations Imposed by Grid

Further limitations are imposed by the Grid data model. It is the hope of many GIS modelers that these limitations will be overcome as Grid matures into an environmental modeling environment. Grid is limited in its ability to

represent space to a rectangular two-dimensional domain of square cells of uniform size. The shape of the domain can be modified by the use of NODATA values for cell data, so that an irregular array of cells is bounded by the rectangle (the grid) and cells for which no data exist are assigned a special value well outside the range of normal data. Calculations involving grids still use all of the cells, however, which is some wasted effort on the part of the computer. The constraint of uniformly-sized square cells also results in wasted computer time, since the model mesh is just as fine in areas of minor interest as in those of intense interest. The modeling mesh is therefore a uniform grid of square cells, with the convention that in the case of a surface representation, cell values represent the value of the surface at the cell center. Although Grid data may be either integer or floating point, this research uses the latter exclusively.

The selection of a development platform and programming language was narrowed by the constraints of ESRI's development environment. Core code development was done on a 486-based personal computer, but the integration to Grid required a Sun Microsystems workstation running SunOS, with code encryption done by the native C compiler, cc. A new version of Grid was produced by linking my objects with the Grid objects supplied with ESRI's Software Development Library (SDL).

3.3 Porous Medium Flow and Transport

Flow modeling in porous media can involve dozens of parameters to characterize both the fluid and the medium, depending on the degree of detail of descriptive data. In the absence of detailed information, however, there are still a few essential properties of the materials involved which must be specified. As

discussed above, this research focuses on the simplest forms of flow and transport models, and so the number of parameters has been reduced to a minimum. For flow modeling using Darcy's law, the essential elements are the porous medium material properties of hydraulic conductivity (or transmissivity and saturated thickness), effective porosity, and the state variable of hydraulic head. These properties are defined continuously over the gridded two-dimensional domain.

It is important to make a distinction between saturated thickness and the thickness of the geologic formation. If the aquifer is completely saturated, these two values are the same, but an unconfined aquifer will have a saturated thickness less than the total thickness of the transmissive unit. In steady-state modeling, the saturated thickness will remain constant over time. In any case, what is critical to the flow modeling is the thickness of the flowing, saturated part of the medium. This value should always be used with these functions.

Effective porosity n is that void space within the material which contributes to flow. This excludes pores which are isolated from the flow and those, which while in fluid connection, do not involve fluid flow, such as dead-end pores. This is not to be confused with the effective porosity used in the context of specific yield (Bear, 1972).

While porosity is a property of the medium alone, the hydraulic conductivity K is dependent not only on the material's intrinsic permeability but on the density and viscosity of the fluid as well. We will assume that fresh water is the transporting fluid, and that its density and viscosity are constant throughout the domain. The hydraulic conductivity field for a uniform fluid varies only with the intrinsic permeability, making it in effect a porous medium property. Since the transmissivity T is more relevant to formation-scale transport, it is used with the saturated thickness and effective porosity to characterize the geologic materials.

Peizometric head h (hereafter called simply head) describes the sum of pressure and potential energies of the fluid per unit weight. In unconfined aquifers, the head is reflected in the elevation of the water table (the top of the saturated zone). In saturated media, h is measured by the elevation to which water rises in a well which penetrates the aquifer. A fully screened and penetrating well will reflect a head value averaged over the thickness of the formation.

The GIS description of the geologic medium is reduced to a few important spatially distributed parameters: transmissivity to a particular fluid, saturated thickness and effective porosity of the formation, and head. The problem can then be abstracted to a slab of resistive material permitting fluid flow within it. Darcy's law, discussed below, can be used to solve for head values and fluid fluxes within the slab with the additional constraint of the conservation of mass.

With these simplifications, transport modeling likewise may be reduced to a few critical parameters. Advective transport obviously requires a fluid velocity field and a time of interest. The gaussian dispersion model discussed below requires dispersivity values to scale the variance of the gaussian distribution. These dispersivity coefficients may be related to the time or distance of travel as well, which are related in the flow velocity field. In addition to the dispersivity, Contaminant transport uses retardation and decay of a constituent carried in the flow field. The implementation of all of these factors in the Grid functions is discussed in detail later in this chapter.

3.3.1 Continuity

We are interested in describing the flow within porous media, involving the flow vectors, head values, and volumetric mass balance. The usual formulation of a groundwater problem is to prescribe boundary conditions and solve for the flow vectors and head values. Assuming that the density and viscosity of the fluid are constant, this involves three equations (the three components of Darcy's law) and four unknowns (the specific discharge components and the head value). A fourth equation is available from the principle of conservation of mass, which states that the time rate of change of the mass of a substance within a control volume equals the net flux of mass into the volume plus the rate of production of mass within the volume. For a confined aquifer, the continuity equation takes the form (Bear, 1972):

$$S \frac{\partial h}{\partial t} + \nabla \cdot \mathbf{U} = 0 \quad (1)$$

where

S \equiv coefficient of storage, defined as that volume of water released from a vertical column of aquifer of unit horizontal area, per unit decline in peizometric head (dimensionless)

h \equiv peizometric head (units of length)

t \equiv time (units of time)

\mathbf{U} \equiv the aquifer flux vector, representing the volumetric flow rate across a unit thickness of aquifer per unit time, normal to the surface

$\nabla \cdot \mathbf{U}$ is the divergence of the aquifer flux (\mathbf{U} has units of length²/time, or length³/length time)

For an unconfined aquifer, we write (Bear, 1972):

$$S_y \frac{\partial H}{\partial t} + \nabla \cdot \mathbf{U} = W \quad (2)$$

where

S_y \equiv specific yield, defined as that volume of water released from a vertical column of aquifer of unit horizontal area, per unit decline in the phreatic surface (dimensionless)

H \equiv head (units of length)

W \equiv rate of recharge and other net fluxes into the aquifer (units of length/time)

In the case of steady flow, both $\partial h/\partial t$ and $\partial H/\partial t$ are zero. If we disallow pumping, recharge, and leakage to and from other aquifers, W is also zero, and the equations both reduce to $\nabla \cdot \mathbf{U} = 0$. With the addition of initial and boundary conditions, the heads and flux vectors can be determined, generally by numerical techniques. The problem of solving for a field of heads given material properties and initial and boundary conditions may be computationally intensive, and so is not appropriate for modeling within the GIS itself (McKinney and Tsai, 1993).

For the purposes of this research, I have posed the problem differently. Assuming that the heads are known (either from field data or by application of an iterative method), we solve for the fluid fluxes and the source terms. This calculation requires no iteration and so is fast on the computer.

3.3.2 Darcy's Law

Darcy's law was discovered as a governing equation which predicts average fluid volume flux rates through porous media. The proportionalities it expresses

were first investigated by Henri Darcy while studying the behavior of flow through packed columns of sand (Darcy, 1856), and are concisely expressed as

$$Q = -\frac{K A \Delta h}{L} \quad (3)$$

where

Q \equiv rate of flow through the column (volumetric discharge per unit time) (units of length³/time)

K \equiv a proportionality constant, defined as the hydraulic conductivity (units of length/time)

A \equiv cross-sectional area (units of length²)

Δh \equiv difference in hydraulic head as measured at each end of the column (units of length)

L \equiv length of the column (units of length)

Further, the specific discharge (or Darcy flux) q is the volumetric discharge divided by the cross-sectional area, $q = Q/A$, so that

$$q = -\frac{K \Delta h}{L} \quad (4)$$

These equations reflect the one-dimensionality of Darcy's experiments, but they can be modified in several ways to work on a two-dimensional grid of values for head h , transmissivity T , thickness b , and effective porosity n . To make use of the variation in thickness, the hydraulic conductivity is replaced by transmissivity and thickness using the relation $T = K \cdot b$. By dividing the right-hand side of equation (4) by the material's effective porosity, the fluid flux q is converted to a fluid or seepage velocity $v = q/n$, which represents the average velocity of the fluid moving through the pores. Darcy's law must also be extended to two dimensions in order to calculate fluid fluxes through the side walls of a

rectangular cell which extends over the thickness of an aquifer. We are interested in calculating the fluid flux between cells adjacent to a central cell, that is, the flux through each face, as shown in Figure 1.

The head gradient between adjacent cells is estimated by a difference in heads assigned to the two cells, divided by the distance between cell centers, as shown in Figure 2. Since the values for T , b , and n are also defined only at the grid cell centers, values at the cell walls are estimated by averaging. The intercell transmissivity is approximated by the harmonic average⁵ of the transmissivities of the two adjacent cells, which produces an equivalent transmissivity for flow

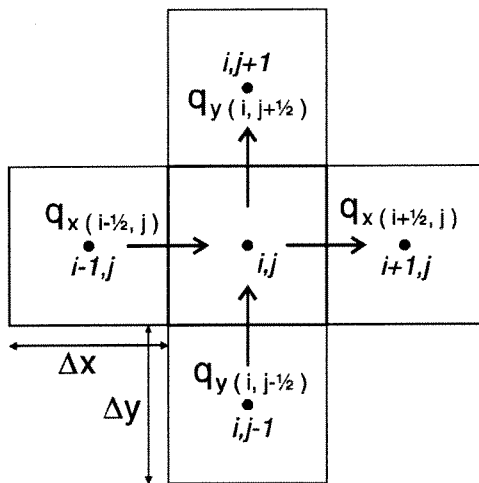


Figure 1 Darcy's law is used to calculate fluid fluxes into adjacent cells.

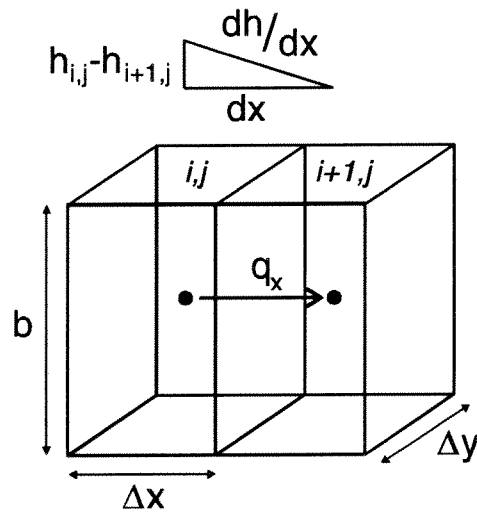


Figure 2 Darcy's Law applied to one side of a grid cell.

⁵If a series l of blocks of varying hydraulic conductivity are in series with the flow (flow crosses from one block into the other), the effective conductivity is given by the law of harmonic composition (Marsily, 1986):

$$\frac{\sum l_i}{K_{\text{mean}}} = \sum \frac{l_i}{K_i}$$

orthogonal to the cell wall. Both cell thicknesses and porosities are averaged arithmetically.

The following equations summarize the mathematical calculation of the volumetric discharge Q through the cell walls for an individual cell based on the local and surrounding four values of head and averaged transmissivity:

$$\begin{aligned}
 Q_{x(i-\frac{1}{2},j)} &\doteq - \frac{2 T_{i-1,j} T_{i,j}}{T_{i-1,j} + T_{i,j}} (h_{i,j} - h_{i-1,j}) \frac{\Delta y}{\Delta x} \\
 Q_{x(i+\frac{1}{2},j)} &\doteq - \frac{2 T_{i,j} T_{i+1,j}}{T_{i,j} + T_{i+1,j}} (h_{i+1,j} - h_{i,j}) \frac{\Delta y}{\Delta x} \\
 Q_{y(i,j-\frac{1}{2})} &\doteq - \frac{2 T_{i,j-1} T_{i,j}}{T_{i,j-1} + T_{i,j}} (h_{i,j} - h_{i,j-1}) \frac{\Delta x}{\Delta y} \\
 Q_{y(i,j+\frac{1}{2})} &\doteq - \frac{2 T_{i,j} T_{i,j+1}}{T_{i,j} + T_{i,j+1}} (h_{i,j+1} - h_{i,j}) \frac{\Delta x}{\Delta y}
 \end{aligned} \tag{5}$$

The corresponding darcy flux q is calculated by dividing Q by the area of the cell wall. For example,

$$q_{x(i+\frac{1}{2},j)} = \frac{Q_{x(i+\frac{1}{2},j)}}{b_{i+\frac{1}{2},j} \Delta y} \tag{6}$$

When q is divided by the averaged effective porosity of the cells, we obtain the x and y components of the average seepage velocity vector v :

$$\mathbf{v}_{x(i\pm 1/2,j)} = \frac{q_{x(i\pm 1/2,j)}}{n_{i\pm 1/2,j}} \quad \mathbf{v}_{y(i,j\pm 1/2)} = \frac{q_{y(i,j\pm 1/2)}}{n_{i,j\pm 1/2}} \quad (7)$$

The average x and y components of the flow velocity vector for the cell are therefore:

$$\mathbf{v}_x(i,j) = \frac{1}{2} \left[Q_{x(i-1/2,j)} \frac{1}{\Delta y} \left(\frac{2}{b_{i-1,j} + b_{i,j}} \right) \left(\frac{2}{n_{i-1,j} + n_{i,j}} \right) + Q_{x(i+1/2,j)} \frac{1}{\Delta y} \left(\frac{2}{b_{i,j} + b_{i+1,j}} \right) \left(\frac{2}{n_{i,j} + n_{i+1,j}} \right) \right] \quad (8)$$

$$\mathbf{v}_y(i,j) = \frac{1}{2} \left[Q_{y(i,j-1/2)} \frac{1}{\Delta x} \left(\frac{2}{b_{i,j-1} + b_{i,j}} \right) \left(\frac{2}{n_{i,j-1} + n_{i,j}} \right) + Q_{y(i,j+1/2)} \frac{1}{\Delta x} \left(\frac{2}{b_{i,j} + b_{i,j+1}} \right) \left(\frac{2}{n_{i,j} + n_{i,j+1}} \right) \right]$$

Grid uses the directional convention of geographic coordinates, so a vector should be expressed in direction (degrees from north) and magnitude. This conversion is

$$\mathbf{v}_{\text{mag}} = \sqrt{\mathbf{v}_x^2 + \mathbf{v}_y^2} \quad \mathbf{v}_{\text{dir}} = 90 - \tan^{-1} \left(\frac{\mathbf{v}_y}{\mathbf{v}_x} \right) \cdot \frac{180}{\pi} \quad (9)$$

Two grids are generated to store values of these two components, and these two grids are used together to describe the vector field.

Recall that fluid flow in a porous medium is constrained by continuity: Water cannot be spontaneously produced or destroyed, and the net mass flux into

the control volume must equal the change in mass within the volume. For steady flow, $\partial h/\partial t = 0$ and $\partial H/\partial t = 0$ and equation (2) (page 37) becomes

$$\nabla \cdot \mathbf{U} = \mathbf{W}. \quad (10)$$

If we consider the volumetric fluid flux Q (L^3/T) across the cell, this fluid flow can be expressed as a volume. To impose this equation on a gridded domain, let the control volume be the two-dimensional grid cell. The divergence of the flux vector may then be expressed in a discretized form as

$$Q_{x(i-\frac{1}{2},j)} - Q_{x(i+\frac{1}{2},j)} + Q_{y(i,j-\frac{1}{2})} - Q_{y(i,j+\frac{1}{2})} = \mathbf{W} \cdot \Delta x \cdot \Delta y. \quad (11)$$

This equation is used to calculate the volume balance residual in the cell, i.e. the source strength times the area of the cell: $\mathbf{W} \cdot \Delta x \cdot \Delta y$. If this value is nonzero, either the head field is inconsistent with the material properties or there are sinks or sources within the cell. Sources and sinks may be local wells or recharge and discharge to the surface or another aquifer.

3.3.3 The DarcyFlow Function

Now that the mathematical model for the DarcyFlow function has been developed, it must be coded into a computer program. For integration with the Grid data model, the coding must work with data elements on a regularly-spaced grid (Figure 3), rather than on an abstract (x,y) plane.

The DarcyFlow function is a straightforward exercise in programming for Grid. Since it is a cell-by-cell calculation which depends on only the nearest four neighbors (see Figure 1), it can be executed by looping through the same procedure for each row and each column of the rectangular grid (except for the

boundary cells). For each cell, the values for T , b , n , and h for the center cell and its four neighbors may be read from the grid files on disk, using the `GetWindowCell` function. In general, it is important to allow for grids of arbitrary size, and they may contain millions of cells. For this reason, programs using Grid data files cannot read the entire grid into memory. As an example, DarcyFlow uses seven grids at one time. Even a modest geographic dataset may contain a million cells (as for a 1000 x 1000 cell grid). With values stored as 4 byte floating point variables, we would need 4MB per grid, or

28MB total to run DarcyFlow, in addition to memory occupied by the operating system, Arc, Grid, and other programs. A large dataset stored in memory is beyond the capability of most computers, so data are read "on the fly". As a compromise, another subroutine is available in the Grid library: `GetWindowRow` retrieves an entire row of values from a grid, and its companion function, `PutWindowRow`, writes a row of values to disk. This will rarely overload memory, and enhances performance greatly by reducing the number of disk reads and writes.

Once the volume balance residual and the flow vector component values are calculated (following the equations presented above), they are stored in row

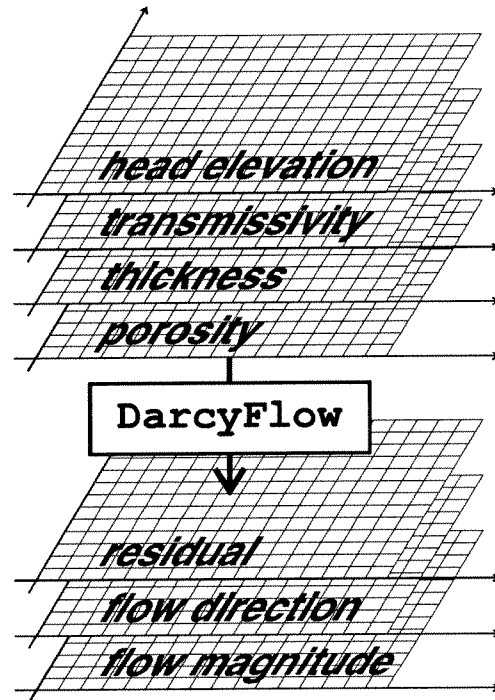


Figure 3 Input and output grids for DarcyFlow.

structures and later written to their respective grids (residual, flow direction, and flow magnitude) with the `PutWindowRow` function.

A special case must be considered for cells which border the grid. There is no information available outside the grid, so a true residual and velocity cannot be estimated. Border cells in the residual grid are assigned the value of `NODATA`, which in Grid generally means that it is unknown. For the vector direction and magnitude grids, `DarcyFlow` assigns each border cell the values of its next interior neighbor. In a smooth field, this will help to preserve the character of the field near the edges of the domain. In a highly variable field, this smooth border may be uncharacteristic, in which case the edge cells could be masked or otherwise ignored.

3.4 Advection Through the Flow Field

Once the steady flow field has been calculated by `DarcyFlow`, we are interested in calculating advection paths through this field, the next step towards advection-dispersion modeling of solute transport. Advection of a fluid particle⁶ through a flow field can be analyzed from various perspectives. A Lagrangian approach is to follow the trajectory of a particular fluid particle through time. The locus of positions in time and space is the pathline for the particle. In an unsteady flow field, two particles which are tracked from the same spatial location but at different times may have different path lines. By contrast, streamlines are an Eulerian concept, defined as the instantaneous curves that are

⁶The concept of a fluid particle arises from the continuum approach to modeling porous media. It is useful to imagine the fluid particle as a small volume of fluid. The motion of the particle is the average motion of the individual fluid elements which make up that volume.

at every point tangent to the direction of the velocity at that point (Bear, 1972). In a steady flow field, the pathline and streamline are coincident. For the flux vector \mathbf{q} and an element of length $d\mathbf{s}$ along the streamline, the equation for the streamline comes from the condition that these vectors are parallel, or that their cross product vanishes:

$$\mathbf{q} \times d\mathbf{s} = \mathbf{0}, \quad (12)$$

or

$$q_y dx - q_x dy = 0. \quad (13)$$

Solving this equation produces a family of streamlines through the flow field. In the case of incompressible flow with no distributed sources or sinks, one may also define a stream function $\Psi(x,y)$, which takes a constant value along a streamline. The advection of a fluid particle follows a streamline, a property used by the DarcyTrack program.

3.4.1 Numerical Approximation of Streamlines

Particle tracking in a steady uniform flow field is a trivial matter, consisting of simply calculating a new position in the direction of flow at a distance $L = v \cdot t$, where v is the magnitude of the flow vector. In a nonuniform flow field, the calculation is more difficult. If we are interested in tracking a fluid particle from a known position, then we are faced with an initial value problem in a domain which is a solution (the flow field) to an ordinary differential equation (Darcy's law) constrained by continuity. The initial location provides the constant value of $\Psi(x,y)$.

A standard numerical approach to solving this type of problem is to use a Runge-Kutta technique. This is a family of techniques which use local values

of the derivative(s) of a function to predict subsequent values of the function. From an initial value, one can then estimate the entire function. Various Runge-Kutta formulations use different degrees of derivatives (truncating a Taylor series expansion) and weight them in different ways. We will address two simple forms here: the first- and second-orders otherwise known as the Euler and Heun methods, respectively (Chapra and Canale, 1988).

Consider a head field defined at grid cell centers over a two-dimensional domain in cartesian coordinates x and y . The derivative of the head field corresponds proportionally to the vector flow field via Darcy's law since the material transmissivity is isotropic. The ParticleTrack function uses the flow velocity vector field to propagate a fluid particle along its streamline. This field can be made continuous by performing a bilinear interpolation between cell center values.

Given an initial point P_i , the location of another point P_{i+1} along the streamline can be approximated by a Taylor series expansion:

$$P_{i+1} = P_i + h_i' L + \frac{h_i''}{2} L^2 + \frac{h_i'''}{3!} L^3 + \dots + \frac{h_i^{(n)}}{n!} L^n \quad (14)$$

where L is a specified step length (which should be smaller than the dimension of a grid cell). If our surface interpolation is bilinear, all derivatives but the first are zero, and equation (14) reduces to

$$P_{i+1} = P_i + h_i' L \quad (15)$$

This is Euler's method, as illustrated in Figure 4. This first-order Runge-Kutta estimate involves starting at a point P_i for the particle and interpolating a flow vector v_i from the surrounding values. Using this flow and a specified step length L , a new position P_{i+1} is located at a distance L from P_i , in the direction

of v_i , which is also the direction of h_i' in an isotropic transmissivity field. The time required to complete this step is simply L/v_i . The process is repeated to obtain positions P_{i+1} , P_{i+2} , P_{i+3} , ..., until some ending criterion is reached (either a requested time is reached, or the particle has migrated into a depression or out of the domain). However, in all but a nearly unidirectional flow field, this estimated path will quickly deviate from the true path.

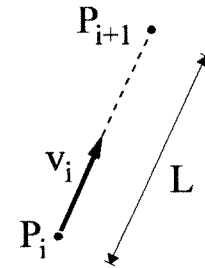


Figure 4 Particle tracking by Euler's method.

A much better estimate can be achieved by Heun's Method, which involves an additional calculation. Let the initial vector v_i at P_i be called the predictor, and let another vector v' be the value of the flow field at point P' , which would be P_{i+1} in Euler's method (see Figure 5). This v' is the corrector. An average of these two vectors, v'' , is used to find the point, P_{i+1} , located at a distance L from P_i , in the direction of v'' . P_{i+1} is then the next point in the path, and the process is repeated to find P_{i+2} , etc. Heun's method is actually a second order Runge-Kutta method with equal weighting given the predictor and the corrector, and produces a much better result than the Euler scheme. Comparison of Heun's method to the semi-analytical technique presented in the next section shows that it produces nearly identical results for smooth flow fields, and is computationally more efficient.

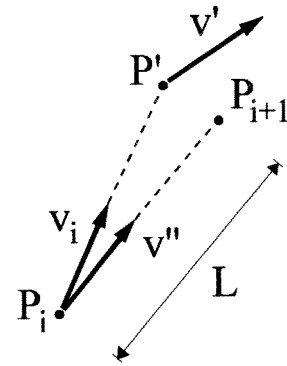


Figure 5 Particle tracking by Heun's method.

ParticleTrack, one of two path generation functions developed for Grid, generates a streamline by applying Heun's method to an existing vector field.

The computer implementation of this model brings a new concept to Grid in that the output is a series of line segments that are in no way tied to the position or spacing of the grid. Gridded data representing flow velocity vectors are used for input data, but the output streamline segments are free to float through the (x,y) space of the grid. The output data are in the form of both a text file and an Arc line coverage. The file contains the (x,y) locations of the beginning points of each segment, the cumulative length and time of the path, and the direction and magnitude of the vector segment. While this file is useful for examining the data, its primary purpose is for data exchange with the PorousPuff and PorousPlume transport functions described below. The Arc coverage contains attributes for these same values, for integration with the Arc/Info system.

As the path migrates from the source point, local data are needed to interpolate the flow vector from the flow direction and magnitude grids. Since only the four nearest cell values are needed, the `GetWindowCell` function is used rather than `GetWindowRow`. These values are interpolated with the `GridVecInterp` function, part of the `INTERP.C` module described in the Appendix. New values are read as needed as the path proceeds across the grid until either the user-specified ending time is reached, the edge of the grid is encountered, or the path is determined to enter a depression, which may exist in the presence of a sink such as a well. In the case of a well, the volume balance residual grid will indicate a negative balance equal to the flow out of a production well.

`ParticleTrack` must continually check for the possibility of a depression. Since each path step is of the same fixed and finite length, the predicted vector will oscillate or “bounce” around the lowest point in the depression. Geometrical analysis shows that as the path bounces, the angle subtending two adjacent segments will be quite acute, and alternate endpoints will be less than one

segment length apart. ParticleTrack tests each pair of segments for this property, and reports a depression if several sequential bounces are found, terminating the path generation.

3.4.2 Analytical Approximation of Streamlines

Particle tracking with the numerical methods described in the previous section is admittedly rather inelegant (if computationally efficient), and a much more satisfying streamline calculation can be derived analytically. It is a piecewise analytical solution to the streamline over a surface, solved over sections of a surface interpolated from a regularly gridded potential or head field. The interpolated surface is defined by a bilinear interpolation from head values specified at the four cell corners. Each cell also uses constant values for porosity, thickness, and transmissivity over its area. Given a starting point either interior to the cell or on an edge (as for a particle emerging from a neighboring cell), a streamline can be determined analytically, resulting in an exit point for the path, a path length, and a travel time over that length. Pollock (1988) has developed an algorithm for calculating streamlines on a finite difference grid using cell face fluid velocities as a primary data source, and Cordes and Kinzelbach (1992) have developed a similar solution for finite elements. We could apply that approach to the fluxes calculated in the DarcyFlow function, but since those fluxes are ultimately based on the heads, calculations can be reduced by working directly from the head field data. It should also be noted that since the transmissivity field is isotropic in this model, the streamline will follow the head gradient.

To define a generic problem of tracing a streamline over an interpolated surface, let us consider a cell extending over a rectangle in cartesian space, so that

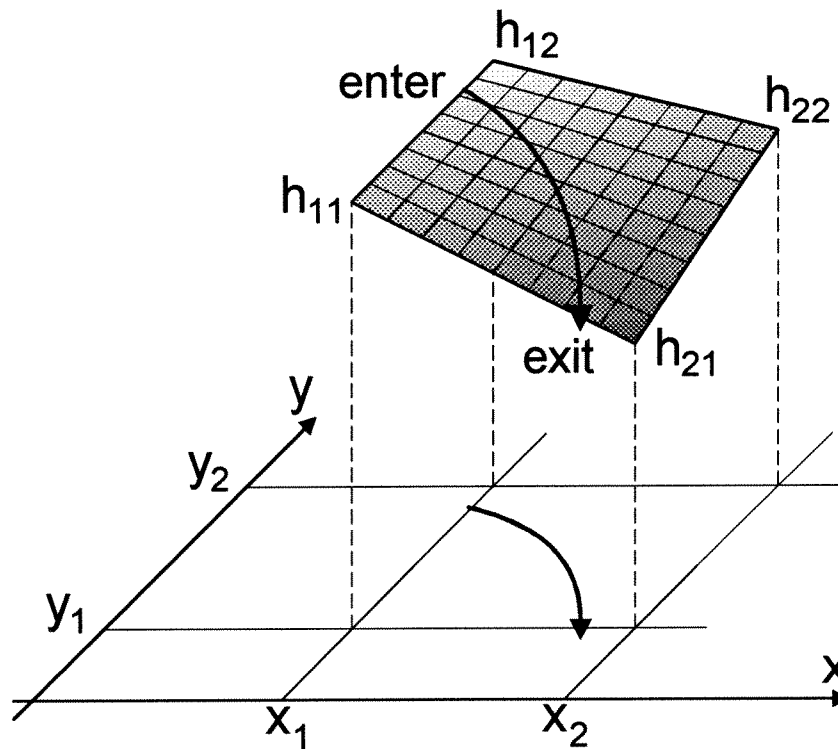


Figure 6 Interpolated bilinear surface over a cell.

the four edges are defined by x_1 , x_2 , y_1 , and y_2 as shown in Figure 6. (Note that this is different from the cell used in the Grid model, which has values assigned at the center — a point we will return to later.) A potential or head surface is interpolated from the head values at the four corners using a straightforward bilinear formula:

$$\begin{aligned}
h(x,y) = & \left(1 - \frac{x-x_1}{x_2-x_1}\right) \left(1 - \frac{y-y_1}{y_2-y_1}\right) h_{11} + \left(1 - \frac{x-x_1}{x_2-x_1}\right) \left(\frac{y-y_1}{y_2-y_1}\right) h_{12} \\
& + \left(\frac{x-x_1}{x_2-x_1}\right) \left(1 - \frac{y-y_1}{y_2-y_1}\right) h_{21} + \left(\frac{x-x_1}{x_2-x_1}\right) \left(\frac{y-y_1}{y_2-y_1}\right) h_{22}
\end{aligned} \tag{16}$$

With the appropriate substitutions, this can be seen to have the general form of a surface defined by

$$h(x,y) = a_0 + a_1 x + a_2 y + a_3 xy \tag{17}$$

where

$$\begin{aligned}
a_0 &= \frac{x_2 y_2 h_{11} - x_2 y_1 h_{12} - x_1 y_2 h_{21} + x_1 y_1 h_{22}}{x_2 y_2 - x_2 y_1 - x_1 y_2 + x_1 y_1} \\
a_1 &= \frac{-y_2 h_{11} + y_1 h_{12} + y_2 h_{21} - y_1 h_{22}}{x_2 y_2 - x_2 y_1 - x_1 y_2 + x_1 y_1} \\
a_2 &= \frac{-x_2 h_{11} + x_2 h_{12} + x_1 h_{21} - x_1 h_{22}}{x_2 y_2 - x_2 y_1 - x_1 y_2 + x_1 y_1} \\
a_3 &= \frac{h_{11} - h_{12} - h_{21} + h_{22}}{x_2 y_2 - x_2 y_1 - x_1 y_2 + x_1 y_1}
\end{aligned} \tag{18}$$

A particle travelling across this surface will follow a streamline along the negative gradient since by definition the streamline is everywhere tangent to the flow vector. Recall the definition of a streamline presented in equation (12):

$$\mathbf{q} \times d\mathbf{s} = 0 \quad (19)$$

and

$$q_y dx - q_x dy = 0. \quad (20)$$

By rearranging terms to collect like variables, $\frac{dx}{q_x} = \frac{dy}{q_y}$, and it follows that

$$\frac{dx}{-\frac{\partial h}{\partial x}} = \frac{dy}{-\frac{\partial h}{\partial y}}. \quad (21)$$

A useful property of the bilinear surface is that the derivative in x is solely a function of y and vice versa: $\frac{\partial h}{\partial x} = \frac{\partial h}{\partial x}(y)$ and $\frac{\partial h}{\partial y} = \frac{\partial h}{\partial y}(x)$. These partial

derivatives are proportional to the components of the flux vector,

$$-q_x \sim \frac{\partial h}{\partial x} = a_1 + a_3 y \quad -q_y \sim \frac{\partial h}{\partial y} = a_2 + a_3 x. \quad (22)$$

By integrating equation (21) and substituting, we see that

$$\int -\frac{\partial h}{\partial y}(x) dx = \int -\frac{\partial h}{\partial x}(y) dy \quad (23)$$

$$\int -(a_2 + a_3 x) dx = \int -(a_1 + a_3 y) dy.$$

Solving the integrals produces the equation for the family of streamlines across the head surface:

$$a_2 x + a_3 \frac{x^2}{2} = a_1 y + a_3 \frac{y^2}{2} + C \quad (24)$$

The constant C is determined given a known (x,y) point which lies along a particular streamline. The stream function $\Psi(x,y)$ is constant along the streamline:

$$\Psi(x,y) = a_2 x + a_3 \frac{x^2}{2} - a_1 y - a_3 \frac{y^2}{2} = C. \quad (25)$$

However, we are interested in more than just the equation of the location of streamline. The length of the path contained within a cell is necessary, as is the travel time across the cell, since we will later use the cumulative length and time along the path. A derivation for these calculations follows. The length of the curved line segment can be calculated as an integral with respect to either x or y , depending on the uniqueness of the function. Since the two formulations are completely analogous, we will examine only one, with the curve expressed as the stream function Ψ in terms of x : $\Psi_y(x)$. Solving equation (24) in terms of $y(x)$ gives two roots:

$$\Psi_y(x) = y(x) = \frac{1}{a_3} \left(-a_1 \pm \sqrt{a_1^2 + 2a_2 a_3 x + a_3^2 x^2 - 2a_3 C} \right). \quad (26)$$

Although these are two distinct curves, the length calculation produces the same result for both⁷, so only one is used here:

$$\Psi_y(x) = \frac{1}{a_3} \left(-a_1 + \sqrt{a_1^2 + 2a_2 a_3 x + a_3^2 x^2 - 2a_3 C} \right). \quad (27)$$

In terms of x , the length $L(x)$ of the curve $\Psi_y(x)$ is given by

⁷The length calculations are the same, since the length integral involves the square of the differentials, which differ only in sign.

$$L(x) = \int_{x_{\text{enter}}}^{x_{\text{exit}}} \sqrt{1 + \left(\frac{d\Psi_y}{dx}\right)^2} dx. \quad (28)$$

The derivative of equation (27) is

$$\frac{d\Psi_y}{dx} = \frac{2a_2a_3 + 2a_3^2x}{2a_3\sqrt{a_1^2 + 2a_2a_3x + a_3^2x^2 - 2a_3C}}. \quad (29)$$

By substituting the derivative into the integrand, $L(x)$ becomes

$$L(x) = \int_{x_{\text{enter}}}^{x_{\text{exit}}} \sqrt{1 + \left(\frac{2a_2a_3 + 2a_3^2x}{2a_3\sqrt{a_1^2 + 2a_2a_3x + a_3^2x^2 - 2a_3C}}\right)^2} dx. \quad (30)$$

This integral has no closed form, and must be approximated numerically. The analogous solution for the stream function in terms of y , $\Psi_x(y)$, is

$$L(y) = \int_{y_{\text{enter}}}^{y_{\text{exit}}} \sqrt{1 + \left(\frac{2a_1a_3 + 2a_3^2y}{2a_3\sqrt{a_2^2 + 2a_1a_3y + a_3^2y^2 + 2a_3C}}\right)^2} dy. \quad (31)$$

The true length is the larger absolute value of $L(x)$ and $L(y)$, since the shorter is the result of nonuniqueness. While there are cases where the two lengths are the same, in general the curve will have a unique $x(y)$ or $y(x)$ but not both. For example, the curve in Figure 8 (page 58) is unique in $y(x)$ but not $x(y)$. An integration from x_{enter} to x_{exit} will produce the correct length, but an integration from y_{enter} to y_{exit} will not.

In order to calculate the travel time of a particle across the cell, we need to understand its velocity function. Although T , b , and n are constant over the cell, the velocity is not, since it varies with the gradient. In general, the x component of the velocity vector is

$$V_x(x,y) = -\frac{T}{bn} \left(\frac{\partial}{\partial x} h(x,y) \right), \quad (32)$$

with an analogous equation for the y component. The travel time with respect to x , $t(x)$, is determined by integrating the inverse of the velocity over the length of the path segment and substituting the partial derivative from equation (22):

$$t(x) = \int \frac{1}{v_x(x,y)} dS = \int_{x_{\text{enter}}}^{x_{\text{exit}}} \frac{1}{\frac{T}{bn} \left(\frac{\partial}{\partial x} h(x,y) \right)} dx = \int_{x_{\text{enter}}}^{x_{\text{exit}}} \frac{1}{\frac{T}{bn} (a_1 + a_3 y)} dx. \quad (33)$$

Since we can write y in terms of x using equation (27), a substitution makes $t(x)$ a function of x alone:

$$t(x) = \int_{x_{\text{enter}}}^{x_{\text{exit}}} \frac{1}{\frac{T}{bn} \left[a_1 + a_3 \left(\frac{1}{a_3} \left(-a_1 + \sqrt{a_1^2 + 2a_2 a_3 x + a_3^2 x^2 - 2a_3 C} \right) \right) \right]} dx \quad (34)$$

$$t(x) = \int_{x_{\text{enter}}}^{x_{\text{exit}}} \frac{bn}{T} \left(\frac{1}{\sqrt{a_1^2 + 2a_2 a_3 x + a_3^2 x^2 - 2a_3 C}} \right) dx.$$

This integral equation does have a closed form, and the travel time can be calculated directly from its solution,

$$t(x) = \frac{bn}{T} \frac{1}{a_3} \ln \left[\sqrt{a_1^2 + a_3^2 x^2 + 2a_2 a_3 x - 2a_3 C} + a_3 x + a_2 \right] \Bigg|_{x_{\text{enter}}}^{x_{\text{exit}}}, \quad (35)$$

or the analogous equation in y ,

$$t(y) = \frac{bn}{T} \frac{1}{a_3} \ln \left[\sqrt{a_2^2 + a_3^2 y^2 + 2a_1 a_3 y + 2a_3 C} + a_3 y + a_1 \right] \Bigg|_{y_{\text{enter}}}^{y_{\text{exit}}}. \quad (36)$$

The true travel time is determined by a real solution of equation (35) or (36). As long as the argument of the logarithm is real and positive, a correct result exists, so one of these equations will always produce the correct result.

The methods presented above work well in the general case, but result in singularities in the case of a planar surface. For planes, $a_3 = 0$, and the head surface equation becomes

$$h(x,y) = a_0 + a_1 x + a_2 y. \quad (37)$$

Following a derivation completely analogous to that of the general case, the family of (straight) streamlines is the solution of

$$a_2 x = a_1 y + C. \quad (38)$$

Finding the exit point involves a similar solution for the intersections of the streamline with the edges of the cell. Since flow always follows the negative head gradient, the exit point will be that intersection with the lowest head value. The length of this line is simply obtained from the pythagorean theorem,

$$L = \sqrt{(x_{\text{exit}} - x_{\text{enter}})^2 + (y_{\text{exit}} - y_{\text{enter}})^2} \quad (39)$$

and since the velocity is constant, the time calculation is simply

$$t = \frac{L}{v}, \quad (40)$$

More special cases lurk in the case where the slope of the plane is exactly in the x or y direction, i.e. when either a_1 or a_2 is zero. The length of a path is then simply $L(x) = |x_{\text{exit}} - x_{\text{enter}}|$ or $L(y) = |y_{\text{exit}} - y_{\text{enter}}|$, and the time is again $t = L/v$. If both a_1 and a_2 are zero, the surface is the horizontal plane $h(x,y) = C$, and no flow occurs.

3.4.3 Streamlines on a Grid

Now we shall consider the implementation of this model on a regular grid: the DarcyTrack function. The Grid data model contains one floating point (or integer) value for each cell in the grid. The head elevation grid represents points on a surface, and by convention the values are assumed to be at the center of the cell: a block-centered finite-difference model. The interpolated bilinear surface which DarcyTrack uses for streamline generation has four of these cell values at its corners, resulting in an offset grid block (see Figure 7). This introduces another problem, since the material properties (T , b , and n) are assumed to be constant over the original grid cell. By subdividing the bilinear surface into four quadrants, whose corner values are easily interpolated, we get a smaller cell with constant properties and a bilinear head surface. This cell is what the DarcyTrack function uses, and its construction is entirely internal to the program.

Let a known point on the path line be $(x_{\text{enter}}, y_{\text{enter}})$. This point of entrance will generally be on an edge of the cell, as the particle path crosses a cell boundary, but may also be interior or even at a corner, or vertex. To determine the exit point, we must find all intersections (roots) of the streamline equation

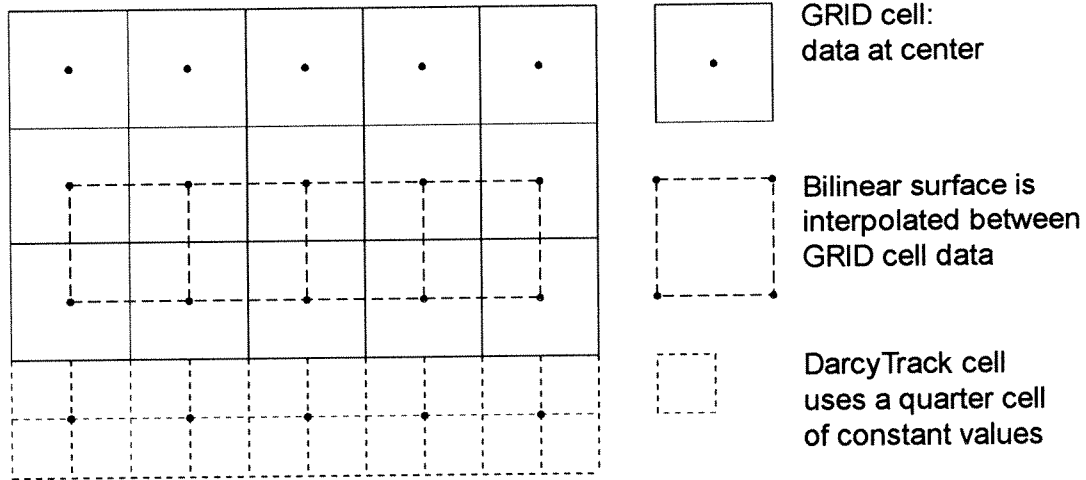


Figure 7 Cells are redefined for the DarcyTrack function.

(25) with the edges of the cell, that is with $x = x_1$, $x = x_2$, $y = y_1$, and $y = y_2$. Of the possible roots, we then need to determine which of those roots is real and lies within the boundaries of the cell. Those roots will define the intersections of the curve with the cell edges (see Figure 8).

Given $(x_{\text{enter}}, y_{\text{enter}})$, the equation for the streamline is completed by solving for C , rearranging the equation as

$$C = a_2 x + a_3 \frac{x^2}{2} - a_1 y - a_3 \frac{y^2}{2}. \quad (41)$$

The streamline equation has two roots in x (adding and subtracting the radical)

$$\frac{1}{a_3} \left(-a_2 \pm \sqrt{a_2^2 + 2a_1 a_3 y + a_3^2 y^2 + 2a_3 C} \right) \quad (42)$$

and two in y

$$\frac{1}{a_3} \left(-a_1 \pm \sqrt{a_1^2 + 2a_2 a_3 x + a_3^2 x^2 - 2a_3 C} \right). \quad (43)$$

Each of these four roots will have one intersection each with the upper and lower cell edges for the corresponding coordinate. By substituting x_1 , x_2 , y_1 , and y_2

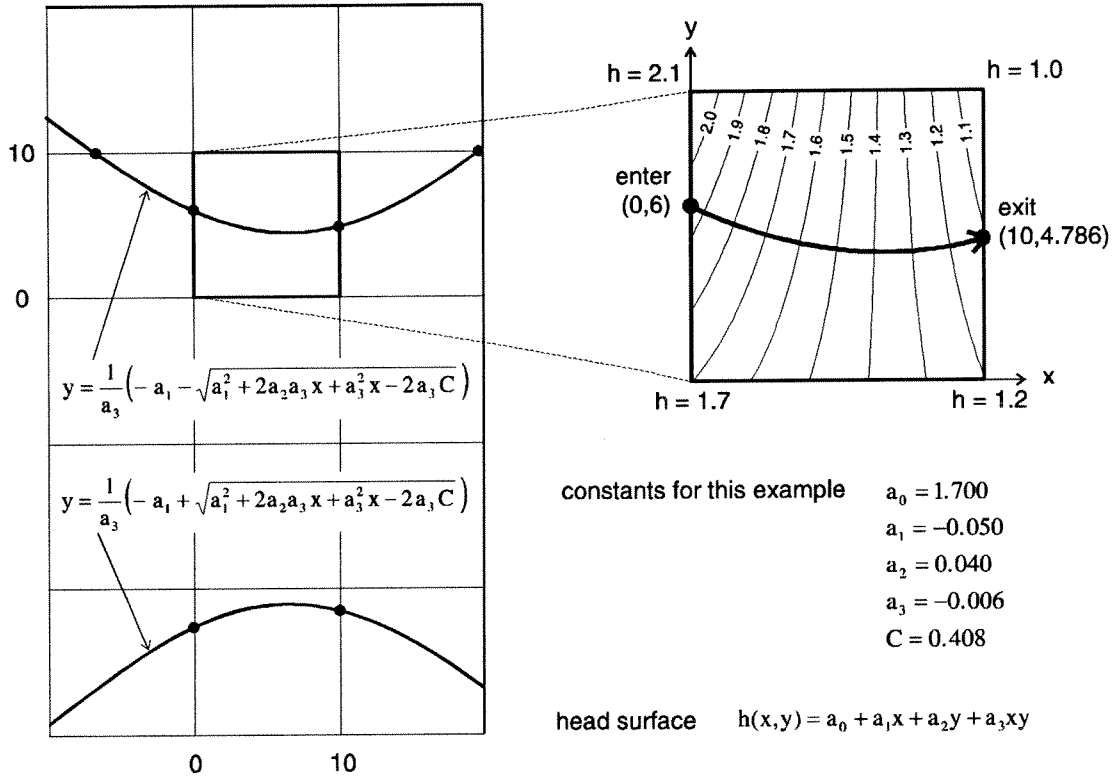


Figure 8 Example of streamline intersections with cell boundaries for the DarcyTrack function.

into equations (42) and (43), we find eight roots corresponding to intersections with the lines $x = x_1$, $x = x_2$, $y = y_1$, and $y = y_2$. The nature of these roots varies with the character of the surface and the location of $(x_{\text{enter}}, y_{\text{enter}})$, and in general some of the roots will have imaginary parts (when the equation takes on an elliptical form), and some will intersect the bounding lines outside of the edges of the cell. At least two roots will be found which intersect the edge, though in the case of a tangential path these two roots may be copunctual. The root point which has the lower head value will always be the exit point, $(x_{\text{exit}}, y_{\text{exit}})$. Now

that the two endpoints of the path are known, we can proceed with calculating the length and travel time of the path segment.

