

Utah State University

DigitalCommons@USU

---

All Graduate Theses and Dissertations

Graduate Studies

---

5-1978

## A Mathematical Model of the Dispersion of a Concentrated Substance for Use in the Great Salt Lake's South Arm

Anthony O. Rughellis  
*Utah State University*

Follow this and additional works at: <https://digitalcommons.usu.edu/etd>

 Part of the [Civil Engineering Commons](#)

---

### Recommended Citation

Rughellis, Anthony O., "A Mathematical Model of the Dispersion of a Concentrated Substance for Use in the Great Salt Lake's South Arm" (1978). *All Graduate Theses and Dissertations*. 3271.

<https://digitalcommons.usu.edu/etd/3271>

This Thesis is brought to you for free and open access by the Graduate Studies at DigitalCommons@USU. It has been accepted for inclusion in All Graduate Theses and Dissertations by an authorized administrator of DigitalCommons@USU. For more information, please contact [digitalcommons@usu.edu](mailto:digitalcommons@usu.edu).



A MATHEMATICAL MODEL OF THE DISPERSION OF A CONCENTRATED SUBSTANCE

FOR USE IN THE GREAT SALT LAKE'S SOUTH ARM

by

Anthony O. Righellis

A thesis submitted in partial fulfillment  
of the requirements for the degree

of

MASTER OF SCIENCE

in

Engineering

UTAH STATE UNIVERSITY  
Logan, Utah

1978

378.2  
R449m  
C-2

ACKNOWLEDGMENTS

My gratitude to Dr. Gary Watters, my major advisor, for his suggestions and guidance during the development of the model and the writing of this paper. I would also like to thank my other committee members, Dr. Roland Jeppson and Dr. Calvin Clyde.

My special thanks goes to Charles Kincaid for his numerous hours of guidance in learning and applying the finite element method. A note of appreciation goes to Dr. David Watkins for his insights into the finite element method and the diffusion process.

Anthony O. Righellis

## TABLE OF CONTENTS

|   | Page |
|---|------|
| ACKNOWLEDGMENTS . . . . .   | ii   |
| LIST OF TABLES . . . . .  | iv   |
| LIST OF FIGURES . . . . .   | v    |
| ABSTRACT . . . . .  | vi   |
| Chapter   |      |
| I INTRODUCTION . . . . .  | 1    |
| II LITERATURE REVIEW . . . . .  | 6    |
| Dispersion and Diffusion . . . . .  | 6    |
| Diffusion Coefficient . . . . .   | 6    |
| Numerical Method. . . . .   | 8    |
| III THEORETICAL DEVELOPMENT . . . . .   | 10   |
| Diffusion . . . . .   | 10   |
| Use of Convection-Dispersion Equation. . . . .                                    | 13   |
| Assumptions Used in Model Application. . . . .                                    | 14   |
| Finite Element Representation of the Convection-Dispersion<br>Equation . . . . .  | 15   |
| Representation of Terms in the Convection-Dispersion Equation. . . . .            | 16   |
| IV MODELING TECHNIQUE AND PARAMETER DISCUSSION . . . . .                          | 22   |
| Computer Model's Parameters . . . . .   | 22   |
| Example Convection-Dispersion Problems . . . . .                                  | 27   |
| V PROBLEMS ASSOCIATED WITH THE APPLICATION OF THE MODEL . . . . .                 | 41   |
| Velocity Field . . . . .  | 41   |
| Dispersive Transport at the Boundaries . . . . .                                  | 41   |
| Introduction of Substance. . . . .  | 42   |
| Discretization of the Domain . . . . .  | 43   |
| VI CONCLUSIONS AND RECOMMENDATIONS . . . . .                                      | 45   |
| REFERENCES . . . . .  | 48   |
| APPENDICES . . . . .  | 49   |
| Appendix A: Units of Variables in the Convection-Dispersion<br>Equation . . . . . | 50   |
| Appendix B: User's Guide to Convection-Dispersion Model . . . . .                 | 52   |
| Appendix C: FORTRAN IV - Listing of Computer Model . . . . .                      | 61   |



## LIST OF TABLES

| Table |  | Page |
|-------|--|------|
| 1.    | Description of elements. . . . .       | 20   |
| 2.    | Basic units. . . . .                   | 51   |
| 3.    | Variables used in model. . . . .       | 51   |
| 4.    | Types of modeling techniques . . . . . | 54   |

## LIST OF FIGURES

| Figure   | Page |
|--|------|
| 1. Map of Great Salt Lake . . . . .  | 2    |
| 2. Bi-directional flow through causeway . . . . .  | 3    |
| 3. Differential control volume. . . . .  | 11   |
| 4. Domain for examples. . . . .  | 29   |
| 5a. Linear discretized domain for steady-state examples. .   | 30   |
| 5b. Quadratic discretized domain for steady-state examples   | 30   |
| 6. Solution of example 1. . . . .  | 31   |
| 7a. Linear Galerkin solution of example 2 along reflective<br>boundary . . . . .   | 32   |
| 7b. Quadratic Galerkin solution of example 2 along reflec-<br>tive boundary. . . . .   | 32   |
| 7c. Linear "upwinding" solution of example 2 along reflec-<br>tive boundary. . . . .   | 32   |
| 8. Solution of example 2. . . . .  | 34   |
| 9. Solution of example 3. . . . .  | 35   |
| 10. Solution of example 4. . . . .   | 37   |
| 11. Linear discretized domain for unsteady-state examples.   | 38   |
| 12. Concentration at a position 30 ft downstream of the<br>influent boundary along the reflective boundary for<br>example 5. . . . . | 38   |
| 13. Concentration profile along the reflective boundary<br>for example 6. . . . .  | 40   |

## ABSTRACT

A Mathematical Model of the Dispersion of a Concentrated Substance  
For Use in the Great Salt Lake's South Arm

by

Anthony O. Righellis, Master of Science

Utah State University, 1978

Major Professor: Gary Z. Watters  
Department: Civil Engineering

The ability to predict the dispersion of substances in the Great Salt Lake is a requisite towards making responsible management decisions relating to uses of the lake. The lake is a complex terminal body of water and will require a fairly sophisticated mathematical model to properly simulate the dispersion process in the lake. This finite element convection-dispersion model is a first step towards developing a comprehensive model.