There are a few special cases which complicate these calculations, related to depressions and to initial locations on an edge or vertex. These have been addressed by structuring the execution of the code around them (see the flow chart in Figure 9). The initial location can in general be anywhere on the grid, and must be either interior to a cell, on an edge, or on a vertex. If the point is on an edge or vertex, the surrounding surfaces must be examined to find the most negative gradient. If none of the surrounding surfaces has a negative gradient, then the point is in a "valley" or in a hole. If the valley has some slope to it, the particle migrates to the downslope vertex. If the path leads to a vertex with no way out (a depression) the process is terminated.

The output from the DarcyTrack function is in the same format as that from the ParticleTrack function: a text file and an Arc coverage.

3.5 Advection-Dispersion in Porous Media

Once the average fluid velocities have been calculated throughout the domain, we can examine the behavior of fluid mixing in the porous medium. Mixing is caused by the advection and dispersion of fluid elements, and the two processes are intimately intertwined. While the mean advection of a fluid particle follows the velocity field, dispersion occurs on several scales. It is an amalgamation of diffusion induced by local concentration gradients, of pore-scale mixing from non-uniform velocity profiles, and of dispersion introduced by the variety of travel times through different tortuous pore paths and other fluid conduits. These complex phenomena are assumed to have the common property of random

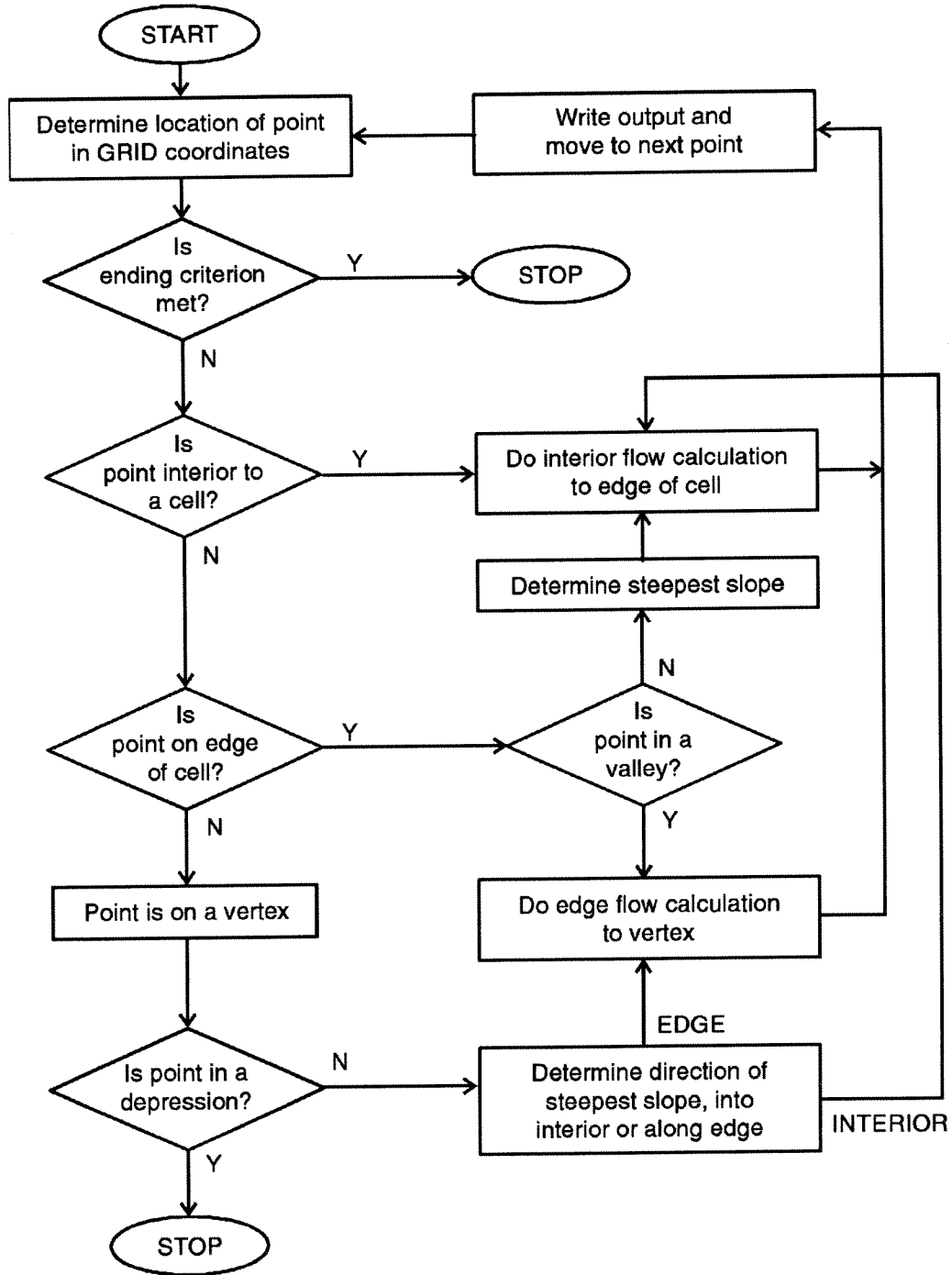


Figure 9 Flow chart for the DarcyTrack program.

redistribution of fluid, and so they are lumped together into the dispersion coefficients. In combining these phenomena, we are assuming that our ignorance of their detail and of other processes is also unbiased.

Einstein (1926) showed that the random and independent movement of microscopically visible particles, termed Brownian motion, can be characterized

by a normal probability distribution $f(x,t)$ of the resulting displacements in a given time t :

$$f(x,t) = \frac{N}{\sqrt{4\pi D}} \frac{e^{-\frac{x^2}{4Dt}}}{\sqrt{t}} \quad (44)$$

where N is the total number of particles, x is the spatial dimension, and D is the coefficient of diffusion. The universality of this observation is easily demonstrated. An experiment can be constructed in the form of a computer program which tracks the position of a set of initially copunctual particles which are each displaced by random (or, for the computer, pseudorandom) distances over a series of time steps. Their distribution can be analyzed, and is seen to be normal (Hathhorn, 1990). Figure 10 illustrates a series of time steps from just such a computer program, where the points are moved by random distances within specified ranges. The variance of the gaussian distribution defined by equation (44) increases with time at a predictable rate. Einstein (1926) arrived at the result

$$\sigma^2 = 2 D t \quad (45)$$

where σ^2 is the arithmetic mean of the squares of particle displacements with time (the variance). Thus, the distribution of particles which initially made up



Figure 10 Diffusion modeled as a random process produces a normal (gaussian) distribution.

a discrete parcel can be modeled assuming a gaussian distribution with a time-dependent variance.

If we extend this notion of random and independent displacements to the movement of fluid flow through a porous medium, we can use the gaussian model to describe dispersion. We must recognize, however, that the variance depends not on time as for diffusion, but is a function of the distance travelled, since dispersion is induced by the flow itself. In order to express the variance in terms of length of travel, we will use the mean displacement of the particles: the advection path of the centroid of mass of the particles.

The value of the dispersion coefficient D varies with the magnitude v of the mean seepage velocity \mathbf{v} (Marsily, 1986). For common groundwater velocities where kinematic dispersion dominates, the dispersion tensor D_{ij} and the velocity vector \mathbf{v} can be related by the two components of dispersivity, α_L and α_T in the general form of a second-order tensor which is proportional to the velocity:

$$D_{ij} = \alpha_T v \delta_{ij} + (\alpha_L - \alpha_T) \frac{v_i v_j}{|v|} . \quad (46)$$

If we align the coordinate system with the velocity in the longitudinal direction, then $v_j = 0$. For the PorousPuff function, we use an orthogonal coordinate system which is always aligned with the velocity field. In this case, the dispersivity has spatial components which correspond to the longitudinal and transverse components of the dispersion coefficient, so that

$$D_L = \alpha_L v \quad \text{and} \quad D_T = \alpha_T v . \quad (47)$$

Since the dispersion coefficient is calculated from it, the dispersivity must be supplied as a parameter to the PorousPuff function. The best estimates of dispersivity will be obtained by *in situ* testing such as a two well tracer test (Fetter, 1993), and if these data are available they should be used. However, aquifer tests are expensive and often these data must be estimated by less sophisticated means. A series of simulations can be used to capture the uncertainty in the choice of a value for dispersivity.

By referring to published values of dispersivity estimated from field data, one may get estimates for specific geologic media and study scales. In a recent compilation of published values, Gelhar, et al. (1992) have shown that dispersivity varies with size, or problem scale, as shown in Figure 11. This relationship, known as the scale effect, has also been noted by Lallemand-Barres and Peaudecerf (1978), Anderson (1979), Pickens and Grisak (1981), Gelhar, et al. (1985), Neuman (1990), Xu and Eckstein (1993), and others. The trend which at first seems apparent in these data seems less so when one considers the quality of the data. If we consider only those points which Gelhar et al. (1992) have classified as highly reliable, there is no clear trend whatsoever, a point which they make most emphatically. These most reliable dispersivity values are available

only in a limited range of scale (10 to 300 meters) and show no correlation to scale.

Although these data are insufficient for defining any universal functional relationship between scale and dispersivity, one must be generated for the purposes of implementing the PorousPuff function in the Arc/Info GIS. A default value for dispersivity is necessary for keeping with the style of other Grid functions, and for the prevention of grave errors of ignorance on the part of the user. A functional relationship is here devised for relating dispersivity to scale without the need for further information, and is intended as a method of generating plausible values in the absence of other data. Allowing this default to routinely make the calculations is not recommended — it is merely a safety net.

Attempts have been made to fit simple functions to the scale effect. Neuman (1990) suggests a linear regression of $\log(\alpha_L)$ vs $\log(L)$ over two ranges, with a break at 100 meters. There is no justification for the 100 m break *a priori*, and I can imagine no reason for its existence in the field. Since the data for scales over 100 m are of poor quality, I also see no need to condition a model on them. Xu and Eckstein (1993) recognized that the data plotted in Figure 11 suggest another relationship, and propose linearizing $\log(\alpha_L)$ with $\log(\log(L))$. They performed a linear regression of $\log(\alpha_L)$ to $\log(\log(L))$, using data from Neuman (1990), presented in meters:

$$\alpha_L = 0.902 [\log(L)]^{2.875} \quad (48)$$

Although this transformation does linearize the data to a remarkable extent, Xu and Eckstein's choice of units influences the nature of the fit, which excludes all data below one meter and causes the lower tail of the curve to asymptotically approach $L = 1$ m. This is due to the double log of L (the log of any number between 0 and 1 will produce a negative, which is beyond the range of the log

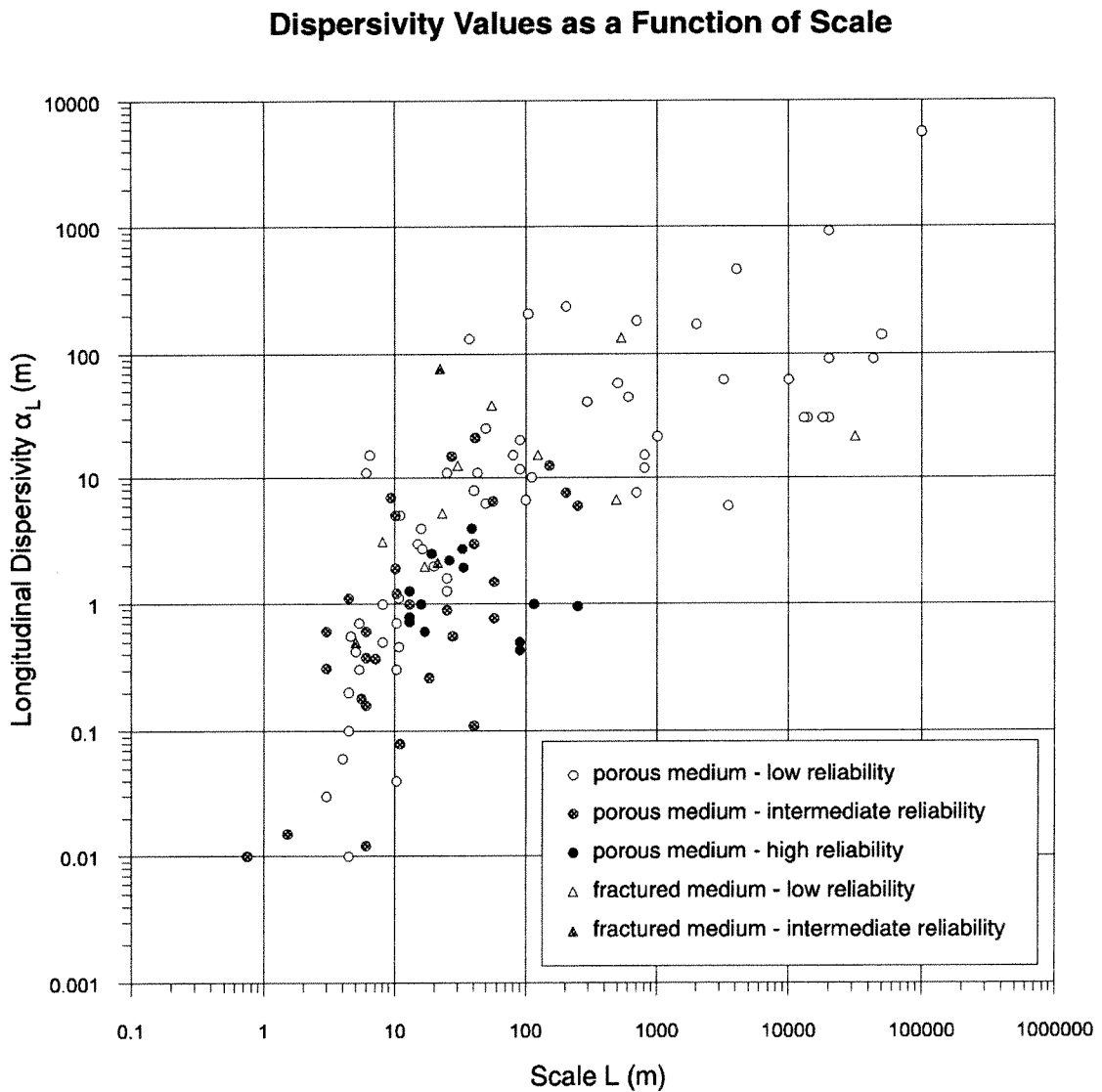


Figure 11 Field dispersivity data show a rough correlation of α_L to L. (After Gelhar et al., 1992)

function). To ameliorate this problem, I converted the best available data (Gelhar et al., 1992) to centimeters, and then performed a linear regression of $\log(\alpha_L)$ to $\log(\log(L))$ as suggested by Xu and Eckstein. This produced a satisfactory fit (Figure 12), but it is still discomforting that the fit should depend on the units chosen. This regression produces the equation

$$\alpha_L = 0.0239 [\log(L)]^{7.33} \quad (49)$$

as the best fit (data in centimeters), with a regression coefficient of $r^2 = 0.64$. In addition to the problem of units defining the shape of the function, there is the added problem that to use equation (49), one must use units of centimeters to preserve the values of the constants. In keeping with the Grid functions developed here, it is highly desirable to develop a dimensionless function, so that any consistent set of units will work throughout the analysis.

One simple dimensionless function is a linear function of the form $\alpha_L = a \cdot L + b$, where a and b are constants. By setting $b = 0$, this fitting function is forced through the origin and nondimensionality is preserved. A regression produces the equation

$$\alpha_L = 0.0363 L = L / 27.535 \quad (50)$$

shown in Figure 13. This simple linear fit has a regression coefficient of $r^2 = 0.65$, at least as good as the double log fit, with the additional feature that it works for any linear units. Other statistics shown in the figures also show that equation (50) is a better overall fit. and the larger t statistic shows a better representation by a normal distribution. For these reasons, equation (50) is used as the default calculation of longitudinal dispersivity. Simplicity has served well, and the low reliability of the more extreme dispersivity data do not warrant a more elaborate model.

Dispersivity Values as a Function of Scale
Linear Regression of $\log(\alpha_L)$ vs $\log(\log(L))$

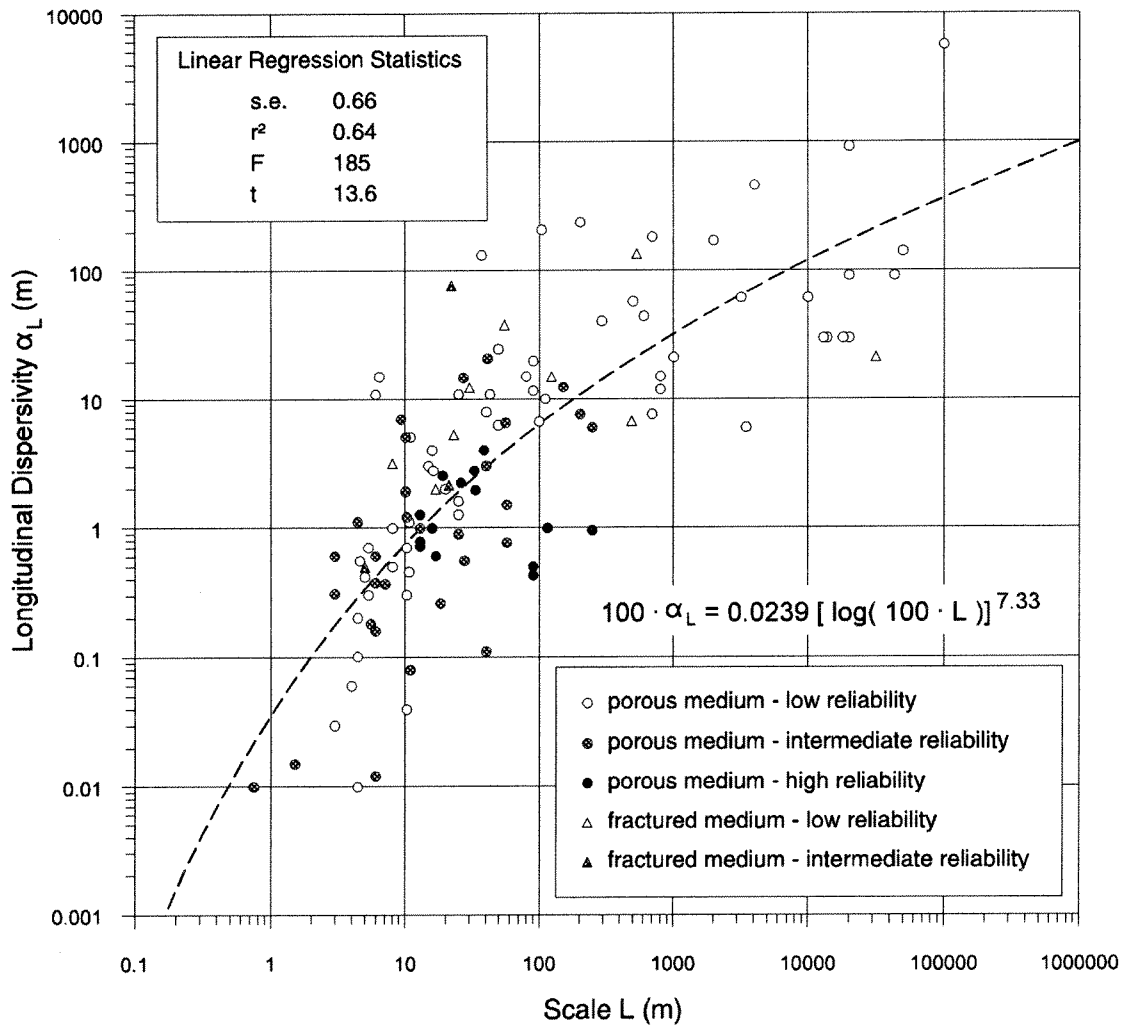


Figure 12 Field dispersivity data fit of $\log(\alpha_L)$ to $\log(\log(L))$.

Dispersivity Values as a Function of Scale
Linear Regression of α_L vs L

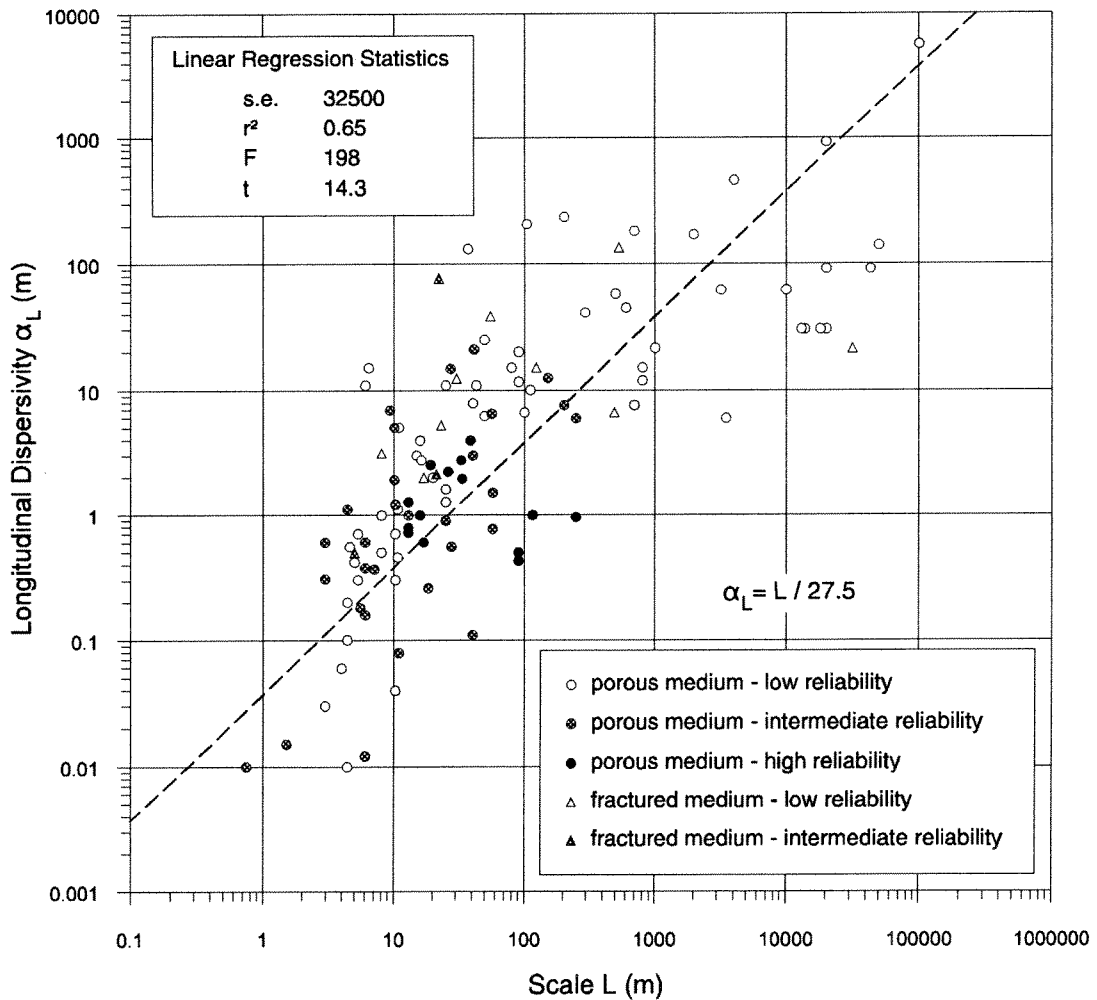


Figure 13 Field dispersivity data with a linear regression of α_L vs L.

Transverse dispersivity is also required for our two-dimensional calculation. Again, by examining the data from Gelhar (1992) we can discern a nearly linear trend (Figure 14). The dashed line represents a popular value of the dispersivity ratio of

$$\frac{\alpha_L}{\alpha_T} = 3 \quad (51)$$

(Federal Register, 1986), which seems quite adequate. This value is used as a default calculation of α_T given α_L by equation (50). Again, these defaults have no theoretical basis, and the modeler is encouraged to use better data or methods if available.

So far, this discussion has applied to the description of generalized fluid flow through a porous medium. The physical concepts are valid for any fluid or medium, and occur at every spatiotemporal location in the domain. Every part of the flow field is undergoing continuous advection and dispersion, and our imaginary parcels of fluid can be constructed at any location. In the case where σ^2 is a linear function of L (as when a constant value of α_L is used), the dispersion of that parcel is superposed with the dispersion of all other possible parcels in space and time, creating a continuous dispersion field. Let us now focus on the fate and transport of a particular constituent of the fluid, which expresses a more complex behavior.

In this conceptual model, we consider two additional effects on a fluid constituent: decay and retardation. Decay may result from a variety of processes, and in general its estimation is complex. Chemical reaction, biological degradation, and nuclear processes all contribute to decay, but only the latter depends solely on the amount of the constituent present. Biological and chemical effects require the presence of additional species such as electron donors or bacteria, and

Comparison of Longitudinal and Transverse Dispersivities

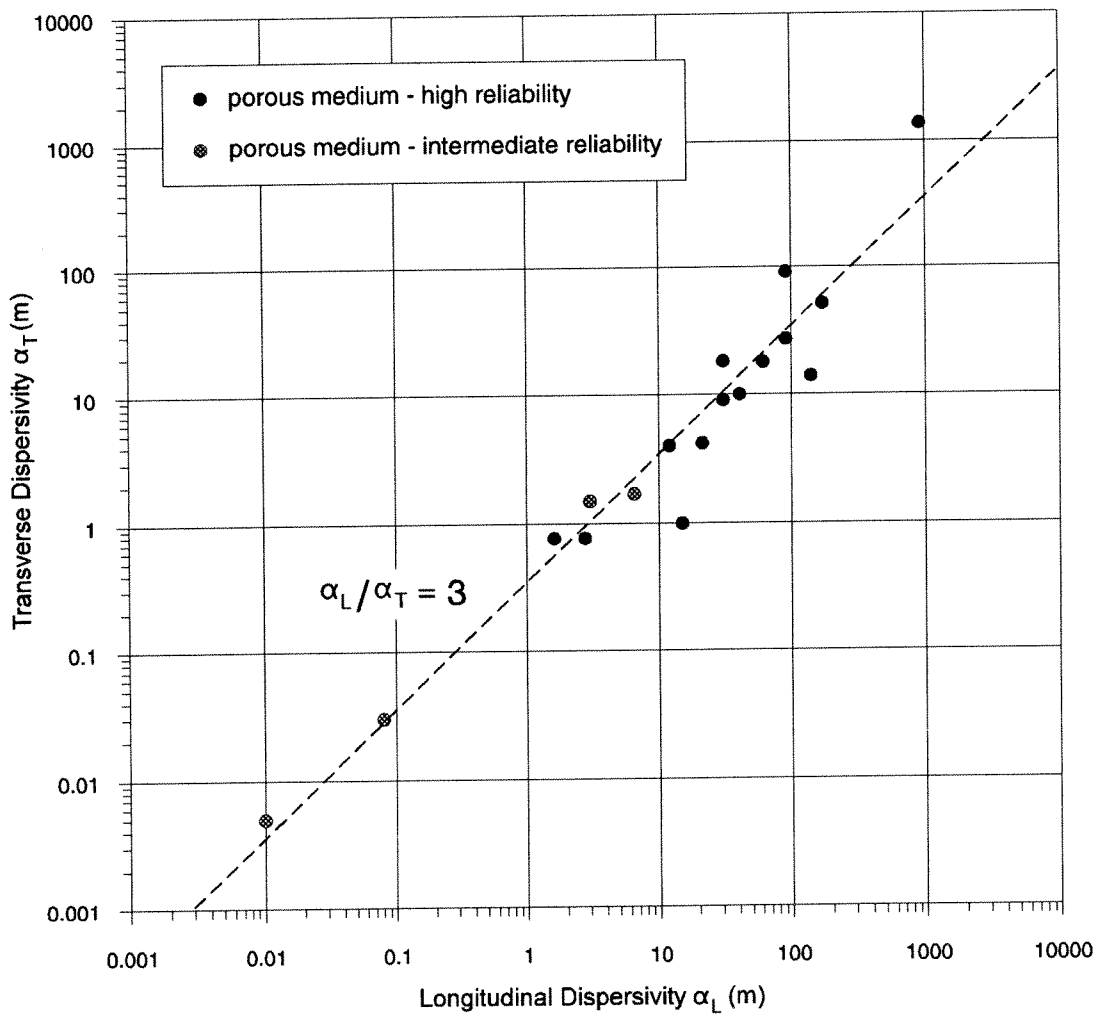


Figure 14 Transverse vs longitudinal dispersivity. (Data from Gelhar, et al., 1992.)

in general these effects cannot be estimated *a priori*. Although biological degradation may be modeled as a first-order decay process by using the Monod kinetic model, the assumption that bacterial growth is substrate-limited over the extent of the contaminant plume may be difficult to justify. Again, our ignorance of subsurface conditions allows us to lump all of these decay effects into one parameter: a first-order decay coefficient. This assumes that the rate of decay of a constituent is a linear function of its bulk concentration, which is strictly true for radioactive decay. The rate of this decay is determined by the first-order decay constant λ .

In an attempt to capture the essence of the chromatographic nature of constituent flow through a porous medium, the conceptual model presented here also includes chemical retardation. An adsorbing constituent will be retarded in its migration through the porous matrix due to these chromatographic effects. The process of adsorption and desorption, controlled in this model by the distribution coefficient K_d and the bulk density ρ_b , are assumed to be everywhere occurring in equilibrium with the aqueous concentration. The effect of this behavior is to retard movement of the entire mass M , and it is captured in the retardation factor, R :

$$R = \frac{M}{c \cdot n} = \frac{n + \rho_b K_d}{n} = 1 + \frac{\rho_b K_d}{n} . \quad (52)$$

R has a minimum value of 1, corresponding to no retardation. Increasing values of R imply greater retardation of the constituent.

3.5.1 The Advection-Dispersion Equation

Transport equations are founded on the principle of the conservation of mass: The net rate of increase of mass in a control volume is equal to net mass flux into the volume plus the increase of mass within the volume. This statement of continuity is expressed in the equation:

$$\frac{\partial \mathbf{m}}{\partial t} + \nabla \cdot \mathbf{J} = \mathbf{S}^+ \quad (53)$$

where

\mathbf{m} \equiv bulk concentration (units of mass/length³)

t \equiv time (units of time)

\mathbf{J} \equiv the mass flux vector, representing the mass transport across a unit surface per unit time, normal to the surface (units of mass/length²/time)

\mathbf{S}^+ \equiv strength of mass production within the control volume (units of mass/length³/time)

and $\nabla \cdot \mathbf{J}$ is the divergence of the flux vector. The bulk concentration \mathbf{m} of the constituent includes all of the mass in the volume, be it sorbed or in solution. Assuming local equilibrium and a linear sorption isotherm, the sorbed concentration c_s can be expressed as a function of the aqueous concentration c_{aq} :

$$c_s = K_d c_{aq} \quad (54)$$

where K_d is the distribution coefficient for the constituent. The bulk concentration is the sum of the aqueous and sorbed concentrations:

$$\mathbf{m} = n c_{aq} + \rho_b c_s \quad (55)$$

or

$$\mathbf{m} = (\mathbf{n} + \rho_b \mathbf{K}_d) \mathbf{c}_{\text{aq}} \quad (56)$$

The mass flux vector \mathbf{J} includes both advective and dispersive components as described above:

$$\mathbf{J} = \mathbf{q} \mathbf{c} - \mathbf{n} \mathbf{D} \cdot \nabla \mathbf{c} . \quad (57)$$

The right hand side of equation (53) contains the source term, S^+ . The only process we have considered which affects this value is decay as a first-order function of mass, so that

$$\mathbf{S}^+ = -\lambda \mathbf{m} . \quad (58)$$

Substituting these expressions into equation (53) gives

$$\frac{\partial}{\partial t} [(\mathbf{n} + \rho_b \mathbf{K}_d) \mathbf{c}] + \nabla \cdot [\mathbf{q} \mathbf{c} - \mathbf{n} \mathbf{D} \cdot \nabla \mathbf{c}] = -\lambda [(\mathbf{n} + \rho_b \mathbf{K}_d) \mathbf{c}] . \quad (59)$$

To simplify, let us assume steady flow, with no volumetric sources or sinks. In this case,

$$\nabla \cdot \mathbf{q} = 0 . \quad (60)$$

By the chain rule,

$$\nabla \cdot (\mathbf{q} \mathbf{c}) = \mathbf{c} \nabla \cdot \mathbf{q} + \mathbf{q} \cdot \nabla \mathbf{c} , \quad (61)$$

the second term of which disappears by equation (60). Assuming also a constant fluid density and porosity (using the value of porosity found at the centroid of the dispersion ellipse), equation (59) can be written

$$(\mathbf{n} + \rho_b \mathbf{K}_d) \frac{\partial \mathbf{c}}{\partial t} + \mathbf{q} \cdot \nabla \mathbf{c} - \mathbf{n} \nabla \cdot (\mathbf{D} \cdot \nabla \mathbf{c}) = -\lambda \mathbf{c} (\mathbf{n} + \rho_b \mathbf{K}_d) . \quad (62)$$

The definition of retardation (equation (52)) can simplify this equation further:

$$nR \frac{\partial c}{\partial t} + \mathbf{q} \cdot \nabla c - n \nabla \cdot (\mathbf{D} \cdot \nabla c) = -\lambda c n R . \quad (63)$$

Dividing through by the porosity n gives the general form

$$R \frac{\partial c}{\partial t} - \nabla \cdot (\mathbf{D} \cdot \nabla c) + \mathbf{v} \cdot \nabla c + \lambda R c = 0 . \quad (64)$$

By orienting the coordinate system so that the local flow direction is $+x$, (the only nonzero component of the velocity vector is v_x), we can rearrange to get

$$R \frac{\partial c}{\partial t} = \frac{\partial}{\partial x} \left(D_{xx} \frac{\partial c}{\partial x} \right) + \frac{\partial}{\partial y} \left(D_{yy} \frac{\partial c}{\partial y} \right) + \frac{\partial}{\partial z} \left(D_{zz} \frac{\partial c}{\partial z} \right) - v_x \frac{\partial c}{\partial x} - \lambda R c . \quad (65)$$

At this point, it is convenient to introduce a new coordinate system to facilitate the discussions which follow. Let us define a two-dimensional coordinate space oriented with the flow, with two orthogonal directions: longitudinal and transverse, corresponding to the principal directions of the dispersion tensor. These directions are indicated by the subscripts L, and T. For example, displacements are denoted by X_L and X_T , and components of the dispersivity tensor are D_L and D_T . As we proceed along the advection path, this coordinate system follows the curve, so that the longitudinal axis is tangent to the local flow direction, v_L . By further dividing through by R , we can express equation (65) in the X_L, X_T coordinate system as:

$$\frac{\partial c}{\partial t} = \frac{D_L}{R} \frac{\partial^2 c}{\partial X_L^2} + \frac{D_T}{R} \frac{\partial^2 c}{\partial X_T^2} - \frac{v_L}{R} \frac{\partial c}{\partial X_L} - \lambda c . \quad (66)$$

This is the principal governing equation for the following discussion.

By application of various source terms, different solutions to this equation are obtained. We shall consider two idealized sources: the instantaneous point

source and the continuous point source. These are the gaussian puff and plume models described below.

3.5.2 Solution for an Instantaneous Point Source

An instantaneous pulse input (analogous to the Dirac delta function) of a soluble constituent into an initially clean (zero concentration) ideal porous medium will result in a puff of concentration which migrates with the seepage velocity of the fluid medium (advection) and spreads out in all directions (dispersion) in two processes which may be mathematically decoupled into advection and dispersion, as shown in Figure 15.

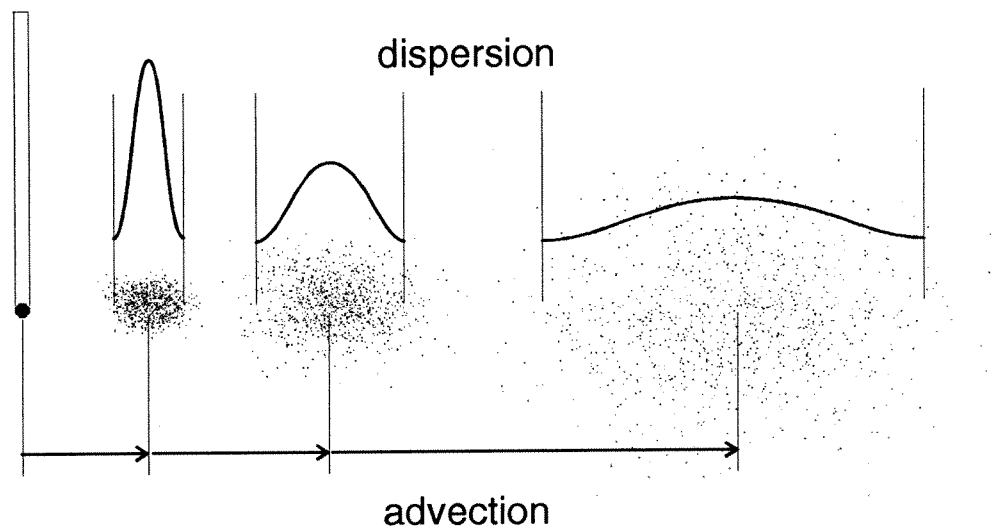


Figure 15 Advection and dispersion of a point source.

By decoupling these processes, the entire mass M is advected as the centroid of mass, along the particle path streamline. The dispersion is calculated as a separate process, with dispersion scaled to path length. Advection then drops out of the solution, since it is incorporated in the change of coordinates and the

dispersion coefficient. The origin of the (X_L, X_T) coordinate system is at the centroid of mass after advection has been accounted for.

With the aid of Green's function for the partial differential operator (Friedman, 1956) in the directions X_L and X_T , the concentration as a function of spatial location can be derived by redistributing the initial mass, M . If we consider an aquifer of uniform thickness b , the vertically averaged solution for an instantaneous point source in a homogeneous and isotropic porous medium subject to a uniform velocity in the L direction is (after Wilson and Miller, 1978)

$$c(x_L, x_T, t) = \frac{\exp\left(\frac{-X_L^2}{2\sigma_L^2}\right)}{\sqrt{2\pi\sigma_L^2}} \cdot \frac{\exp\left(\frac{-X_T^2}{2\sigma_T^2}\right)}{\sqrt{2\pi\sigma_T^2}} \cdot \frac{M e^{-\lambda t}}{n b R} \quad (67)$$

where

- c \equiv solute concentration (units of mass/length³)
- X_L \equiv longitudinal coordinate (units of length)
- X_T \equiv transverse coordinate (units of length)
- t \equiv time (units of time)
- σ_L^2 \equiv longitudinal variance (units of length²)
- σ_T^2 \equiv transverse variance (units of length²)
- M \equiv initial mass (units of mass)
- λ \equiv decay coefficient (units of inverse time)
- n \equiv effective porosity (dimensionless)
- b \equiv saturated thickness (units of length)
- R \equiv retardation factor (dimensionless)

The variances σ_i^2 govern the gaussian distribution, and are functions of the dispersion coefficients as described in equation (45) on page 62. This equation is the basis for the puff model. The concentration distribution is bivariate gaussian, and varies with time as the puff travels with seepage velocity v . This velocity enters the equation indirectly through time t and variances σ_i^2 , which are commonly calculated from dispersivities α_i and travel path length L , or from retarded dispersion coefficients D_i/R and time (Bear, 1972):

$$\sigma_i^2 = 2 \frac{D_i}{R} t = 2 \frac{\alpha_i v}{R} t = 2 \alpha_i L. \quad (68)$$

It should be noted that if we allow α to vary with L , as discussed above, then the problem becomes nonlinear. This is an important consideration when applying the model over varying times and distances.

A dispersion ellipse is calculated around the centroid, with the major axis oriented along the flow direction. The longitudinal and transverse displacements can readily be transformed from global (x, y) grid coordinates to the local (X_L, X_T) coordinates. This bivariate gaussian distribution is illustrated in Figure 16. The ellipses are projections of isocons onto the analysis plane, with sections through the concentration field also shown.

The PorousPuff function is an application of equation (67), in the (X_L, X_T) coordinate system. To execute this transformation on the (x,y) grid, the origin is identified as that point along the path which corresponds to the position of a particle which has travelled for the length of time supplied by the user. For example, if we are interested in the concentration distribution after 100 days, PorousPuff reads the path file (generated by either ParticleTrack or DarcyTrack) to find the time data which bracket 100 days. The corresponding locations are used to interpolate a location for advection of the centroid of mass of the source

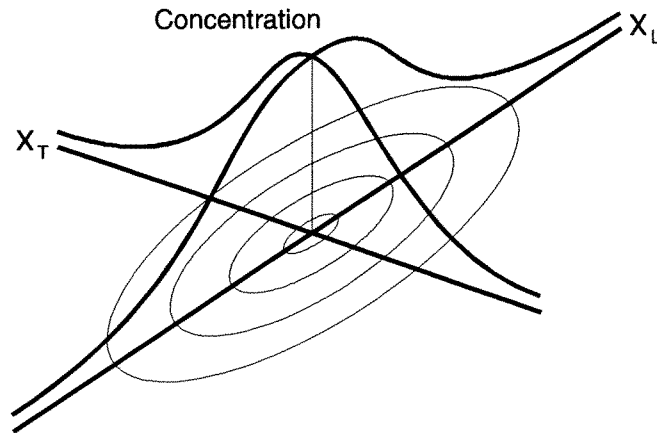


Figure 16 Perspective illustration of the gaussian dispersion of a puff.

after 100 days. This point is the origin of the (X_L, X_T) coordinate system. X_L is oriented in the direction of the path at that point, and X_T is perpendicular. A concentration is then calculated for each cell in the grid, based on that cell's (X_L, X_T) coordinates, and the material properties at the centroid of mass. Figure 17 shows a series of three separate calculated puffs, each with its own (X_L, X_T) coordinates.

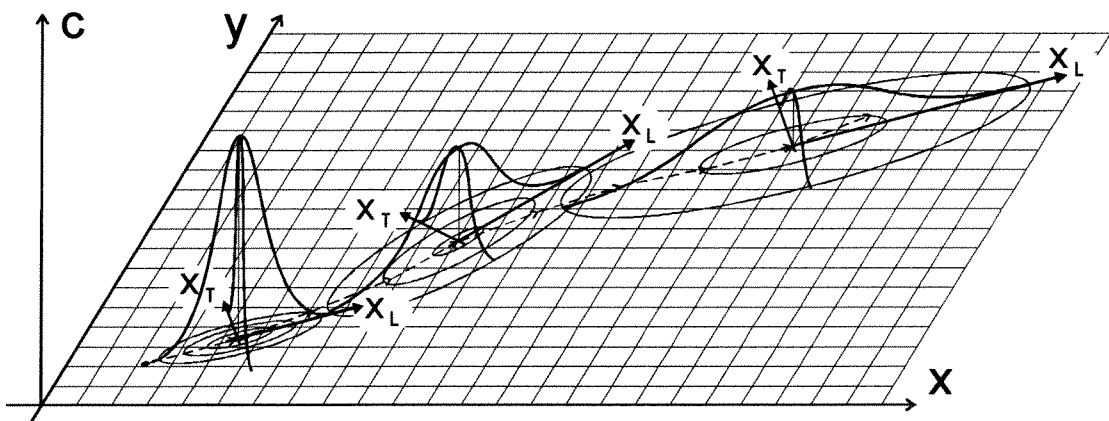


Figure 17 A sequence of three puffs along a path line is shown superimposed on the underlying grid.

The calculation of the concentration for each cell requires an integration of equation (67) over the area of the cell (see Figure 18). That integral is easily obtained in (X_L, X_T) coordinates:

$$c_{\text{cell}} = \int_{X_{L1}}^{X_{L2}} \int_{X_{T1}}^{X_{T2}} \left[\frac{\exp\left(\frac{-x_L^2}{2\sigma_L^2}\right)}{\sqrt{2\pi\sigma_L^2}} \cdot \frac{\exp\left(\frac{-x_T^2}{2\sigma_T^2}\right)}{\sqrt{2\pi\sigma_T^2}} \cdot \frac{M e^{-\lambda t}}{n b R} \right] dX_T dX_L \quad (69)$$

$$c_{\text{cell}} = \frac{1}{4} \left[\operatorname{erf}\left(\frac{X_{L2}}{\sqrt{2\sigma_L}}\right) - \operatorname{erf}\left(\frac{X_{L1}}{\sqrt{2\sigma_L}}\right) \right] \cdot \left[\operatorname{erf}\left(\frac{X_{T2}}{\sqrt{2\sigma_T}}\right) - \operatorname{erf}\left(\frac{X_{T1}}{\sqrt{2\sigma_T}}\right) \right] \cdot \frac{M e^{-\lambda t}}{n b R}$$

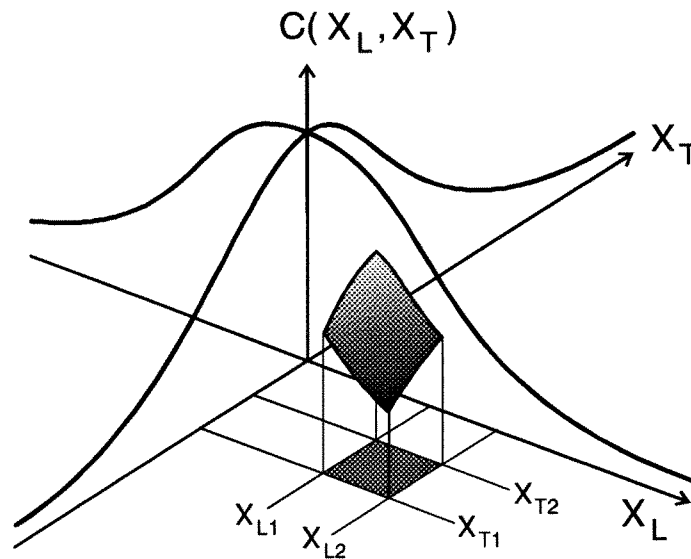


Figure 18 Integration of the concentration function in (X_L, X_T) coordinates.

Unfortunately, this integral is not so easily obtained for the (x,y) coordinate system, in which the grid cell is fixed. We must resort to some sort of numerical approximation. The naïve approach of choosing the value at the

cell center will produce large mass balance errors if the variance σ^2 is small in comparison to the cell size, ℓ . In this case, a numerical integration scheme is necessary. Since the integration domain is limited to the edges of the cell, x_1 , x_2 , y_1 , and y_2 , Gaussian-Legendre quadrature is appropriate (Abrahamowitz and Stegun, 1964). This method makes use of the property of polynomials that they can be approximated exactly by evaluating the integrand at a number n^m of integration points, where n is the degree of the polynomial and m is the number of spatial dimensions (for our purposes, $m=2$). Equation (67) is not a polynomial, but it is a very smooth and well-behaved function.

Each cell is mapped into the (ξ, η) space used for the integration, so that the mapped cell extends from $\xi = -1$ to $\xi = +1$ and $\eta = -1$ to $\eta = +1$. Within this area are located the integration points, whose number and location are defined by the order of integration chosen. In this example, a second-order scheme is chosen ($n = 2$), resulting in $n^m = 2^2 = 4$ integration points, located at $(\pm\sqrt{1/3}, \pm\sqrt{1/3})$ in (ξ, η) space. Each of these points is also assigned a weight w_{ij} , which for order $n = 2$ equals 1 for all four points but is different for higher orders. (All the values for locations and weights are assigned in the module QUADRAT.C, listed in the Appendix.) The integral over the cell in (ξ, η) space is approximated by summing the values of the concentration function at each point times the weight for each point (Becker, et al., 1981):

$$\int_{-1}^1 \int_{-1}^1 c(\xi, \eta) d\xi d\eta \approx \sum_{i=1}^n \left[\sum_{j=1}^n c(\xi_i, \eta_j) w_j \right] w_i. \quad (70)$$

However, we do not know the concentration function $c(\xi, \eta)$ but rather $c(X_L, X_T)$. We can map the integration points from (ξ, η) to (x, y) and then to (X_L, X_T) space, and perform the summation

$$\int_{X_{L1}}^{X_{L2}} \int_{X_{T1}}^{X_{T2}} c(X_L, X_T) dX_T dX_L \approx \sum_{i=1}^n \left[\sum_{j=1}^n c(X_{Li}, X_{Tj}) w_j \right] w_i. \quad (71)$$

This operation is illustrated in Figure 19.

We can evaluate the accuracy of the Gaussian-Legendre quadrature integration by comparing its results to the exact solution in equation (69) by aligning the (X_L, X_T) coordinate system with (x, y) . In doing so, it becomes apparent that the accuracy of the quadrature integration is heavily dependent on the relative sizes of the standard deviation of the gaussian distribution and the grid cell size, as illustrated in Figure 20. It is useful to examine this relationship further so that PorousPuff may automate the choice of integration order. Since the smaller σ (given longitudinal and transverse) is the controlling factor, PorousPuff tests the two values and uses the smaller in the following calculations.

By varying the ratio of standard deviation over cell size σ/ℓ , (or variance over cell area σ^2/ℓ^2) we can examine at which values of the ratio a higher order of integration is needed. A graph of the percent error of the estimate for various orders of integration vs the σ/ℓ ratio is presented in Figure 21. Thus, for any desired percent error and a given σ/ℓ , an appropriate order of integration can be chosen. For example, order 2 is good to within 0.1% for any value of $\sigma/\ell \geq 1$. For the same accuracy, if $\sigma/\ell = 0.01$, an order 32 approximation is necessary to accommodate the more complex surface. Of course, that calculation involves $32^2 = 1024$ points per cell, so it may be more efficient to try the analysis with a smaller cell size to increase σ/ℓ .

The PorousPuff function checks the value of σ/ℓ and chooses an order to achieve an integration error of less than 0.1%. The total mass balance is calculated as the percent quotient of the source mass and the sum of the

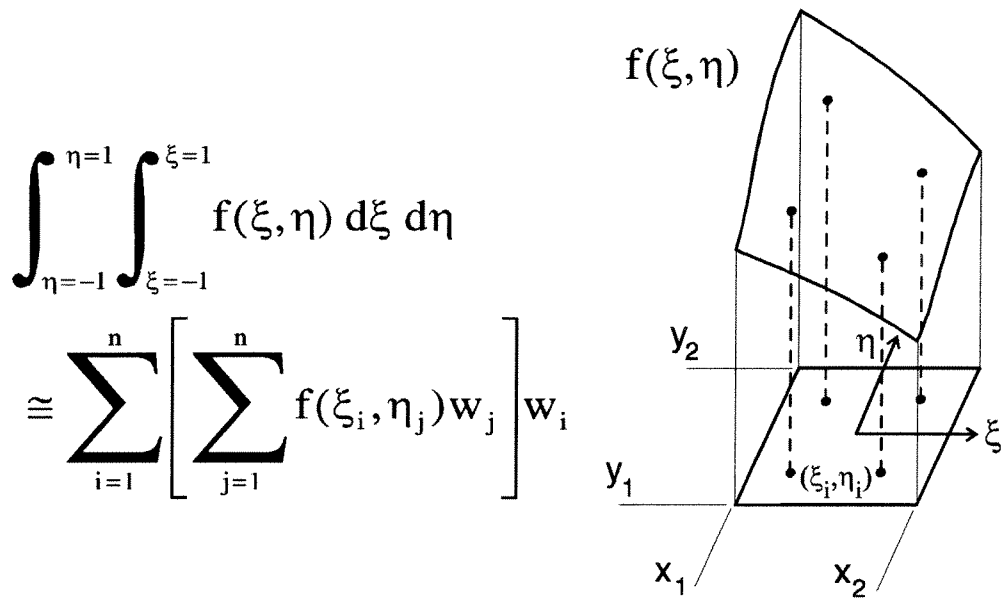
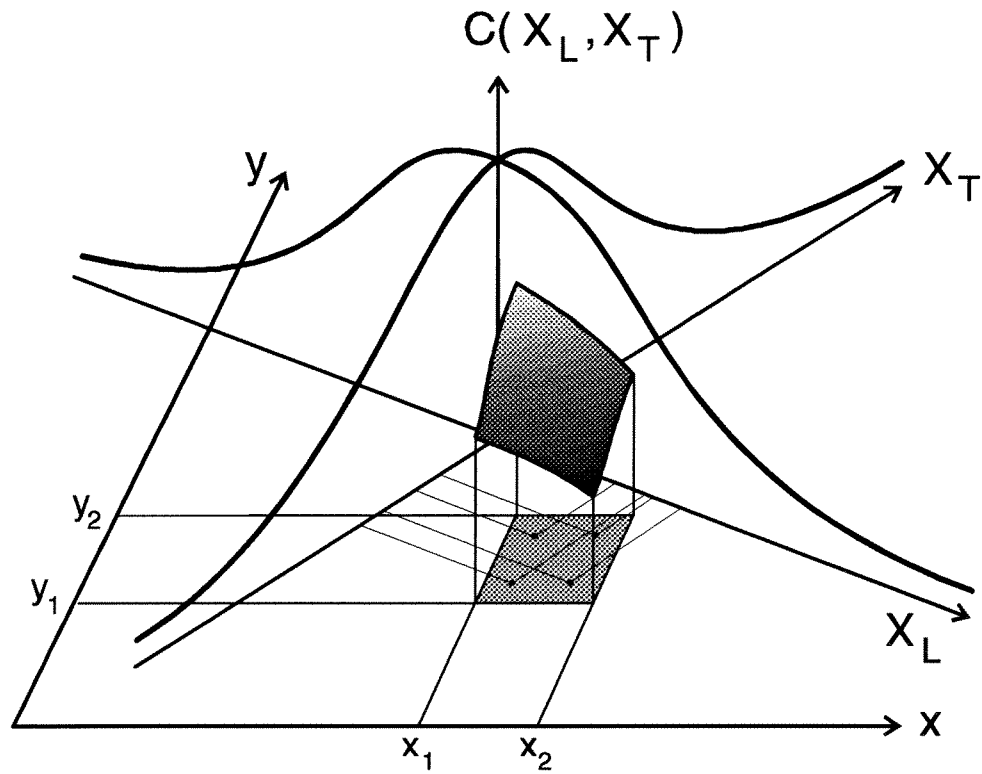


Figure 19 Gauss-Legendre quadrature integration is used to estimate the average concentration over each cell.

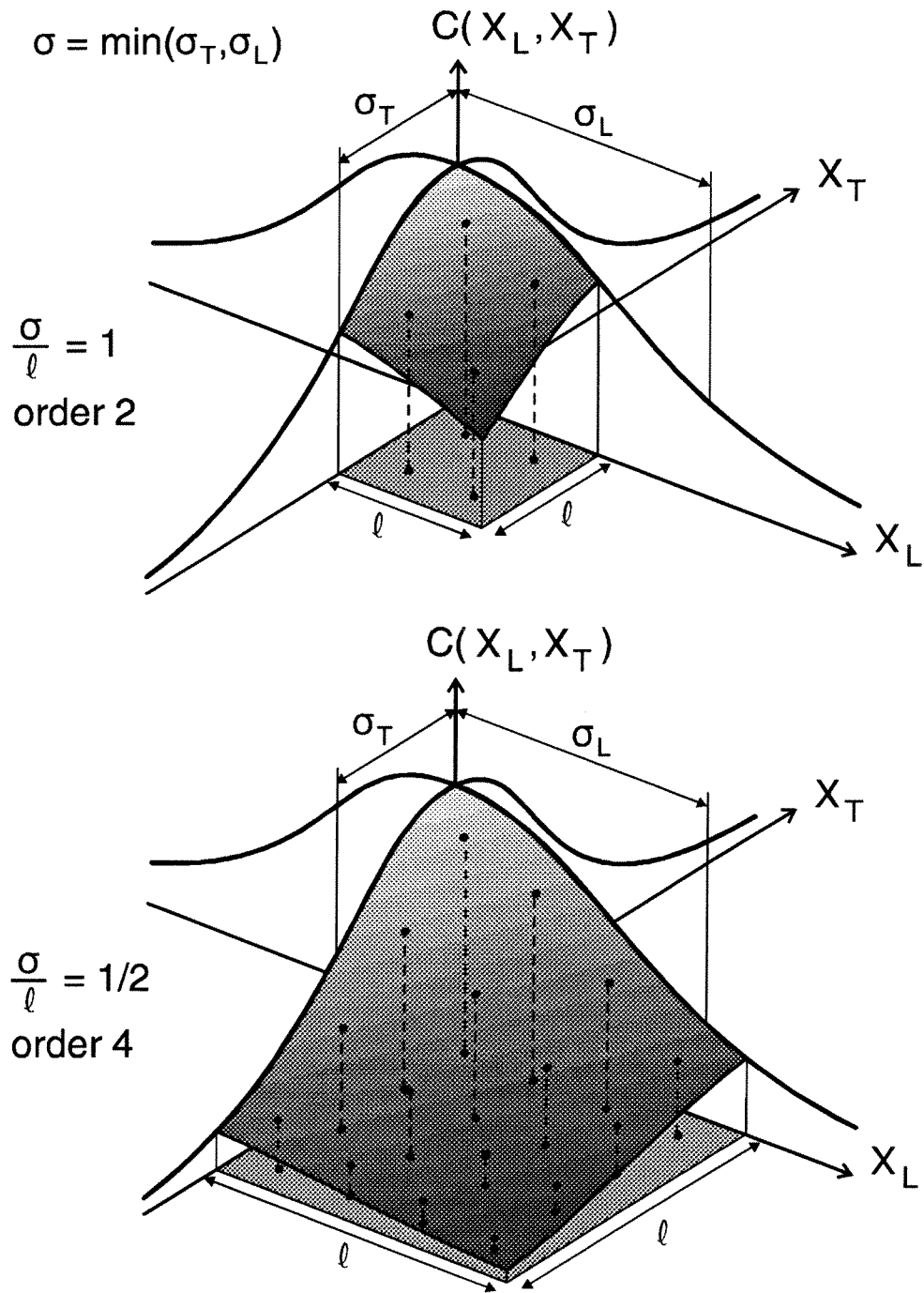


Figure 20 The order of integration is defined by the ratio of standard deviation of the distribution to the grid cell size.

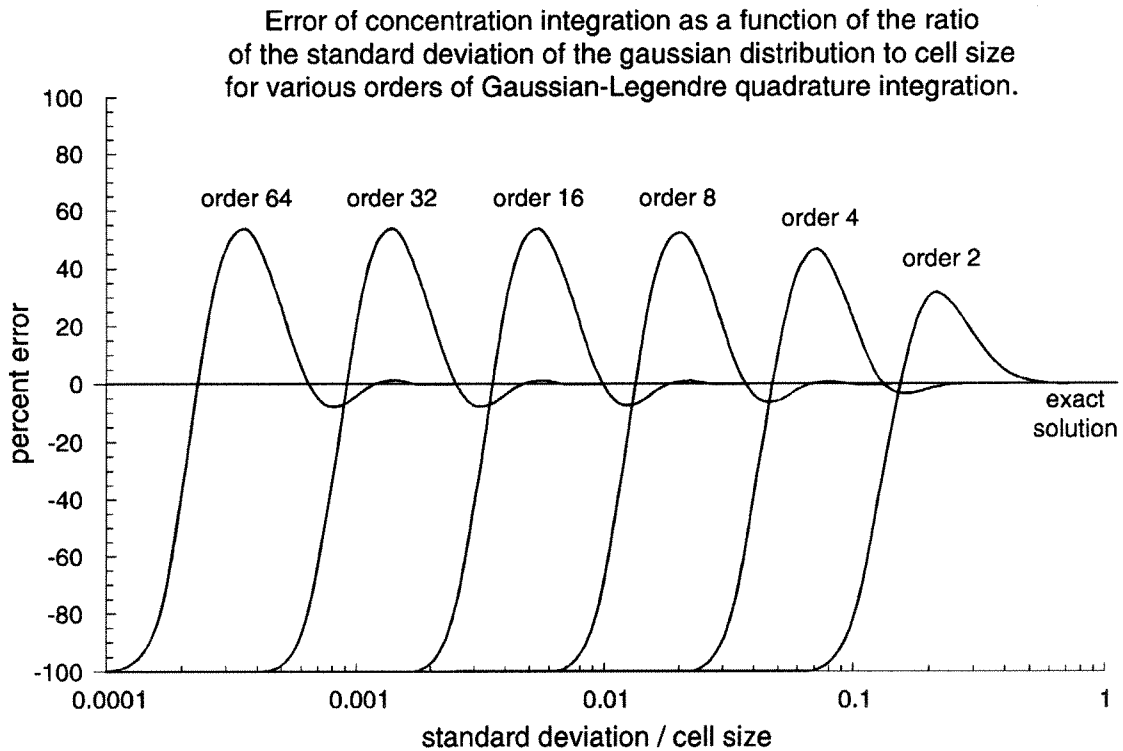


Figure 21 Integration error as a function of standard deviation vs cell size for Gaussian-Legendre quadrature integration.

individual cell masses after distribution:

$$\text{Mass Balance} = 100 \times \frac{M_{\text{initial}}}{\sum_{\text{cells}} M_{\text{cell}}} .$$

The Gaussian-Legendre quadrature integration works so well that this mass balance is very nearly 100% after accounting for the mass which has left the grid.

3.5.3 Solution for a Continuous Point Source

The continuous point source, or plume model has a similar development to that of the puff model. The differences are in the nature of the source, which is now described by a step input, rather than a pulse, and in the construction of the (X_L, X_T) coordinate system, discussed below. The solution is obtained by convolving the puff equation (67) over time. Bear (1972, 1979) presents the steady state solution ($t = \infty$)

$$c(X_L, X_T) = \frac{\dot{m} \times \exp\left(\frac{v_L X_L}{2 D_L}\right)}{\sqrt{2\pi D_L} \sqrt{2\pi D_T}} K_0 \left[\sqrt{\frac{v_L^2}{4 D_L} \left(\frac{X_L^2}{D_L} + \frac{X_T^2}{D_T} \right)} \right] \quad (73)$$

where \dot{m} is the mass release rate (mass/time) and K_0 is the Bessel function of the second kind of order zero. Wilson and Miller (1978) improved on this solution by accounting for the effects of porosity n , decay λ , and retardation R using

$$c(X_L, X_T) = \frac{\dot{m} \times \exp\left(\frac{v_L X_L}{2 D_L}\right)}{n \sqrt{2\pi D_L} \sqrt{2\pi D_T}} K_0 \left[\sqrt{\left(\frac{v_L^2}{4 D_L} + \lambda R \right) \left(\frac{X_L^2}{D_L} + \frac{X_T^2}{D_T} \right)} \right]. \quad (74)$$

However, both of these solutions assume a unit thickness, and both neglect to use the retarded velocities and dispersivities in the Bessel function. Accounting for these differences gives the equation

$$c(X_L, X_T) = \frac{\dot{m} \times \exp\left(\frac{v_L X_L}{2 D_L}\right)}{nb \sqrt{2\pi D_L} \sqrt{2\pi D_T}} K_0 \left[\sqrt{\left(\frac{(v_L/R)^2}{4 D_L/R} + \lambda R \right) \left(\frac{X_L^2}{D_L/R} + \frac{X_T^2}{D_T/R} \right)} \right]. \quad (75)$$

Charbeneau (unpublished notes) derives a simple solution by assuming that in steady state the longitudinal mixing is negligible, so that only transverse mixing is considered.

$$c(X_L, X_T) = \frac{in \times \exp\left(-\lambda R \frac{X_L}{v_L}\right) \times \exp\left(-\frac{X_T^2}{4 D_T \frac{X_L}{v_L}}\right)}{V_L n b \sqrt{4 \pi D_T \frac{X_L}{v_L}}}. \quad (76)$$

The differences between this solution and equation (75) diminish rapidly with distance from the source. Charbeneau's equation is computationally more efficient (without the numerically approximated Bessel function) and forms the basis for the plume model, illustrated in Figure 22.

Very similar to the structure of PorousPuff is the PorousPlume program. Most of the programming differences are in the construction of the (X_L, X_T) coordinates for each cell in the grid.

The plume solution given in equation (76) can be modified to accommodate a curving advective path by replacing X_L with the distance along the path and X_T with the perpendicular distance from the path line as shown in Figure 23.

For a given cell, an (X_L, X_T) coordinate system is constructed with an origin located at that point along the path which is closest to the cell. X_L is the distance along the path to the origin, and X_T is the distance from the origin to the cell, in effect molding the plume to the advection path. These values are used in the plume equation to calculate the concentration for the cell. Again, the integrated value for the cell is obtained by quadrature, using σ_T for the standard deviation. Although these solutions have been developed for homogeneous media, and are exact only for a uniform flow field, they are used in practice to

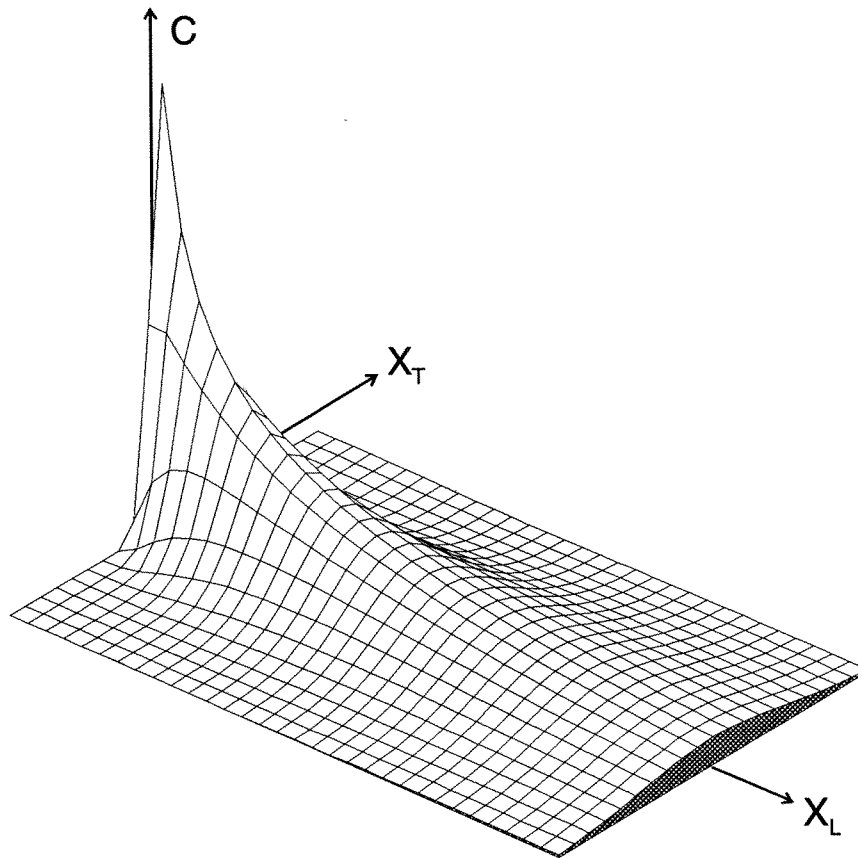


Figure 22 Surface representation of the plume model.

approximate contaminant transport in the general case. The PorousPuff and PorousPlume functions work with heterogeneous media in the form of grids of transmissivity, porosity, thickness, and head. The user must be aware of the behavior of these functions when applying them. They do not give an accurate representation of concentration distributions in strongly heterogeneous media, and their approximation is only as good as the region of application is homogeneous.

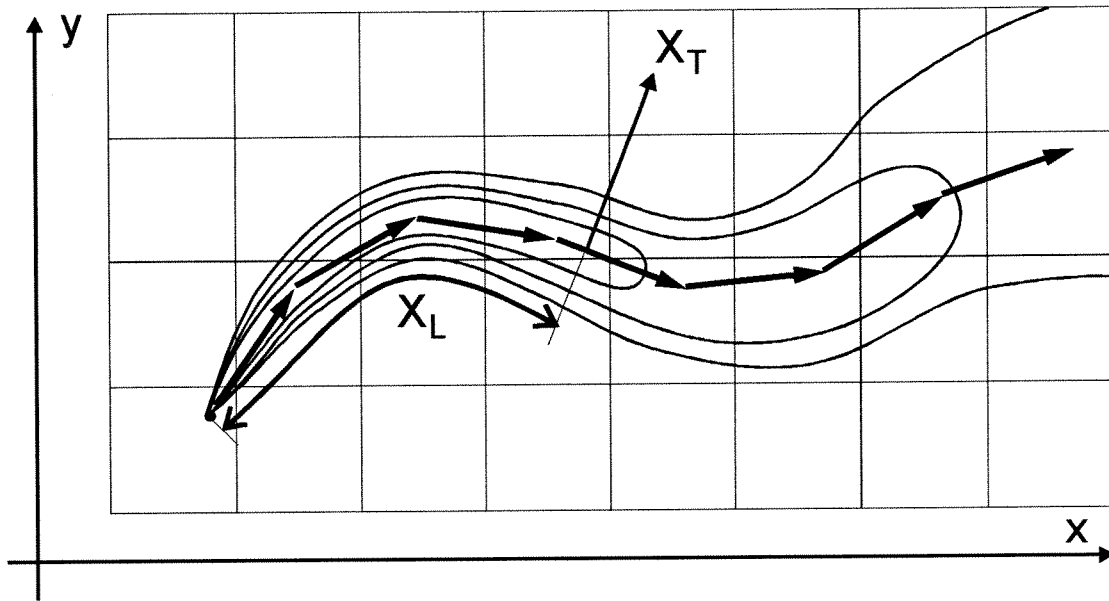


Figure 23 Dispersion plume calculated by equation (76).

Chapter 4

Programming for the GIS

Writing new functions for Grid is not a trivial undertaking. Even the simplest function requires the construction of additional modules of code in order to link the function together with Grid, a task which will probably require some help by the programmers at ESRI. This chapter documents the process of adding the DarcyFlow function to Grid, and it is hoped that by example readers will be able to add their own functions. The codes for all of the functions presented in this research are presented in the Appendix.

It is best to write a completely functional and debugged prototype of a function before building the associated programs for integrating it with Grid. This will give the programmer an idea of exactly what information is required by the function, and will have to be shuttled from the Grid command line to the subroutine which does the work. In the case of DarcyFlow, the names of four input grids and three output grids are needed. It is instructive to follow this information from the Grid command line to the `DarcyFlowProcess` subroutine, presenting the relevant code fragments along the way. The context of the fragments is found by referring to the complete codes in the Appendix.

4.1 Implementing the Functions In Grid

Arc/Info is a complex collection of programs, with various modules embedded in or linked to the main program. Associated programs, such as Grid, are initiated from Arc but have their own command prompts and functions. The linkage is often tight enough that functions native to one program can be called

from another. For example, the graphical display tools of the ArcPlot program can be used from within Grid to create objects in a display window.

Building these functions into Grid requires an intimate knowledge of the inner workings of Arc and Grid, and how to get these programs to share data and execution protocol. These protocols are complicated by the fact that Arc is written in FORTRAN and Grid, a newer program, in C (Menon, 1992).

This is the first time that functions have been written for Grid outside of ESRI (Menon, 1992), so it has opened the door of Grid programming to anyone with the Arc SDL (Software Development Libraries). Development is assumed to take place on a Sun workstation, using the Kernighan and Ritchie C compiler, cc.

4.2 Constructing the Grid Integration Programs

This section discusses the coding necessary to integrate a function into Grid, using DarcyFlow as an example. There are two ways to integrate a function: it may be linked into the Grid executable, GRID, or it may exist as a standalone program which is called by Grid. The method chosen is completely transparent to the user, but each has advantages and disadvantages. A function linked into GRID requires that considerably fewer intermediate subroutines need be written, but it requires relinking GRID during development and debugging. A separate program (DARCYFLOW is an example) is compiled and linked separately and requires that GRID be relinked only once. However, it becomes necessary to write four additional modules to make it run from Grid. Since this second method is more difficult, it is used here as an example. The DarcyFlow,

PorousPuff, and PorousPlume functions run as standalone programs, and ParticleTrack and DarcyTrack are built into the GRID executable.

The GRID program uses two important collections of subroutine modules, `aclisub` and `gclisub`, which contain codes for integrating functions like DarcyFlow with Arc and Grid, respectively. Since DARCYPFLOW is a separate program launched by Grid, and Grid is itself launched by Arc, the `aclisub` subroutines are necessary to translate between the C function calls of Grid and the FORTRAN calls of Arc. In short, the information typed by the user at the Grid prompt is parsed into strings by Grid, translated into FORTRAN strings, pushed onto the stack, and passed to Arc. These arguments are popped off the stack, checked for validity, translated back to C strings, pushed again onto the stack, and control is passed to the `gclisub` subroutine. (Functions which are linked into the GRID executable do not need `aclisub` modules since the `gclisub` subroutines are called directly from Grid). This pops the arguments off the stack, again checks their validity, and calls the pre-processor. The grids are opened (or created) and everything is set up for the function which does the work, `DarcyFlowProcess`.

Each subroutine in these modules has a unique name and is listed in Table 1 with its respective module (source file). Flow of control and information is from top to bottom (Menon, 1993).

In greater detail, let's follow the flow of information from the Grid prompt. The user invokes the DarcyFlow function at the Grid prompt by entering, for example,

Table 1 Subroutines and associated modules for the DarcyFlow function.

| program | module | routine | purpose | file name |
|-----------|---------|------------------|--|--------------------------|
| GRID | gclisub | GCLDarcyFlow | gets arguments from Grid executive | gclisub/ darcyflow.c |
| | | USGDarcyFlow | provides usage text for Grid executive | gclisub/ darcyflow.c |
| | aclisub | DRCYFL | uses standard Arc/Info subprogram execution method to launch DARCYFLOW | aclisub/ darcyflow.f |
| DARCYFLOW | darcy | DARCYFLOW | receives arguments from DRCYFL, calls DarcyFlow via FORTRAN77 binding | darcy/ darcy.f |
| | | darcyf_ | performs FORTRAN77 binding to DarcyFlow | darcy/ darcyflowf77.c |
| | | DarcyFlow | opens communications to data grids | darcy/ darcyflow.c |
| | | DarcyFlowProcess | performs calculations discussed in dissertation | darcy/ darcyflow.c |

Grid: **usage darcyflow**

```
Usage: (F) DARCYFLOW ( <head_elevation_grid>,
                        <porosity_grid>, <thickness_grid>,
                        <transmissivity_grid>,
                        <velocity_direction_grid>,
                        <velocity_magnitude_grid>)
```

Grid: **residual = DARCYFLOW(head, poro, thick, trans, dir, mag)**

The format of this command is specified in the usage text, which is coded into USGDarcyFlow. Grid calls the GCLDarcyFlow subroutine, supplying the name of the output grid (residual, in this case) and the number of arguments *n* which it has loaded onto the stack. The following code fragment is a standard C function declaration, in the style of Kernighan and Ritchie (1978):

```
GCLDarcyFlow( residual, n )
char *residual;
int n;
```

The members of the stack correspond to the names of the input grids. GCLDarcyFlow checks to assure that the number is correct, pops the arguments from the stack, checks to see that they are valid symbol names and that the input grids exist (and that output grids do not already exist), and converts the grid names to FORTRAN character strings. Using these strings, GCLDarcyFlow calls the FORTRAN subroutine DRCYFL and explicitly adds the lengths of the character strings as additional arguments (FORTRAN expects this since it does so implicitly):

```
drcyfl_( f residual, f headelev, f porosity, f thickns, f transmy, f flowdir,
         f flowvel, &ierr,
         MAXNCHARS, MAXNCHARS, MAXNCHARS, MAXNCHARS,
         MAXNCHARS, MAXNCHARS, MAXNCHARS );
```

and the control is passed to the FORTRAN subroutine DRCYFL:

```
SUBROUTINE DRCYFL(RESID,HEADEL,POROS,THICK,TRANSM,FLWDIR,FLWVEL,IERR)
```

DRCYFL then checks again the existence of the grids and executes the main FORTRAN program, DARCYPFLOW. The standard Arc/Info argument-passing mechanism using the ARGSYS module is used to pass the arguments and Grid environment to the DARCYPFLOW program:

```
CALL ARGSD
CALL ARGSCH (RESID)
CALL ARGSCH (HEADEL)
CALL ARGSCH (POROS)
CALL ARGSCH (THICK)
CALL ARGSCH (TRANSM)
CALL ARGSCH (FLWDIR)
```

```

CALL ARGSCH (FLWVEL)
CALL ARGSGR

CALL ARGEXE ('darcyflow', IERR)

```

This causes DARCYFLOW to execute as a child process spawned from GRID. DARCYFLOW then receives its arguments using the ARGSYS module, including the Grid environment variables (window, cellsize, and mask), and calls the C subroutine darcyf_, the FORTRAN binding to the entry point of the DarcyFlow function:

```

+ CALL DARCYF(RESID,HEADEL,POROS,THICK,TRANSM,FLWDIR,FLWVEL,
+           MASK,WINDOW,CELLSZ,IERR)

```

and the control is passed to darcyf_. Again, the lengths of the character string arguments are explicitly added to the argument list.

```

darcyf_(resid,headel,poros,thick,transm,flwdir,flwvel,
        mask>window,cell,ierr,lre,lhe,lpo,lth,ltr,lfd,lfv,lma)

char    *resid;
char    *headel;
char    *poros;
char    *thick;
char    *transm;
char    *flwdir;
char    *flwvel;
char    *mask;
double  *window;
double  *cell;
int     *ierr;
int     lre,lhe,lpo,lth,ltr,lfd,lfv,lma;

```

darcyf_ converts the character grid names from FORTRAN to C, and calls the DarcyFlow subroutine (which is in the C source file darcy/darcyflow.c):

```

*ierr = DarcyFlow ( c_resid, c_headel, c_poros, c_thick, c_flwdir, c_flwvel,
                  c_transm, c_mask, window, *cell );

```

and the control is passed to DarcyFlow:

```

int DarcyFlow( szResidual, szHeadElev, szPorosity, szThick, szTrans,
              szFlowDir, szFlowMag, mask, dfWinbox, dfCellsize )

char  *szResidual;    /* output grid: volume balance residual */
char  *szHeadElev;   /* grid of groundwater head elevations */
char  *szPorosity;   /* grid of formation porosities */
char  *szThick;     /* grid of formation thicknesses */
char  *szTrans;     /* grid of formation transmissivities */
char  *szFlowDir;   /* output grid of flow directions */
char  *szFlowMag;   /* output grid of flow velocities */
char  *mask;        /* grid environment : mask */
double dfWinbox[];  /* grid environment : window */
double dfCellsize; /* grid environment : cellsize */

```

(The change in programming style reflects the change in principal authorship. Tauxe is the main author for `darcy/darcyflow.c`, and the other codes were first coded by ESRI staff.) `DarcyFlow` opens communications channels to the existing input grids (using `CellLyrOpen`), and creates the output grids (using `CellLyrCreate`) which will be supplied with calculated data by the `DarcyFlowProcess` subroutine. The channels are passed as arguments to `DarcyFlowProcess`:

```

DarcyFlowProcess( hResidual, hHeadElev, hPorosity, hThick, hTrans, hFlowDir,
                 hFlowMag );

```

and the control is passed to `DarcyFlowProcess`:

```

void DarcyFlowProcess( gResidual, gHeadElev, gPorosity, gThick, gTrans,
                     gFlowDir, gFlowMag )

int  gResidual;    /* handle to vol. balance residual grid */
int  gHeadElev;   /* handle to head elevation grid */
int  gPorosity;   /* handle to porosity grid */
int  gThick;     /* handle to thickness grid */
int  gTrans;     /* handle to transmissivity grid */
int  gFlowDir;   /* handle to flow direction grid */
int  gFlowMag;   /* handle to flow velocity grid */

```

Here, the grids are filled with data calculated from the input grid data. Grid I/O is handled with functions like `GetWindowRow` and `PutWindowRow`. After `DarcyFlowProcess` has finished performing all of the calculations discussed in previous sections, `DarcyFlow` closes the grid channels and passes the control back to `darcyf_`, which

returns to DARCYPFLOW. Once DARCYPFLOW has completed execution, control is returned to its parent process in DRCYFL, which checks the error values. Finally, DRCYFL returns to GCLDarcyFlow and back to Grid.

The PorousPuff and PorousPlume functions are implemented similarly as separate programs, with the modification that some of the arguments are numerical values rather than names of grids. By contrast, the ParticleTrack and DarcyTrack functions are linked into the GRID executable. Besides a slightly different method of execution, these functions also produce Arc coverages. The coding for creation of these coverages was done by Sud Menon, the Grid Manager at ESRI, and due to its proprietary nature is not included in the Appendix.

4.3 Linking the New Version of Grid

Once all of these files have been created, there is the task of compiling and linking them into new programs. The DARCYPFLOW program is a separate executable, but functions within it use the Grid library. A new version of GRID is necessary as well, so that it can call DARCYPFLOW. After compiling the source codes, the object files must be linked with those supplied by ESRI. The Software Development Library (SDL), available from ESRI for outsiders who wish to develop their own enhancements to Arc/Info and its submodules, contains all of the object codes and make files necessary to reconstruct Grid. With the addition of the object files of the proposed functions, a new version of Grid can be linked which contains all of the functionality of the original program plus the new functions. The object files for the new functions are made by simply compiling (without linking) all of the individual C (or FORTRAN) codes:

```
cc -c filename.c
```

In order for Grid to recognize the new function, the makefile (gridexec/GNUMakefile) must contain references to it. For example, the makefile for the new Grid which includes all five of these new functions has the statements

```
PROGRAM=grid
OTHERS=\
$(NEWGRID)/aclisub/darcyflow.o\
$(NEWGRID)/aclisub/popuff.o\
$(NEWGRID)/aclisub/poplume.o\
$(NEWGRID)/gclisub/darcyflow.o\
$(NEWGRID)/gclisub/darcytrk.o\
$(NEWGRID)/gclisub/ptrack.o\
$(NEWGRID)/gclisub/ppuff.o\
$(NEWGRID)/gclisub/pplume.o\
$(NEWGRID)/dtrack/darcytrk.o\
$(NEWGRID)/ptrack/ptrack.o\
$(NEWGRID)/dispersion/trackcov.o\
$(NEWGRID)/dispersion/tcvsub.o\
$(NEWGRID)/dispersion/aatsub.o\
$(NEWGRID)/dispersion/interp.o\
$(NEWGRID)/dispersion/geometry.o\
$(NEWGRID)/dispersion/quadrat.o\
$(ARCOBJ)/slice/sliceentry.o\
$(ARCOBJ)/costpath/*.o\
$(ARCOBJ)/remaprtm/*.o\
$(ARCOBJ)/gfile/gfilesvf.o\
$(ARCOBJ)/arcplot/netexec.o\
$(ARCOBJ)/arcplot/angdis.o\
$(ARCOBJ)/arcplot/apusag.o\
...
```

where the top several lines (bolded text) have been added to the existing ones beginning with "\$(ARCOBJ)". Function declarations are added to the sym.h file:

```
extern int    GCLDarcyFlow(), GCLDarcyTrack(), GCLParticleTrack();
extern int    GCLPorousPuff(), GCLPorousPlume();
extern int    GCLAdjust();
extern int    BExpandShrink(), BuildSta(), BuildVat();
extern int    Combine(), ConvertRemap();
extern int    GCLCopy();
extern int    Corridor();
extern int    CostDistance(), CostBackLink(), CostAllocation();
...
```

Usage subroutine references are likewise added:

```
extern int    USGDarcyFlow(), USGDarcyTrack(), USGParticleTrack();
extern int    USGPorousPuff(), USGPorousPlume();
```



```

extern int    USGAdjust();
extern int    USGBExpandShrink(),USGBuildSta(),USGBuildVat();
extern int    USGCombine(), USGConvertRemap();
extern int    USGCopy();
extern int    USGCorridor();
extern int    USGCostDistance(), USGCostBackLink(), USGCostAllocation();
...

```

Function names used by Grid are also linked for their respective subroutines:

```

/*
 * Usage declaration of per layer functions
 */
struct {
    char          *name;
    int           (*fun)();
    int           (*usg)();
} swaps[] = {
    "DARCYFLOW",      GCLDarcyFlow,      USGDarcyFlow,
    "PARTICLETRACK", GCLParticleTrack, USGParticleTrack,
    "DARCYTRACK",    GCLDarcyTrack,    USGDarcyTrack,
    "POROUSPUFF",    GCLPorousPuff,    USGPorousPuff,
    "POROUSPLUME",   GCLPorousPlume,   USGPorousPlume,
    "ADJUST",        GCLAdjust,        USGAdjust,
    "COMBINE",       Combine,          USGCombine,
    "CORRIDOR",      Corridor,         USGCorridor,
    "COSTDISTANCE", CostDistance,     USGCostDistance,
    ...

```

With these modifications and the additional `aclisub` and `gclisub` modules, the makefile will generate a new version of Grid.

Chapter 5

Model Applications

This chapter puts into practice the new GIS groundwater modeling functions. First, a verification is performed by comparing the results generated by the functions to analytical results. This exercise is followed by an example of using the GIS to analyze an active groundwater problem: calculating the groundwater travel time and the potential location of a contaminant plume from a proposed radioactive waste repository.

5.1 Groundwater Analysis in Grid

In general, a groundwater flow and transport analysis using the new Grid functions will consist of several steps, outlined below. The following sections discuss the methods used for analysis of a particular groundwater problem.

- Prepare the hydrogeological datasets.
 - Select a base grid, with location, dimension, and cell size appropriate to the problem.
 - From field data or approximations, construct transmissivity, thickness, and porosity fields.
 - Geologic samples from bore holes are examined to estimate effective porosity, or estimates are based on measurements from similar strata.

- Saturated thickness is estimated from geologic maps, bore holes, and measurements of head elevation (in the case of unconfined flow).
- Pumping tests are usually used to calculate transmissivities.
- Construct a grid of regional steady state head elevations.
 - The head grid will be interpolated from existing point values measured from wells, or will be generated by an external groundwater simulation program.
- Import the head, transmissivity (or log transmissivity), thickness, and porosity data into Grid. Log transmissivity grids are most easily converted to transmissivity with map algebra within Grid.
 - If data already exist as an Arc polygon or point coverage, they may be converted to grid via POLYGRID or POINTGRID (ESRI, 1991b).
 - If data are prepared as regular block-centered finite difference values, conforming to the Grid data model, format the data files into "asciigrid" files⁸ and import via ASCIIGRID. Corner-centered data, such as a Surfer grid file, will require a shift in coordinates.
 - If data are scattered points, they may be formatted into "generate" files and cell values calculated by one of Grid's surface interpolation functions such as IDW, KRIGING, SPLINE, or TREND.

⁸The "asciigrid" format is provided in Arc/Info for transferring information in and out of Grid, which uses a proprietary file structure. The format of the text file is straightforward, and discussed in the documentation for Arc (ESRI, 1991a).

- Test the data for consistency.
 - Use DarcyFlow to generate a volume balance residual grid, and examine this for unaccounted-for sources or sinks. Sinks may be removed with Grid's FILL function.
 - Modify the head or transmissivity grids to achieve consistency.
- Determine the flow field over the region with DarcyFlow.
 - The grid fluid velocities may be used with thickness and porosity to generate fluid flux data, which is useful in determining the severity of volume balance errors.
- Use the particle-tracking function to determine path lines and travel times.
 - From a selected source location, use ParticleTrack (or DarcyTrack) to approximate the streamline through that point. A particular time of interest may be specified, but by default the path is followed until it leaves the grid.
 - If a “field” of path lines is desired, the ParticleTrack function may be included in a DOCELL loop with source locations positioned at the center of each cell.
- Perform advection-dispersion modeling with PorousPuff or PorousPlume.
 - Select values for the additional parameters required for advection-dispersion modeling, as discussed in section 3.5.
 - Dispersivities are best determined from tracer test data. Often these tests are not available depending on the nature of the site or the budget, so one may turn to published values for a similar geological environments. In the absence of all other information, PorousPuff and Porous-

Plume are equipped with default values in an attempt to preclude gross errors.

- Retardation factors are also best determined from tracer test data at the site, but would involve introducing the contaminant to the aquifer. Estimates may be made based on theory or on published values (Mercer and Waddell, 1992).
- Decay coefficients for radioactive substances are well known. Other uses of the decay coefficient are discussed in section 3.5.
- Creative use of Grid's map algebra can enhance the use of these functions. Command files or programs may be written in AML to perform complex and repetitive tasks for advanced modeling.

5.2 Model Verification

Tests were run to examine the accuracy and internal consistency of the proposed functions. This section discusses four tests which compare the results of the Grid functions to analytical solutions: 1) flow and travel times to a single production well in a uniform flow field (capture well problem), 2) flow and travel times to an injection/production well pair (well dipole problem), 3) dispersion of a puff through a series of time steps in a uniform flow field (stepped puff simulation), and 4) comparison of a plume to a convolved series of puffs (convolved plume simulation).