The model provides a finite element solution to the two-dimensional convection-dispersion equation and is capable of simulating steady or unsteady-state situations. It utilizes a known velocity field, dispersion coefficients, an introduced substance concentration, substance decay rates, and the region geometry to produce a solution to a given convection-dispersion problem.

At the present time, a quantitative verification of the model has not been done, but qualitative use of the model indicates that it yields reasonable solutions satisfying continuity to convection-dispersion problems. Problems tested utilize a uniform flow field and various methods of introducing a substance, such as internal injections, established

concentration gradients, and diffusers. This model affords the options in the approximating techniques of linear or quadratic interpolation functions, the Galerkin or "upwinding" methods of weighted residuals, and a linearly or quadratically varying velocity field. The model must use a continuous flow field to produce a credible solution. The model does need improvement in its ability to conserve mass in unsteady-state problems when introducing a substance into the modeled region and allowing dispersive transport at the boundaries. Proper nodal spacing (mesh size) is also important because a relatively coarse mesh size can result in poor approximations in some areas of the region modeled.

CHAPTER I  
INTRODUCTION

The Great Salt Lake is a terminal lake about 75 miles in length and 30 miles in width. In 1959, the Southern Pacific Railroad Company causeway was constructed, dividing the lake into two distinctive parts or arms (Figure 1). The north arm consists essentially of unstratified salt-saturated water (brine). The south arm is composed of two layers of different densities. The upper layer is unsaturated salt water (dilute brine) and the more dense lower layer is brine. This phenomena occurs because nearly all of the fresh water surface inflow is into the south arm. As a result, the south arm's surface is higher in elevation than the north arm's surface causing the diluted brine to flow into the north arm, mixing and becoming salt-saturated as evaporation occurs. This north arm brine is now more dense causing a backflow into the south arm at the lake bottom. Figure 2 illustrates this bi-directional flow.

Before the construction of the railroad causeway the lake was much more homogeneous in nature. The causeway construction has caused a series of problems between the salt companies bordering the south and north shores and the Southern Pacific Railroad Company. The south shore salt companies claim to be at a disadvantage in their operations because their source of salts is from this newly diluted south arm. These companies claim the causeway created the dilution. However, firm knowledge is lacking concerning the actual mechanics of the lake's circulation and its effect on salinity distribution.

This study is part of a coordinated effort to develop a knowledge of the hydraulic characteristics of Great Salt Lake to assist in solving

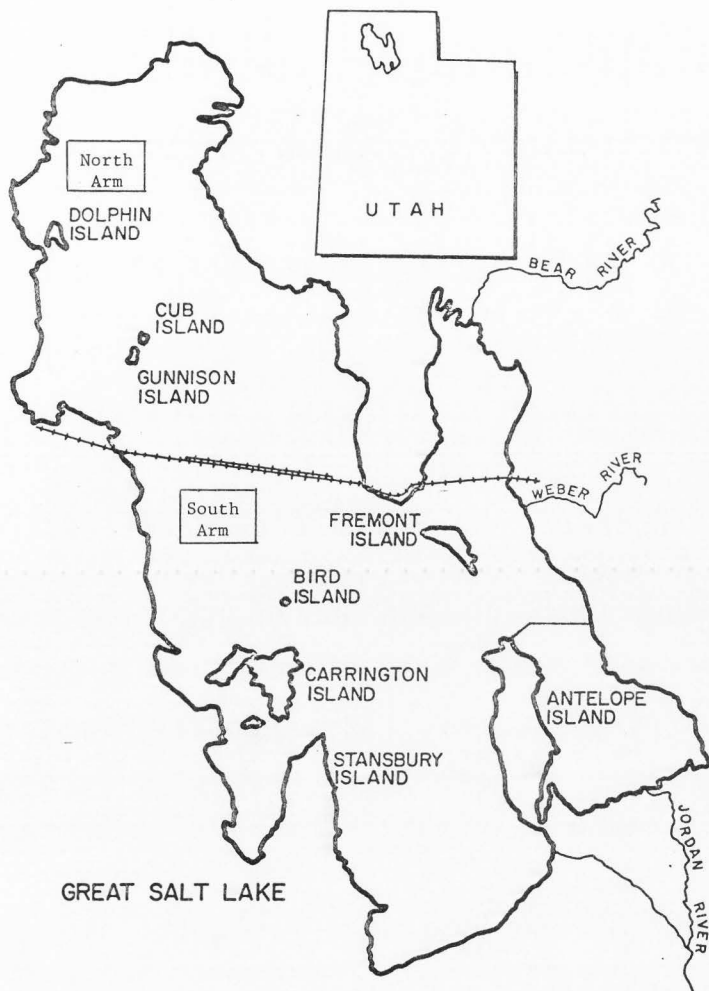


Figure 1. Map of Great Salt Lake.