5.2.1 The Capture Well Simulation

A classic groundwater hydraulics problem with an analytical solution concerns the placement of a single injection well in an infinite uniform confined aquifer. Bear and Jacobs (1965) and Bear (1979) discuss this model in detail, and the results are useful in our similar problem of a production well in a uniform confined aquifer, or capture well. By inverting the sign of pumpage Q , we can modify Bear's equations, as presented below. Analytical solutions exist for the head potential field, the stream function, and the travel time from any point to the well. These solutions are well suited for testing the DarcyFlow, ParticleTrack, and DarcyTrack functions, since they will test the proper construction of the flow field and volume balance residual by DarcyFlow, the propagation of flow paths by ParticleTrack, and the approximation of streamlines by DarcyTrack.

For simplicity, let us place the well at the origin, and orient the x axis parallel with the direction of the uniform flow field, so that with no pumping the head field would dip to $+x$ as a planar surface. Transmissivity, porosity, and saturated thickness are constant for the uniform field in a confined aquifer. The head value at any (x,y) location is given by superposition of the well function and the uniform flow field (modified from Bear, 1979):

$$h(x,y) = \frac{Q}{4\pi T} \ln(x^2 + y^2) - \frac{Ux}{T}, \quad (77)$$

where

h \equiv head (units of length)

Q \equiv pumping rate of the production well (units of length³/time)

T \equiv aquifer transmissivity (units of length²/time)

$U \equiv$ the aquifer flux vector (units of $\text{length}^2/\text{time}$, or $\text{length}^3/\text{length time}$).

Figure 24 is a mesh plot the head surface. Now we can construct the problem in Grid. The head field described by equation (77) is calculated independently (for example, by a spreadsheet) and formatted into an ASCII text file which is imported into Grid with the ASCIIGRID command:

```
Arc: ASCIIGRID head.txt head float
```

In this case, a grid of 100×100 cells was used, with a cell size of 10 meters square, centered at the origin. The following values were selected to simulate a uniform aquifer:

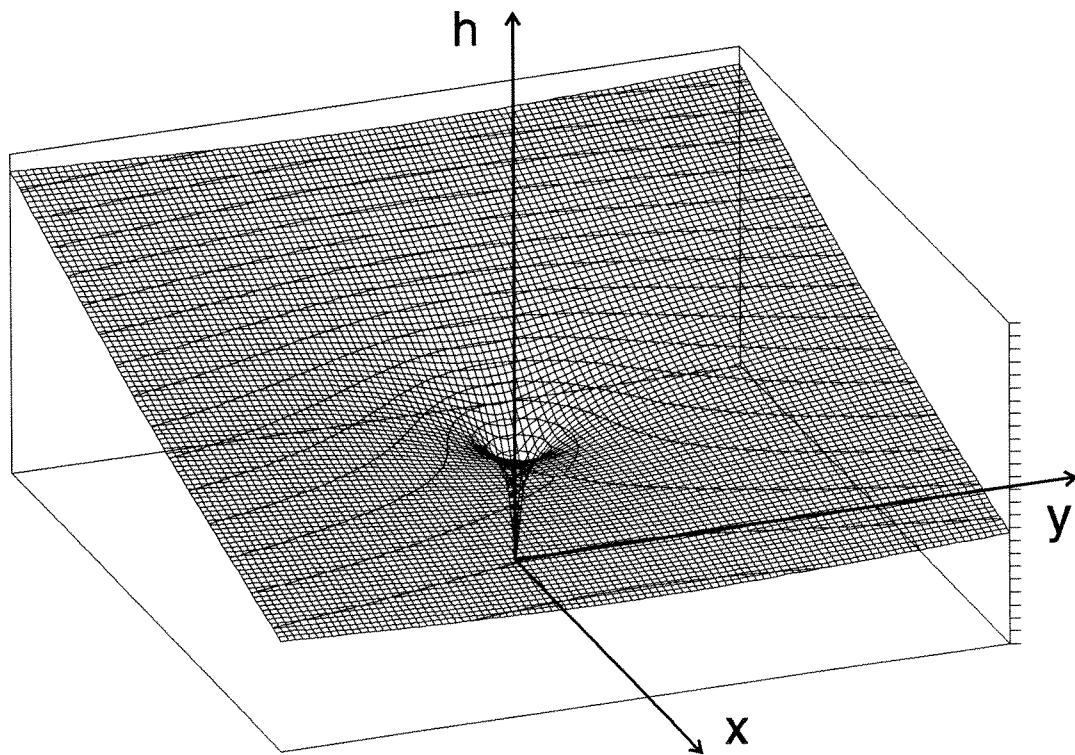


Figure 24 Head field of the capture well problem. Vertical extent 0 to 25 m.

| | | |
|---|--------------------------|-------------------|
| Q | = 20.3 m ³ /d | well pumping rate |
| T | = 1.32 m ² /d | transmissivity |
| U | = 0.02 m ² /d | aquifer flux |
| n | = 0.35 | porosity |
| b | = 11 m | thickness |

The constant-valued grids T, n, and b are easily generated by map algebra:

```
Grid: poro = 0.35
Grid: trans = 1.32
Grid: thick = 11.0
```

DarcyFlow is used to calculate the flow field and volume balance residual from these with the command

```
Grid: res = DARCYFLOW( head, poro, thick, trans, dir, mag )
```

which generates the grids res, dir, and mag, containing values for each cell's volume balance residual, flow direction, and flow magnitude, respectively. The residual grid is everywhere zero except at the origin (at the well) as expected, and the flow direction and magnitude are illustrated in Figure 25 and Figure 26. Units are consistent with those used for the input grids, so residual is in m³/d and flow magnitude in m/d. Flow directions are recorded in geographic coordinates (degrees clockwise from north) in keeping with Arc/Info conventions.

The stream function for the capture well problem is also modified from Bear (1979):

$$\Psi(x,y) = \frac{Q}{2\pi T} \tan^{-1}\left(\frac{y}{x}\right) - \frac{U}{T}y. \quad (78)$$

The ParticleTrack function approximates a path line, producing particle tracks from initial locations in the flow field. Since the flow is steady-state, the path line and streamline are coincident. The results of several applications of

Flow Field Surrounding the Capture Well

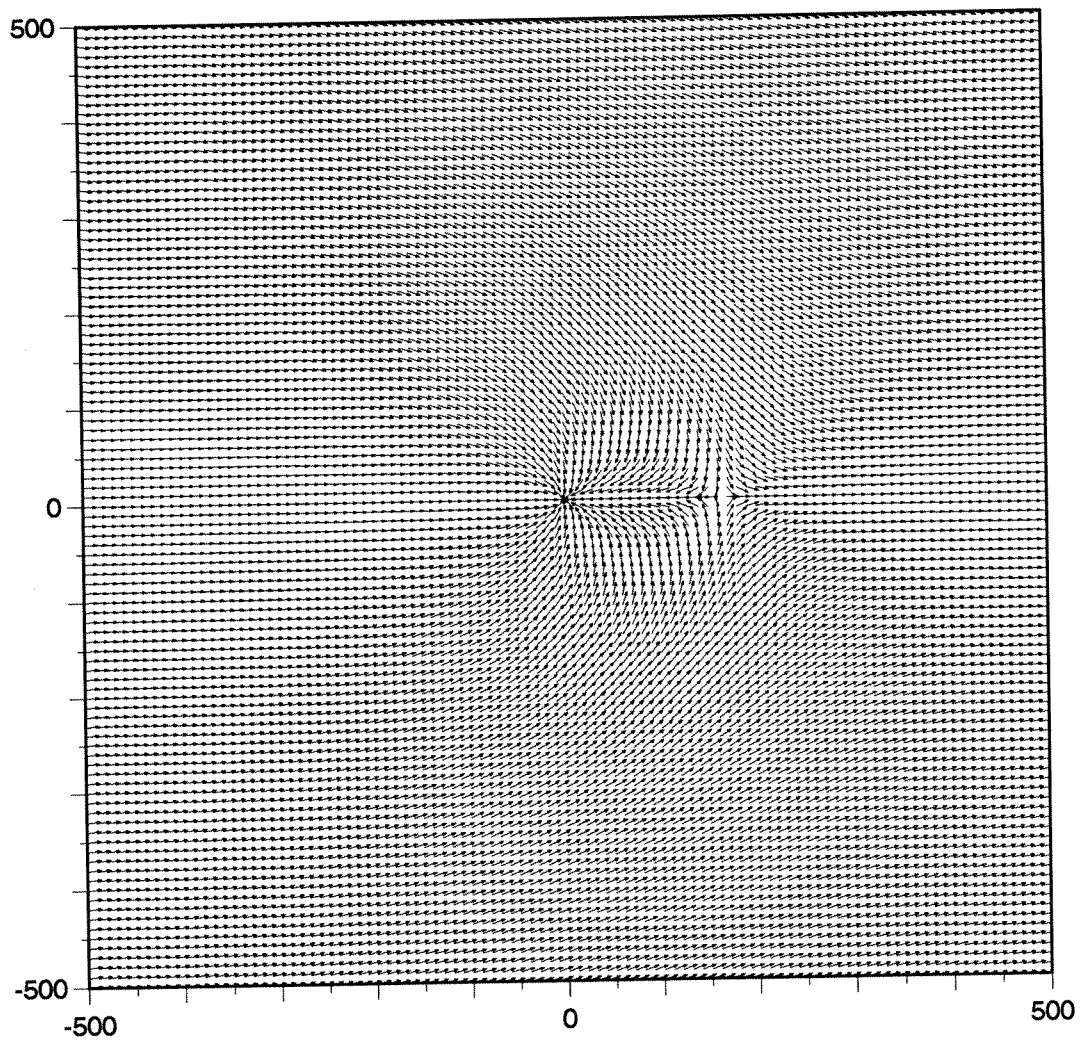


Figure 25 Flow direction vectors for the capture well, generated by DarcyFlow.

Flow Magnitude Surrounding the Capture Well

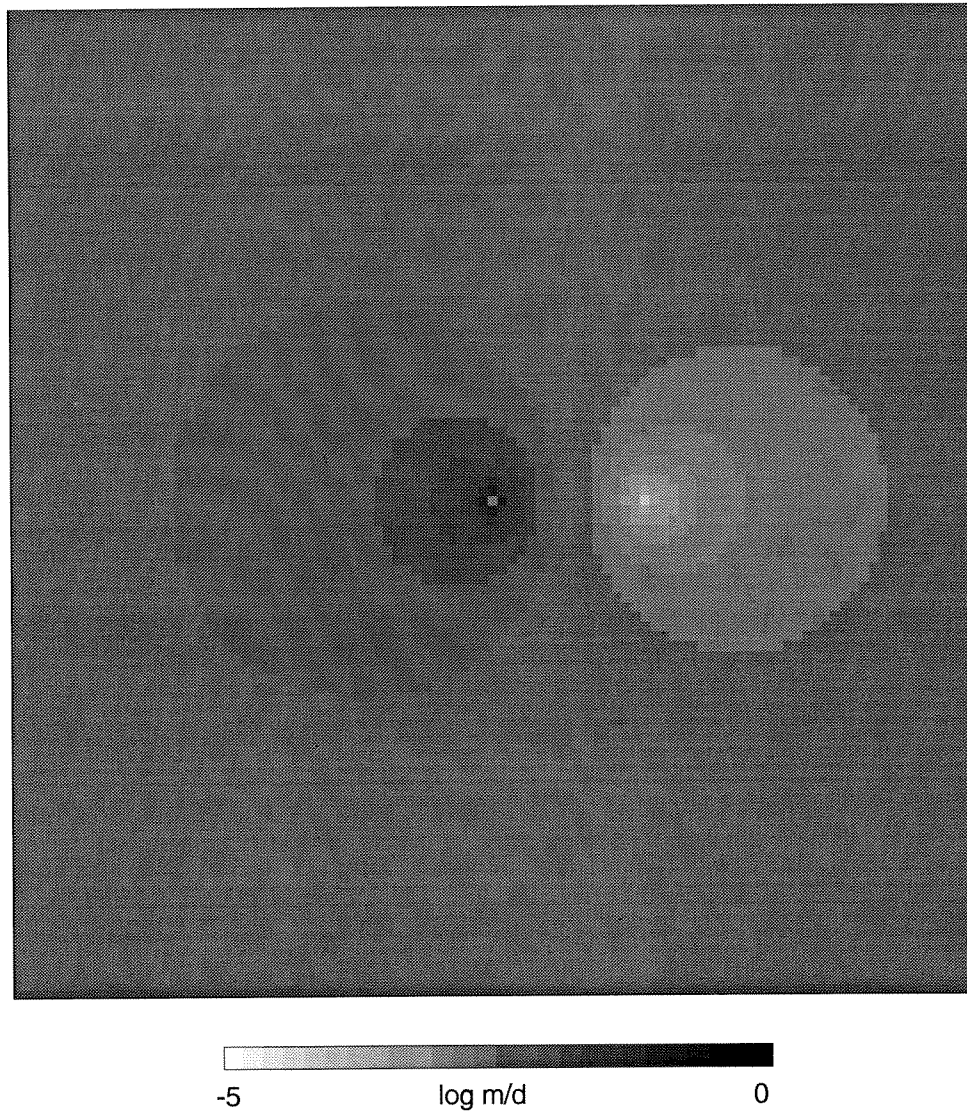


Figure 26 Flow magnitude for the capture well, generated by DarcyFlow.

ParticleTrack to the capture well problem are presented in Figure 27. The paths generated by DarcyTrack, which calculates piecewise streamlines, are essentially identical, confirming both functions (Figure 28). It is interesting to examine how well these paths approximate the analytical streamlines, since the error will accumulate along the sequentially computed path. Since every (x,y) point along the streamline should by definition correspond to a constant value of Ψ , we can examine how the actual value of Ψ according to equation (78) varies as the path progresses. Figure 29 shows the deviation from the streamline of a path generated by ParticleTrack, and Figure 30 by DarcyTrack, expressed as percent error. The path chosen is the fourth from the top in the streamline figures, the outermost path to be captured by the well. Until the last several points, where the flow field becomes extreme near the well and is not well approximated by a bilinear interpolation, the error is less than 0.01%, and most of the path is below 0.001%. It is also worth noting that ParticleTrack produces a smoother deviation than DarcyTrack, which oscillates. In both cases, the degree of error is quite acceptable for most purposes.

Another test of the tracking functions is to compare the predicted vs analytical travel times from a point (x,y) to capture by the production well at the origin. This tests the calculation of the magnitude of the flow velocity vectors, rather than the location of consecutive points. The analytical solution for travel times,

$$t(x,y) = \frac{nbQ}{2\pi U^2} \left[\ln \left(\frac{\sin \left[\tan^{-1} \left(\frac{y}{-x} \right) \right]}{\sin \left[\tan^{-1} \left(\frac{y}{-x} \right) + \frac{2\pi Uy}{Q} \right]} \right) - \frac{2\pi Ux}{Q} \right], \quad (79)$$

Streamlines Surrounding Capture Well
Calculations by ParticleTrack

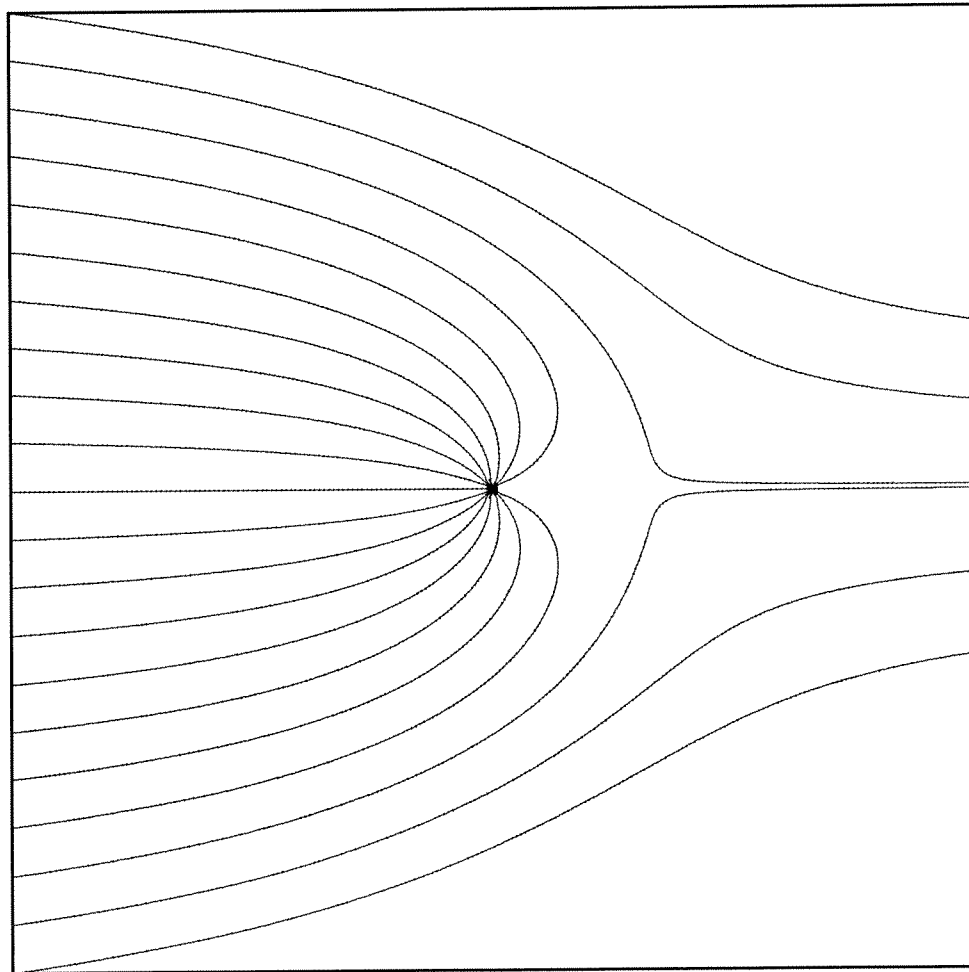


Figure 27 Streamlines for the capture well, generated by ParticleTrack.

Streamlines Surrounding Capture Well
Calculations by DarcyTrack

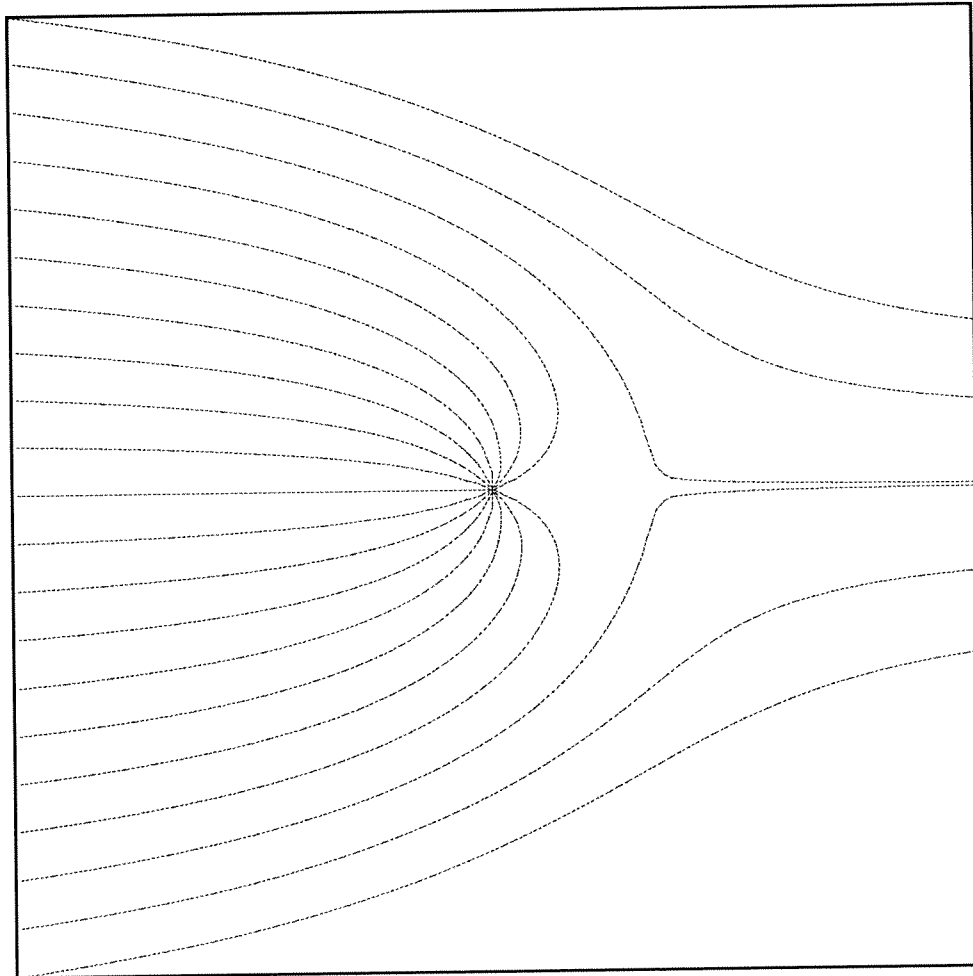


Figure 28 Streamlines for the capture well, generated by DarcyTrack.

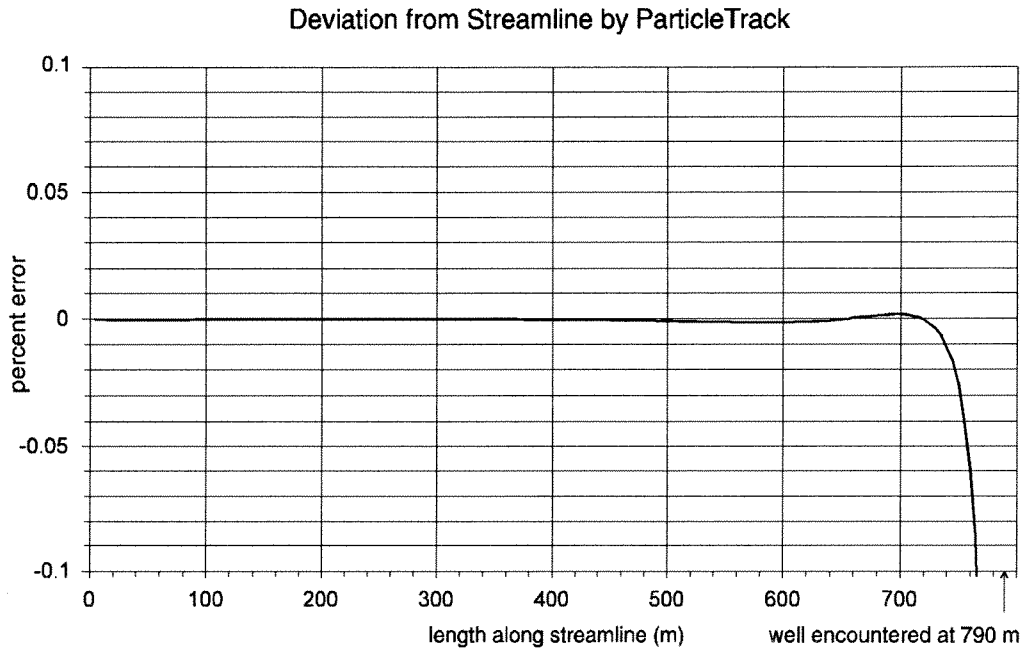


Figure 29 Deviation of ParticleTrack streamlines for the capture well.

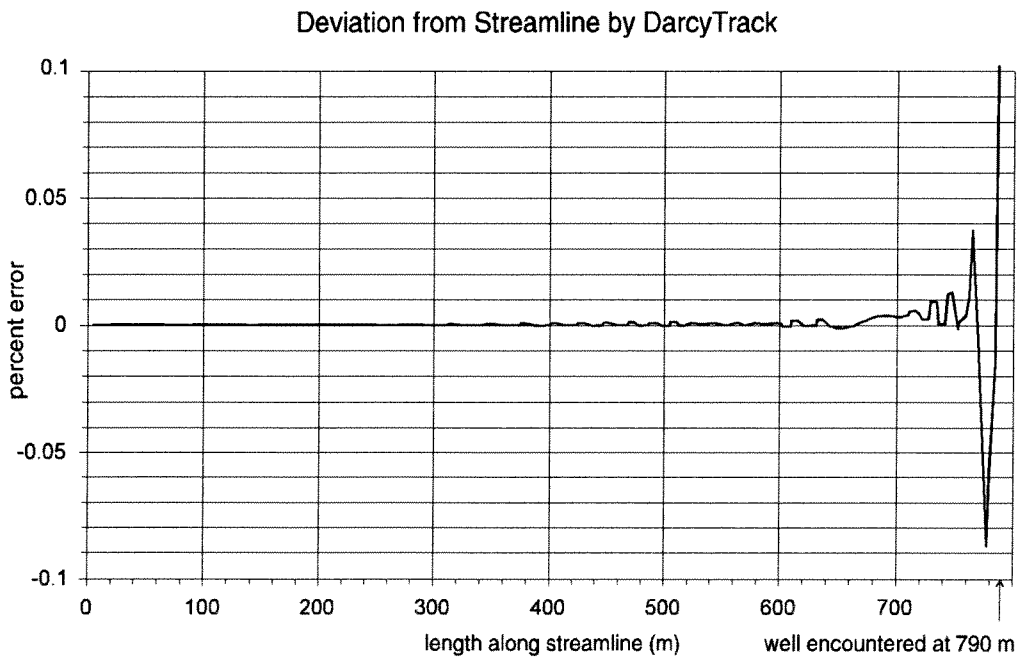


Figure 30 Deviation of DarcyTrack streamlines for the capture well.

is similar to that given by Bear and Jacobs (1965) for travel time from a injection well. The results of a comparison of travel times for the streamlines shown in Figure 27 and Figure 28 are shown in Figure 31. The accuracy of the travel time calculations and the small deviations from the streamlines presented above are a positive verification of both the ParticleTrack and DarcyTrack functions, which are based on independent algorithms. ParticleTrack, even though algorithmically and computationally much simpler than DarcyTrack, does an excellent job at calculating streamlines. It also serves as a general-purpose method for tracking particles through any flow field, while DarcyTrack tracks them through a potential field.

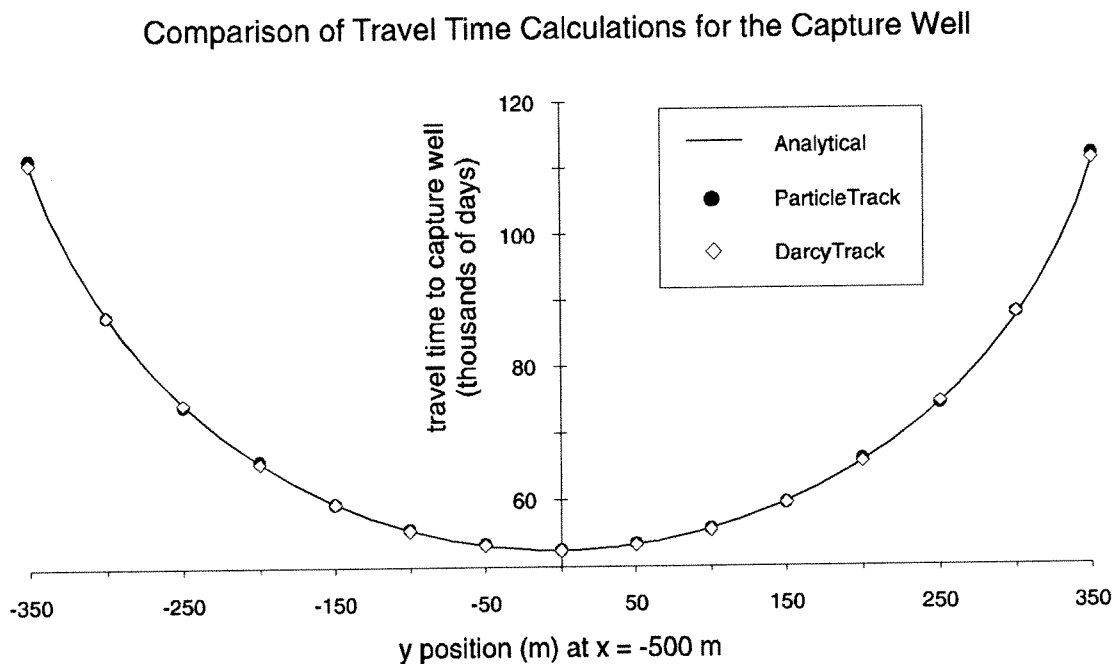


Figure 31 Comparison of capture well travel times by ParticleTrack, DarcyTrack, and the analytical solution.

5.2.2 The Well Dipole Simulation

Another classic problem in well hydraulics is examined to further verify these functions: the injection/production well pair, or dipole. Like the capture well problem, analytical solutions exist for the head potential, streamlines, and travel times.

For the purposes of the simulation, let us place the wells on the x axis, equidistant from the origin; the injection well at (-200, 0) and the production well at (200, 0) on the same grid layout used for the capture well. Again, transmissivity, porosity, and thickness are constant. The head value at any (x,y) location is given by superposition of the two well functions:

$$h(x,y) = \frac{Q}{4 \pi T} \ln \left[\frac{(x-L)^2 + y^2}{(x+L)^2 + y^2} \right], \quad (80)$$

Figure 32 illustrates the shape of this head field. Again, we can construct the problem in Grid by calculating the head field independently and importing into Grid with ASCII GRID. A new set of parameter values is selected:

| | | |
|---|--------------------------|-----------------------------|
| Q | = ±730 m ³ /d | well pumping rate |
| L | = 200 m | half-distance between wells |
| T | = 0.55 m ² /d | transmissivity |
| n | = 0.35 | porosity |
| b | = 11 m | thickness |

DarcyFlow is again used to calculate the flow field and volume balance residual grids res, dir, and mag. Again, the residual grid is everywhere zero except at the wells as expected, and the flow direction and magnitude are illustrated in Figure 33 and Figure 34.

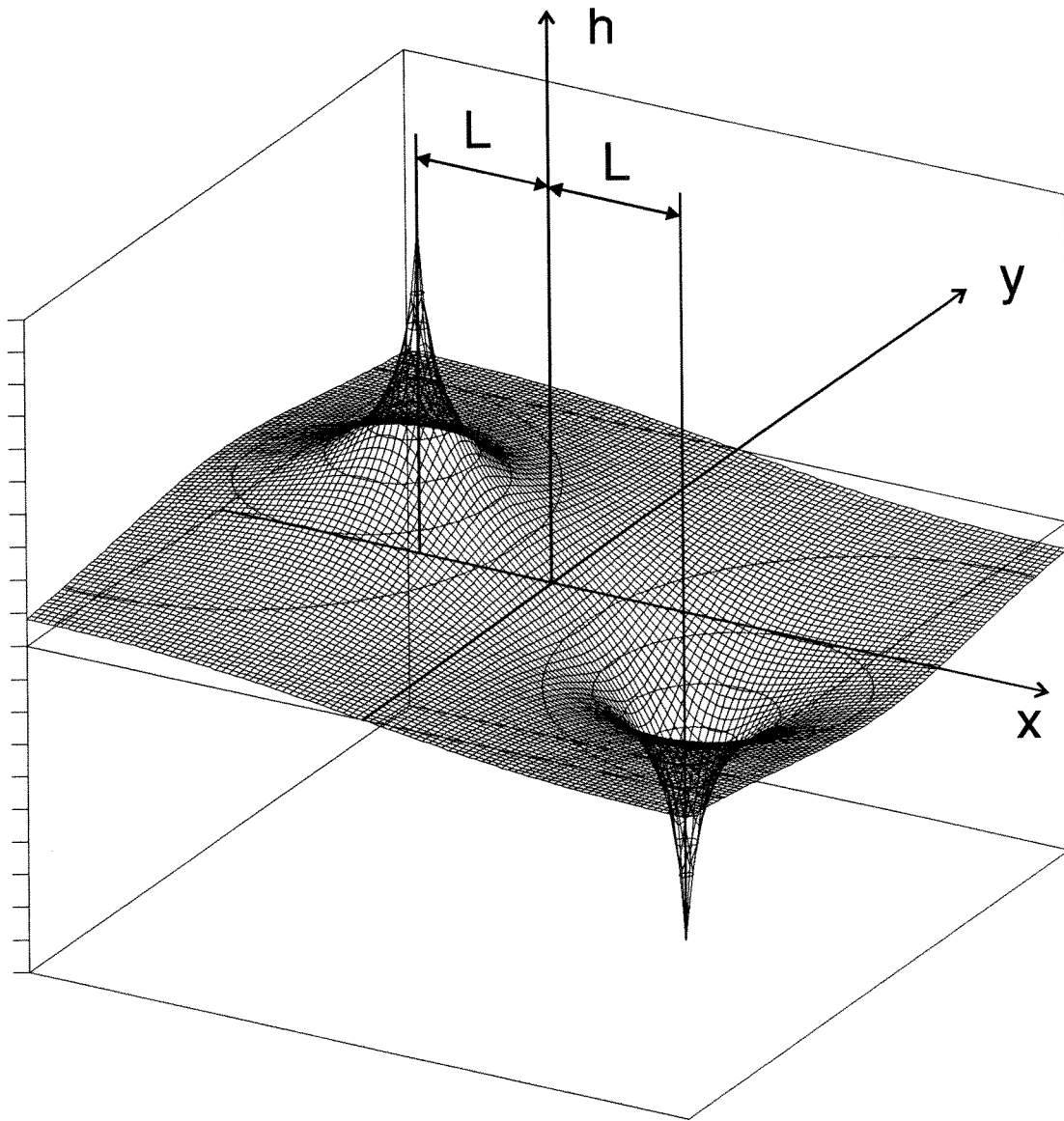


Figure 32 Head field of the well dipole problem. Contour interval 100 m.

Flow Field for Injection/Production Well Dipole

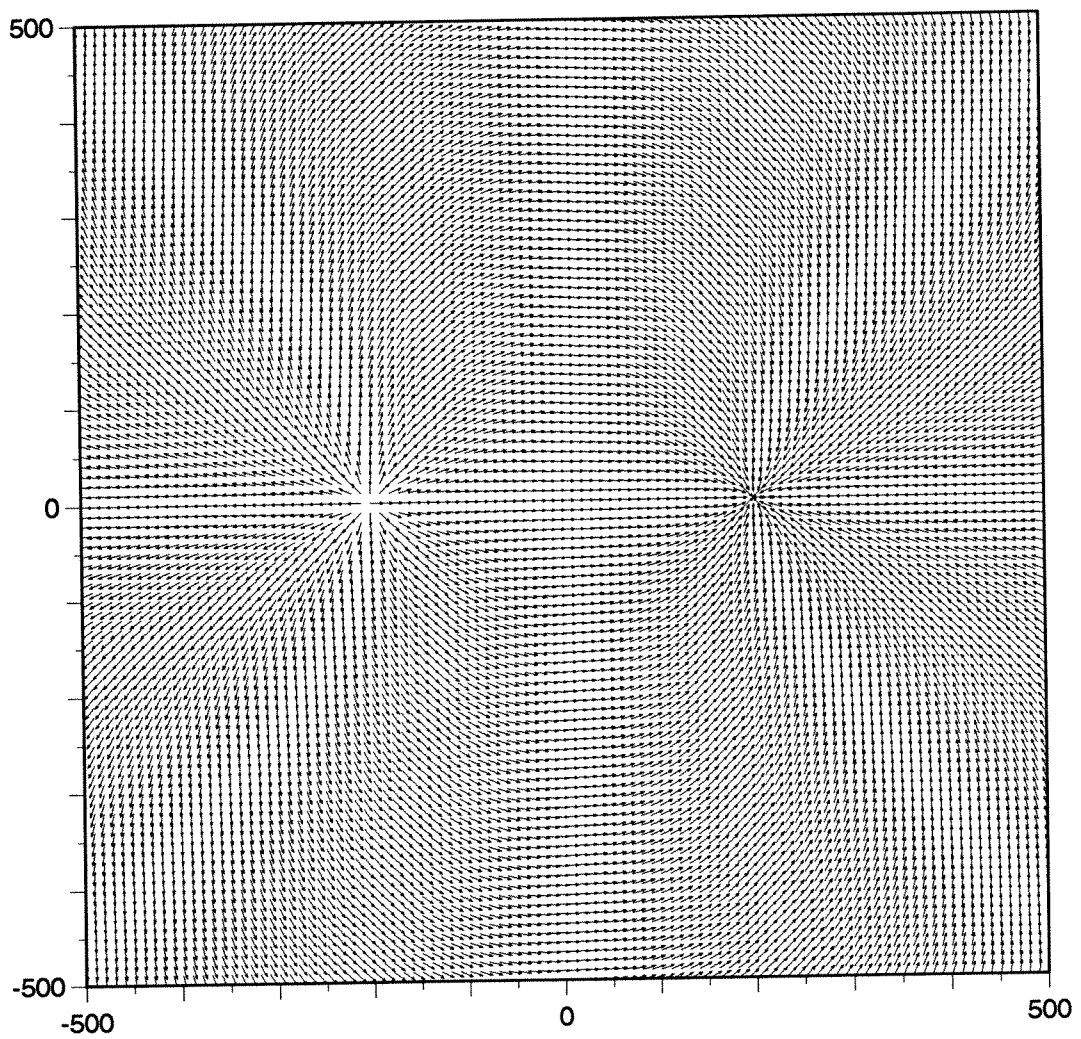


Figure 33 Flow direction vectors for the well dipole, generated by DarcyFlow.

Flow Magnitude Surrounding the Well Dipole

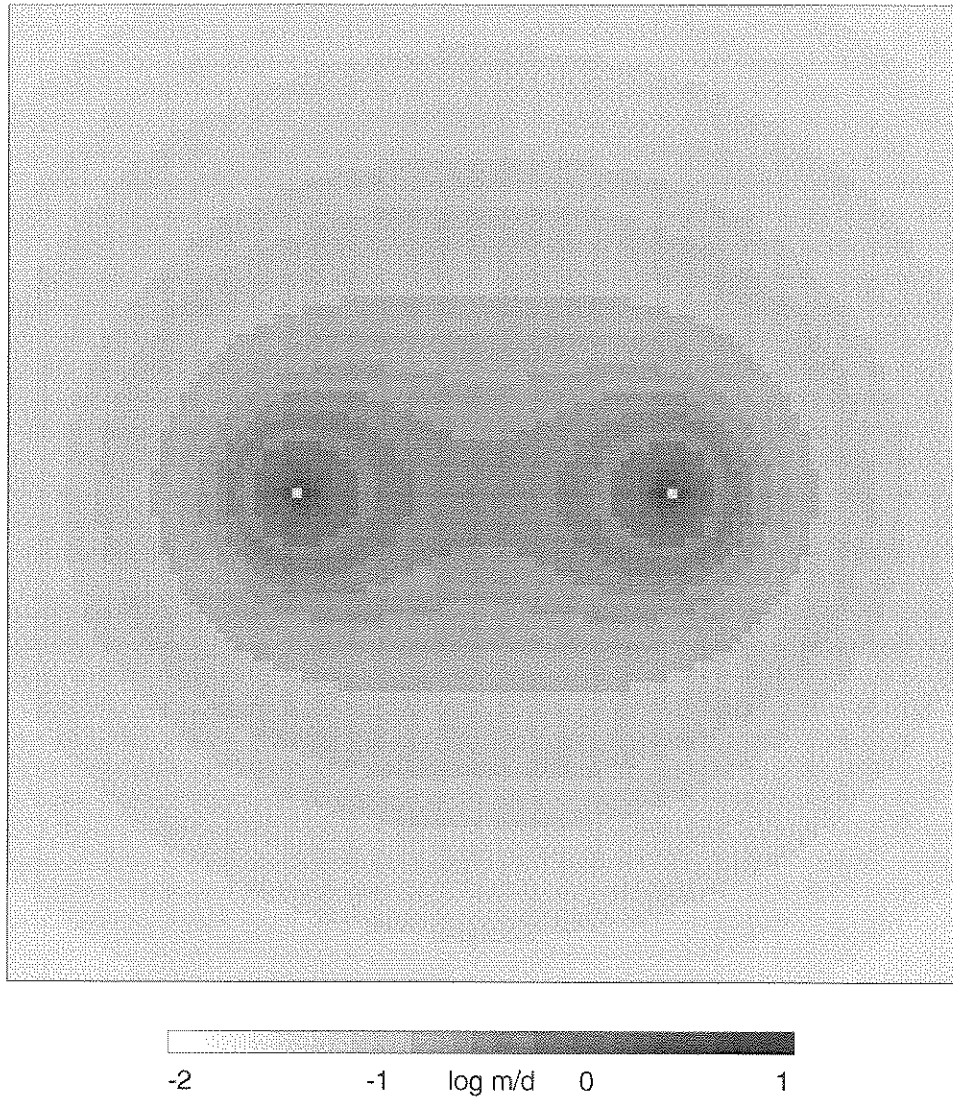


Figure 34 Flow magnitude for the well dipole, generated by DarcyFlow.

In determining streamlines, it is useful to employ the concept of fractional breakthrough. In a steady state dipole system, all the fluid pumped into the injection well will at infinite time find its way to the production well. Imagine a large number N of particles simultaneously introduced via the injection well at $x=-L$. The first fluid particle to complete the circuit to the production well at $x=+L$ will follow the shortest path: a line of length $2L$ along the x axis. As shown in Figure 35, the particle which begins its journey in the $+y$ direction will follow a semicircular path (as do all particles) arriving at the production well traveling in the $-y$ direction. By the time this particle arrives, half of all the particles ($N/2$) have arrived as well. This corresponds to a fractional breakthrough $F = 1/2$. In an infinite medium, F will asymptotically approach unity as time approaches infinity.

For each value of F , there is a corresponding value of the stream function

$$\Psi(F) = \frac{Q}{4T} (1 - F), \quad (81)$$

and a corresponding time of breakthrough $t_F(F)$, which is the travel time of a particle which has followed the streamline $\Psi(F)$ from injection to withdrawal (after Grove and Beetem, 1971):

$$t_F(F) = \frac{4\pi n b L^2}{Q} \left(\frac{1 - \pi \cdot F \cdot \cot(\pi \cdot F)}{\sin^2(\pi \cdot F)} \right). \quad (82)$$

or, in terms of Ψ ,

$$t_F(\Psi) = \frac{4\pi n b L^2}{Q} \left(\frac{1 - \left(\pi - \frac{4\pi T \Psi}{Q} \right) \cdot \cot \left(\pi - \frac{4\pi T \Psi}{Q} \right)}{\sin^2 \left(\pi - \frac{4\pi T \Psi}{Q} \right)} \right). \quad (83)$$

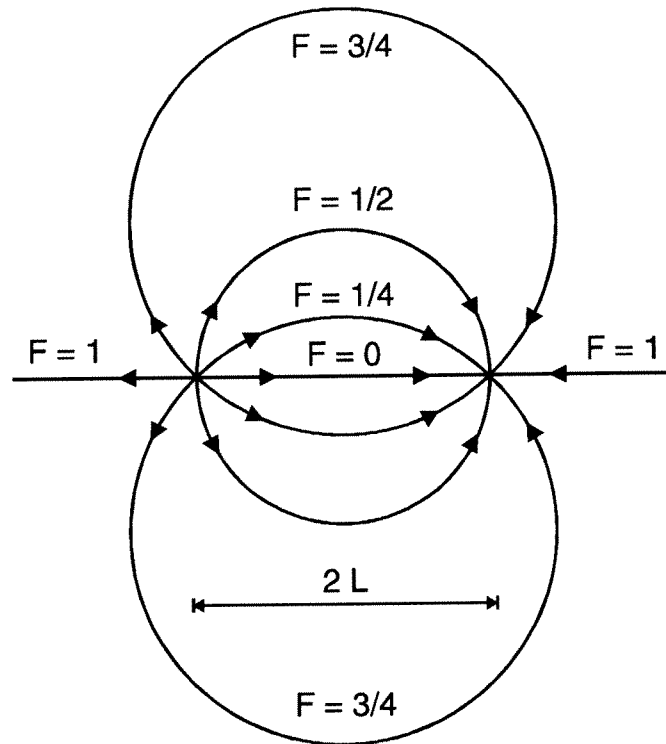


Figure 35 Flow streamlines and corresponding fractional breakthrough for the well dipole.