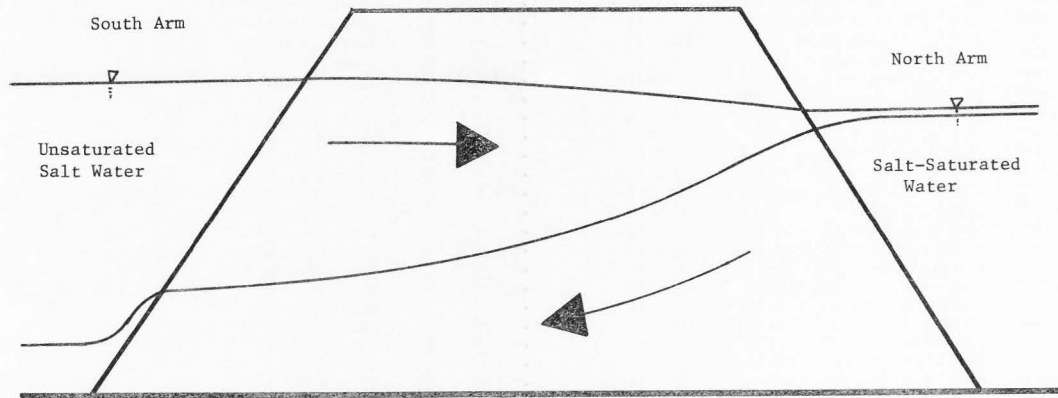


Figure 2. Bi-directional flow through causeway.

management problems of this nature. The work will be directed to developing a model which can be used to simulate the dispersion of a substance in the south arm. In addition to the question as to how salinity is transported and distributed throughout the south arm, the convection and dispersion of pollutants is of interest. To address these problems, it is necessary to know the mixing (dispersion) characteristics of Great Salt Lake's south arm as well as the circulation currents. By making some simplifying assumptions and using a differential mass balance a convection-dispersion equation can be developed to represent dispersion processes.

This research on convection-dispersion is being carried out in conjunction with three other works. A hydrodynamic lake model is being developed to yield the needed velocity-field data for input into the governing convection-dispersion equation. Studies concerned with the flow through the causeway culverts and the porous embankments will provide needed information for the boundary conditions necessary to complete the hydrodynamic lake model.

The objective of this study is to develop a mathematical model which can be used to simulate the convection and dispersion of a substance in the dilute brine of the south arm's upper layer. This study is considered as a first step towards developing a more comprehensive two-layer model. It is possible that the main function of the lower layer within the lake may be to act as a reservoir for passing salinity from the north arm to the surface layer in the south arm. If this is the case, then an upper layer model could be a low-cost preliminary solution to the Great Salt Lake circulation problem.

Specifically, the objectives of this work are:



1. To develop a steady-state numerical model of the dispersion of a concentrated substance.
2. To develop an unsteady-state numerical model of the dispersion of a concentration substance. This model will include a source-sink term and a decay term.
3. Present examples illustrating the use of the model.
4. Determine the limitations of the model.

The dispersion of a concentrated substance will be modeled utilizing the finite element method in numerically solving a convection-dispersion equation. Two different methods of weighted residuals will be applied to the steady state solution. They are the Galerkin method and the "upwinding" method. The unsteady state will use the same methods as the steady state except for the addition of the Crank-Nicolson technique for time stepping. The boundary conditions and source-sink terms will be used to simulate effects of external influences.

## CHAPTER II

### LITERATURE REVIEW

There are three distinct areas of concern that require researching. The first is the theory of diffusion and the governing equations. Second is the determination of practical values for the diffusion coefficient when it becomes clear that the governing equations require the use of such coefficients. The third area is the use of the finite element method in developing a numerical solution to the governing equations.

#### Dispersion and Diffusion

The current views on the concept and theory of dispersion and diffusion are expressed by Holley (6). He explains the different mechanisms that cause a substance to be dispersed and presents the governing equation in various forms relating the use of specific coefficients to each method of diffusion. Ippen (9) shows the mechanics of applying a differential mass balance to obtain a convection-dispersion equation.

Many authors, such as Harleman (4) and Narayanan and Shankar (10), deal with diffusion in estuaries, lakes or bays that are influenced by tidal action. These works are closely related to this study, but because of the influence of tidal actions, their results are of limited use. The literature has been searched to find a method for calculating a value for the diffusion coefficient.

#### Diffusion Coefficient

The search for methods to determine the diffusion coefficient was somewhat unproductive. Adequate methods or sources of information are

lacking. Various authors present different equations for determining the values of the diffusion coefficient but none seem to be properly applicable to this study because of tidal factors, irrelevant roughness coefficients, and other nonapplicable parameters. For example, Narayanan and Shankar (10) used Taylor's equation as given by Harleman (4). It shows that the diffusion coefficient in the x and y directions are calculated using the equations

$$D_x = 77n|u|h^{5/6}$$

$$D_y = 77n|v|h^{5/6}$$

where n is Manning's roughness coefficient and u and v are the velocities averaged over a tidal cycle in the x and y directions, respectively. The value of h represents the water depth. This approach is not quite applicable, except as a check on the order of magnitude, because it deals with tidal oscillations, which do not occur in the Great Salt Lake. These types of oscillations can greatly affect the diffusion coefficients and do not bear directly on this study.

George (3) presents many methods in his dissertation including Richardson's, "4/3 Law", Taylor's equation utilizing Darcy-Weisbach's friction factor, and other works. They all fall short in practical value in relation to this work because of their dependence on factors that can not be readily evaluated, such as assorted unknown coefficients and vague length parameters.

Huyakorn (8) shows by an example that

$$D = \frac{Re \cdot \nu}{Pe}$$

$$Pe = Re \cdot Pr$$

where  $Pr$  is the Prandtl number,  $Pe$  is the Peclet number,  $Re$  is the Reynolds number,  $\nu$  is the kinematic viscosity of the fluid, and  $D$  is the diffusion coefficient. This might be of some use, but in a complex flow field the Reynolds and Peclet numbers can be difficult to evaluate.

The diffusion coefficient is something that must be evaluated in order to present an accurate solution to a modeling problem. Narayanan and Shanker (10) show that for short-term transport, the process is convection-dominated and an accurate diffusion coefficient is not necessary. At present, it seems that the only way to determine an accurate value for the diffusion coefficient is to monitor the dispersion of a substance in the prototype then model the event with varying values for the diffusion coefficient until agreement between the prototype and the model occurs.

#### Numerical Method

The finite element method is the numerical technique used to develop a solution to the governing equation within a region. It is chosen over the finite difference method because of its flexibility in node placement, simple boundary conditions, and capabilities in approximating irregular physical boundaries (Zienkiewicz (11), Huebner (7)). In applying the finite element method, the Galerkin method of weighted residuals is employed.

The Galerkin method of weighted residuals is chosen over the variational method because of its clarity. The variational method cannot be applied to all situations. In some cases it is not valid. Avoiding this possibility is another reason for choosing the method of weighted residuals. Zienkiewicz (11) and Huebner (7) explain the basics of the Galerkin method

of weighted residuals as applied to the finite element method. They show the mechanics of the method from the simple linear elements to the complex quadratic isoparametric elements and more. A variation of the method known as "upwinding" is a recent development in the finite element method.

Huyakorn (8) and Christie, et al. (2) show "upwinding as applied to a dispersion problem. It is an improvement over the standard Galerkin procedure because it puts more weighting or emphasis on certain portions of an element rather than weighting according to the shape function. This tends to remove oscillatory solutions in a convection-dominated domain. These methods are applied to both the steady and unsteady state solutions.

The time-varying solution is developed using the Crank-Nicolson technique as found in Carnahan-Luther-Wilkes (1) or others. The unsteady state solution is necessary for situations where a substance is only introduced for a finite period of time. This is a realistic case for a pollutant. Salinity modeling probably will not require an unsteady state solution since the source will not vary with time.

CHAPTER III  
THEORETICAL DEVELOPMENT

Diffusion

Fick's law

A substance will diffuse proportionally to its concentration gradient. This process is known as Fick's law of diffusion. This first law states,

$$\dot{m} = - D_m \frac{\partial c}{\partial s}$$

where

- $\dot{m}$  = the rate of transport of substance
- $c$  = concentration of diffusive substance
- $D_m$  = molecular diffusion coefficient
- $s$  = coordinate normal to the unit area through which it passes.

This phenomena is caused by molecular diffusion. If there is no turbulence this basic law applied to a mass balance within a differential control volume would be the governing equation for the process of diffusion.

Turbulent diffusion

When a flow field (see Figure 3) is turbulent, the substance mixes at a much greater rate than under Fickian diffusion. This is because of the fluid's variance from its average motion. Applying a mass balance to the differential control volume in Figure 3 and then time-averaging the terms and utilizing an analogy to Fick's law yields the following convection-diffusion equation,

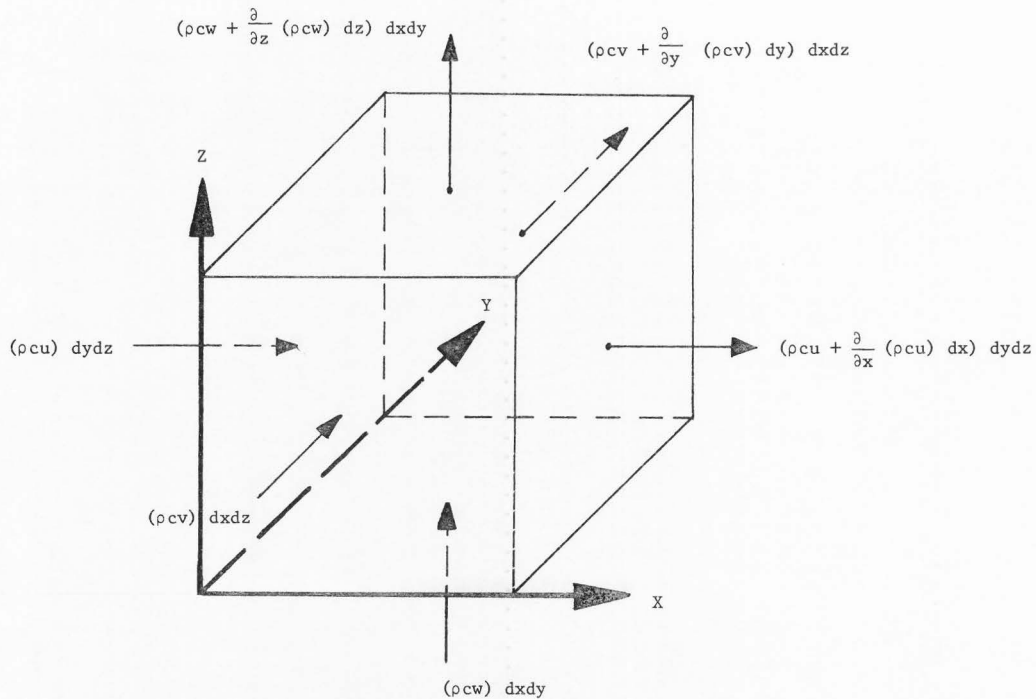


Figure 3. Differential control volume.

$$\frac{\partial \bar{c}}{\partial t} + \bar{u} \frac{\partial \bar{c}}{\partial x} + \bar{v} \frac{\partial \bar{c}}{\partial y} + \bar{w} \frac{\partial \bar{c}}{\partial z} = (D_m + e_x) \frac{\partial^2 \bar{c}}{\partial x^2} + (D_m + e_y) \frac{\partial^2 \bar{c}}{\partial y^2} + (D_m + e_z) \frac{\partial^2 \bar{c}}{\partial z^2} \quad (1)$$

where,

- $\bar{c}$  = time averaged concentration of diffusive substance
- $x, y, z$  = orthogonal coordinates (Cartesian)
- $\bar{u}, \bar{v}, \bar{w}$  = time averaged velocities in  $x, y,$  and  $z$  directions, respectively
- $D_m$  = molecular coefficient
- $e_x, e_y, e_z$  = turbulent diffusion coefficients in  $x, y,$  and  $z$  directions, respectively.