In testing ParticleTrack and DarcyTrack against equations (81) and (82), it is necessary to choose particle starting locations far from the injection well, since the head field is poorly represented very near the well (due to the bilinear interpolation). Starting positions along the $+y$ axis were chosen to correspond to particular F values. In order to determine the y values of the stream function at $x = 0$ for various values of F , we begin by defining the stream function for the well dipole:

$$\Psi(x,y) = \frac{Q}{2\pi T} \left[\tan^{-1}\left(\frac{y}{x-L}\right) - \tan^{-1}\left(\frac{y}{x+L}\right) \right]. \quad (84)$$

The stream function may also be written as an equation for a circle with a center

at $x = 0$, $y = \frac{L}{\tan\left(\frac{2\pi\Gamma\Psi}{Q}\right)}$ and a radius $r = L\sqrt{1 + \frac{1}{\tan^2\left(\frac{2\pi\Gamma\Psi}{Q}\right)}}$ (Grove and Beetem,

1971):

$$x^2 + \left[y - \frac{L}{\tan\left(\frac{2\pi\Gamma\Psi}{Q}\right)} \right]^2 = \left[L\sqrt{1 + \frac{1}{\tan^2\left(\frac{2\pi\Gamma\Psi}{Q}\right)}} \right]^2. \quad (85)$$

We can reformulate this as a curve expressing Ψ in terms of y : $\Psi_x(y)$. This gives

$$\Psi_x(y) = \sqrt{\left[L\sqrt{1 + \frac{1}{\tan^2\left(\frac{2\pi\Gamma\Psi}{Q}\right)}} \right]^2 - \left[y - \frac{L}{\tan\left(\frac{2\pi\Gamma\Psi}{Q}\right)} \right]^2}. \quad (86)$$

The intersections with the y axis are given by the roots of equation (86):

$$y_{(x=0)} = L \left(\frac{1 \pm \sqrt{1 + \tan^2\left(\frac{2\pi\Gamma\Psi}{Q}\right)}}{\tan\left(\frac{2\pi\Gamma\Psi}{Q}\right)} \right). \quad (87)$$

Equation (81) relates Ψ and F so that we may write this in terms of F . The starting locations along the $+y$ axis for the streamlines are given by positive values of

$$y_{(x=0)} = L \left(\frac{1 \pm \sqrt{1 + \tan^2(\pi - \pi F)}}{\tan(\pi - \pi F)} \right). \quad (88)$$

The travel times (to the production well) of particles released from these locations are by symmetry one half of t_F . F values of 1/20, 1/8, 1/4, 3/8, 1/2, 5/8, and 3/4 were selected to test a range of travel paths. The results for paths in the first quadrant calculated by ParticleTrack and DarcyTrack are presented in Figure 36 and Figure 37, respectively. Similar paths for the fourth quadrant may be generated by reflecting across the x axis. A comparison of calculated vs predicted travel times shows very good agreement in Figure 38.

Streamlines for Injection/Production Well Dipole Calculations by ParticleTrack

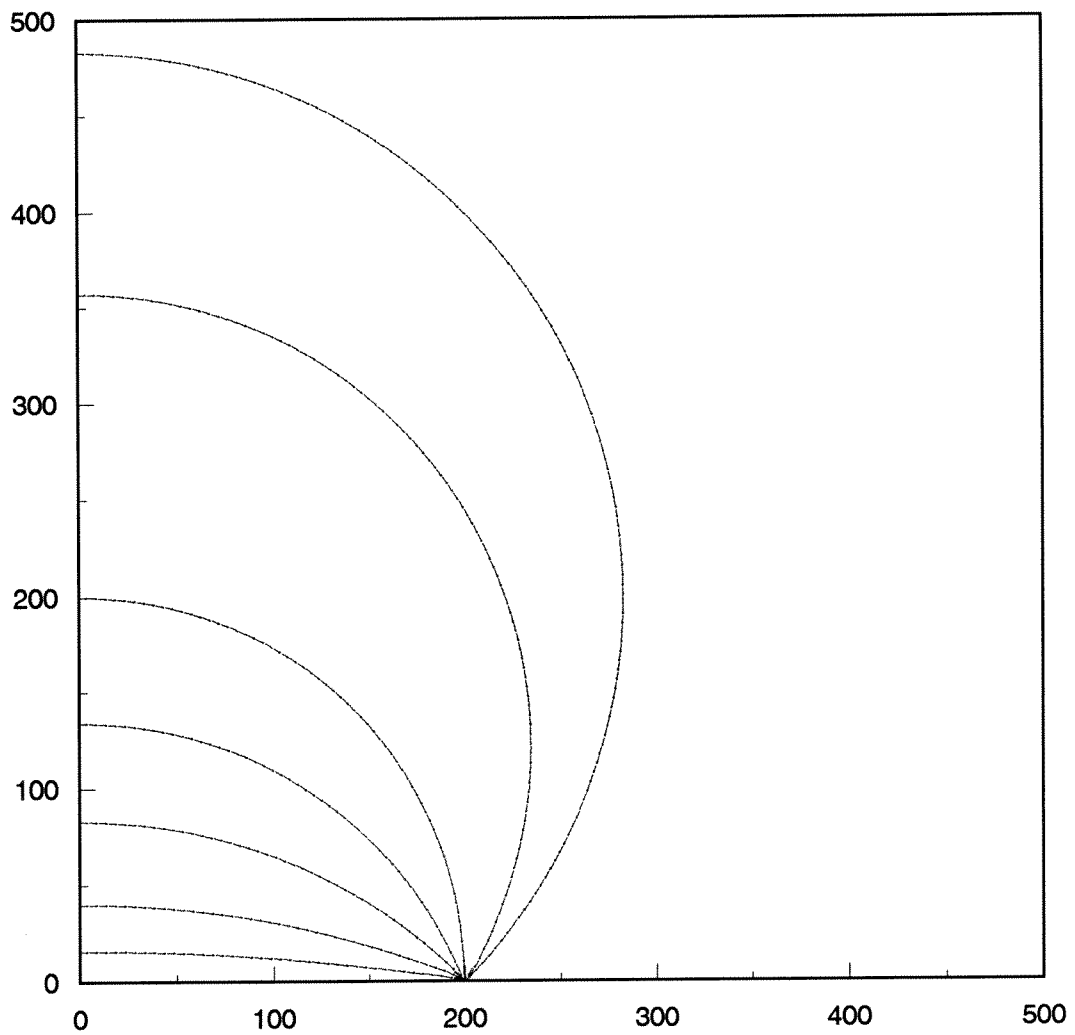


Figure 36 Streamlines for the well dipole, generated by ParticleTrack.

Streamlines for Injection/Production Well Dipole Calculated by DarcyTrack

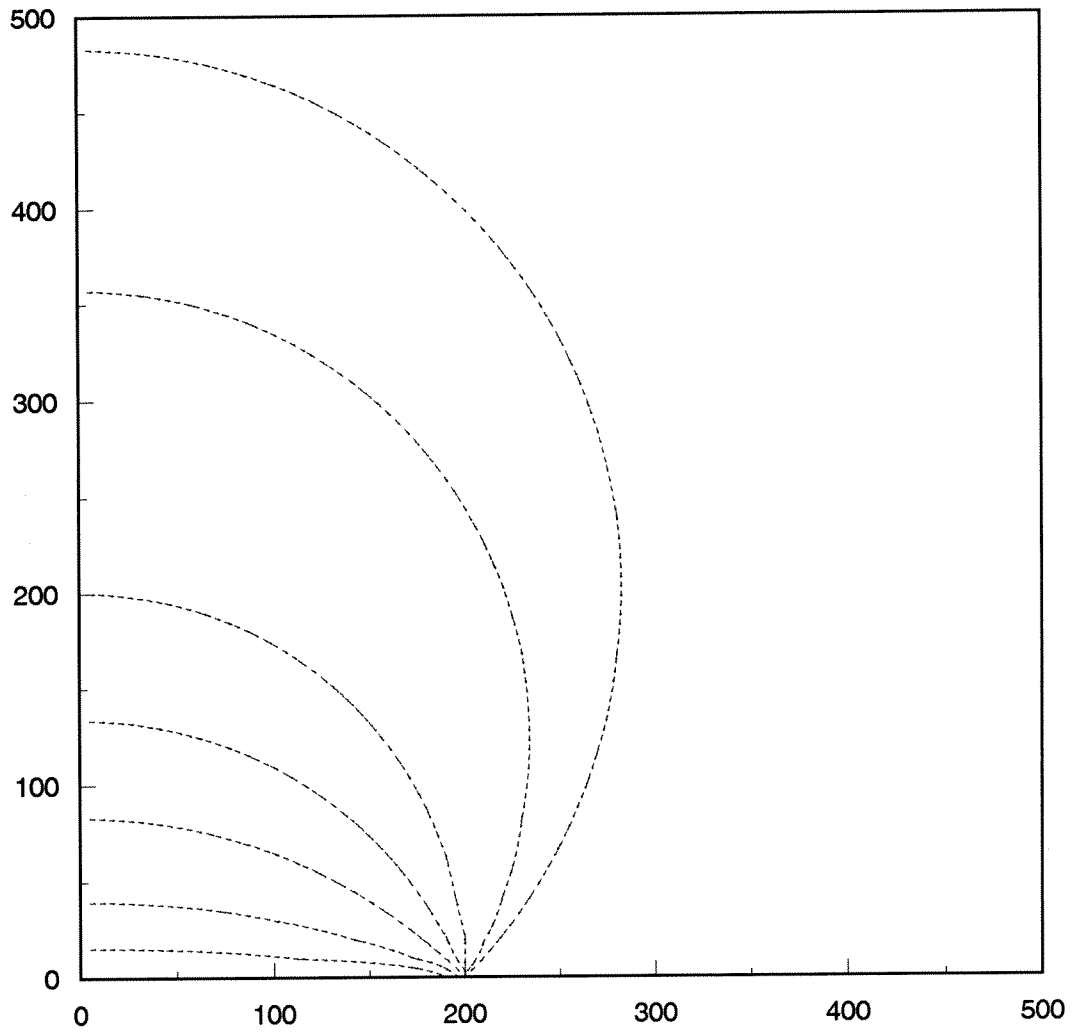


Figure 37 Streamlines for the well dipole, generated by DarcyTrack.

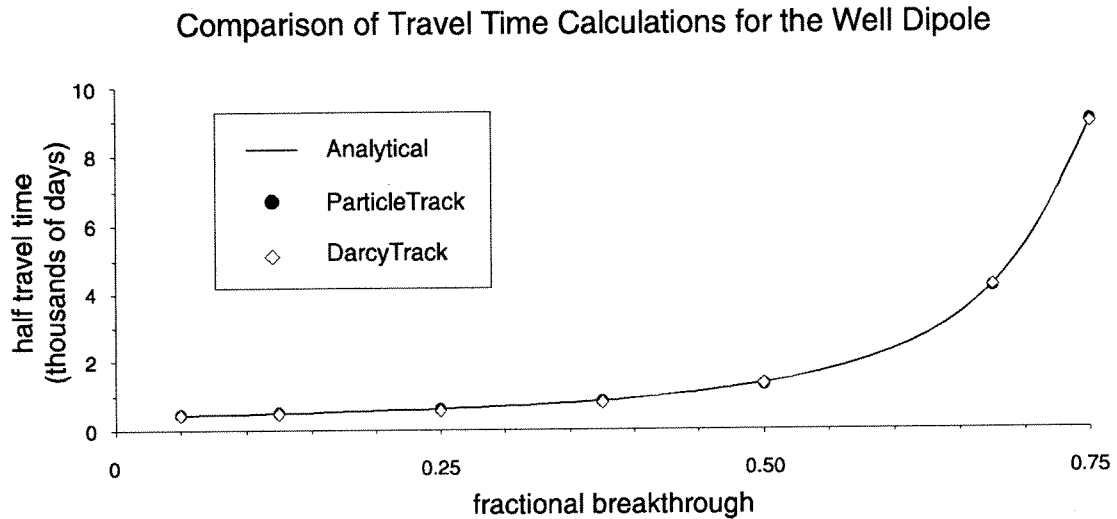


Figure 38 Comparison of well dipole travel times by ParticleTrack, DarcyTrack, and the analytical solution.

5.2.3 The Stepped Puff Simulation

This simulation is intended to test the internal consistency of the PorousPuff function, but also serves to illustrate the technique of superposition of masses distributed from distinct sources. In its most basic application, PorousPuff distributes the mass from a point source location to other cells in the grid. An example of a single puff in a uniform flow field is shown in Figure 39. The concentration distribution from several point sources can be superposed, as illustrated in Figure 40. Of course, this can be done any number of times.

As a test of internal consistency of PorousPuff, a concentration distribution like the one shown in Figure 39 was converted to a grid of point sources, one in each cell, with the map algebra command

```
mass = conc * poro * thick * %GRD$DX% * %GRD$DY%
```

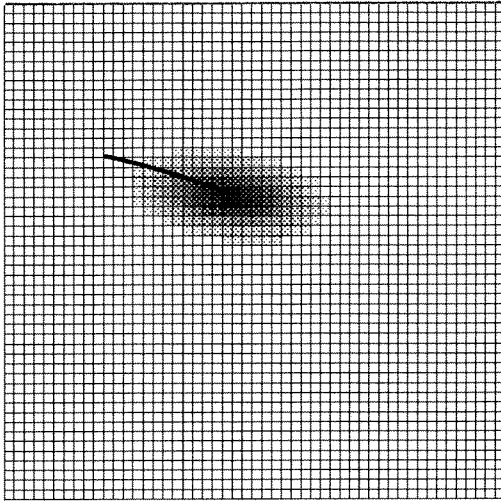


Figure 39 Basic application of PorousPuff to distribute a point source.

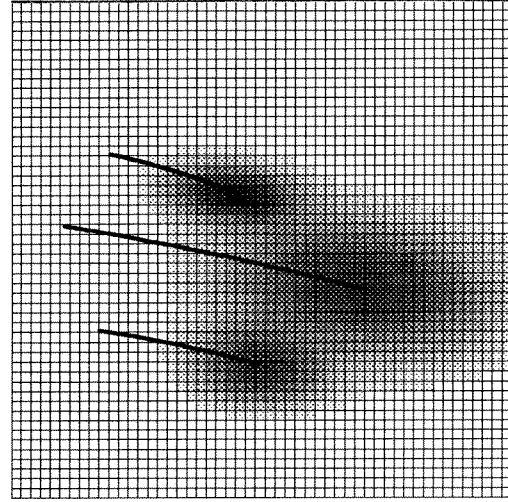


Figure 40 Repeated application of PorousPuff to distribute several sources.

where $GRD\$DX$ and $GRD\$DY$ are internal variables containing the grid cell dimensions (which are identical in the current implementation of Grid). Here, the mass has been calculated from the aqueous concentration by multiplying by the volume of fluid within the cell (cell area times thickness times porosity), assuming no sorption (Battaglin, 1993). Each point source is then distributed through the flow field by application of ParticleTrack, to determine the advective component, and PorousPuff, for the dispersive. The results of all these calculations are superposed to produce a composite distribution. In the following examples, a domain of 50×50 cells (each $20\text{m} \times 20\text{m}$) centered at the origin is given a uniform head field generated by

$$\mathbf{h} = \mathbf{h}_0 - \frac{1}{T} (\mathbf{U}_x \mathbf{x} + \mathbf{U}_y \mathbf{y}) \quad (89)$$

with the arbitrary values $h_0 = 10$ m, $T = 1.42$ m²/d, $U_x = 0.023$ m/d, and $U_y = -0.003$ m/d. With the additional information that $b = 5.7$ m and $n = 0.33$,

DarcyFlow produces uniform flow field with a direction of 97.4° and a flow magnitude of 0.0123 m/d. The initial source is a slug of 10^6 mg located at (-200, 50).

If we do two steps with time increments of, say, 10,000 days each, then the final distribution can be compared to a single step of 20,000 days. In a uniform field, the results of a single puff distribution and the two-step distribution should be identical. The following example uses five steps of 10,000 days each, and compares the result with one step of 50,000 days. For the PorousPuff function, we will need values for dispersivities, retardation factor, and decay coefficient. The dispersivities are $\alpha_L = 15$ m and $\alpha_L/\alpha_T = 4.3$, in line with an intermediate travel distance of about 400 m. A value of $R = 1.35$ is selected arbitrarily and $\lambda = 1.4 \times 10^{-5} \text{ d}^{-1}$ corresponds to a half life of about 50,000 d.

Figure 41 shows the results of the five-step redistributed puff after 50,000 days. From this we can subtract the one-step 50,000 day puff with map algebra:

$$\text{diff5} = \text{step5} - \text{puff5}$$

The difference is shown in Figure 42, with a maximum absolute value of 153 mg/m^3 at the peak, which is underpredicted by the redistribution method. The single-step puff has a peak value of 6221 mg/m^3 , so the error even after 5 steps is only $100 \times (-153 / 6221) = -2.46 \%$. It is important to note that although all of the mass is preserved in the redistribution steps, the distribution has changed slightly from the pure gaussian. This is due to small relocations in the discretized mass, since the integrated total mass for each cell is placed at the cell's geometric center, which may not be the center of mass for the cell. This source of error can be alleviated by finer discretization, though the additional calculations required will increase round-off error. This problem is an example

Redistributed 5-Step Puff

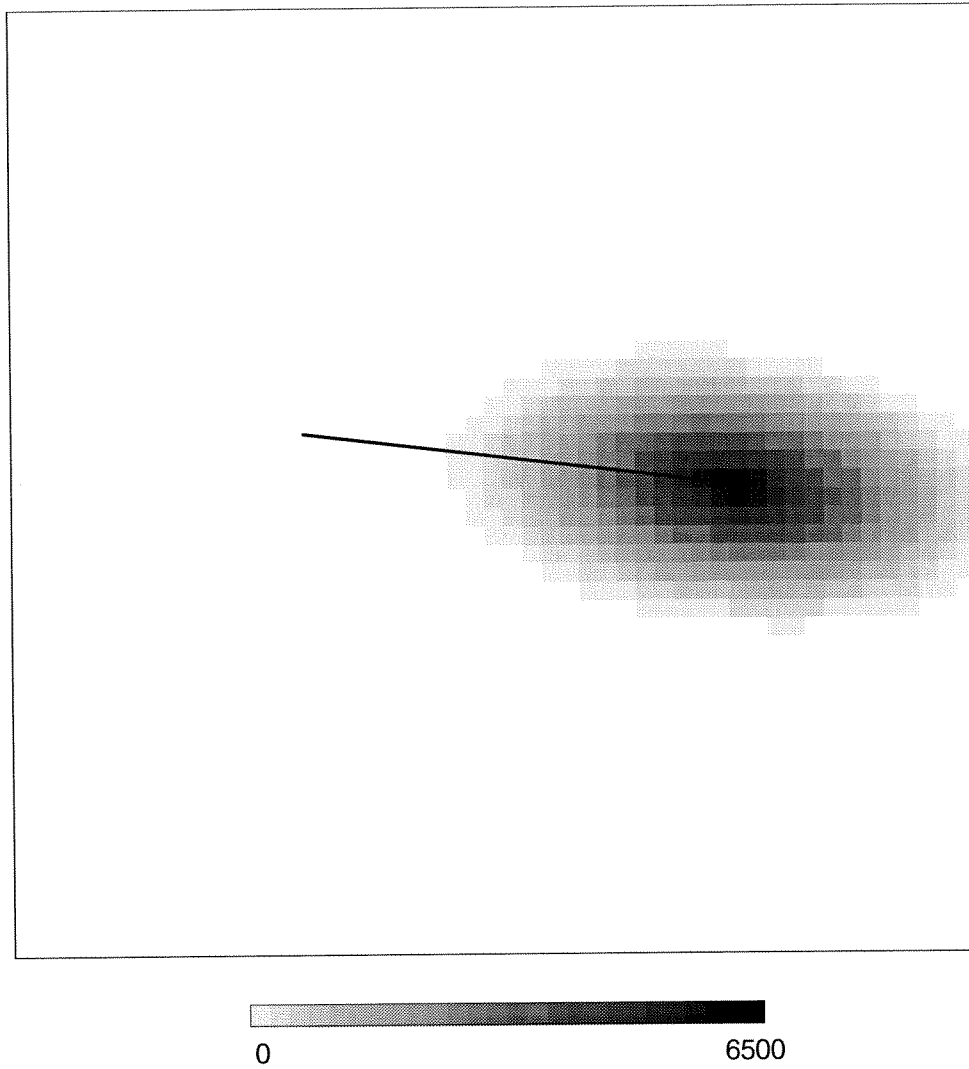


Figure 41 The concentration distribution resulting from the five-step redistributed puff.

Difference Between Single Puff and 5-Step Redistributed Puff

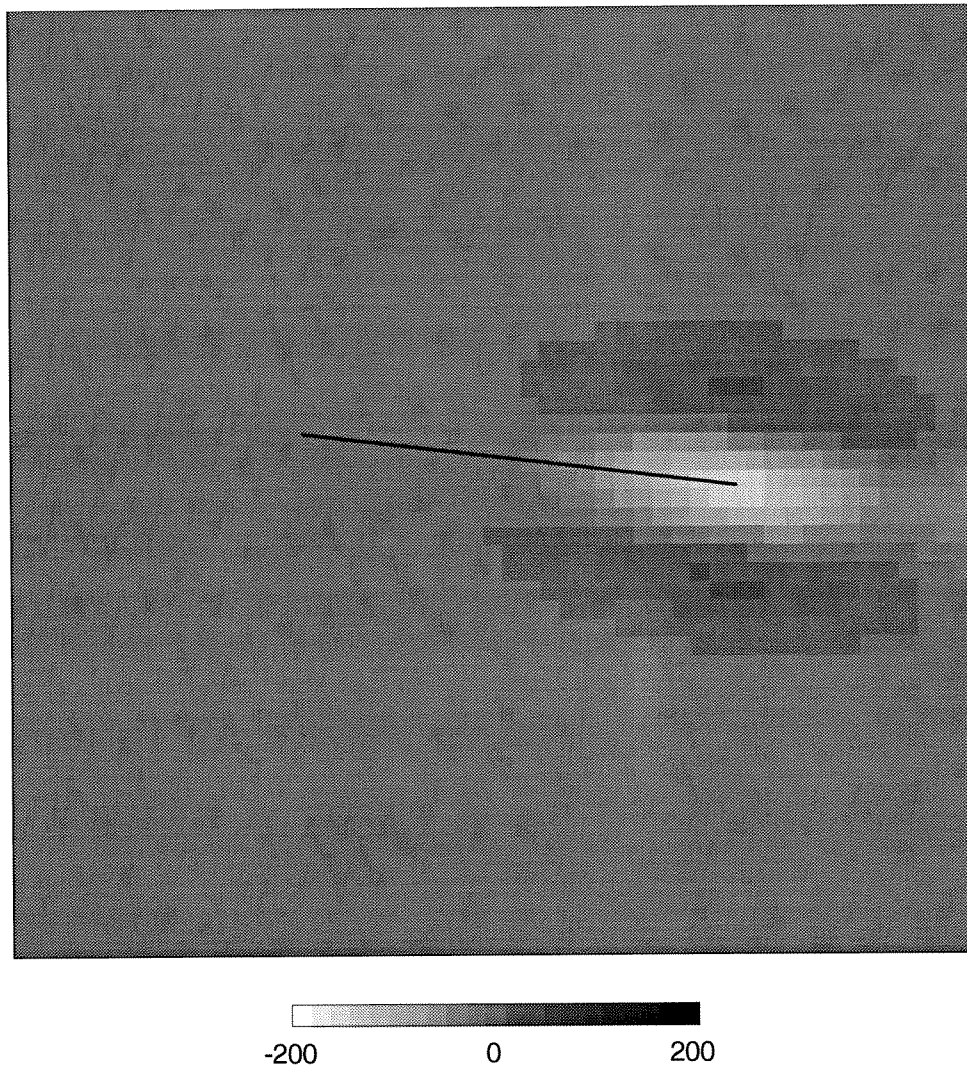


Figure 42 Difference in concentration distributions from the five-step redistributed puff and the single-step puff.

of the greater problem of trying to represent continuous processes on continuous fields with an inherently discrete space model. This is a difficult and pervasive problem in the field of modeling in GIS (Kemp, 1993).

A much greater source of roundoff errors is the `CELLVALUE` function, which is the only way to get the value of a particular cell into an AML. This function returns values only to three decimal places and does not use scientific notation, so that any value less than 0.001 is lost. Another effect contributing to the error is mass lost by migrating from the grid, apparent in Figure 42 as a small longitudinal asymmetry. The effect of losing mass by losing the contribution of puffs which have migrated off of the grid can be avoided by selecting a domain of sufficient dimensions that all travel times of interest are contained within the grid.

Using Arc Macro Language, a program may be written to execute this stepped puff technique (see `MULTIPUFF.AML` in the Appendix). The algorithm is simple: 1) calculate an initial puff from the source location, 2) convert the grid of masses to a grid of concentrations, 3) for each cell in the grid, calculate the puff from a source centered in the cell and, 4) sum the concentration distributions from each cell. Repeat this for each time step as desired.

In a uniform flow field, this result is not very interesting except for verification of the `PorousPuff` function. However, it is very useful in nonuniform fields, where material properties vary in space. Since the thickness and porosity which help to define the concentration distribution are obtained only at the centroid of the puff, `PorousPuff` may distribute mass into areas which are inappropriate, such as regions of low transmissivity or different flow direction. The single puff assumes that the properties are uniform over the bulk of the distribution, so `PorousPuff` should be applied only in small increments over which the field is essentially uniform. If the stepped puff technique is employed, more

widespread information is used from the grids of material properties, and the advection and dispersion of mass follows a more natural progression. The only disadvantage to this technique is the prohibitive computer time required. Since each puff involves an integration for each cell, a stepped puff will make calculations for the square of the number of cells, running both ParticleTrack and PorousPuff for each cell in the grid. A modest 100×100 cell grid contains 10^4 cells, so each step in the stepped puff would require 10^4 applications of ParticleTrack and 10^4 of PorousPuff, each of which involves 10^4 integrations to determine the concentration in each cell, totalling 10^8 integrations. Each quadrature integration will require at least 2^2 but perhaps as many as 64^2 (4096) sampling points, so that at a minimum, the concentration calculation is looped through 4×10^8 times. The five-step puff described above involved at least 2×10^9 loops. This would not ordinarily be an excessive demand of the computer, but Grid insists on keeping data on disk rather than in RAM, so that, for example, the thickness value for each centroid must be acquired through a disk read. This is done as a conservative approach to preserving data integrity. Grid is so slow that one stepped puff on a 100×100 cell grid takes about thirty hours to complete on a Sun SPARC IPX. This cost is the Achilles' heel of Grid, and inhibits using PorousPuff effectively. However, PorousPuff is prepared to work well once Grid is programmed to work more efficiently.

5.2.4 The Convolved Plume Simulation

To test the compatibility between PorousPuff and PorousPlume, we may perform a convolution. This is a natural way to examine the plume, since it is how equation (76) (page 87) was derived. We proceed by generating a series of puffs which approximate the constant mass release rate used for the plume. We start with the same uniform flow field used in the last example, and generate a plume (Figure 43) with a source strength of 10^3 mg/d located at (-350, 50), and use the same longitudinal dispersivity $\alpha_L = 15$ m, dispersivity ratio $\alpha_L/\alpha_T = 4.3$, retardation factor $R = 1.35$, and decay coefficient $\lambda = 1.4 \times 10^{-5} \text{ d}^{-1}$:

```
Grid: path = PARTICLETRACK( dir, mag, -350, 50, #, # )
Grid: plume = POROUSPLUME( path, poro, thick, 1e3, 1000, 15, 4.3, 1.35, 1.4e-5 )
```

The plume is truncated upstream of the source and downstream of the point where the path exits, which corresponds to a retarded travel time of just over 95,000 d.

To approximate this plume by convolution, a series of 95 puffs is generated using the same source location and parameters. If these puffs are released at 1000 day intervals, each should start with a mass of 10^6 mg:

```
Grid: puff01 = POROUSPUFF( path, poro, thick, 1e6, 1000, 15, 4.3, 1.35, 1.4e-5 )
Grid: puff02 = POROUSPUFF( path, poro, thick, 1e6, 2000, 15, 4.3, 1.35, 1.4e-5 )
Grid: puff03 = POROUSPUFF( path, poro, thick, 1e6, 3000, 15, 4.3, 1.35, 1.4e-5 )
...
Grid: puff93 = POROUSPUFF( path, poro, thick, 1e6, 93000, 15, 4.3, 1.35, 1.4e-5 )
Grid: puff94 = POROUSPUFF( path, poro, thick, 1e6, 94000, 15, 4.3, 1.35, 1.4e-5 )
Grid: puff95 = POROUSPUFF( path, poro, thick, 1e6, 95000, 15, 4.3, 1.35, 1.4e-5 )
```

The resulting puffs are summed into a convolved plume, shown in Figure 44.

Concentration Plume by PorousPlume

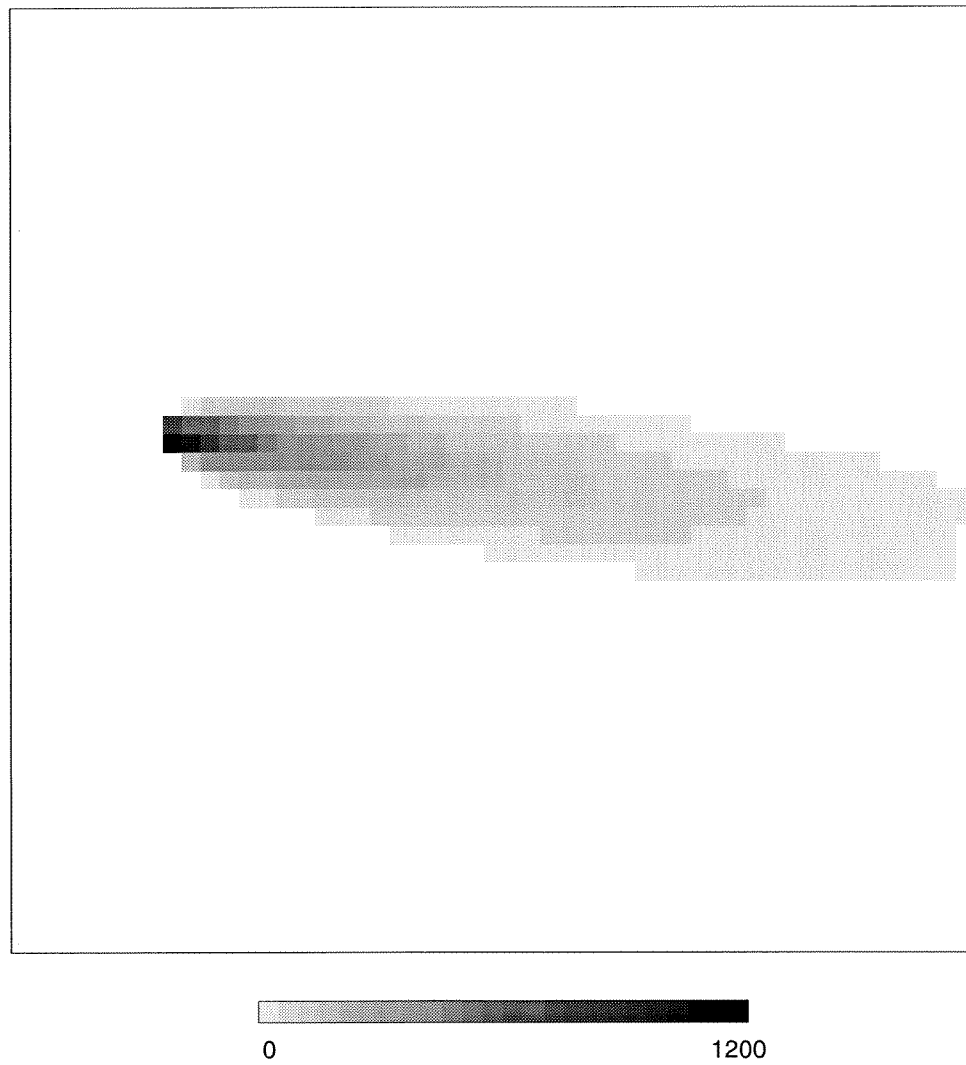


Figure 43 Plume generated by PorousPlume.

Plume Convolved by PorousPuff

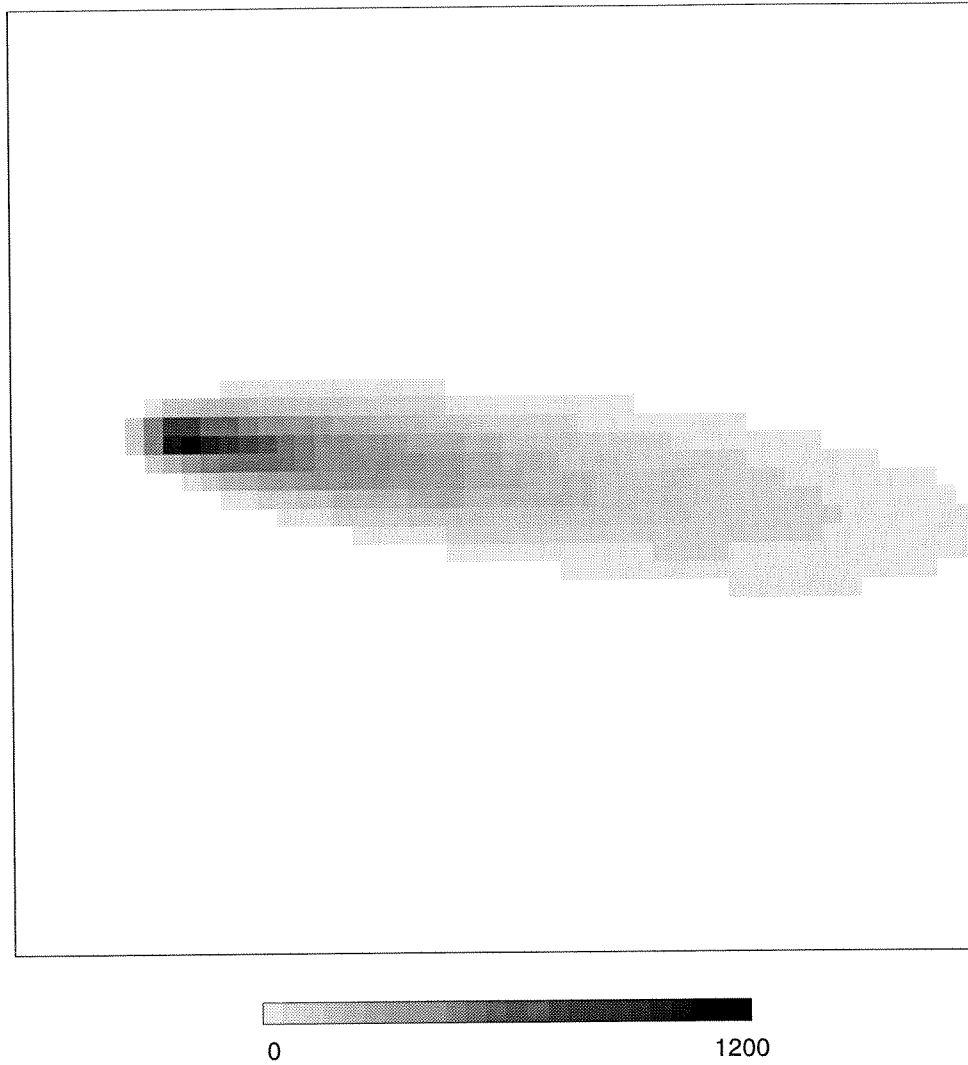


Figure 44 Convolved plume generated by repeated application of PorousPuff.

Subtracting this convolved plume from the one calculated directly by Porous-Plume produces the difference shown in Figure 45:

```
Grid: diffplume = conplume - plume
```

This has several notable features. Just upstream of the source, where Porous-Plume has truncated itself, the convolved plume has placed an amount of mass, due to the infinite tails of the bivariate gaussian distribution resulting in positive values for the difference. At the downstream end, the convolved plume underpredicts the concentration since it does not include a contribution by those puffs whose centroids would be off of the grid, further downstream. Between the end effects, however, the bulk of the plume matches quite well, validating the convolution technique. This has implications for use in plumes from transient sources, which could be modeled as a series of puffs released over a particular window of time.

Difference of PorousPlume and
Plume by Convolution of PorousPuff

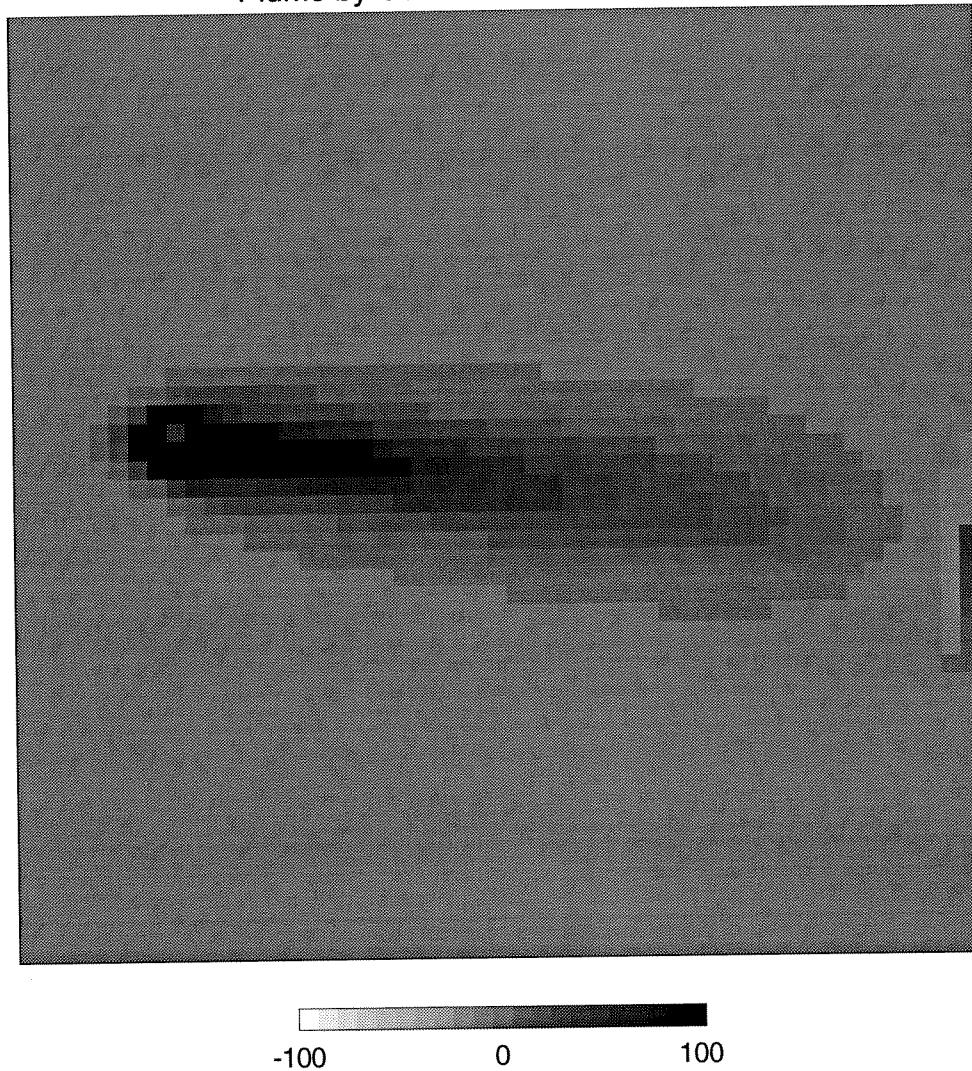


Figure 45 Difference between the plume convolved by PorousPuff and that calculated by PorousPlume.

5.3 Model Demonstration

We can at once make use of the GIS by generating maps of the study area for this demonstration. For example, Figure 46 is a map of the rivers of New Mexico, derived from a readily available coverage of the rivers of the United States. The WIPP Site is located in southeastern New Mexico, with geographical features and the WIPP study area identified in Figure 47. This study area measures 25×30 km, and has been subdivided into 50×60 cells, each 500 m square, for the purposes of modeling in Grid. A local coordinate system originates at the western corner of the study area, with x following the southwest edge and y the northwest. The base grid, shown in Figure 48, is rotated 38° east of north to align it with local geological features. The WIPP Site, outlined in the figure, is the boundary of the repository property, considered the critical boundary for the 10,000 yr travel time specified by the EPA in 10 CFR 191 Subpart B (U.S.E.P.A., 1985). We will be examining travel times to that boundary from the location of the mined area at (10000, 16000) in local coordinates.

5.3.1 GIS Model of the Culebra Dolomite

The performance assessment of the WIPP includes scenarios of the release of contaminants from the repository to the accessible environment. The most troubling scenarios involve a containment breach caused by future exploratory drilling through the repository into the high-pressure brines of the Castille Formation below (Bertram-Howery, et al., 1990). These brines may become saturated with radioactive materials in the repository and migrate up the bore



Figure 46 Arc/Info coverage of rivers in New Mexico.

Regional Map of Southeastern New Mexico

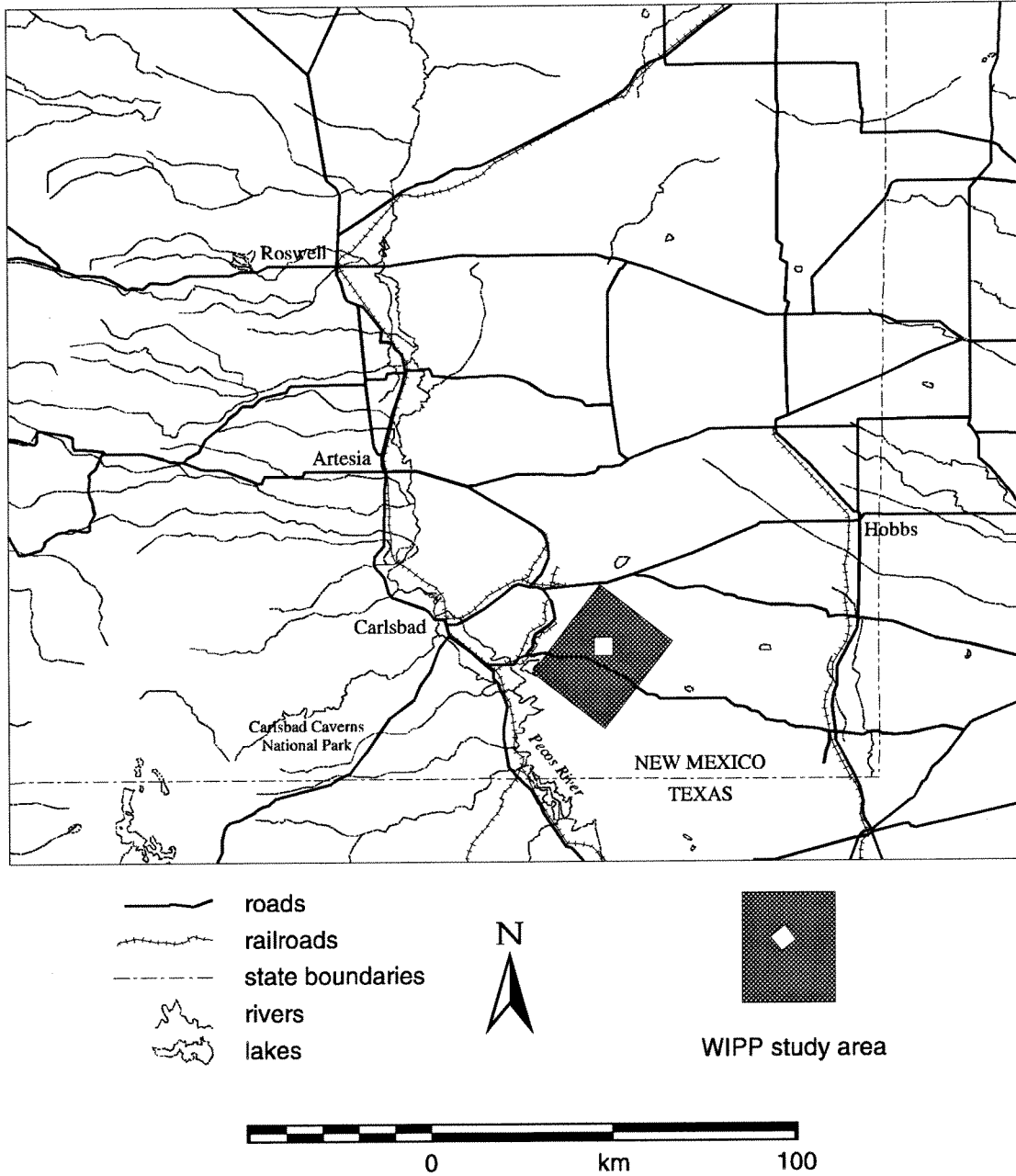


Figure 47 Arc/Info coverage of features in southeastern New Mexico, identifying the location of the WIPP study area.