The turbulent diffusion coefficients are the result of turbulent mixing and designate how the turbulent flow affects the diffusive process.

It can readily be seen how the combined diffusion coefficients ( $D_m + e_x, D_m + e_y, D_m + e_z$ ) for a turbulent flow field must be greater than the diffusion coefficient in Fickian diffusion ( $D_m$ ). This means that turbulence increases the rate at which a substance will diffuse. The turbulent diffusion coefficient is much larger than the Fickian diffusion coefficient, so generally the Fickian coefficient is neglected in turbulent flow.

### Dispersion

In integrating Equation (1) in the  $z$ -direction to form a two-dimensional equation, the effect of the vertically varying velocity on convection must be accounted for. The velocity has been vertically



averaged instead of represented in its actual varying profile. This profile causes an additional lateral spreading when viewed in the x-y plane and the phenomenon is referred to as dispersion. To compensate for this event, the diffusion coefficients must be increased to accurately represent what occurs. Rewriting Equation (1) in two-dimensions, dropping the bar notation, and adding a source-sink term and a decay term while changing the diffusion coefficient to a dispersion coefficient gives the following convection-dispersion equation,

$$\frac{\partial c}{\partial t} + u \frac{\partial c}{\partial x} + v \frac{\partial c}{\partial y} - \frac{\partial}{\partial x} \left( D_x \frac{\partial c}{\partial x} \right) - \frac{\partial}{\partial y} \left( D_y \frac{\partial c}{\partial y} \right) + \alpha + \beta = 0 \quad (2)$$

where,

- c = concentration of dispersive substance
- x,y = orthogonal coordinates (Cartesian)
- $D_x, D_y$  = dispersion coefficients in x and y directions, respectively
- u,v = vertically averaged component velocities in x and y directions, respectively
- $\alpha$  = source-sink term
- $\beta$  = decay term (function of c)
- t = time

The dispersion coefficient (also commonly referred to as "diffusion" coefficient) is a combination of the molecular, turbulent, dispersive effects. The above equation is what has been referred to in the previous chapters as the governing equation.

#### Use of Convection-Dispersion Equation

Equation (2) is used to model the dispersion of a concentrated substance in a fluid. The only unknown variable is the concentration. All

of the other parameters are considered known throughout the entire domain. The physical geometry supplies all of the spatial information. The substance has its own decay rate. The source-sink term simulates physical injections of substance. The velocities are supplied by either field or laboratory measurements or the numerical steady state hydrodynamic model referred to in Chapter I.

#### Assumptions Used in Model Application

Because this model is being developed to apply to the upper layer of Great Salt Lake's south arm, any simplifying assumptions made should not greatly detract from the model's ability to simulate the prototype. In accordance with these objectives, three basic assumptions made about the south arm's upper layer are incorporated in the model. They are as follows:

1. The south arm's upper layer has no variation in depth and as a two-dimensional process is assumed. Since the layer's depth is very small compared to its areal extent, the use of a two-dimensional model seems justified.
2. The layer is of constant density and the dispersive substance does not change the layer's density.
3. The dispersive substance is well-mixed in the vertical direction. The wind and wave action on the shallow layer cause the mixing.

These assumptions allow the use of Equation (2) in developing a solution in the upper layer of Great Salt Lake's south arm.

Finite Element Representation of the  
Convection-Dispersion Equation

The finite element method is chosen over other numerical methods because of its variable mesh size, natural boundary conditions, and capabilities in approximating irregular physical boundaries. The method of weighted residuals as applied to the finite element method is a way of determining an approximate solution to a discretized continuum problem.

Method of weighted residuals

The method of weighted residuals is a technique for obtaining an approximate solution to a differential equation by weighting the approximate solution and minimizing a residual error between the true and the approximate solution. This approach in conjunction with the finite element method uses a discrete element as its foundation. Setting the sum of the 'weighted residuals' over all of the elements to zero one finds a solution for a set of nodal values over the domain. The nodal values, when substituted into the interpolation functions, yield an approximate solution which best satisfies the problem within the limits of the discretization and approximation schemes chosen.

Galerkin and "upwinding" methods

The Galerkin method is a common technique used to weight the governing equation over an element and Huebner (7) and Zienkiewicz (11) give a good account of the method for various types of elements. In the Galerkin method, the weighting functions are identical to the interpolation (or shape) functions. This is not the case in the "upwinding" method.

The "upwinding" technique, as shown by Christie et al. (2), Heinrich et al. (5), and Huyakorn (8), for convection-dispersion equations uses a skewed weighting scheme. The interpolation functions are of the standard form as used in the Galerkin method. The difference is in the equation's weighting functions which are skewed with more emphasis (weighting) on the portion of the element that is upwind (upstream). For a convection-dominated problem this technique finds its justification in removing the severe longitudinal oscillations which can occur with the Galerkin method. The result is a solution with a better degree of accuracy, fewer numerical oscillations, without too many additional computations.

#### Representation of Terms in the Convection-Dispersion Equation

Not all of the terms of the governing equation (Equation (2)) are developed in quite the same manner. There are five different groups of terms. They are as follows:

1.  $u \frac{\partial c}{\partial x}, v \frac{\partial c}{\partial y}$  (convective terms)
2.  $-\frac{\partial}{\partial x} \left( D_x \frac{\partial c}{\partial x} \right), -\frac{\partial}{\partial y} \left( D_y \frac{\partial c}{\partial y} \right)$  (dispersive terms)
3.  $\alpha$  (source-sink term)
4.  $\beta$  (decay term)
5.  $\frac{\partial c}{\partial t}$  (time rate of change term)

The basic development of each group will be shown as it is used in the model. The matrix  $\{W\}$  represents the weighting functions. The matrices  $\{N_v\}$  and  $\{N_c\}$  are the element's velocity and concentration shape functions, respectively. Standard notation as found in Zienkiewicz (11) and Huebner (7) will be used.

Convective terms. All of the variables in the convective terms vary from node to node, thus,

$$u^{(e)}(x,y) = [Nv] \{u\}^{(e)} = [Nv_1, Nv_2, \dots, Nv_n] \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ u_n \end{bmatrix}$$

$$\frac{\partial c}{\partial x}^{(e)}(x,y) = [Nc_x] \{c\}^{(e)}$$

The superscript (e) indicates the values of the variable for a particular element but will be dropped in future equations as a convenience in writing them. The subscripts on the expanded notation indicate the local node numbers of the element. Combining the variables and applying the weighting functions to create the inner product yields,

$$\text{Weighted Residual of } \left[ u \frac{\partial c}{\partial y} \right]^{(e)} = \int_{D(e)} \{W\} [Nv] \{u\} [Nc_x] dx dy \{c\} \quad (3)$$

Dispersive terms. The dispersion coefficients are assumed constant over an element and require no interpolation functions. The dispersion terms are integrated using Green's Theorem to create boundary terms within the governing equation and reduce the order of the differential equation. After this process, the inner product is taken with the weighting functions and the weighted residual of these terms become:

$$\text{Weighted Residual of } \left[ -\frac{\partial}{\partial x} \left( D_x \frac{\partial c}{\partial x} \right) \right]^{(e)} = \int_{D(e)} D_x \{W_x\} [Nc_x] dx dy \{c\} - \int_{S(e)} D_x [Nc_x] \{c\} \{W\} dy \quad (4)$$

Further mention of the line integral in the above equation will be made later when the boundary conditions are discussed.

Source-sink term. The source-sink term is handled in the same manner as the dispersion coefficients as it is constant over an element. This term's formulation results in the following equality.

$$\begin{array}{l} \text{Weighted} \\ \text{Residual of} \end{array} \left[ \alpha^{(e)} \right] = \int_{D(e)} \{W\} \alpha \, dx dy \quad (5)$$

Decay term. The decay term is a first order reaction of the form  $\beta = KC$  where  $K$  is the coefficient of decay. The coefficient will be constant over an element and the term's formulation results in the following equality.

$$\begin{array}{l} \text{Weighted} \\ \text{Residual of} \end{array} \left[ \beta^{(e)} \right] = \int_{D(e)} K \{W\} [Nc] \, dx dy \{c\} \quad (6)$$

Time rate of change term. The time rate of change term cannot be represented by the Galerkin or "upwinding" methods. Since the interpolation functions are dependent only on spatial information, the derivative of concentration with respect to time ( $\frac{\partial c}{\partial t}$  or  $\dot{c}$ ) cannot be expressed with the interpolation functions ( $[N_t] = 0$ ). This dilemma is solved by using a finite difference representation of the rate of concentration change.

$$\dot{c} = \frac{c^{t+\Delta t} - c^t}{\Delta t}$$

Now, applying the interpolation functions, weighting functions, and taking the inner product gives

$$\frac{\partial c}{\partial t} = \frac{1}{\Delta t} \int_{D(e)} \{W\} [Nc] \, dx dy \{c\}^{t+\Delta t} - \frac{1}{\Delta t} \int_{D(e)} \{W\} [Nc] \, dx dy \{c\}^t \quad (7)$$

This model does not simply add the rate of change term to the convection-dispersion equation as shown in Equation (2) for its unsteady state solution. This implicit method of time stepping is inherently stable but not exceptionally accurate. A more accurate and inherently stable method is the Crank-Nicolson method. This technique adds the time rate of change term to the convection-dispersion equation and then averages the terms of this.

$$[M] \{ \dot{c} \} + [K] \{ c \} = \{ f \} \quad (8)$$

where,

$$\dot{c} \approx \frac{c^{t+\Delta t} - c^t}{\Delta t}$$

$$c \approx \frac{c^{t+\Delta t} + c^t}{2}$$

$$f \approx \frac{f^{t+\Delta t} + f^t}{2}$$

(M) and (K) are the so-called "mass" and "stiffness" matrices, respectively. This approach is better than the implicit method because it reduces the numerical influence of the size of the time increment (Carnahan-Luther-Wilkes (1)).

#### Types of elements used

The computer model developed has the capability to use three types of quadrilateral elements in its solution process. They are described in Table (1). The first type is an isoparametric element with quadratically varying interpolation functions. This allows the element to have curvilinear sides. The second type of element has linearly varying interpolation functions with linear sides. Zienkiewicz (11) gives a complete

Table 1. Description of elements.

| Element No. | Type of Concentration Interpolation Function | Type of Velocity Interpolation Function | No. of Nodes on Element | Remarks |  |
|-------------|--|---|-------------------------|---------|--|
| 1           | Quadratic                                    | Quadratic                               | corner                  | 4       | All information processed at all 8 nodes.  |
|             |  |   | mid-side                | 4       |  |
|             |  |   | TOTAL                   | 8       |  |
| 2           | Linear                                       | Linear                                  | corner                  | 4       | All information processed at all 4 nodes.  |
|             |  |   | mid-side                | 0       |  |
|             |  |   | TOTAL                   | 4       |  |
| 3           | Linear                                       | Quadratic                               | corner                  | 4       | Velocity information processed at all 8 nodes. Concentration information processed at corner nodes only. |
|             |  |   | mid-side                | 4       |  |
|             |  |   | TOTAL                   | 8       |  |

explanation of the first two types of elements that are used in the model. Element No. 3 is a blend of the first two types of elements.