Base Grid for WIPP Study Area

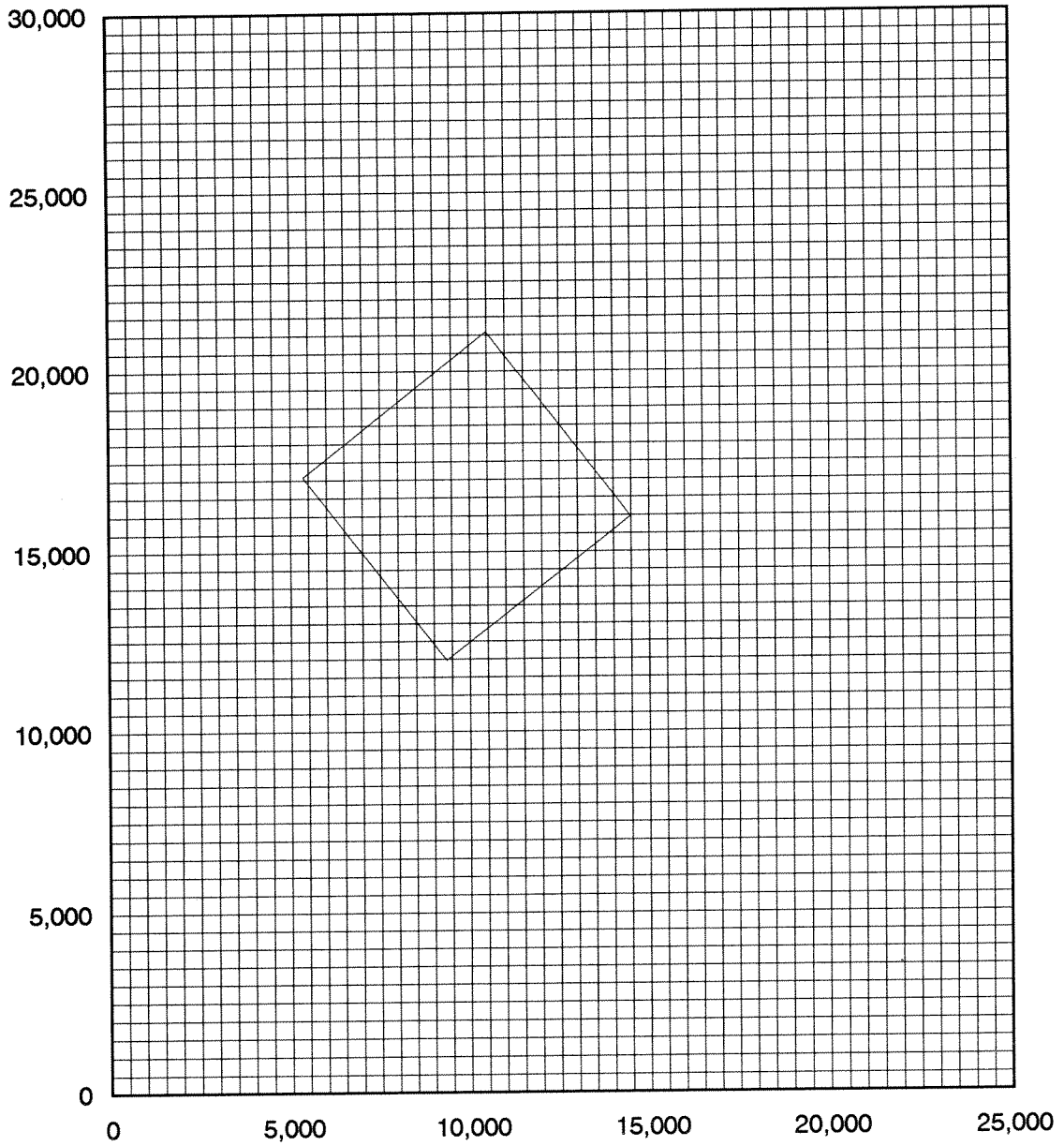


Figure 48 Base grid for analysis of the Culebra Dolomite in Grid, with an outline of the WIPP Site boundary.

hole into the most transmissive layer above: the Culbra Dolomite, a member of the Rustler Formation (Brinster, 1991). The Culebra varies from 6 to 12 m in thickness, and is saturated with groundwater. It is an important part of the pathway from the repository to the environment and so has been the subject of intense hydrogeological investigation.

The Culebra has been modeled in a variety of ways, but the best calibrated analysis has been performed by LaVenue and RamaRao (1992), who have conditionally simulated about 70 transmissivity fields calibrated to both steady-state and transient head data. The steady-state head field is used with the transmissivity field here as a starting point for the GIS analysis. LaVenue's best calibrated log transmissivity field and steady-state head field were reformatted into a regular grid, and imported to Grid via Arc's ASCIIGRID command,

```
Arc: ASCIIGRID head.txt head float  
Arc: ASCIIGRID logt.txt logt float
```

and are shown in Figure 49 and Figure 50. The transmissivity grid is calculated from the log transmissivity grid by map algebra:

```
Grid: trans = exp10( logt )
```

The grids for saturated thickness and effective porosity are constant-valued. Porosity in the Culebra has been estimated at well H-6 (5584, 16926 in local coordinates) by Walter (1983) as 0.11, and LaVenue has chosen an historically entrenched value of 0.16 for the entire Culebra member (LaVenue, Cauffman, and Pickens, 1990, LaVenue and RamaRao, 1992). LaVenue has also chosen a constant thickness of 7.7 m in spite of the wealth of thickness data, with measurements from scores of boreholes. The true thickness, however, is incorporated into the transmissivity. These grids are easily generated in Grid,

Steady State Freshwater Head Elevation (m amsl)

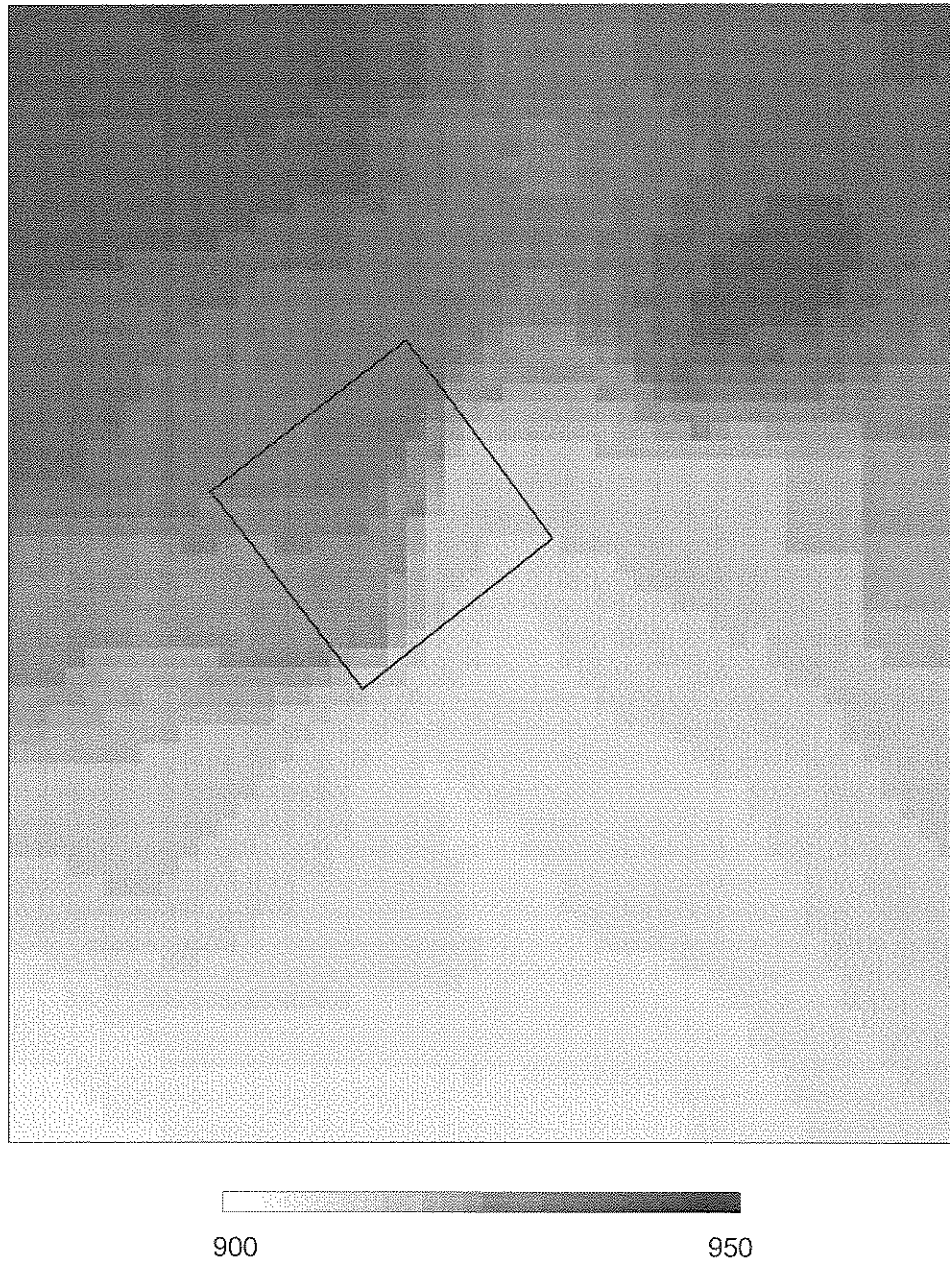


Figure 49 WIPP regional steady-state freshwater head elevations (after LaVenue and RamaRao, 1992).

Log_{10} Transmissivity (m^2/d)

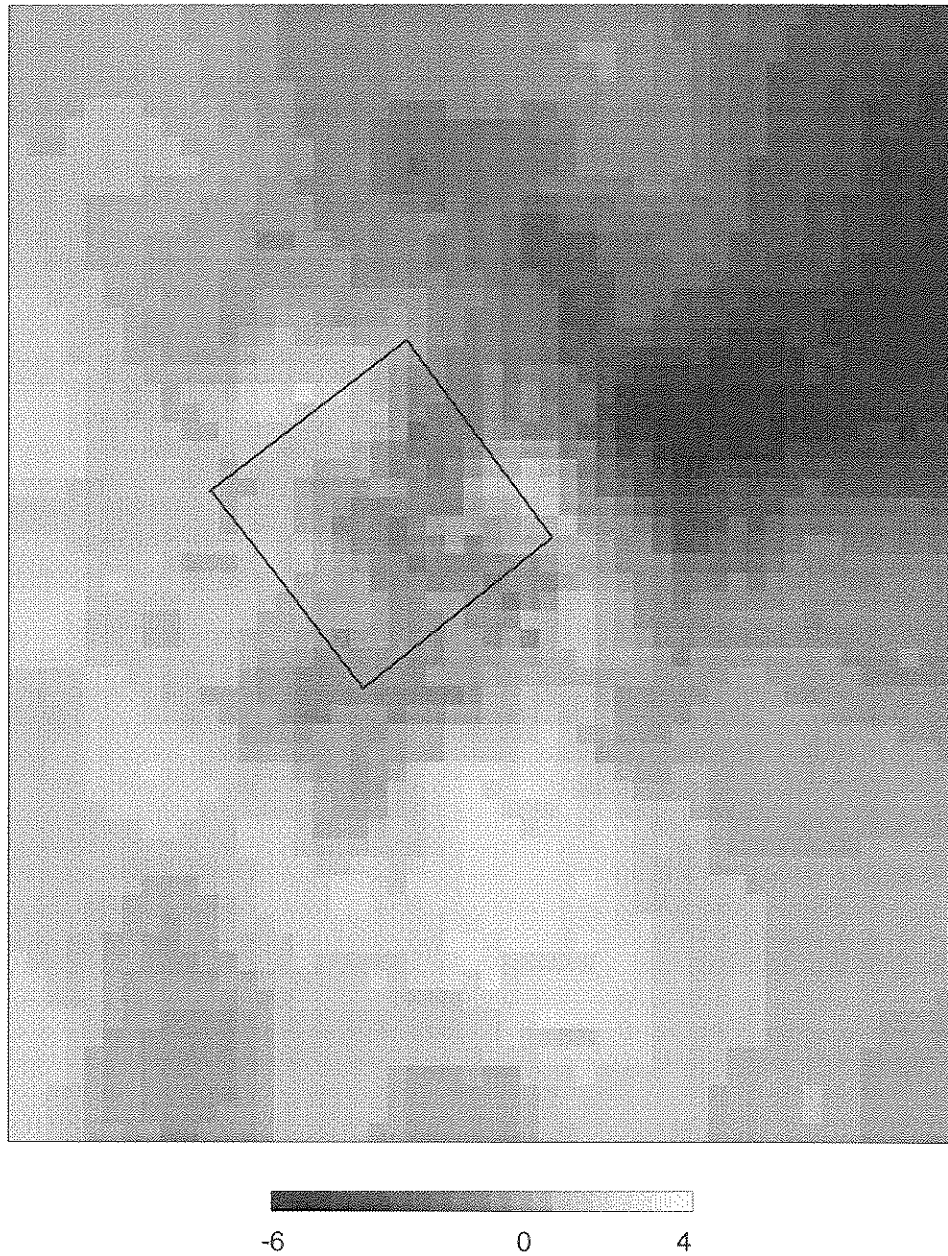


Figure 50 WIPP regional log transmissivities (after LaVenue and RamaRao, 1992).

once the current window (defining the location and dimensions of the grid) and cell sizes are adopted from an existing grid. In this case, the head grid is used for a pattern:

```
Grid: SETCELL head
Grid: SETWINDOW head
Grid: poro = 0.16
Grid: thick = 7.7
```

Now we have the four grids necessary to run DarcyFlow, so the command

```
Grid: res = DARCYFLOW( head, poro, thick, trans, dir, mag )
```

generates the flow field (contained in the *dir* and *mag* grids, combined in Figure 51) and the volume balance residual grid, *res* (Figure 52). The residual shows us to what extent the calibration of LaVenue's model is successful. There are some problems, notably in Nash Draw which runs along the left side of the grid, with mass balance errors approaching 100 m³/d per cell. This is an area of dissolution of the Culebra, with widely varying transmissivities, so the difficulty in modeling is to be expected. However, while there are areas of relatively higher mass balance losses, the critical region of flow immediately south of the WIPP Site has been well calibrated, showing a good mass balance with a maximum of only 1 m³/d per cell. Since each cell has an area of (500m)² = 250,000m², a loss of 1 m³/d over this area amounts to a rate of 4×10^{-6} m/d = 4 μm/d.

Flow Field Surrounding WIPP Site

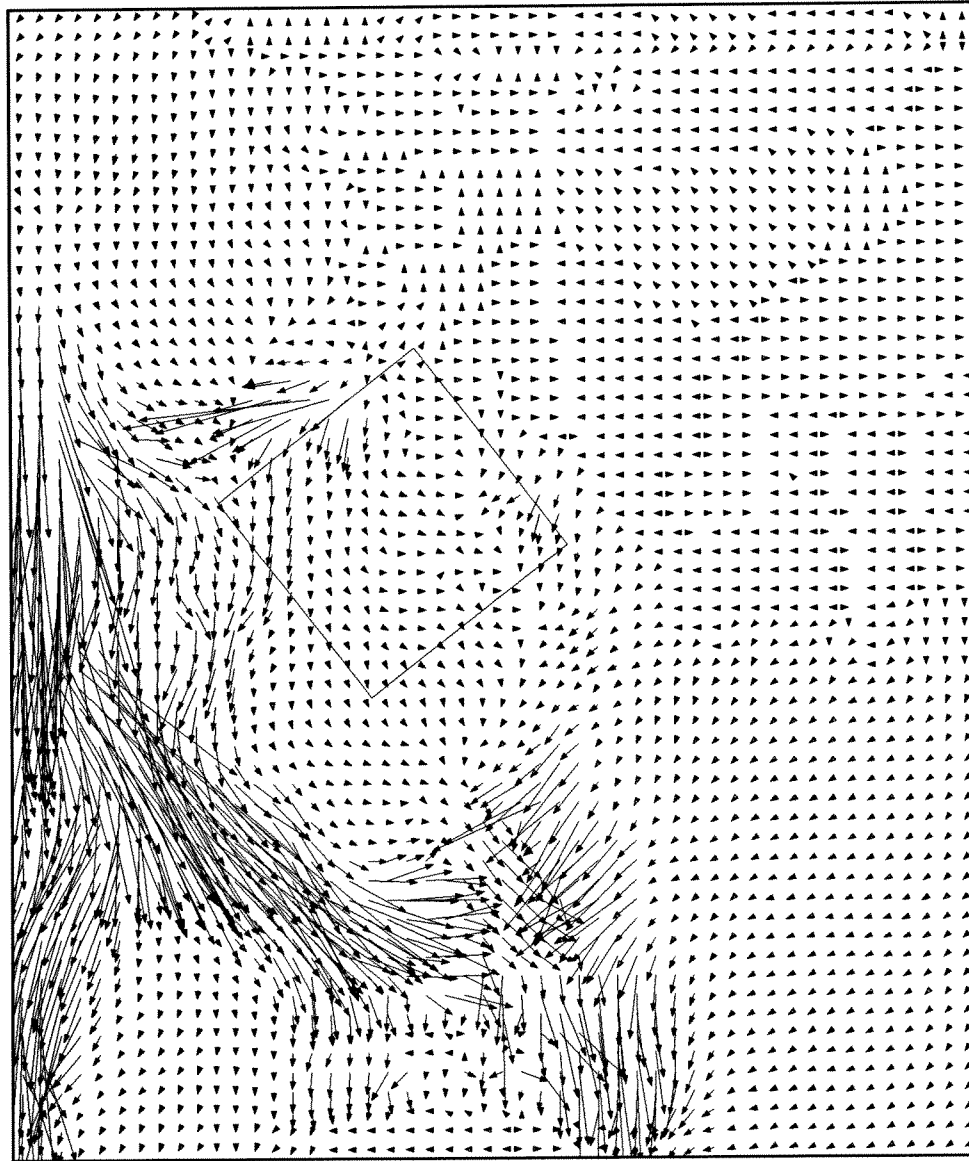


Figure 51 WIPP regional flow field generated by DarcyFlow.

Volume Balance Residual (m³/d)

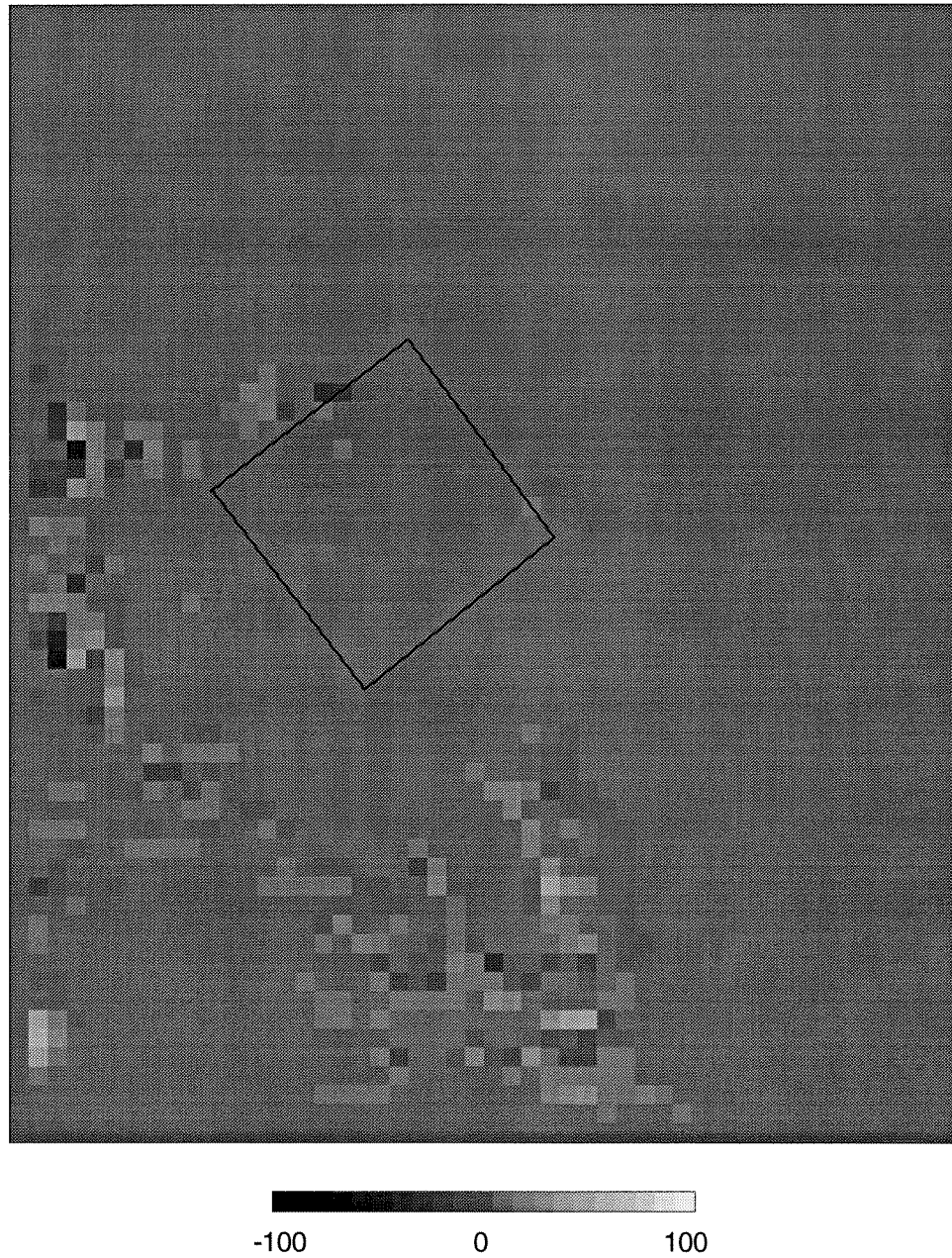


Figure 52 WIPP regional volume balance residual generated by DarcyFlow.

5.3.2 Travel Time Calculation

We can now use ParticleTrack (and DarcyTrack) to generate travel paths through the field:

```
Grid: ptrack = PARTICLETRACK( dir, mag, 10000, 16000, #, # )  
Grid: dtrack = DARCYTRACK( head, poro, thick, trans, 10000, 16000, # )
```

The source location used in the WIPP performance assessment models is the center of the waste panels at (10000, 16000) in local coordinates, and is the starting point for the paths shown in Figure 53. The long range paths have very different travel times, resulting from DarcyTrack's more literal view of the head surface which closely follows small fluctuations and "valleys" at cell boundaries. A nearly flat valley will require a great deal of time to traverse. In contrast, ParticleTrack works from an inherently smoother field, smoothed once by the averaging in DarcyFlow and again by the approximations used by ParticleTrack. Zooming in on the region surrounding the WIPP Site (Figure 54), where the field is smoother, the short range travel paths are seen to be more consistent, with travel times approaching the longest time estimated by LaVenue and RamaRao (1992) at 56,000 years to exit the site. The 70,000 year travel times from ParticleTrack and DarcyTrack are well within the distribution of travel times calculated by all of the performance assessment modeling methods (Marietta, personal communication).

The immediate area of the site may be re-examined at a finer scale. By repeating the process using 100 m cells, we import the head elevation and log transmissivity grids shown in Figure 55 and Figure 56. DarcyFlow is used to generate the residual grid (Figure 57) and the flow directions (Figure 58). Travel paths from the center of the waste panels are generated by both ParticleTrack and

Calculated Travel Paths

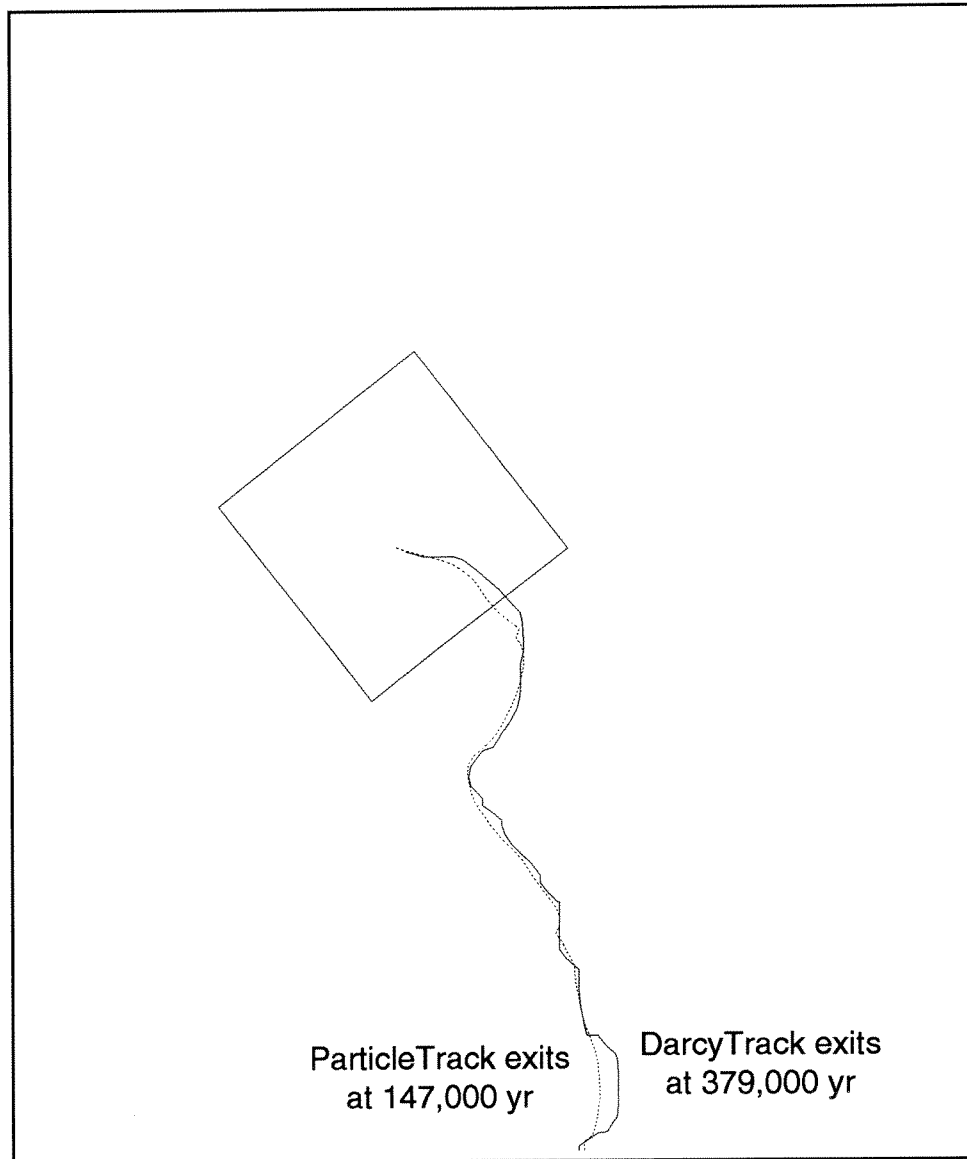


Figure 53 Long range travel paths from the WIPP.

Flow Paths from the WIPP Site

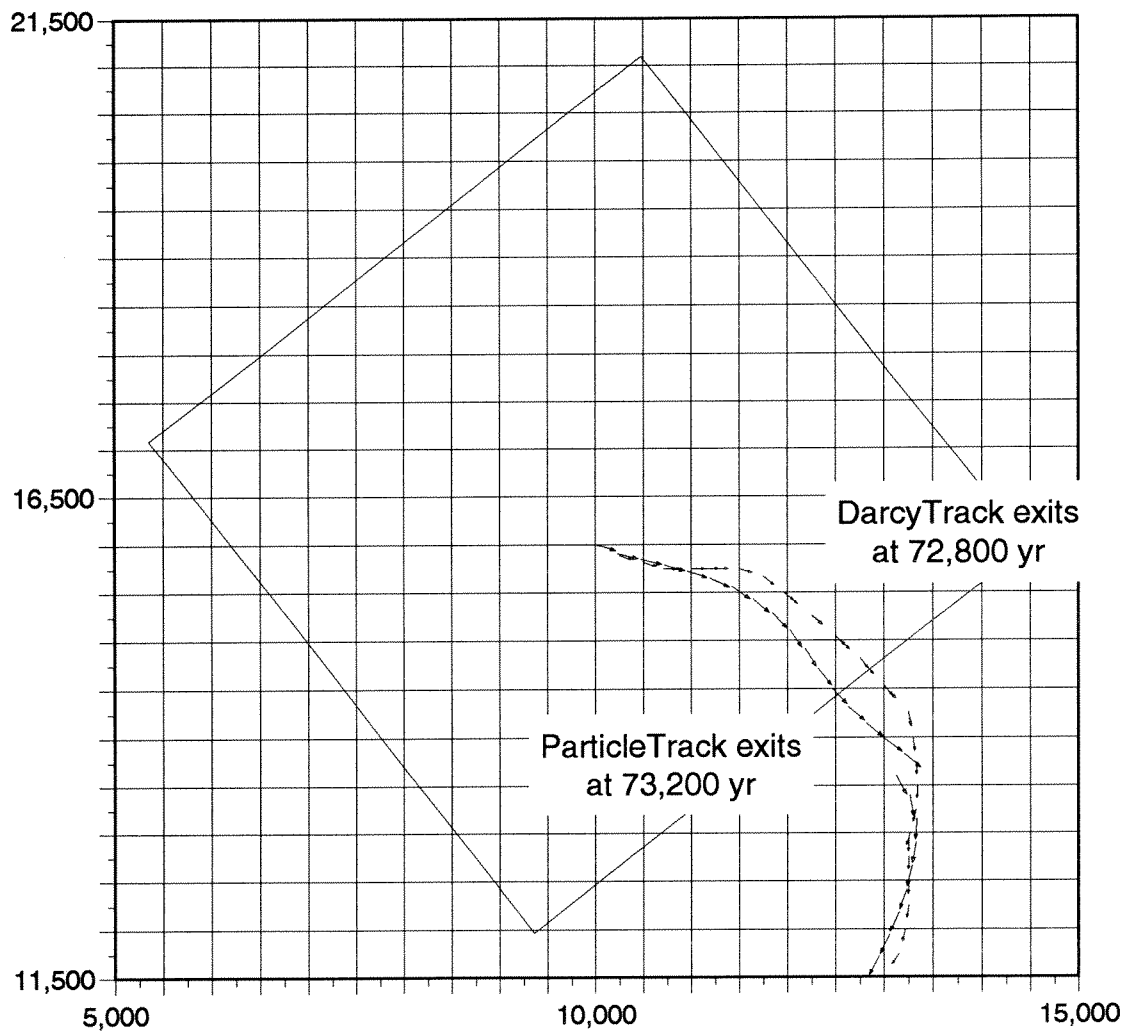


Figure 54 Short range travel paths from the WIPP.

DarcyTrack, shown in Figure 59. It is interesting to note the increase (by a factor of 2.5 to 3.5) in travel times to the WIPP boundary in the fine-scale study. At the finer resolution, heterogeneities in transmissivity are increased. Since DarcyFlow and ParticleTrack use a harmonic average of the transmissivities of neighboring cells, a finer resolution will inevitably result in lower values, and the velocities will be correspondingly reduced. DarcyTrack uses block transmissivities which are constant for each segment of the track, a difference which is reflected in the different travel times predicted by the two functions. It is worth noting that changing the step size used by ParticleTrack has little effect on travel time.

The ability to switch scales and resolutions is a strength of the GIS. Using the RESAMPLE function in Grid, one can redefine the cell size and boundaries of the grid given a variety of interpolation techniques. Unfortunately, performing a rotation of axes is still not possible within Grid, although it can be done in Arc one the grid cells have been exported as polygons. The rotated polygons may then be imported back into Grid.

Freshwater Head Elevation (m amsl)

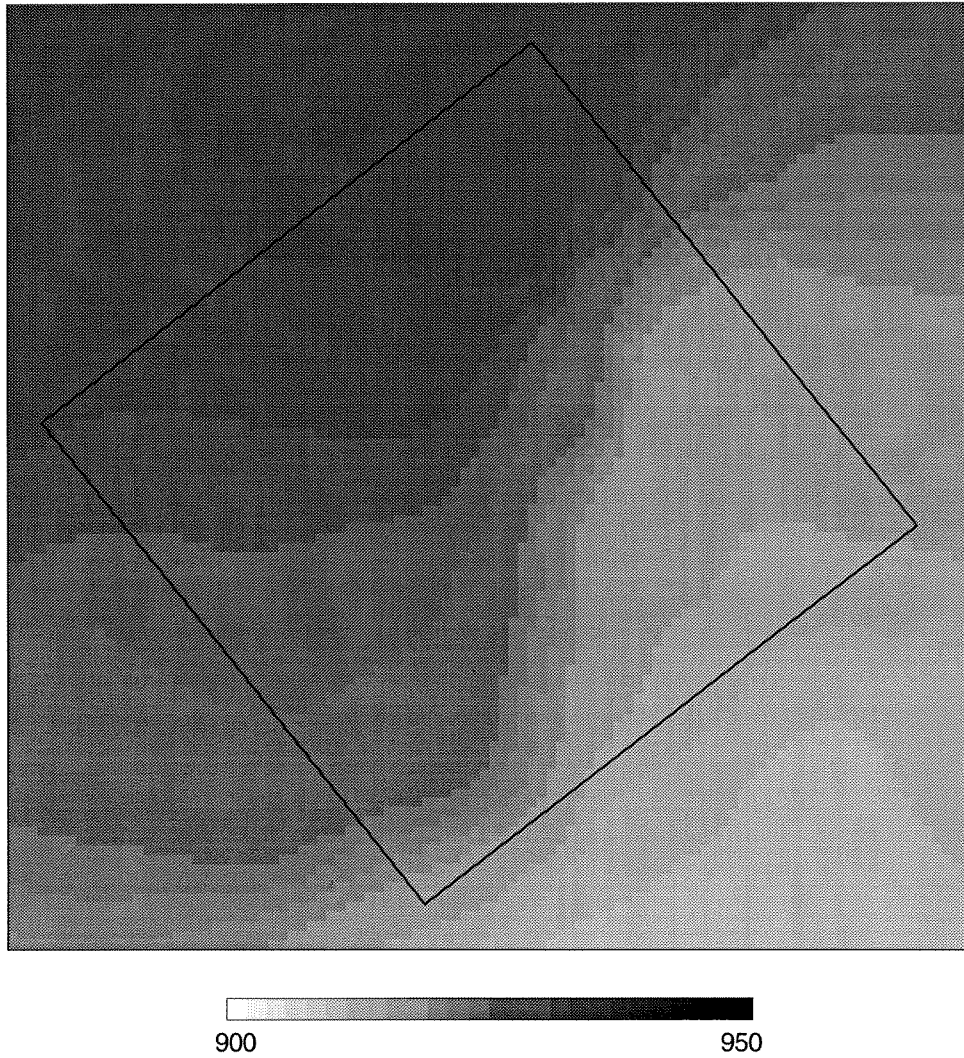


Figure 55 WIPP Site steady-state head elevations. Compare to Figure 49.

Log_{10} Transmissivity (m^2/d)

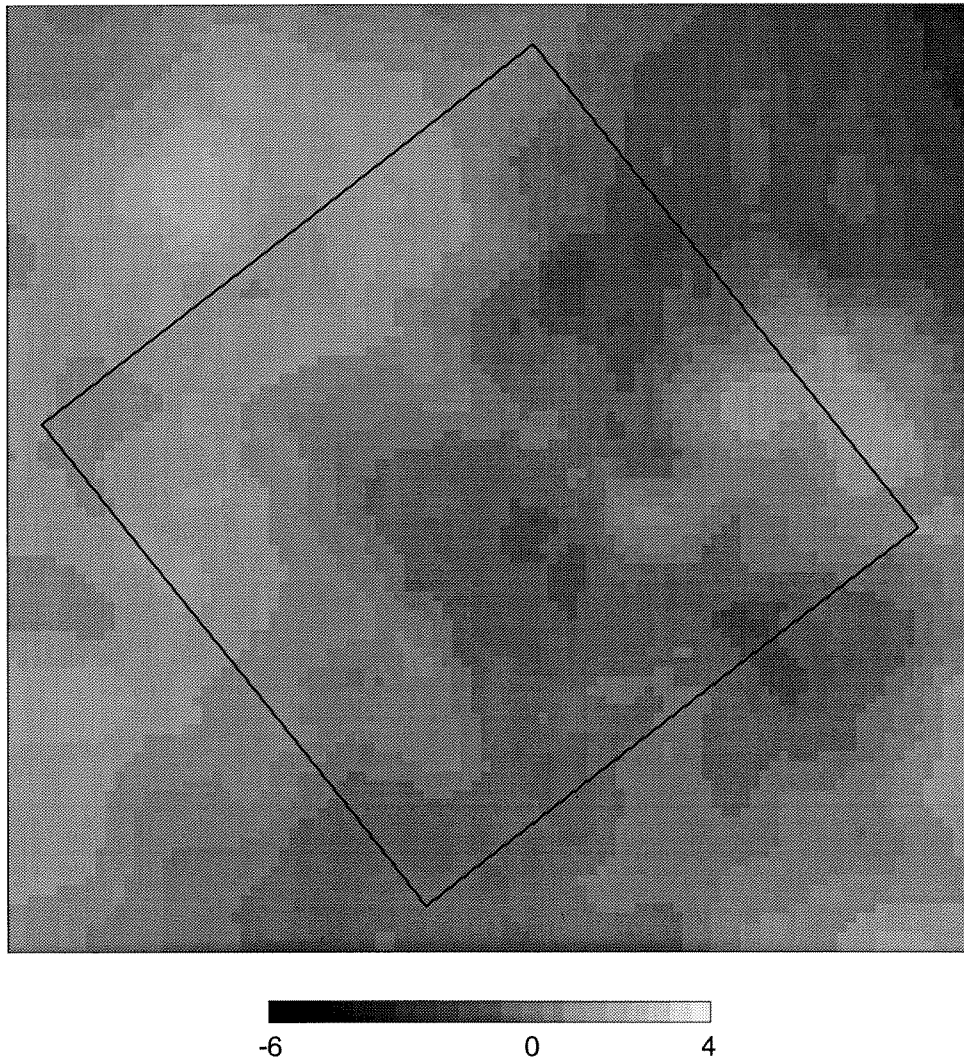


Figure 56 WIPP Site log transmissivities. Compare to Figure 50.

Volume Balance Residual (m^3/d)

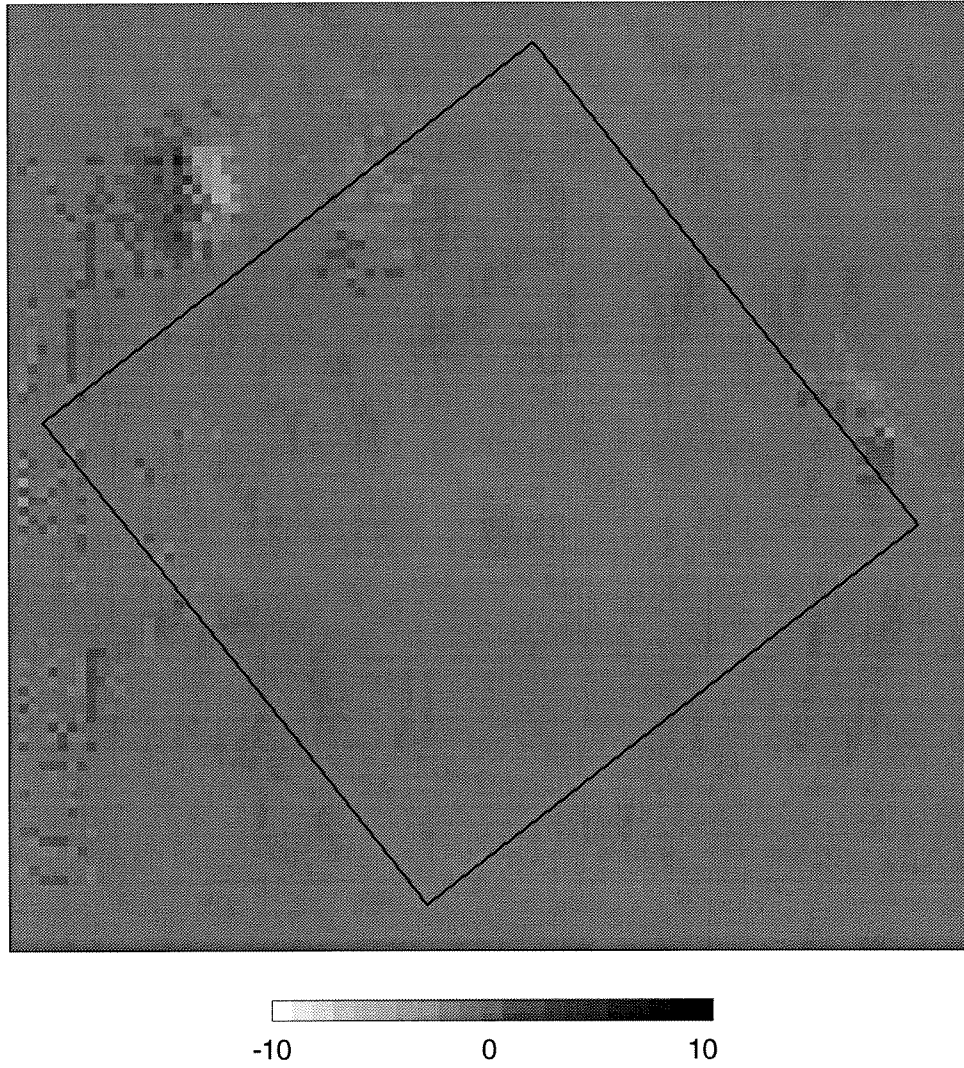


Figure 57 WIPP Site volume balance residual grid.

Detailed Flow Field at the WIPP Site

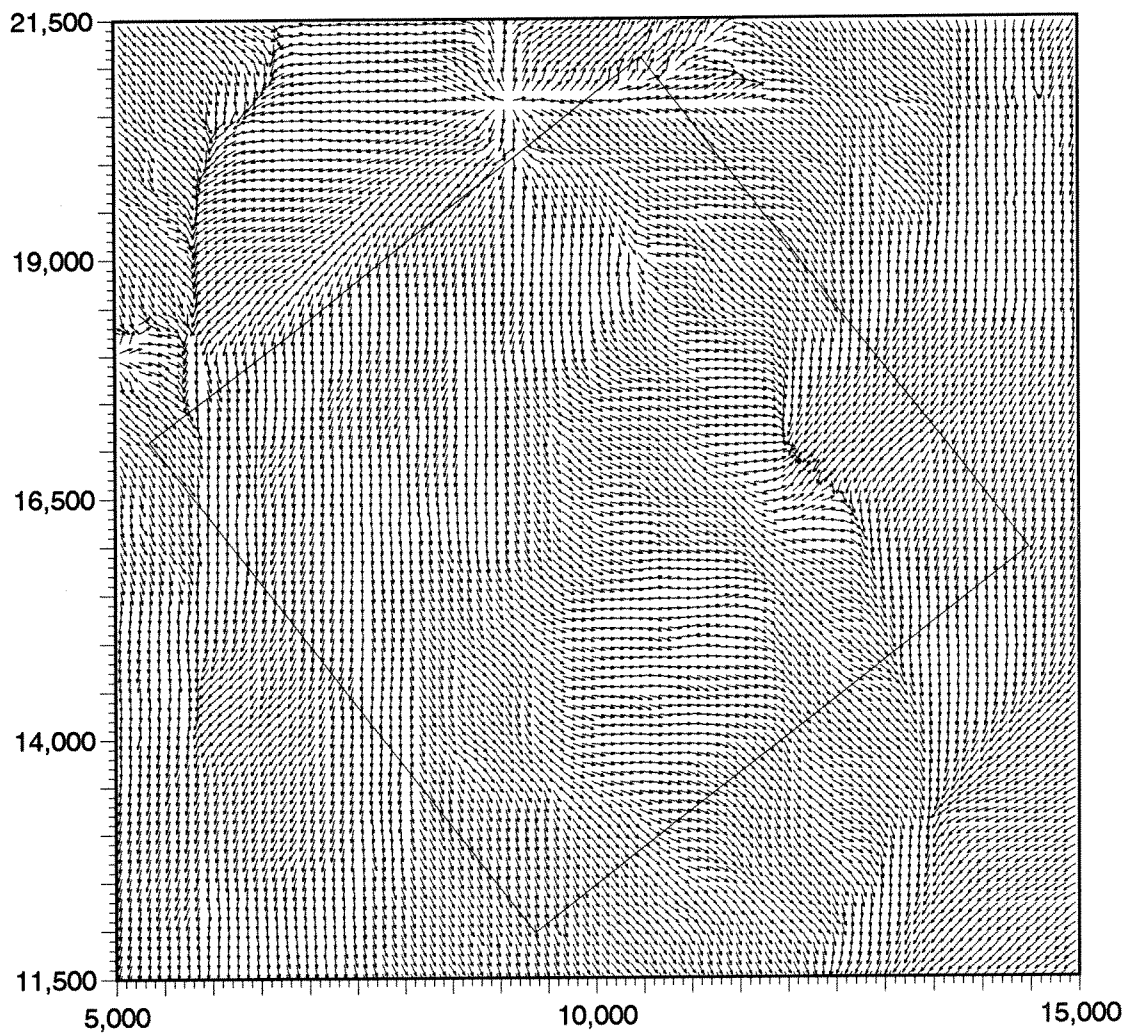


Figure 58 WIPP Site flow directions generated by DarcyFlow.