#### Boundary conditions

There are two types of boundary conditions incorporated in the model -- Dirichlet and Neumann conditions. The Dirichlet condition is a specified-value condition wherein a fixed value of concentration is assigned to a boundary node. This means that the equation for this node is eliminated, since no calculation is required. The Neumann condition is the natural boundary condition. It takes the form,  $D_x \frac{\partial c}{\partial x} 1_x + D_y \frac{\partial c}{\partial y} 1_y$ .



In our situation, these line integrals in the dispersive terms (see Equation (4)) become zero and do not have to be incorporated in the governing equation or in the computer program. They are naturally satisfied.

#### Finite element convection-dispersion equation

To put all this information into perspective, the following two equations below show the finite element formulation of the steady and unsteady state versions of the governing equation.

##### Steady state.

$$\left[ \int_{D(e)} \{W\} [Nv] \{u\} [Nc_x] + \{W\} [Nv] \{v\} [Nc_y] + D_x \{W_x\} [Nc_x] + D_y \{W_y\} [Nc_y] dx dy \right] \{c\} + \alpha [W] = \{R\} \quad (9)$$

##### Unsteady state.

$$\left\{ \frac{1}{2} \int_{D(e)} \{W\} [Nv] \{u\} [Nc_x] + \{W\} [Nv] \{v\} [Nc_y] + D_x \{W_x\} [Nc_x] + D_y \{W_y\} [Nc_y] + K \{W\} [Nc] dx dy \right\} \{c\}^t + \Delta t + \frac{1}{\Delta t} \int_{D(e)} \{W\} [Nc] dx dy \{c\}^t + \Delta t = \left\{ \frac{\alpha^t + \Delta t + \alpha^t}{2} \{W\} \right\} + \frac{1}{\Delta t} \int_{D(e)} \{W\} [Nc] dx dy \{c\}^t - \left\{ \frac{1}{2} \int_{D(e)} \{W\} [Nv] \{u\} [Nc_x] + \{W\} [Nv] \{v\} [Nc_y] + D_x \{W_x\} [Nc_x] + D_y \{W_y\} [Nc_y] + K \{W\} [Nc] dx dy \right\} \{c\}^t \quad (10)$$

## CHAPTER IV

## MODELING TECHNIQUE AND PARAMETER DISCUSSION

The computer model uses a variety of parameters to simulate various problem situations. This chapter will explain the use of these parameters and demonstrate their application in solving a variety of convection-dispersion problems.

Computer Model's Parameters

The model has seven main parameters for simulating various conditions. They are as follows:

1. Degree of approximation - velocity and concentration
2. Dispersion coefficient
3. Velocity field
4. Boundary conditions
5. Steady or unsteady-state
6. Time varying source-sink terms
7. Time dependent decay term.

The effect of these parameters is independent of one another but some thought is required when combining them to create a solution to a problem. The following sections explain their purposes and uses.

Degree of approximation

The model can calculate an approximate solution to a problem using either a linear or quadratic variation of the nodal concentration values. The quadratic approach requires fewer nodes and elements to obtain a degree of accuracy similar to that found with the linear case, but each element requires more calculations. The choice of the degree of variation is

dependent on the specific problem and the computer facility available to execute the program. The quadratic variation can only use the Galerkin method, whereas the linear variation has the option of using either the Galerkin or "upwinding" methods. In convection-dominated problems, using the linear "upwinding" method can give much more reasonable results than the higher order quadratic Galerkin method.

#### Dispersion coefficient

The computer model approximates the dispersion process by allowing the orthogonal components of the dispersion coefficient to vary from element to element. The two component coefficients are constant over each individual element, but need not be equal.

#### Velocity field

The velocity information is represented by nodal values of the orthogonal velocity components. They can vary either linearly or quadratically within a given element. If the velocities variation is linear then the model's degree of approximation to the exact solution must be linear in concentration (see Table 1). The quadratically varying velocities can be used with either the linear or quadratic variation of concentration in the model. The type of approximating method that should be used depends on whether the problem is convection or dispersion-dominated.

The Galerkin method will give a good approximation to the exact solution of dispersion-dominated problems. If the domain is convection-dominated then the "upwinding" method will give a better approximation to the solution than the Galerkin method. The Galerkin method fails to accurately approximate the solution because of the severe longitudinal (direction of the velocity) oscillations which occur.

### Boundary conditions

The use of the two types of boundary conditions, Dirichlet and Neumann, can represent a multitude of conditions. The Dirichlet condition can simulate constant concentration boundaries, varying concentration boundaries, and single node specifications. The Neumann condition can simulate solid boundaries, reflective boundaries, and transport boundaries. This section deals with their application as related to the computer model.

The Dirichlet boundary condition is a specified-value condition. It can simulate a constant concentration boundary condition by setting all of the nodes on that boundary equal to a specific value. No nodes on that boundary should be excluded from having that specific value. If there are some nodes on that boundary that are not specified, then a serious discontinuity problem can appear in the solution process. This constant concentration boundary condition has an identically self-imposed tangential gradient.