Flow Paths from the WIPP Site

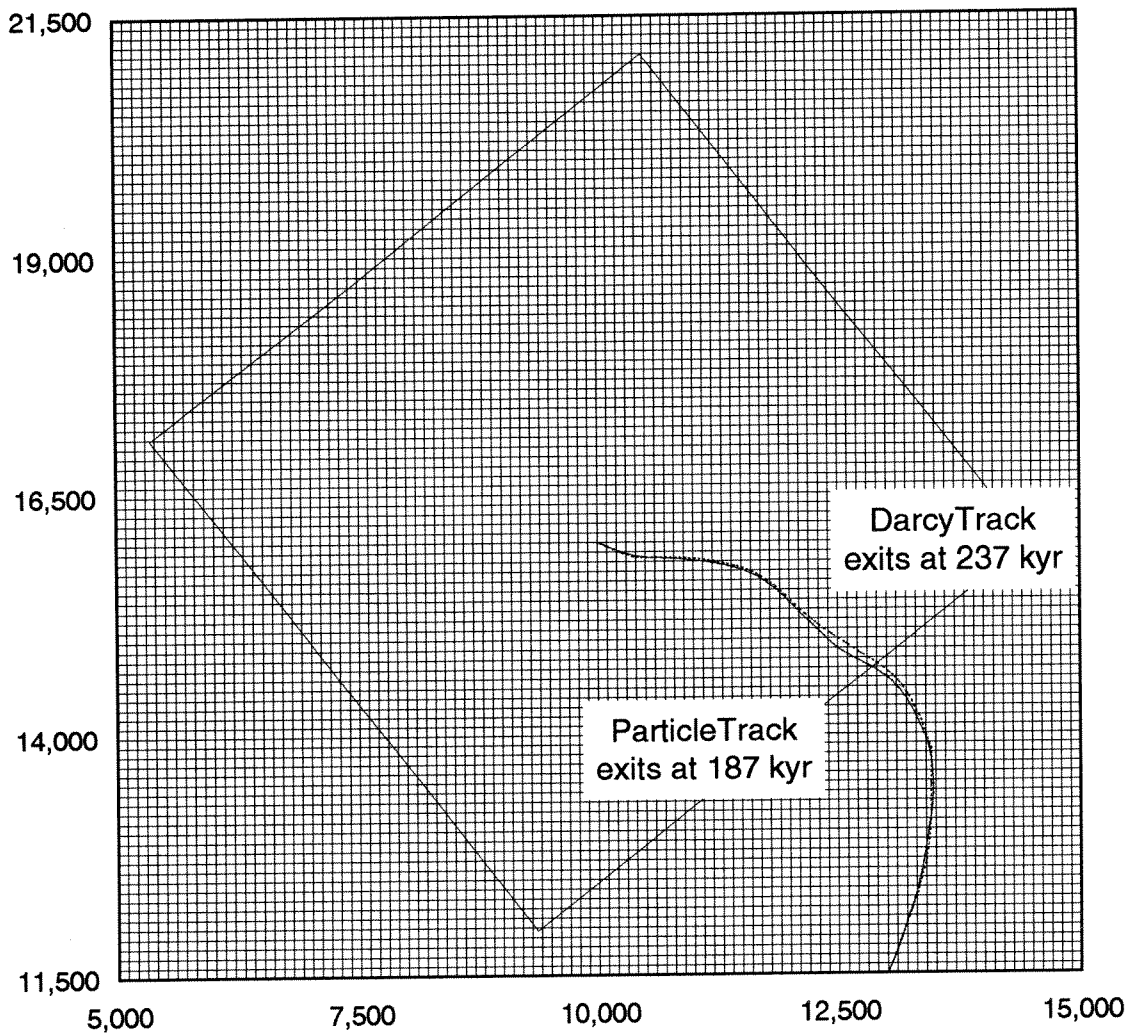


Figure 59 Travel paths from the WIPP Site, generated by ParticleTrack and DarcyTrack.

5.3.3 Transport Modeling Using PorousPuff

PorousPuff can assist in studying the transport of contaminants from the WIPP. The transport parameters required by PorousPuff and PorousPlume have been estimated by researchers involved in the performance assessment and have been discussed at length by a panel of experts (Gallegos, Marsily, and Marietta, 1994). For the purposes of this demonstration, we can select reasonable values for dispersivity, retardation, and decay coefficient based on these discussions. Dagan (1994) and Beyeler (1994) have shown that longitudinal dispersivities of 50 m to 300 m and a longitudinal/transverse ratio of 10 are appropriate for transport to the WIPP boundary. Here we use $\alpha_L = 200$ m and $\alpha_L/\alpha_T = 10$. If we assume an instantaneous release of 10^9 mg and a travel time of 2×10^6 d, PorousPuff generates concentrations in mg/m^3 , shown in Figure 60:

```
Grid: ptrack = PARTICLETRACK( dir, mag, 10000, 16000, #, # )  
Grid: plume = POROUSPUFF( ptrack, poro, thick, 1e9, 2e6, 200, 10, 1, 0 )
```

Using the stepped puff technique discussed above, we can follow this puff through time, allowing it to evolve in shape and character as it disperses into the heterogeneous domain. The following sequence of figures illustrates this evolution. This method produces much more realistic results than simply doing a series of one-step puffs. Notably, the mass is contained in the narrow channel of high transmissivity, whereas for a single puff it would have been dispersed regardless of the neighboring transmissivity values. However, this improvement comes at a price. Each stepped puff required about 24 hours of computer time on a 486/50 PC using the C version of PorousPuff, or 30 hours running within Grid on a Sun SPARC IPX.

Puff Step 1 for the WIPP Site

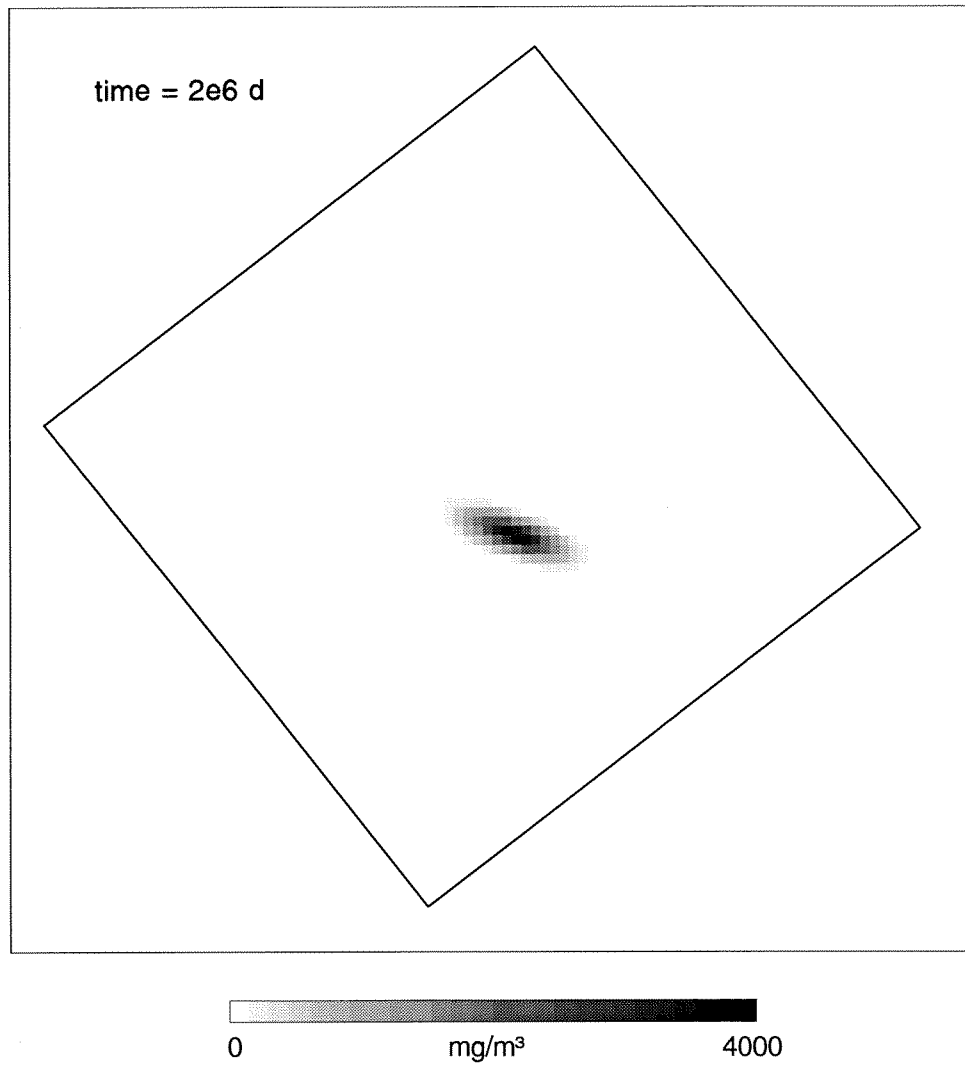


Figure 60 Initial puff released from the WIPP Site.

Puff Step 2 for the WIPP Site

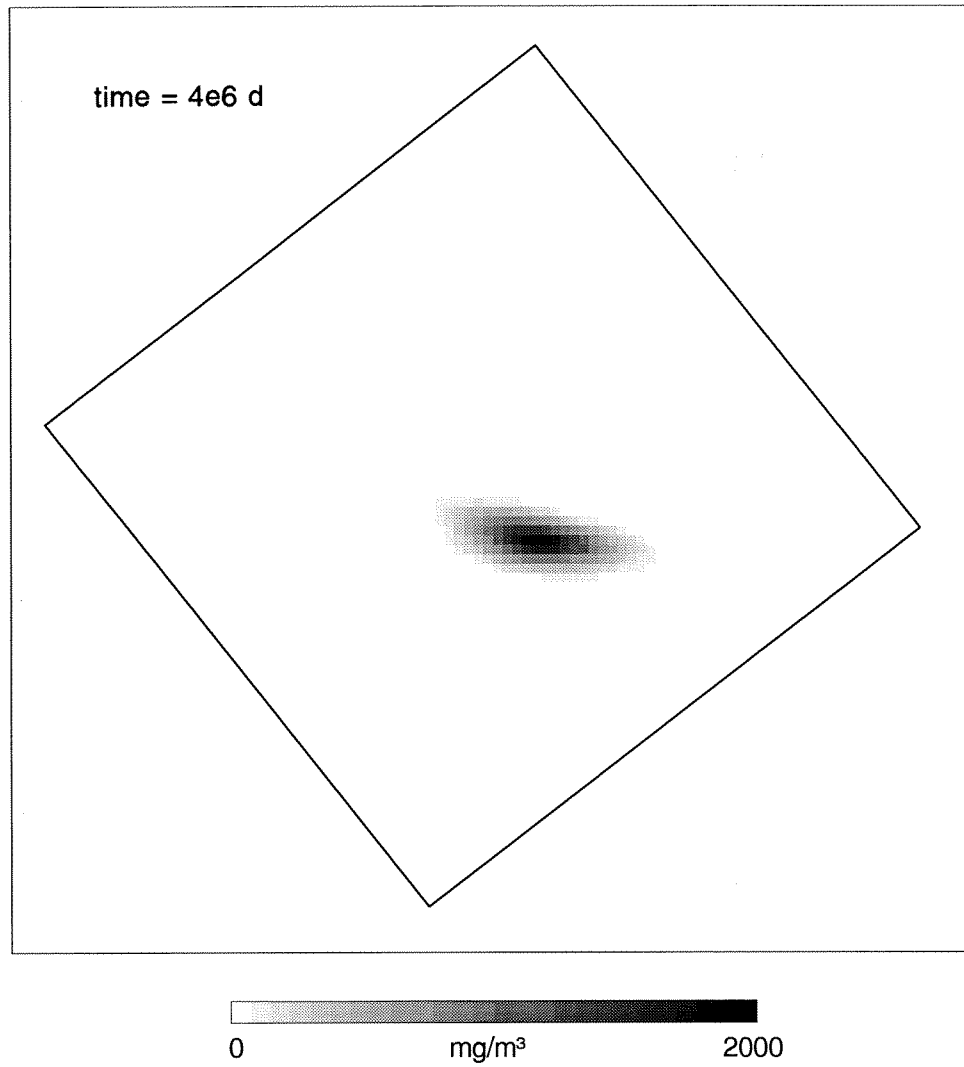


Figure 61 Step 2 in the evolving puff released from the WIPP Site.

Puff Step 3 for the WIPP Site

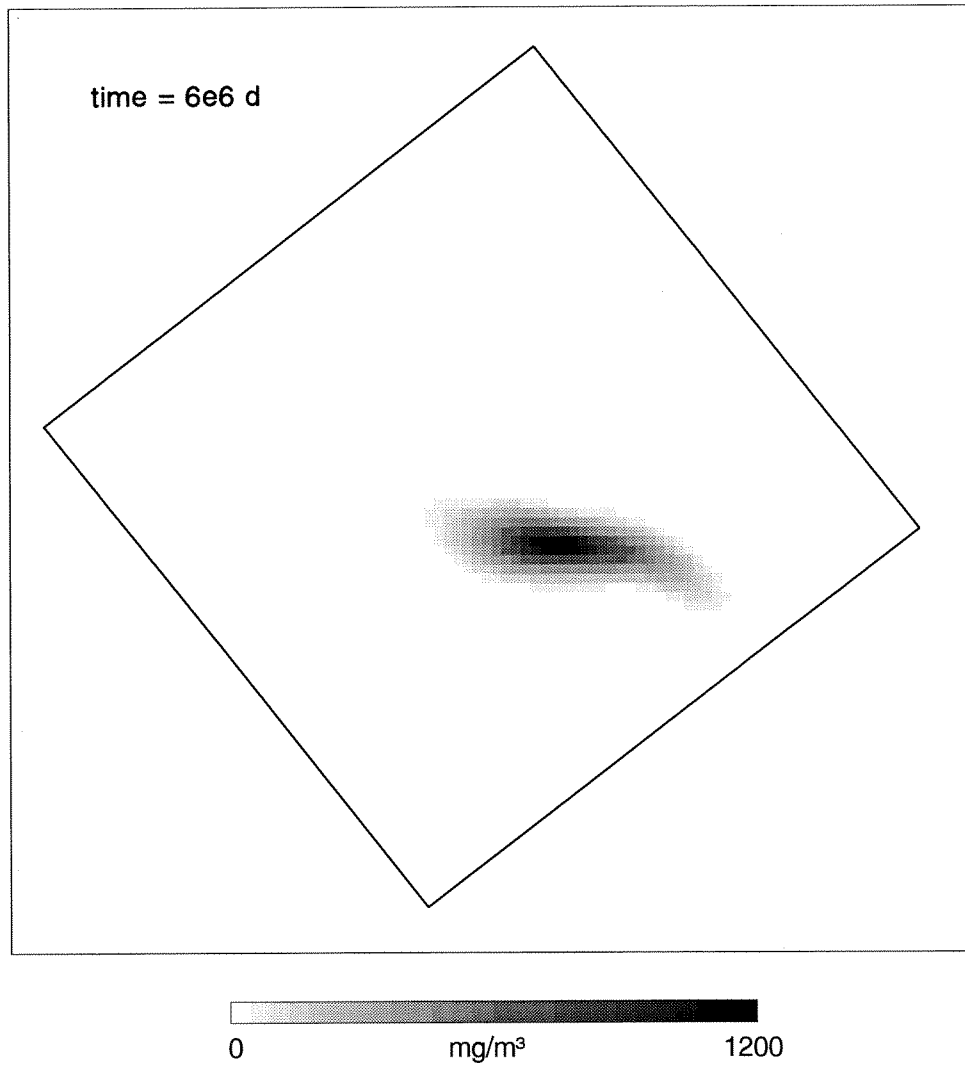


Figure 62 Step 3 in the evolving puff released from the WIPP Site.

Puff Step 4 for the WIPP Site

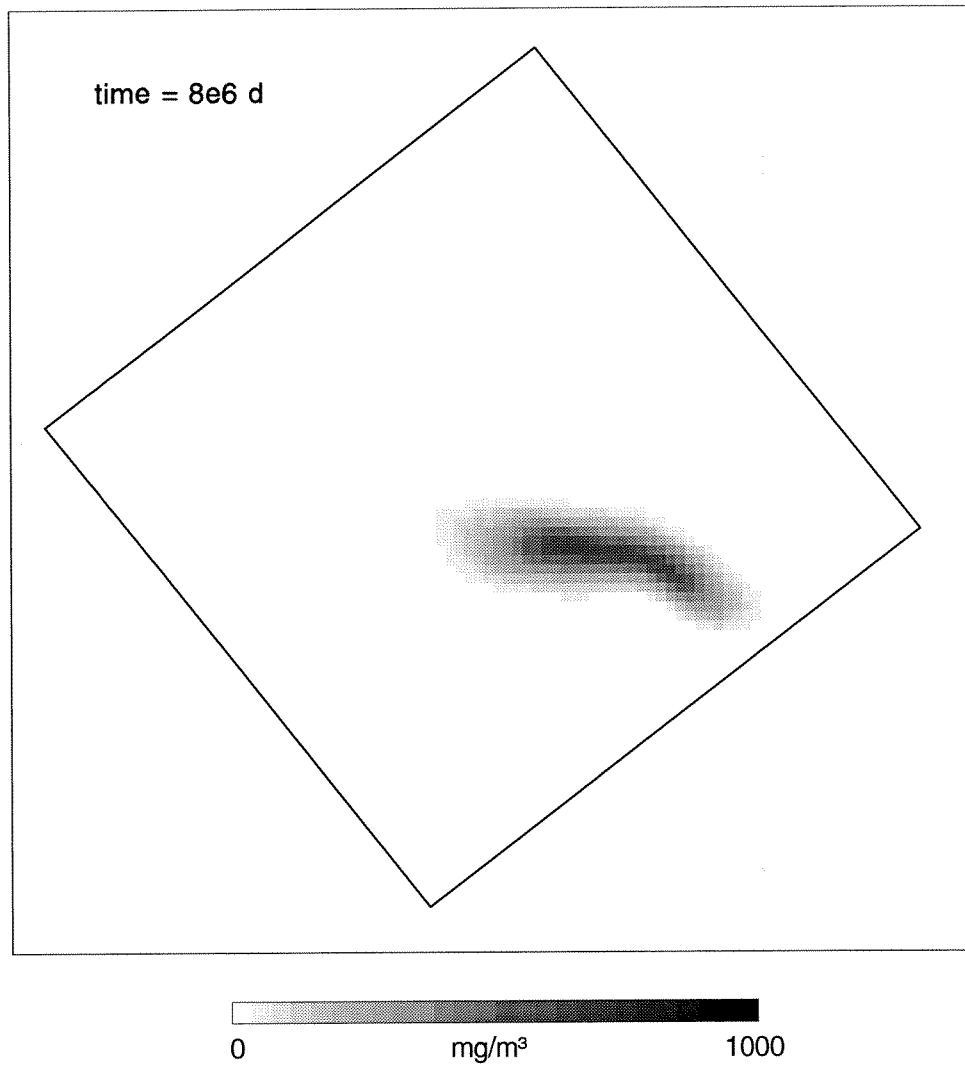


Figure 63 Step 4 in the evolving puff released from the WIPP Site.

Puff Step 5 for the WIPP Site

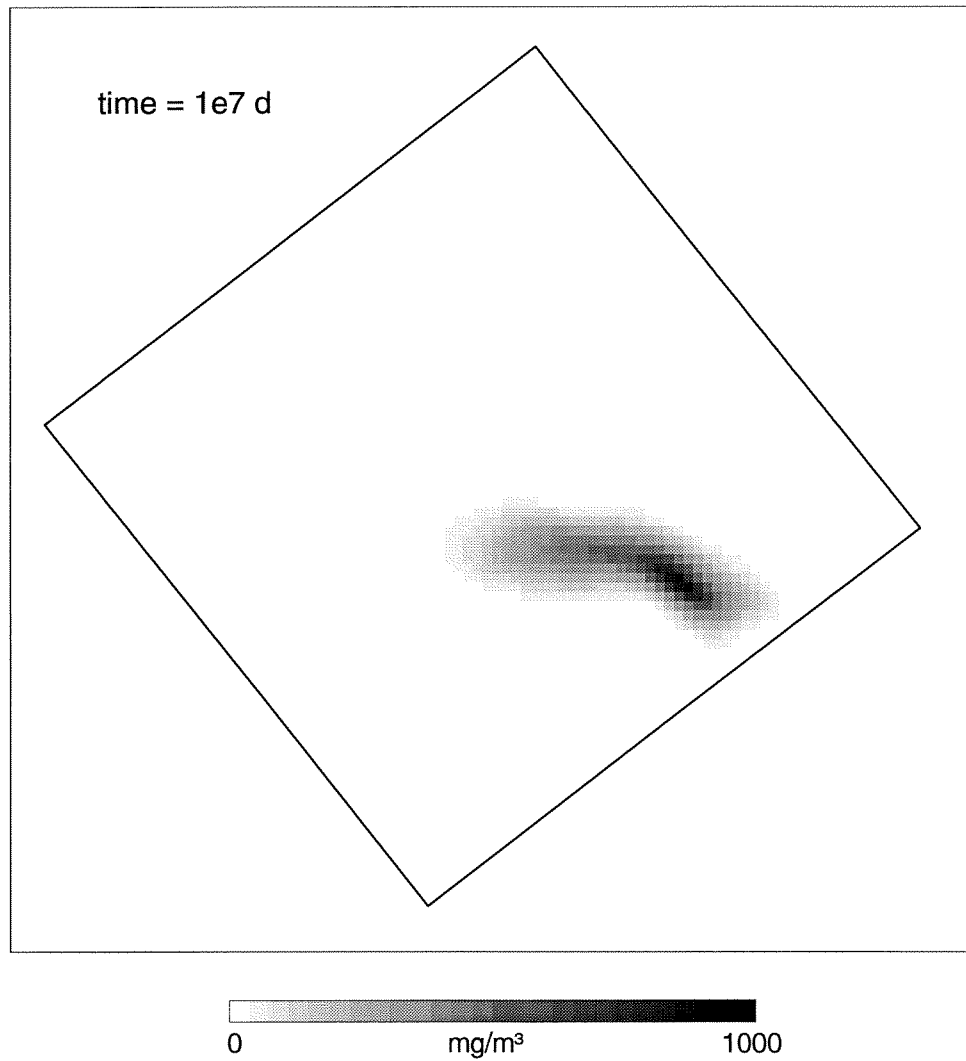


Figure 64 Step 5 in the evolving puff released from the WIPP Site.

Puff Step 7 for the WIPP Site

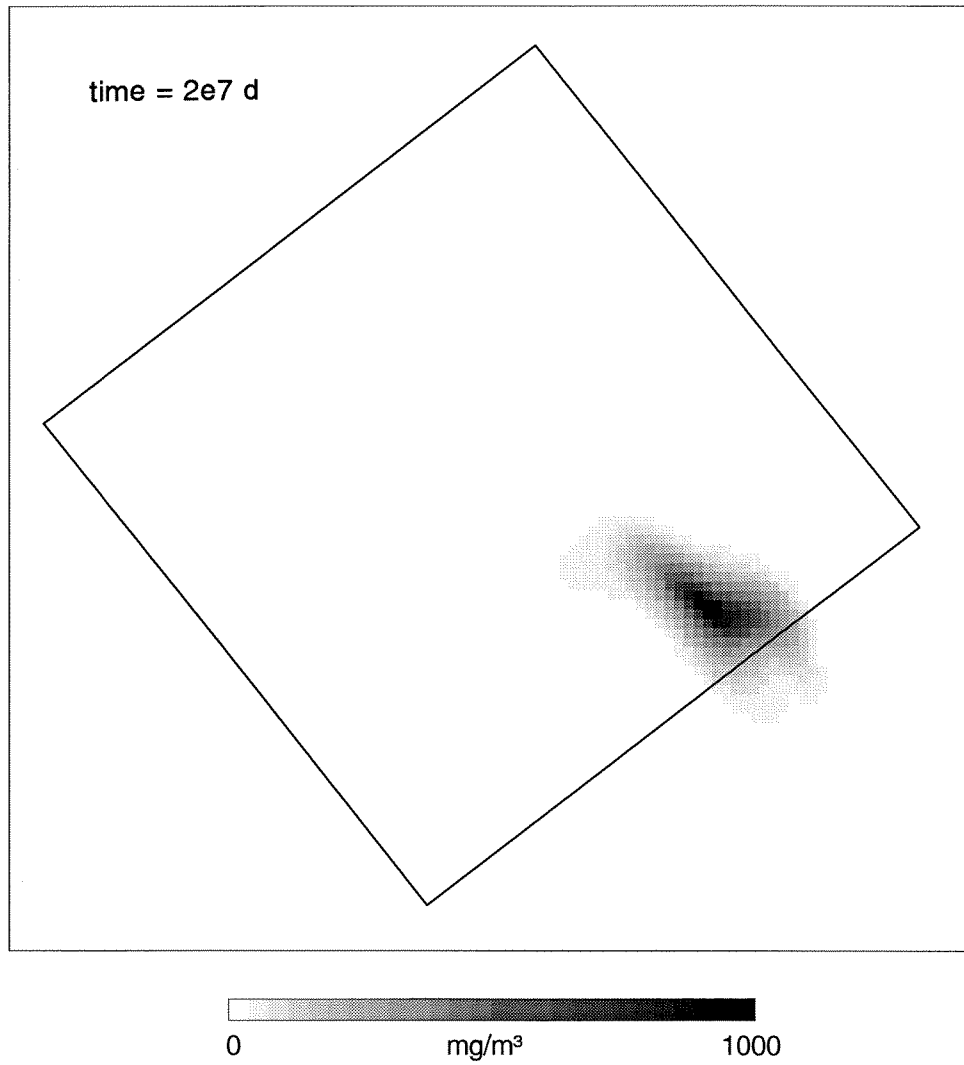


Figure 65 Step 7 in the evolving puff released from the WIPP Site.

Puff Step 9 for the WIPP Site

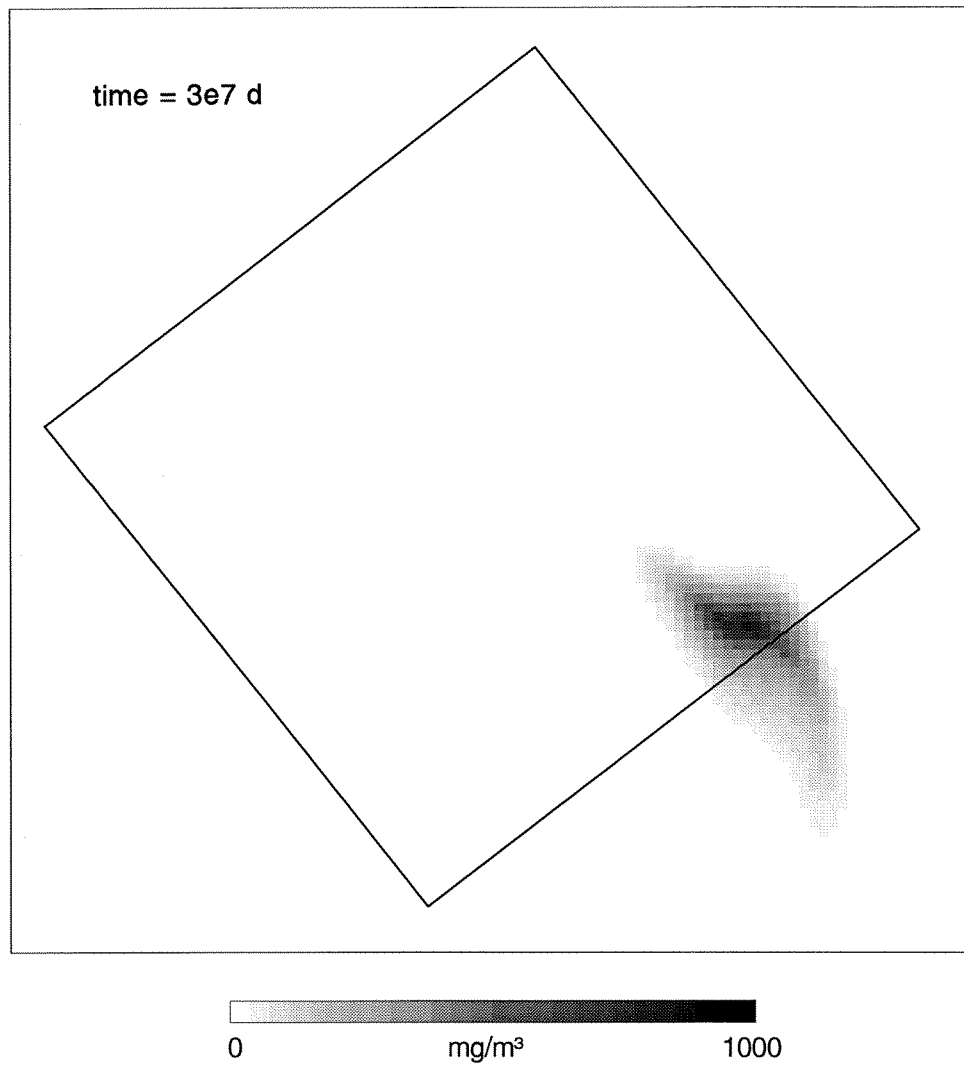


Figure 66 Step 9 in the evolving puff released from the WIPP Site.

Puff Step 11 for the WIPP Site

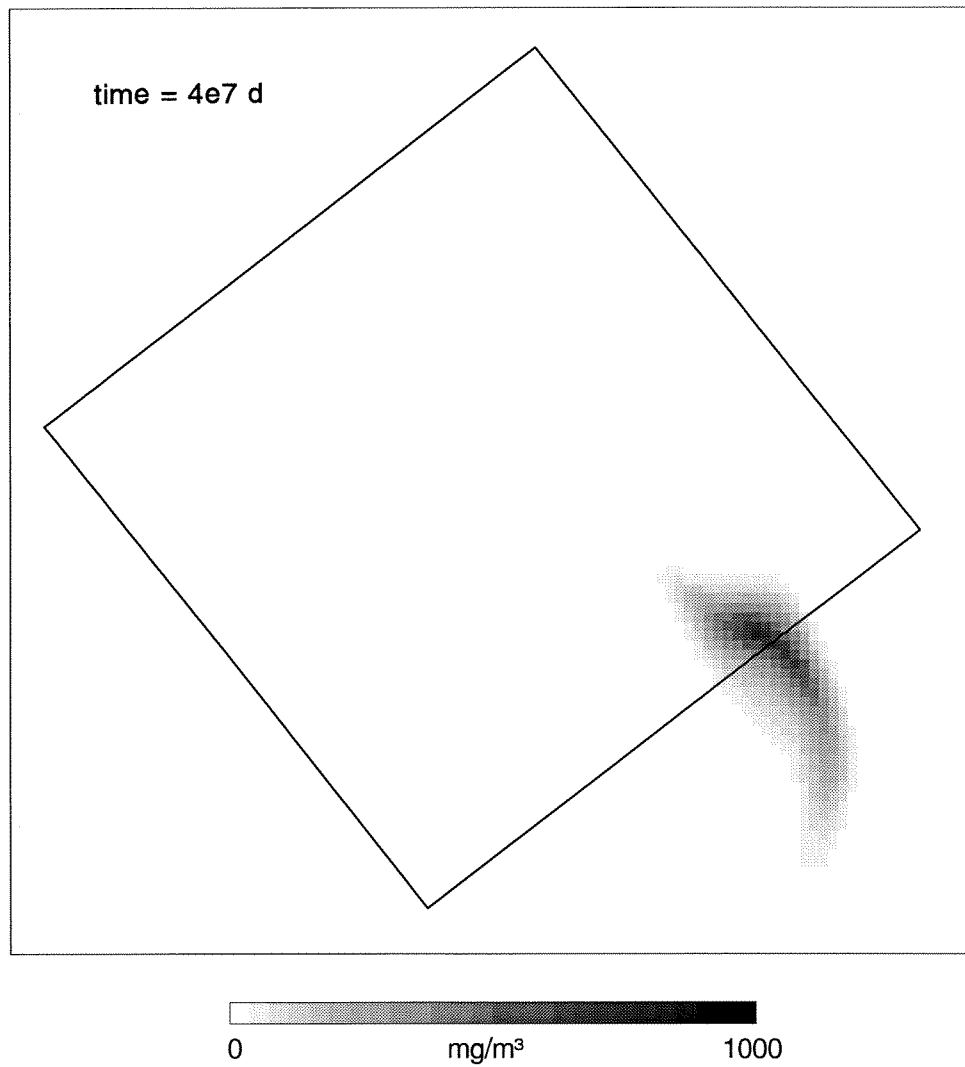


Figure 67 Step 11 in the evolving puff released from the WIPP Site.

5.3.4 Transport Modeling Using PorousPlume

Plume generation is also of interest to the WIPP study, since a release of contaminants may be more like a step than a pulse, depending on the repository breach scenario used (Bertram-Howery, et al., 1990). We will use the same transport parameters used in the preceding PorousPuff simulation. If we assume a release rate of 1000 mg/d, PorousPlume generates the plume with concentrations in mg/m³, shown in Figure 68:

```
Grid: ptrack = PARTICLETRACK( dir, mag, 10000, 16000, #, # )  
Grid: plume = POROUSPLUME( ptrack, poro, thick, 1000, 200, 10, 1, 0 )
```

Since the WIPP problem involves radioactive materials, we may add information for a particular radionuclide. Physical retardation from matrix precipitation in radionuclide transport can be as extreme as 10000 (for Pu, a high priority element), but is conservatively estimated at $R = 100$ (Ostensen, 1994). The decay coefficient for ²³⁹Pu with a half life of 2.41×10^4 years is $\lambda = 7.87 \times 10^{-8} \text{ d}^{-1}$. The corresponding plume (Figure 69) is generated with the commands:

```
Grid: ptrack = PARTICLETRACK( dir, mag, 10000, 16000, #, # )  
Grid: plume = POROUSPLUME( ptrack, poro, thick, 1000, 200, 10, 100, 7.87e-8 )
```

In comparing these two figures, the significance of retardation and decay of radioactive contaminants is apparent. We may easily generate these what-if scenarios using Grid's new groundwater modeling functions, and doing so puts them to their best use: as preliminary screening tools for the analysis of contaminant transport in groundwater.

Concentration Plume from the WIPP Site

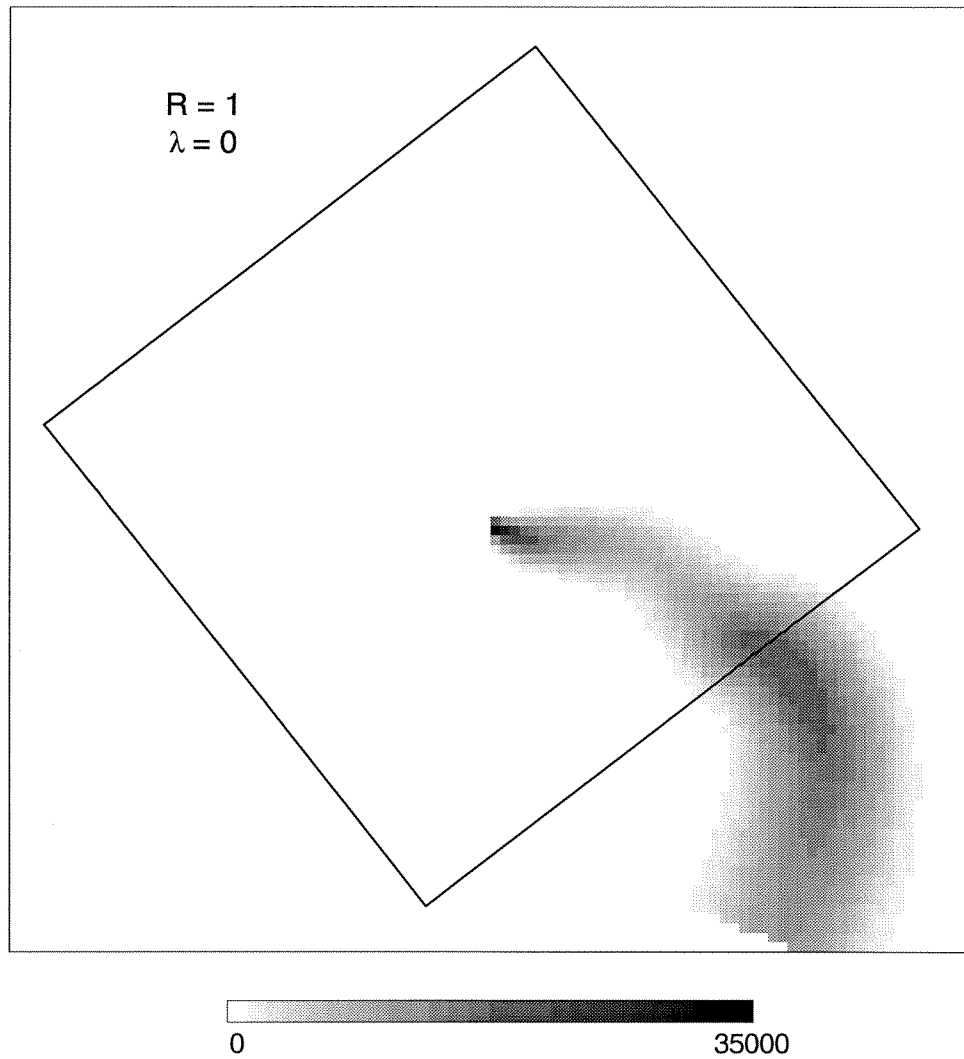


Figure 68 Plume of a conservative, nonreactive tracer from the WIPP Site, generated by PorousPlume.

Concentration Plume from the WIPP Site

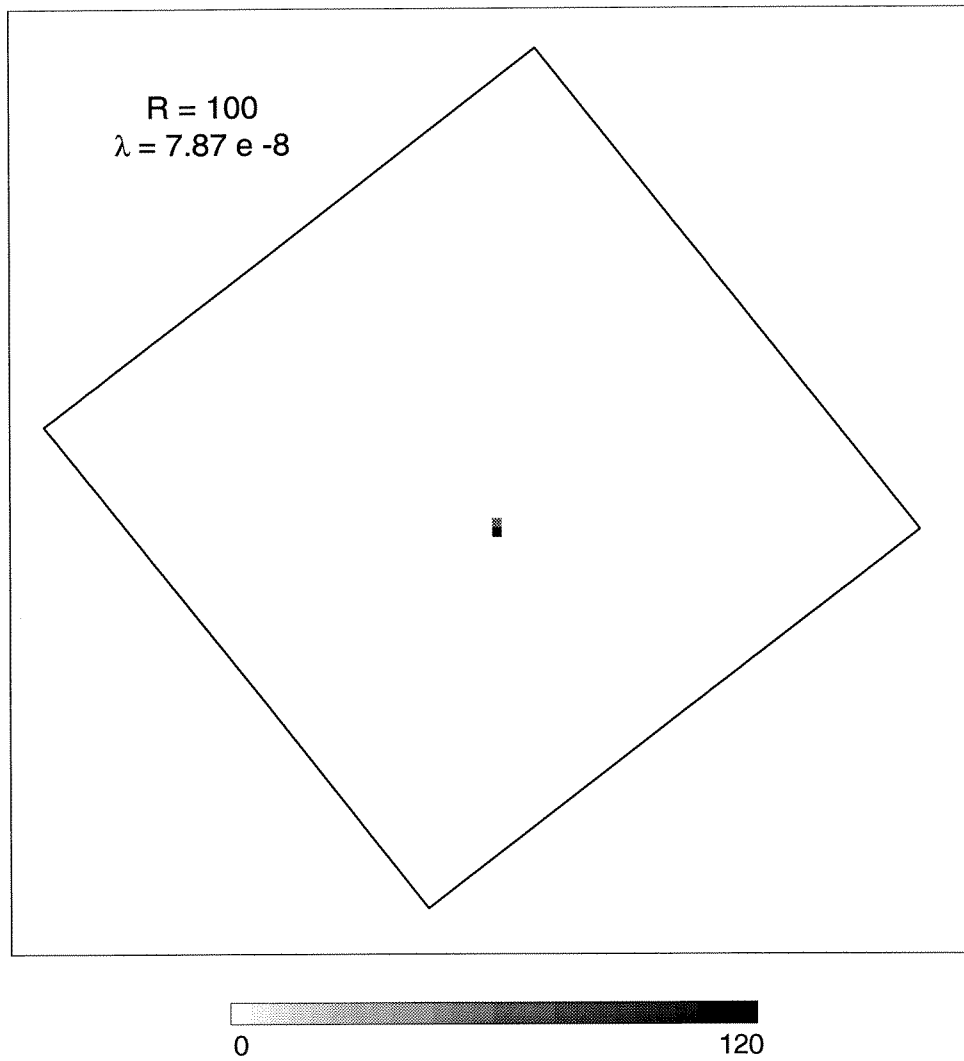


Figure 69 Plume of ^{239}Pu from the WIPP Site, generated by PorousPlume.

Chapter 6

Conclusions

6.1 Research Objectives Revisited

Chapter 1 presented several goals for this research project, which are restated here for the convenience of the reader. Each of these objectives has been met, and I believe this project to have been quite successful.

- Determine the most appropriate form of the solutions to the advection-dispersion equation applied to the conditions of instantaneous and continuous point source input of constituents into an aquifer.

The analytical solutions derived in Chapter 3 are computationally efficient and have a strong theoretical foundation. Since `PorousPuff` and `PorousPlume` are intended to be functions which execute rapidly rather than complex programs which involve a measure of numerical iteration, the analytical solutions are well suited for the purpose.

- Achieve the implementation of new functions into `Grid`. Once this has been demonstrated, the addition of other perhaps more sophisticated functions should be straightforward.

All of the five functions, `DarcyFlow`, `DarcyTrack`, `ParticleTrack`, `PorousPuff`, and `PorousPlume`, were successfully integrated into `Grid`. This was a cooperative effort, with the `Grid` developers working from their end and I from

mine. The details of the development are presented in Chapter 4, using the DarcyFlow function as an example. The codes for all the functions, presented in the Appendix, may be used as templates for additional functions with the aid of the ESRI Software Development Libraries (ArcSDL). The DarcyFlow, ParticleTrack, and PorousPuff functions have been adopted by ESRI for imminent release in Grid version 7.0.

- Integrate the new Grid model with existing geologically appropriate Arc/Info coverages to create a smoothly working modeling environment. This may involve the creation of Arc Macro Language (AML) scripts and dialogs to facilitate repetitive tasks.

A verification of the research was performed using hydrodynamic porous medium models with analytical solutions (the capture well and well dipole simulations in Chapter 5). A hydrogeological flow and transport problem, the modeling of the Culebra Dolomite overlying the Waste Isolation Pilot Plant, was also examined to illustrate the use of Grid as a modeling environment. AMLs were constructed to execute time-stepped simulations and to simplify the use of the functions, which have a complex argument structure.

- Assess the effectiveness of this approach to groundwater modeling. This is obviously critical to determining if this new technology will be useful to the modeling community.

The DarcyFlow, DarcyTrack, and ParticleTrack functions execute with sufficient speed that they are easily used with Grid and within AML programs.

Grid is also well suited for performing single-step simulations using *PorousPuff* and *PorousPlume* and map algebra analyses, but is too slow for effective time-stepped simulations, which are better done external to the GIS.

- Provide through this work a blazed trail for others interested in implementing new functions into Grid. If others see the utility of modeling in the GIS, I invite them to add more functions of general interest to the Grid program.

By closely following the discussion of programming for the GIS, the ambitious programmer may use this research to implement his own tools for environmental analysis in Grid. It must be admitted, however, that embarking on such a path is not for the timid. Depending on one's familiarity with integrating C and FORTRAN codes, and with GIS concepts, the development time for a simple working program may be on the order of days to weeks. By examining the change logs of the codes in the Appendix, the reader can see that the amount of time required to get these modeling functions developed, coded, running, and debugged was about two years.

- Test the hypothesis that a geographic information system can be used as an effective platform for performing groundwater modeling analysis.

This research, as well as the research of many other workers, confirms that groundwater modeling in GIS is easily performed, to an extent. The remaining difficulties, such as simultaneous solution of systems of equations, and the integration of time into GIS, are discussed in the following sections.

6.2 Discussion

The ability to explore groundwater flow and transport in a geographic information system has been demonstrated. The inherently geographic datasets which hydrogeologic analyses require are comfortably at home in the spatially oriented world of GIS, and the dynamics of porous medium flow implied by these data are brought to life with the new GIS functions DarcyFlow, DarcyTrack, ParticleTrack, PorousPuff, and PorousPlume. The traditionally static expression of geographic and environmental data has become dynamic, and the possibility of effective environmental modeling in GIS is imminent. However, before GIS can become a practical platform for sophisticated modeling, its developers must acquire a more numerical perspective. A programming strategy which has been sufficient for spatial data consisting primarily of integers and indices is inadequate for the demands of floating-point numbers and continuous surfaces and functions. Bringing the two viewpoints together in the software will be difficult and will require rehashing many core routines, but it is prerequisite to effective environmental modeling in the GIS.

As the GIS matures, more functions should be developed for modelers, for solving systems of equations to determine potential surfaces, for generating random fields by statistical or fractal algorithms, and for management and visualization of three-dimensional datasets, such as seismic lines and collections of well logs. This research has been a contribution to the world of GIS groundwater modeling. I hope that the results will be useful to and used by the modeling community. Anyone with access to Arc/Info datasets of geologic materials and source information will be interested, and I am eager to get feedback from users.

6.3 Refining the Technique

As this research progressed from concepts to working models, alternative methods of development have come to mind. There are some changes which I would make in retrospect, and there are some additional enhancements which could be made. For example, DarcyFlow was developed as a function which reads and writes only grids, and performs the same operation on each cell, using data from the surrounding cells. As such, it could be rewritten using existing map algebra commands, as an AML containing a DOCELL loop. This would be easier to maintain as a program, since it is not integrated into Grid, but there would be a performance loss, since DarcyFlow works more efficiently as a compiled program.

The units of the volume balance residual grid created by DarcyFlow are $\text{length}^3/\text{time}$, since it originated as a balance of volume fluxes into and out of the grid cell. Unfortunately, the value is dependent on the dimensions of the grid cell, which is also involved in the calculation. To remove this dependence, the volume balance residual grid should be replaced with a residual with volume per area units of $\text{length}/\text{time}$. This per area flux is easily calculated by dividing the volume residual by the cell area. These values would be independent of the grid cell size.

There is a conceptual inconsistency in using a variable porosity and thickness fields in the dispersion calculations. The gaussian distributions of concentration use the single value of porosity and thickness of the cell at the centroid of the dispersion ellipse in scaling the concentration, so these values should be forced to be constant over the entire domain. A further inconsistency is found in having a variable porosity but a constant retardation factor, since

porosity enters into its calculation. In practice, the porosity and thickness are often given constant values, as they were for the WIPP example problems, but perhaps they should be provided to the function as values rather than grids. This would also improve performance since these grids would not have to be read. As a workaround, one can specify constant-valued grids for these parameters.

PorousPlume uses a curvilinear coordinate system for shaping the gaussian plume along a particle path, but PorousPuff does not. The puff uses a rectilinear coordinate system which is at its origin tangent to the path. This results in the curious behavior of swinging around bends in the path and sending its tails into possibly inappropriate areas. PorousPuff could be redesigned to use the curvilinear system used by PorousPlume, so that the puff would be molded to the path. This would also produce a better fit in heterogeneous fields between PorousPlume and the plume convolved from repeated applications of PorousPuff.

6.4 Enhancing Grid

There are several specific problems with the current configuration of Grid which need to be addressed. Once these obstacles have been overcome, Grid will be a much more interesting and flexible platform for environmental modeling.

To enhance processing speed, Grid should allow more data to reside in memory, thereby reducing the frequency of disk reads and writes. Currently, grid data can be accessed either on a per-cell or per-row basis. For small problems, there may be sufficient RAM to read an entire grid into an array. This should be done when possible by checking available memory, resorting to the per-row reads only if necessary. To address this problem, recently developed Grid functions have been better optimized to minimize disk access (Gao, 1991).

Processing in AML could be enhanced a great deal by providing a function to return the floating point value of a particular cell in a grid. As of version 6.1.1, this most basic of needs is not met, and one is reduced to capturing string output from `CELLVALUE` in a watch file which must then be opened, read, closed, and deleted (see `MULTIPUFF.AML` in the Appendix). This problem has been addressed in version 7.0 by adding the `SHOW CELLVALUE` function (which puts the value of a cell into an AML-accessible variable), but the precision of a floating point number is limited to three points after the decimal, and exponential notation is not supported.

It would also be of great benefit to be able to control the order of execution in a `DOCELL` loop, which performs a set of commands for each cell in the grid, but does not guarantee the order in which it is done. For example, the Gauss-Seidel method for solving a system of linear equations (as in a head surface) makes use of neighboring cells which have already been through one processing step. This works well for orderly processing like left-to-right, row-by-row, but will not work if done out of order. The user should be able to impose this order on `DOCELL`, which might otherwise operate in its current fashion, which is commendably optimized to reduce disk access.

Modeling efficiency can be enhanced by relaxing the constraint of working on a regular grid. The regular grid, an artifact from Grid's origins in raster image processing, imposes the same density of calculations in areas of little interest as those of great interest. Allowing a nonuniform grid, where data are still stored in rows and columns, but each of independent width, will reduce the number of unnecessary calculations. Unfortunately, it would require a complex rewrite of virtually all of the Grid functions, commands, and operators, and would require a more complicated data structure (Fortner, 1992).

The advent of symmetric multiprocessor (SMP) computers brings to Grid the possibility of great performance enhancements. Since many of the processes done by map algebra are repetitive independent tasks, like adding two grids on a cell-by-cell basis, the SMP machine can easily subdivide the grids and work on the pieces in parallel. DarcyFlow, PorousPuff, and PorousPlume could all benefit greatly from this technique, though the inherently one-dimensional particle tracking routines would not. Nonetheless, most of Grid's work could make good use of SMP technology.

Another major enhancement to Grid has been long discussed: going 3D. A limitation which Grid shares with many other modeling platforms is its two-dimensionality. As the modeling world comes to realize that 2D is conceptually inadequate for modeling most natural processes, the demand for 3D increases (Ventura, et al., 1993, Raper, 1993, Fisher, 1993, and Crosbie, 1993). There is also a demand for implementing time into the GIS, perhaps as a fourth dimension (Dodson, 1992, Maidment, 1991, 1993b). Moving to three and four dimensions would also be another major rewrite for Grid, but would make it much more interesting for groundwater modeling.

6.5 The Future of Environmental Modeling in GIS

Now that adding simple environmental modeling functions has been demonstrated, we can imagine more sophisticated techniques. GIS seems to be a natural environment for analytical element modeling, for example. Simple numerical techniques for contaminant transport, such as Lagrangian particle tracking, are also intriguing, but numerically demanding processes like solving systems of equations, are still inefficient in the GIS environment (McKinney and

Tsai, 1993). As another example, to complement DarcyFlow a more sophisticated technique has been presented (just at the time of this writing, Zhang, et al., 1994) for generating a continuous velocity field, given the flow field, by cubic splines. Since it serves a similar purpose, the technique could complement the DarcyFlow function, or could be combined with it so that the user could select which method to use. Unfortunately, there was no time to evaluate the technique for this publication.

The implementation of models into Grid can be extended to other areas of environmental modeling. Schimoller (1992) has prepared an air pollution model which predicts the concentration distribution from multiple smoke stack plumes, based on the Industrial Source Complex model (U.S.E.P.A., 1979). This model could readily be implemented into Grid and would prove a useful tool for air pollution monitoring and exposure assessment. Flow and transport of surface waters, the modeling of coastal processes, and ecological modeling are also areas rich with possibility.

The most fruitful work for the near future, however, will be outside of the GIS. In developing the functions presented in this research, the author had to resort to an alternative computer platform. Working with the GIS during the development phase was simply too time-consuming, so the codes were originally written, debugged and fine-tuned as stand-alone programs working with data in ASCII GRID files. This was found to be so much faster and more stable that most of the work in developing examples and running simulations for this dissertation was done with these C codes, running on a 486/33 PC. This work was later confirmed to give the same results using the GIS, albeit more slowly.

This development strategy gave birth to the idea that the entire library of spatial analysis functions and map algebra routines available in the GIS could be

written as programs to be run from the system prompt which work on independent datasets. These datasets could well be in a more flexible format than the ASCII GRID files, allowing for variable cell widths and three dimensions. For future work, I imagine a complete set of programs, or better, a complete library of map algebra and spatial analysis subroutines which could be used by any C programmer for any purpose, including further GIS development. I have already developed several of these programs for performing groundwater modeling functions, arithmetic operations, mathematical transformations, and data translations from ASCII GRID to other data formats. This library can be enhanced as new functional demands arise, and may well spin off many new environmental modeling functions which can be incorporated into a GIS.

To create a more flexible and generalized modeling system, the mathematics behind the functions should be made more explicit. For example, the DarcyFlow function employs two fundamental mathematical concepts: the gradient and the divergence of the vector field. This suggests that the functions GRAD and DIV would be useful as primitives (Maidment, 1993b), and could easily replace DarcyFlow. PorousPuff uses Green's functions, which could be coded as primitive operators. There is a host of advanced mathematical constructs which could be employed at the same level as map algebra addition, and multiplication, operating on vector fields and surfaces. Kemp (1993) has recognized the limitations of current GIS in representing continuous fields, bringing to the imagination new data types which do not demand discretization. This map calculus may seem far removed from the GIS world, but as we recognize the need for direct mathematical modeling of geographical and spatial data, the union seems natural.

LIST OF SYMBOLS

| | |
|--------------|---|
| a | coefficient or constant; also a_0, a_1, a_2, a_3 |
| A | area |
| b | saturated thickness of an aquifer |
| c | bulk concentration of a solute |
| c_{aq} | aqueous concentration of a solute |
| c_s | sorbed concentration |
| C | generic constant |
| D | dispersion coefficient; also D_{xx}, D_{yy}, D_L, D_T |
| \mathbf{D} | dispersion tensor |
| F | fractional breakthrough |
| h | peizometric head |
| H | head of an unconfined aquifer |
| i | index along x coordinate |
| j | index along y coordiante |
| \mathbf{J} | mass flux vector |
| K | hydraulic conductivity of a porous medium |
| K_d | distribution coefficient |
| ℓ | cell size |
| log | base 10 logarithm; \log_{10} |
| ln | natural logarithm |
| L | length; step length; dimension of length |
| m | mass |
| \dot{m} | mass release rate |
| mg | milligram |
| M | dimension of mass |
| n | porosity |
| N | total number; number density |
| P | point label, also P_i, P_{i+1}, \dots |
| q | specific discharge; Darcy flux |
| Q | volumetric discharge rate |
| r | radius |

| | |
|-----------|--|
| r^2 | regression coefficient |
| R | retardation factor |
| s | spatial coordinate along a curve |
| S | coefficient of storage; storativity |
| S_y | specific yield |
| S^+ | source strength; strength of mass production within a control volume |
| t | time |
| t_F | time of fractional breakthrough |
| T | transmissivity of a porous medium; dimension of time |
| U | aquifer flux vector |
| v | seepage velocity; fluid velocity; also v_x, v_y, v_L, v_T |
| \bar{v} | seepage flow average velocity vector |
| w | weights for Gauss-Legendre quadrature integration |
| W | rate of recharge |
| x | spatial coordinate in (x,y) space |
| X_L | longitudinal coordinate in (X_L, X_T) space |
| X_T | longitudinal coordinate in (X_L, X_T) space |
| y | spatial coordinate in (x,y) space |
| α | dispersivity; also α_L, α_T |
| λ | first-order decay constant |
| ξ | coordinate for Gauss-Legendre quadrature integration |
| η | coordinate for Gauss-Legendre quadrature integration |
| ρ_b | bulk density |
| σ | standard deviation; also σ_L, σ_T |
| Ψ | stream function |

BIBLIOGRAPHY

- Abramowitz, M., and I.A. Stegun, 1964, Handbook of Mathematical Functions, Dover, New York
- Acevedo, M.F., D.L. Urban, and M. Alban, 1993, "Landscape Scale Forest Dynamics: GIS, GAP, and Transition Models", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Allewijn, R., 1986, "Methodology for Integrating Satellite Imagery and Field Observations for Hydrological Regionalisation in Alpine Catchments", *Symposium on Remote Sensing for Resources Development and Environmental Management, Enschede, August 1986, Vol. 2, pp 693-697*
- Ambrose, R.B. Jr., T.A. Wool, J.L. Martin, J.P. Connolly, and R.W. Schanz, 1987, WASP4: A Hydrodynamic and Water Quality Model, *Environmental Research Laboratory, U.S. Environmental Protection Agency, Athens, GA*
- Anderson, M.P., 1979, "Using Models to Simulate the Movement of Contaminants Through Groundwater Flow Systems", *CRC Critical Review of Environmental Control*, 9, pp 97-156
- Baker, C.P. and E.C. Panciera, 1990, "Geographic Information System for Groundwater Protection Planning", *Journal of Soil and Water Conservation*, 45 (2), pp 246-248
- Barbé, D.E., H. Miller, and S. Jalla, 1993, "Development of a Computer Interface Among GDS, SCADA, and SWMM for Use in Urban Runoff Simulation", Symposium on Geographic Information Systems and Water Resources, Mobile, AL, March 14-17, 1993, *American Water Resources Association Technical Publication TPS-93-1, AWRA, Bethesda, MD*

- Barrocu, G., and G. Biallo, 1993, "Application of GIS for Aquifer Vulnerability Evaluation", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211*, IAHS Press, Wallingford, Oxfordshire, UK
- Batelaan, O., F. deSmedt, M.N. Otero Valle, and W. Huybrechts, 1993, "Development and Application of a Groundwater Model Integrated in the GIS GRASS", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211*, IAHS Press, Wallingford, Oxfordshire, UK
- Battaglin, W.A., 1989, "Method for Estimating the Water-Table Altitude in a Coastal-Plain Aquifer Using a Geographic Information System", Geoscientific Information Systems Applied to Exploration and Research, Proceedings of a Conference held in Lakewood, CO, September 23-26, 1989, pp 76-84
- Battaglin, W.A., 1993, "Use of Volume Modelling Techniques to Estimate Agricultural Chemical Mass in Groundwater, Minnesota, USA", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211*, IAHS Press, Wallingford, Oxfordshire, UK
- Bear, J., 1972, Dynamics of Fluids in Porous Media, *Dover*, New York, NY
- Bear, J., 1979, Hydraulics of Groundwater, *McGraw Hill*, New York, NY
- Bear, J., and M. Jacobs, 1965, "On the Movement of Water Bodies Injected into Aquifer", *Journal of Hydrology*, 3 (1), pp 37-57
- Becker, E.B., G.F. Carey, and J.T. Oden, 1981, Finite Elements, an Introduction Volume I, *Prentice-Hall*, Englewood Cliffs, NJ

- Beller, A., 1993, "Temporal GIS Requirements for Environmental Modeling", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Beyeler, W., 1994, "Evaluation of Dispersivity Estimates Used in the 1992 WIPP PA", in Summary of the WIPP Performance Assessment Geostatistics Expert Consultatnt Group Meeting of March 23-25, 1993, memo dated 7 Feb 1994, *Sandia National Laboratories, Albuquerque, NM*
- Bertram-Howery, S.G., M.G. Marietta, R.P. Rechard, P.N. Swift, D.R. Anderson, B.L. Baker, J.E. Bean, Jr., W. Beyeler, K.F. Brinster, R.V. Guzowski, J.C. Helton, R.D. McCurley, D.K. Rudeen, J.D. Schreiber, and P. Vaughn, 1990, Preliminary Comparison with 40 CFR Part 191, Subpart B for the Waste Isolation Pilot Plant, December 1990, SAND90-2347, *Sandia National Laboratories, Albuquerque, NM*
- Bissex, D., 1991, "EPA Uses GIS Software to Manage Environmental Data", *Water Environment & Technology*, 3 (12), pp 22-23
- Blodgett, J.C., M.E. Ikehara, and G.E. Williams, 1990, "Monitoring Land Subsidence in Sacramento Valley, CA, Using GPS", *Journal of Surveying Engineering (ASCE)*, 116 (2), pp 112-130
- Brinster, K., 1991, Preliminary Geohydrologic Conceptual Model of the Los Medaños Region Near the Waste Isolation Pilot Plant for the Purpose of Performance Assessment, SAND89-7147, *Sandia National Laboratories, Albuquerque, NM*
- Broten, M., L. Fenstermaker, and J. Shafer, 1987, "Automated GIS for Ground Water Contamination Investigation", Proceedings of the NWWA Conference on Solving Ground Water Problems with Models, *National Water Well Association, Dublin, OH*, pp 1143-1161

- Buckley, D.J., M. Coughenour, C. Blyth, D. O'Leary, and J. Bentz, 1993, "The Ecosystem Management Model Project: Integrating Ecosystem Simulation Modeling and ARC/INFO in the Canadian Parks Service", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Bugliosi, E.F., 1990, "Plan of Study for the Ohio-Indiana Carbonate-Bedrock and Glacial-Aquifer System", *USGS Open-File Report 90-151*
- Burgin, J., in progress, "Automating the Allocation of Water Supplies in Texas Using an Expert System to Control the Interaction of a Geographic Information System and a Linear Programming Solution Algorithm", *Dissertation for the Degree of PhD in Civil Engineering, The University of Texas at Austin, Austin, TX*
- Burrough, P.A., 1992, "Are GIS Data Structures Too Simple Minded?", *Computers and Geosciences* 18(2), pp 395-400
- Burrough, P.A., 1993, "Spatial Data Quality and Error Analysis Issues: GIS Functions and Environmental Modeling", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Cordes, C., and W. Kinzelbach, 1992, "Continuous Groundwater Velocity Fields and Path Lines in Linear, Bilinear, and Trilinear Finite Elements", *Water Resources Research*, 28 (11), pp 2903-2911
- Craig, P.M., 1993, "Basinwide Water Quality Planning Using the QUAL2E Model in a GIS Environment", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*

- Crosbie, P., 1993, "A New Approach to Environmental Modeling", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Cuddy, S.M., J.R. Davis, and P.A. Whigham, 1993, "An Examination of Integrating Time and Space in an Environmental Modelling System", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Dagan, G., 1994, "Upscaling of Dispersion Coefficients in Transport through Heterogeneous Formations", in Summary of the WIPP Performance Assessment Geostatistics Expert Consultatnt Group Meeting of March 23-25, 1993, memo dated 7 Feb 1994, *Sandia National Laboratories, Albuquerque, NM*
- D'Agnese, F.A., A.K. Turner, and C.C. Faunt, 1993, "Using Geoscientific Information Systems for Three-dimensional Regional Ground-Water Flow Modeling in the Death Valley Region, Nevada and California", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Darcy, H., 1856, Les Fontaines Publiques de la Ville de Dijon, *Victor Dalmart, Paris*
- Deckers, F., 1993, "EGIS: A Geohydrological Information System", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211, IAHS Press, Wallingford, Oxfordshire, UK*

- DePinto, J.V., J.F. Atkinson, H.W. Calkins, P.J. Densham, W. Guan, H. Lin, F. Xia, D.W. Rodgers, T. Slawewski, and W.L. Richardson, 1993, "Development of GEO-WAMS: A Modeling Support System for Integrating GIS with Watershed Analysis Models", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Diersch, H.-J., R. Gründler, S. Kaden, and I. Michels, 1992a, "ARC/INFO-based Groundwater Modeling in Environmental Planning and Assessment", in Proceedings of the Twelfth Annual ESRI User Conference, vol. III, p 419, *Environmental Systems Research Institute, Redlands, CA*
- Diersch, H.J.G., R. Gründler, S. Kaden, and I. Michels, 1992b, "Toward GIS-Based 3D/2D Groundwater Contamination Modeling Using FEM", Numerical Methods in Water Resources Vol. 1, Computational Methods in Water Resources IX, *Computational Mechanics Publications, Boston, and Elsevier Applied Science, New York, NY*, pp 749-760
- Dijkstra, E.W., 1959, "A Note on Two Problems in Connection with Graphs", *Numerical Mathematics I*, pp 269-271
- Djokic, D., 1993, "Towards a General Purpose Spatial Decision Support System Using Existing Technologies", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Dodson, R.D., 1992, "Advances in Hydrologic Computation", in Handbook of Hydrology, D.R. Maidment, ed., *McGraw-Hill*
- Einstein, A., 1926, Investigations on the Theory of the Brownian Movement, *Methuen and Co., Ltd.*; republished in 1956 by *Dover Publications*

- Earp, D., 1988, "Ground Water Quality Protection and Monitoring Programs in Albuquerque", Proceedings of the 32nd Annual New Mexico Conference on Ground Water Management, November 5-6, 1987, Albuquerque, NM, pp 45-55
- ESRI, 1991a, Cell-based Modeling with Grid, *Environmental Systems Research Institute, Inc.*, Redlands, CA
- ESRI, 1991b, Grid Command References, *Environmental Systems Research Institute, Inc.*, Redlands, CA
- Evans, B.M. and W.L. Myers, 1990, "GIS-Based Approach to Evaluating Regional Groundwater Pollution Potential with DRASTIC", *Journal of Soil and Water Conservation*, 45 (2), pp 242-245
- Evans, T., D. Djokic, and D.R. Maidment, 1993, "Development and Application of an Expert Geographic Information System", *Journal of Computing in Civil Engineering, ASCE*, 7 (3), pp 339-353
- Faist, M. Personal communication, 1993
- Federal Register, 1986, 51 FR 1649, vol. 51, no. 9, January 14, 1986
- Fedra, K., 1993, "GIS and Environmental Modeling", in Environmental Modeling with GIS, M.F. Goodchild, B.O. Parks, and L.T. Steyaert, eds., *Oxford University Press*
- Fetter, C.W., 1993, Contaminant Hydrgeology, *Macmillan*, New York, NY
- Fisher, T.R., 1993, "Integrated Three-Dimensional Geoscientific Information Systems (GSIS) Technologies for Groundwater and Contaminant Modelling", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211, IAHS Press*, Wallingford, Oxfordshire, UK

- Fitzsimmons, C.K., M.J. Hewitt, and F. Mynar, 1988, "Hazard Ranking: A Job for Geographic Information Systems", HWHM 88: Hazardous Wastes and Hazardous Materials, Proceedings of the 5th National Conference, April 19-21, 1988, Las Vegas, NV, pp 381-385
- Fortner, B., 1992, The Data Handbook, A Guide to Understanding the Organization and Visualization of Technical Data, *Spyglass, Inc.*, Champaign, IL
- Franz, T., and N. Guiguer, 1990, FLOWPATH, Two-Dimensional Horizontal Aquifer Simulation Model, *Waterloo Hydrogeologic Software*, Waterloo, Ontario
- French, D.P., and M. Reed, 1993, "Integrated Environmental Impact Model and GIS for Estimating Natural Resource Damages from Oil and Chemical Spills into Aquatic Systems", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Friedman, B., 1956, Principles and Techniques of Applied Mathematics, *John Wiley and Sons, Inc.*, New York, NY
- Frysinger, S.P., D.A. Copperman, and J.P. Levantino, 1993, "Environmental Decision Support Systems: An Open Architecture Integrating Modeling and GIS", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Gao, P., 1991, GRID Hydrologic Tools, ESRI Internal Technical Report, *Environmental Systems Research Institute Inc.*, Redlands, CA
- Gao, P., C. Zhan, and S. Menon, 1993, "An Overview of Cell Based Modeling with GIS", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*

- Gallegos, D.P., G. de Marsily, M.G. Marietta, 1994, Summary of the WIPP Performance Assessment Geostatistics Expert Consultant Group Meeting of March 23-25, 1993, memo dated 7 Feb 1994, *Sandia National Laboratories*, Albuquerque, NM
- Gelhar, L.W., A. Mantoglu, C. Welty, and K.R. Rehfeldt, 1985, A Review of Field-Scale Physical Solute Transport Processes in Saturated and Unsaturated Porous Media, *Electric Power Research Institute*, EPRI EA-4190 Project 2485-5
- Gelhar, L.W., C. Welty, and K.R. Rehfeldt, 1992, "A Critical Review of Data on Field-Scale Dispersion in Aquifers", *Water Resources Research*, 28 (7), pp 1955-1974
- Goodchild, M.F., 1991, "Spatial Analysis with GIS: Problems and Prospects", *GIS/LIS '91 Proceedings*, October 28 - November 1, 1991, Atlanta, GA
- Goodchild, M.F., 1993, "The State of GIS for Environmental Problem Solving", in Environmental Modeling with GIS, M.F. Goodchild, B.O. Parks, and L.T. Steyaert, eds., *Oxford University Press*
- Grossa, J., R. Klopčič, R. Pape, and R. Yuill, 1988, "Basic Local-Level Water Resource Data Base", *Michigan Institute of Water Research, Technical Report No. 88-G1569-33*
- Grove, D.B., and W.A. Beetem, 1971, "Porosity and Dispersion Constant Calculations for a Fractured Carbonate Aquifer Using the Two-Well Tracer Method", *Water Resources Research*, 7 (1), pp 128-134
- Gupta, S.K., U.P. Singh, and R. Gupta, 1989, "Stringfellow Hazardous Waste Site: Computerized Data Management", Water: Laws and Management, *American Water Resources Association*, Bethesda, MD, pp 13A-1 to 13A-10

- Hathhorn, W.E., 1990, "Diffusion Theory in the First Passage Time Problem in Porous Media", *Dissertation for the Degree of PhD in Civil Engineering, The University of Texas at Austin, Austin, TX*
- Heath, J.P., J.A. Goodrich, and W.M. Grayman, 1993, "GIS for Water Quality Management and Modeling in the Ohio River Basin", Symposium on Geographic Information Systems and Water Resources, Mobile, AL, March 14-17, 1993, *American Water Resources Association Technical Publication TPS-93-1*, AWRA, Bethesda, MD
- Hetrick, W.A., P.M. Rich, and S.B. Weiss, 1993, "Modeling Insolation on Complex Surfaces", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Hinaman, K.C., 1993, "Use of a Geographic Information System to Assemble Input-Data Sets for a Finite-Difference Model of Groundwater Flow", Symposium on Geographic Information Systems and Water Resources, Mobile, AL, March 14-17, 1993, *American Water Resources Association Technical Publication TPS-93-1*, AWRA, Bethesda, MD
- Hlinka, K.J. and J.M. Shafer, 1989, "Statistical-Based Methodology for Prioritizing the Vulnerability of Community Ground-Water Supplies to Contamination", Water: Laws and Management, *American Water Resources Association, Bethesda, MD*, pp 8B-15 to 8B-23
- Haefner, R.J., 1992, "Use of a Geographic Information System to Evaluate Potential Sites for Public-Water-Supply Wells on Long Island, New York", *USGS Open-File Report 91-182*
- Halliday, S.L., M.L. Wolfe, 1991, "Assessing Ground Water Pollution Potential from Nitrogen Fertilizer Using a Geographic Information System", *Water Resources Bulletin*, 27 (2), pp 237-245

- Jankowski, P., and G. Haddock, 1993, "Integrated Nonpoint Source Pollution Modeling System", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Juracek, K.E., 1992., "Use of a Geographic Information System to Assist with Studies of the Availability and Use of Water in Kansas", *USGS Open-File Report 92-142*
- Kemp, K.K., 1993, "Environmental Modelling and GIS: Dealing with Spatial Continuity", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211, IAHS Press, Wallingford, Oxfordshire, UK*
- Kernighan, B., and D. Ritchie, 1978, The C Programming Language, *Prentice-Hall, Englewood Cliffs, NJ*
- Kernodle, J.M. and R.D. Philip, 1988, "Using a Geographic Information System to Develop a Ground-Water Flow Model", Aquifers of the Western Mountain Area, Papers presented at the 23rd Annual American Water Resources Association Conference and Symposium, November 1-6, 1987, Salt Lake City, UT; Regional Aquifer Systems of the United States, *AWRA Monograph Series No. 14., American Water Resources Association, Bethesda, Maryland, p 191-202*
- Kerzner, S., 1989, "EPA/Local Partnership at Work - The Creation of a Ground Water Protection Program", *Protecting Ground Water from the Bottom Up: Local Responses to Wellhead Protection*, October 2-3, 1989, Danvers, Massachusetts, Proceedings of the Conference, Underground Injection Practices Council, Oklahoma City, OK, pp 257-269
- Kilroy, K.C., 1989, "Water-Related Scientific Activities of the U.S. Geological Survey in Nevada, Fiscal Years 1985-89", *USGS Open-File Report 89-264*

- Kittleson, K.M., 1987, "Groundwater Problem in Michigan: An Overview", Rural Groundwater Contamination, *Lewis Publishers, Inc.*, Chelsea, MI, pp 69-84
- Kittleson, K.M. and R.L. Kruska, 1987, Spatial Distribution and Analysis of Groundwater Nitrate Contamination in Kalamazoo County, Michigan, *Michigan Institute of Water Research Technical Report No. G1232-04*
- Kuniansky, E.L., and R.A. Lowther, 1993, "Finite Element Mesh Generation Using Geographic Information Systems", Symposium on Geographic Information Systems and Water Resources, Mobile, AL, March 14-17, 1993, *American Water Resources Association Technical Publication TPS-93-1*, AWRA, Bethesda, MD
- Lam, D.C.L., and D.A. Swayne, 1991, "Integrating Database, Spreadsheet, Graphics, GIS, Statistics, Simulation Models, and Expert Systems: Experiences with the RAISON System on Microcomputers", *NATO ASI Series, vol. G26*, pp 429-459, Springer, Heidelberg
- Lallemand-Barres, A., and P. Peaudcerf, 1978, "Recherche des Relations Entre la Valeur de la Dispersivité Macroscopique d'un Milieu Aquifère, ses Autres Caractéristiques et les Conditions de Mesure", *Bulletin of the Bureau of Research in Geology and Mineralogy, Section 3, Series 2, 4*
- LaVenue, A.M., and B.S. RamaRao, 1992, A Modeling Approach To Address Spatial Variability within the Culebra Transmissivity Field, SAND92-7306, *Sandia National Laboratories*, Albuquerque, NM
- LaVenue, A.M., T.L. Cauffman, and J.F. Pickens, 1990a, Ground-Water Flow Modeling of the Culebra Dolomite Volume I: Model Calibration, SAND89-7068/1, *Sandia National Laboratories*, Albuquerque, NM
- LaVenue, A.M., T.L. Cauffman, and J.F. Pickens, 1990b, Ground-Water Flow Modeling of the Culebra Dolomite Volume II: Data Base, SAND89-7068/2, *Sandia National Laboratories*, Albuquerque, NM

- Lee, G.R. and T.J. Christoffel, 1990, "Clarke County, Virginia's Innovative Response to Groundwater Protection", *Journal of Soil and Water Conservation*, 45 (2), pp 257-259
- Lieste, R., K. Kovar, J.G.W. Verlouw, and J.B.S. Gan, 1993, "Development of the GIS-Based RIVM National Groundwater Model for the Netherlands (LGM)", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211*, IAHS Press, Wallingford, Oxfordshire, UK
- Mallory, M.J., 1990, "Computer Methods for Evaluating Wellhead Protection Areas", Proceedings, Twentieth Mississippi Water Resources Conference, April 10-11, 1990, Jackson, MS
- Marsily, G. de, 1986, Quantitative Hydrogeology, *Academic Press*, San Diego, CA
- Maidment, D.R., 1993a, "Environmental Modeling within GIS", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling*, Breckenridge, CO, September 26-30, 1993
- Maidment, D.R., 1993b, "GIS and Hydrologic Modeling", in Environmental Modeling with GIS, M.F. Goodchild, B.O. Parks, and L.T. Steyaert, eds., *Oxford University Press*
- McDonald, M.G. and Harbaugh, A.W., 1988, "A Modular Three-Dimensional Finite Difference Ground-Water Flow Model", *U.S. Geological Survey Techniques of Water-Resources Investigations Book 6*
- McKinney, D.C., and H.L. Tsai, 1993, "Solving Groundwater Problems Using Multigrid Methods in a Grid-Cell Based GIS", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling*, Breckenridge, CO, September 26-30, 1993

- Menon, S., P. Gao, and C. Zhan, 1991, "GRID: A Data Model and Functional Map Algebra for Raster Geo-Processing", *GIS/LIS '91 Proceedings*, October 28 - November 1, 1991, Atlanta, GA
- Menon, S., 1993, personal communication, *Environmental Systems Research Institutue, Inc.*, Redlands, CA
- Mercer, J.W., and R.K. Waddell, 1992, "Contaminant Transport in Groundwater", in *Handbook of Hydrology*, D.R. Maidment, ed., *McGraw-Hill*
- Mitchell, J.A., B.A. Engel, R. Srinivasan, and S.S.Y. Wang, 1993, "Validation of AGNPS for Small Watersheds Using an Integrated AGNPS/GIS System", *Symposium on Geographic Information Systems and Water Resources, Mobile, AL, March 14-17, 1993*, *American Water Resources Association Technical Publication TPS-93-1*, AWRA, Bethesda, MD
- Mladenhoff, D.J., G.E. Host, and J. Boeder, 1993, "LANDIS: A Spatial Model of Forest Landscape Disturbance, Succession, and Management", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Nachtnebel, H.P., J. Fürst, and H. Holzmann, 1993, "Application of Geographical Information Systems to Support Groundwater Modeling", *HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources*, *International Association of Hydrological Sciences Publication No. 211*, IAHS Press, Wallingford, Oxfordshire, UK
- Nelms, D.L., and D.L. Richardson, 1990, "Geohydrology and the Occurrence of Volatile Organic Compounds in Ground Water, Culpeper Basin of Prince William County, Virginia", *USGS Water-Resources Investigations Report 90-4032*

- Neuman, S.P., 1990, "Universal Scaling of Hydraulic Conductivities and Dispersivities in Geologic Media", *Water Resources Research*, 26 (8), pp 1749-1758
- Nielsen, G.A., J.M. Caprio, P.A. McDaniel, R.D. Snyder, and C. Montagne, 1990, "MAPS: A GIS for Land Resource Management in Montana", *Journal of Soil and Water Conservation*, 45 (4), pp 450-453
- Olsthoorn, T.N., P.T.W.J. Kamps, and W.J. Droesen, 1993, "Groundwater Modeling Using GIS at the Amsterdam Water Supply", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211*, IAHS Press, Wallingford, Oxfordshire, UK
- Orzol, L.L., and T.S. McGrath, 1992, "Modifications of the U.S. Geological Survey Modular, Finite-Difference, Groundwater Flow Model to Read and Write Geographic Information System Files", *USGS Open-File Report 92-50*
- Osborne, T.J., C. Mechenich, and B.H. Shaw, 1987, "New Wisconsin Ground Water Center Tackles Non-Point Source Contamination with Education", Proceedings of the NWWA FOCUS Conference on Midwestern Ground Water Issues, *National Water Well Association*, Dublin, OH, pp 37-58
- Ostensen, R.W., 1994, "Status of Proposed Tracer Tests in the Culebra Dolomite", in Summary of the WIPP Performance Assessment Geostatistics Expert Consultatnt Group Meeting of March 23-25, 1993, memo dated 7 Feb 1994, *Sandia National Laboratories*, Albuquerque, NM
- Peters, L.P., 1989, "Groundwater Applications of Geographic Information Systems--Toward New Ways of Integrating Data Using Digital Methods", Proceedings of the International Groundwater Symposium on Hydrogeology of Cold and Temperate Climates and Hydrogeology of Mineralized Zones, May 1-5, 1988, Halifax, Nova Scotia

- Petty, R.J. and M.A. Hallfrisch, 1989, "Application of the DRASTIC Mapping System for Evaluating Ground Water Pollution Potential in Ohio", Agrichemicals and Groundwater Protection: Resources and Strategies for State and Local Management, *Freshwater Foundation*, Navarre, MN, pp 181-200
- Pickens, J.F., and G.E. Grisak, 1981, "Scale Dependent Dispersion in a Stratified Granular Aquifer", *Water Resources Research*, 17 (4), pp 1191-1211
- Pollock, D.W., 1986, "Semianalytical Computation of Pathlines for Finite Difference Models", *Ground Water*, 26 (6), pp 743-750
- Press, W.H., B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, 1988, Numerical Recipes in C: The Art of Scientific Computing, *Cambridge University Press*, Cambridge
- Raper, J.F., 1989, Three Dimensional Applications in Geographic Information Systems, *Taylor and Francis, Ltd.*, London
- Reitsma, R.F., 1993, "Bootstrapping River Basin Models Through Object Orientation and GIS Topology", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling*, Breckenridge, CO, September 26-30, 1993
- Reyes, C., F. Zamora, and G. Legoretta, 1993, "SIGMA: A Geographic Information System for Atmospheric Modeling of the Mexico City Metropolitan Area", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling*, Breckenridge, CO, September 26-30, 1993

- Richards, C.J., Roaza, H., and R.M. Roaza, 1993, "Integrating Geographic Information Systems in Well Field Design and Aquifer Impact Analysis Using the MODFLOW Finite Difference Modeling Technique", Symposium on Geographic Information Systems and Water Resources, Mobile, AL, March 14-17, 1993, *American Water Resources Association Technical Publication TPS-93-1*, AWRA, Bethesda, MD
- Richards, D.R., N.L. Jones, and H.C. Lin, 1993, "Graphical Innovations in Surface Water Flow Analysis", in Environmental Modeling with GIS, M.F. Goodchild, B.O. Parks, and L.T. Steyaert, eds., *Oxford University Press*
- Roaza, H., R.M. Roaza, and J.R. Wagner, 1993, "Integrating Geographic Information Systems in the Analysis of Ground-Water Flow, Contaminant Transport, and Saltwater Intrusion Using the SWICHA Finite Element Technique", Symposium on Geographic Information Systems and Water Resources, Mobile, AL, March 14-17, 1993, *American Water Resources Association Technical Publication TPS-93-1*, AWRA, Bethesda, MD
- Robinove, C.J., 1986, Principles of Logic and the Use of Digital Geographical Information Systems, *U.S. Geological Survey Circular 977*
- Saghafian, B., 1993, "Implementation of a Distributed Hydrologic Model Within GRASS", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Schimoller, B.J., 1992, "The Development of Air Pollution Dispersion Models for Integration with a Geographic Information System", *Departmental Report in Civil Engineering, The University of Texas at Austin, Austin, TX*

- Scott, J.C., 1990, "Computer Software for Converting Ground-Water and Water-Quality Data from the National Water Information System for Use in a Geographic Information System", *USGS Water-Resources Investigations Report 90-4200*
- Sen, M.A., and B. Kelk, 1993, "Evaluation of a New Integrated Software Platform for Data Storage and Analysis Pertinent to Hazardous Waste Site Investigations", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211*, IAHS Press, Wallingford, Oxfordshire, UK
- Sokol, G., Ch. Leibundgut, K.P. Schulz, and W. Weinzierl, 1993, "Mapping Procedures for Assessing Groundwater Vulnerability to Nitrate and Pesticides", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211*, IAHS Press, Wallingford, Oxfordshire, UK
- Srinivasan, R., J. Arnold, W. Rosenthal, and R.S. Muttiah, 1993, "Hydrologic Modeling of Texas Gulf Basin Using GIS", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Stansbury, J., W. Woldt, I. Bogardi, and A. Bleed, 1991, "Decision Support System for Water Transfer Evaluation", *Water Resources Research*, 27 (4), pp 443-451
- Steppacher, L., 1988, "Ground Water Risk Management on Cape Cod: The Applicability of Geographic Information System Technology", Proceedings of the FOCUS Conference on Eastern Regional Ground Water Issues, September 27-29, 1988, Stamford, Connecticut, pp 427-461, *National Water Well Association*, Dublin, OH

- Steyaert, L.T., 1993, "A Perspective on the State of Environmental Simulation Modeling", in Environmental Modeling with GIS, M.F. Goodchild, B.O. Parks, and L.T. Steyaert, eds., *Oxford University Press*
- Swain, L.A., E.F. Hollyday, C.C. Daniel, and O.S. Zapecza, 1991, "Plan of Study for the Regional Aquifer-System Analysis of the Appalachian Valley and Ridge, Piedmont, and Blue Ridge Physiographic Provinces of the Eastern and Southeastern United States, with a Description of Study-Area Geology and Hydrogeology", *USGS Water-Resources Investigations Report 91-4066*
- Tan, Y.R. and S.F. Shih, 1991, "Geographic Information System for Differentiating Unused Wells", Soil and Crop Science Society of Florida Proceedings, Vol. 50, pp 110-116
- Tauxe, J.D., D.R. Maidment, and R.J. Charbeneau, 1992, "Contaminant Transport Modeling Using New Grid Operators" in Proceedings of the Twelfth Annual ESRI User Conference, vol. III, pp 57-62, *Environmental Systems Research Institute*, Redlands, CA
- Trent, V.P., 1993, "DRASTIC Mapping to Determine the Vulnerability of Ground Water to Pollution", Symposium on Geographic Information Systems and Water Resources, Mobile, AL, March 14-17, 1993, *American Water Resources Association Technical Publication TPS-93-1*, AWRA, Bethesda, MD
- Tomlin, D.C., 1990, Geographic Information Systems and Cartographic Modeling, *Prentice-Hall*, Englewood Cliffs, NJ
- Totman, D., 1989, "Mapping Out a Plan to Protect Arizona's Groundwater", *Water Engineering and Management*, 137 (11), pp 24-26
- U.S. Army Construction Engineering Research Laboratory (CERL), 1991, Geographic Resources Analysis Support System (GRASS), Champaign, IL

- U.S.E.P.A. (Environmental Protection Agency), 1979, Industrial Source Complex (ISC) Model User's Guide, EPA Report No. EPA-450/4-79-030/311 U.S. *Environmental Protection Agency*, Research Triangle Park, NC
- U.S.E.P.A., 1985, "Environmental Standards for the Management and Disposal of Spent nuclear Fuel, High-Level and Transuranic Radioactive Wastes; Final Rule, 40 CFR Part 191", *Federal Register*, 50 (182), 38066-38089
- U.S.E.P.A., 1987, The Enhanced Stream Water Quality Models QUAL2E and QUAL2E-UNCAS, *Environmental Research Laboratory, U.S. Environmental Protection Agency*, Athens, GA
- van der Heijde, P.K.M., 1992, "Developments in Computer Technology Enhancing the Application of Groundwater Models", Environmental Modelling, *Elsevier Applied Science*, New York, NY, pp 23-33
- Van Deursen, W.P.A., and J.C.J. Kwadijk, 1993, "RHINEFLOW: An Integrated GIS Water Balance Model for the River Rhine", HydroGIS 93: Application of Geographic Information Systems in Hydrology and Water Resources, *International Association of Hydrological Sciences Publication No. 211*, IAHS Press, Wallingford, Oxfordshire, UK
- Van Metre, P.C., 1990, "Structure and Application of an Interface Program Between a Geographic-Information System and a Groundwater Flow Model", *USGS Open-File Report 90-165*
- Ventura, S.J., B.J. Irvin, B.K. Slater, K. McSweeney, 1993, "Data Structures for Representation of Soil Stratigraphy", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling*, Breckenridge, CO, September 26-30, 1993

- Vieux, B.E., N.S. Farajalla, and N. Gaur, 1993, "Integrated GIS and Distributed Stormwater Runoff Modeling", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling, Breckenridge, CO, September 26-30, 1993*
- Walter, G.B., 1983, "Convergent Flow Tracer Test at H-6: Waste Isolation Pilot Plant (WIPP), Southeast New Mexico", *Hydro Geochem, Inc., Tuscon, AZ*
- Warner, K.L., J.D. Earle, and M.G. Sherrill, 1991, "Hydrogeologic Information in the Great Lakes Basin, United States, and Application of a Geographic Information System to Public-Supply Wells and Hazardous-Waste Sites", *USGS Open-File Report 91-220*
- Watkins, D., 1993, "Using Geographic Information Systems in Groundwater Modeling", *Departmental Report in Civil Engineering, The University of Texas at Austin, Austin, TX*
- Weaver, J.W., R.J. Charbeneau, J.D. Tauxe, B.K. Lien, and J.B. Provost, 1994, The Hydrocarbon Spill Screening Model (HSSM) Volume 1: User's Guide, EPA/600/R-94/039a, *Robert S. Kerr Environmental Research Laboratory, U.S. Environmental Protection Agency, Ada, OK*
- Weghorst, P.A., J. Cunningham, and B. Mortazavi, 1991, "Geographic Information Systems in Water Resources Management", *Hydraulic Engineering, Proceedings of the 1991 National Conference, American Society of Civil Engineers, New York, NY, pp 888-893*
- Wendland, F., H. Albert, M. Bach, and R. Schmidt (editors), 1993, Atlas zum Nitratstrom in der Bundesrepublik Deutschland, Springer-Verlag, Berlin

- Whittemore, D.O., J.W. Merchant, J. Whistler, C.E. McElwee, and J.J. Woods, 1987, "Ground Water Protection Planning Using the ERDAS Geographic Information System: Automation of DRASTIC and Time-Related Capture Zones", Proceedings of the NWWA FOCUS Conference on Midwestern Ground Water Issues, *National Water Well Association*, Dublin, OH, pp 359-374
- Wilson, J.L. and P.J. Miller, 1978, "Two-Dimensional Plume in Uniform Ground-Water Flow", *Journal of the Hydraulics Division*, ASCE, vol. 104, no. 4, pp 503-514
- Wilson, J.P., R.D. Snyder, C.M. Ryan, W.P. Inskeep, J.S. Jacobsen, and P.R. Rubright, 1992, "Coupling Geographic Information Systems and Models for County Scale Groundwater Pollution Assessments", in Proceedings of the Twelfth Annual ESRI User Conference, vol. III, pp 387-398, *Environmental Systems Research Institute*, Redlands, CA
- Xu, M., and Y. Eckstein, 1993, "Numerical Analysis of the Relationship Between Dispersivity and Field Scale", *1993 Fall Meeting of the American Geophysical Union*, December 6-10, 1993, San Francisco, CA
- Yeh, T-S., and B. de Cambray, 1993, "Time as a Geometric Dimension for Modeling the Evolution of Entities: A 3D Approach", *Second International Conference/Workshop on Integrating Geographic Information Systems and Environmental Modeling*, Breckenridge, CO, September 26-30, 1993
- Yoon, J., G. Padmanabhan, and L.H. Woodbury, 1993, "Linking Agricultural Nonpoint Source Pollution Model (AGNPS) to a Geographic Information System", Symposium on Geographic Information Systems and Water Resources, Mobile, AL, March 14-17, 1993, *American Water Resources Association Technical Publication TPS-93-1*, AWRA, Bethesda, MD

Zhang, Z., Y. Xue, and J. Wu, 1994, "A Cubic-Spline Technique to Calculate Nodal Darcian Velocities in Aquifers", *Water Resources Research*, 30 (4), pp 975-981

APPENDIX

The Appendix contains source code listings for the programs described in this document as well as related computer files. These are available in electronic form by request.