One way of establishing a tangential concentration gradient is by a gradient boundary condition. This is done by specifying different concentration values for the nodes along a boundary. There are a multitude of situations that can be simulated using this type of boundary specification. The values must always vary to the degree of the interpolation functions, which are linear or quadratic. The same caution on continuity applies to the tangential gradient boundary conditions as well as to the constant-concentration boundary condition.

Single node specifications should be used only under unsteady-state conditions. Specifying independent nodes (nodes not sharing a set of

interpolation functions) should only be done with nodes that are assured of receiving negligible concentrations of substance. By specifying these nodes with a value of zero and using a source term (to be explained in a later section of this chapter) to establish a concentration gradient, a solution of reasonable credibility can be obtained. This is because time dependent problems are not only dependent on the boundary conditions but also on the previous time steps information.

Dirichlet boundary conditions must be assigned to at least one node in order for the governing equation to generate a unique solution to a problem. The computer model's other basic type of boundary condition is the Neumann condition and it simulates a completely different class of boundary conditions.

A common use for the Neumann condition is to simulate a solid boundary. Since there can be no transport across a solid boundary, the Neumann boundary condition ( $\frac{\partial c}{\partial n} = 0$ ) is an exact representation of a solid boundary.

A reflective boundary (axis of symmetry) is another common application for the Neumann conditions. A reflective boundary means that a domain must be symmetric about an axis. Using the axis of symmetry as a Neumann boundary condition while modeling only half of the domain results in a savings of computer storage and execution time in obtaining an approximate solution. In this case there is no net transport across this axis of symmetry because of the use of the Neumann boundary condition. The Neumann condition can also be used to approximate transport across boundaries.

When a Dirichlet boundary condition is undesirable at a boundary the Neumann condition ( $\frac{\partial c}{\partial n} = \text{constant}$ ) must be utilized. This condition can

represent the gradient across a boundary. The present computer model is only capable of using a homogeneous Neumann condition,  $\frac{\partial c}{\partial n} = 0$ . This means that the dispersive transport across a boundary will be approximately zero when represented by the Neumann boundary condition. This is a good approximation for a transport boundary if the dispersive transport is small compared to the convective transport. The Neumann condition does not restrict convective transport across a boundary.

#### Steady or unsteady-state

The model has the ability to simulate either steady-state or unsteady-state problems. The steady-state solution process can only utilize the aforementioned parameters (degree of approximation, dispersion coefficients, velocity profile, and boundary conditions). The unsteady-state solution also uses these parameters in addition to source-sink and decay terms. The degree of approximation, dispersion coefficient, velocity profile, and decay terms are static parameters. This means that their values do not change with time, even during an unsteady-state solution process. The only dynamic parameters are the source-sink terms and the boundary conditions. The source-sink and decay terms will be discussed in the next two sections.

The unsteady-state solution process utilizes a discrete time increment to progress through time. This computer model has the capability of changing the time increment during an execution. The time increment can have three different values during a solution process. This allows the use of a small time increment when there is rapid change occurring in the domain and a large time increment when the domain is not experiencing such a rapid change.

### Time varying source-sink terms

The parameter used to introduce or remove an amount of substance over a finite period of time is the source-sink term. This term is constant over an individual element, but can vary from element to element and from time-step to time-step. It has its greatest value in an unsteady-state solution process, but can be used in a steady-state problem.

The source-sink term can be used in a steady-state problem to simulate a physical injection of a continuous constant concentration value. In the unsteady-state the source-sink term can simulate physical injections of slugs of concentrated substance. The unsteady-state simulation is useful for monitoring an introduced pollutant, whereas the steady-state simulation approximates a continuous source.

### Time dependent decay term

The time dependent decay term is used when the substance being monitored has the characteristic of decaying with time. This phenomena is simulated by using a first-order reaction decay term. The rate of decay is dependent on the concentration of substance present. The higher the concentration present, the greater the rate of decay. This term is useful when the model is monitoring concentrated substances such as biological oxygen demand, absorbed or evaporating substances, etc.

### Example Convection-Dispersion Problems

The following is a series of examples illustrating the use of the various parameters in modeling a variety of convection-dispersion problems. The first set of examples (Figures 4 through 10) are steady-state situations intended to show the use and affect of the degree of approximation,

boundary conditions, and the interrelation of the dispersion coefficient and the velocity profile. The second set of examples (Figures 4, 11 through 13) are unsteady-state situations intended to show the use and affect of the source-sink term, boundary conditions, and time interval.

#### Steady-state examples

A standard domain (Figure 4) will be used as a control to show the affect of the changing parameters. Figure 5a shows the node and element placement for the linearly varying domain and Figure 5b shows the node and element placement for the quadratically varying domain in the steady-state examples.

Example 1. This first example has a restrictive set of boundary conditions and a relatively high isotropic dispersion coefficient of  $10.0 \text{ ft}^2/\text{sec}$ . All of the boundaries except for the reflective boundary are constant concentration boundaries. The influent boundary has a value of 0.0 while the solid and effluent boundaries carry a value of 1.0. The reflective boundary is of course a Neumann boundary condition. This example is simulating a diffuser on the solid boundary with the assumption that the domain is long enough to allow the concentration to be uniform when the fluid reaches the effluent boundary. Figure 6 shows the results of the linear Galerkin method's solution to this example. The quadratic Galerkin method gives an almost identical solution to this example.

Example 2. This example is the same as example 1 except that the isotropic dispersion coefficient has been lowered by an order of magnitude to  $1.0 \text{ ft}^2/\text{sec}$ . Figures 7a and 7b show the severe longitudinal oscillations that are generated using the Galerkin method for the linear and quadratic elements, respectively. These results are not a satisfactory



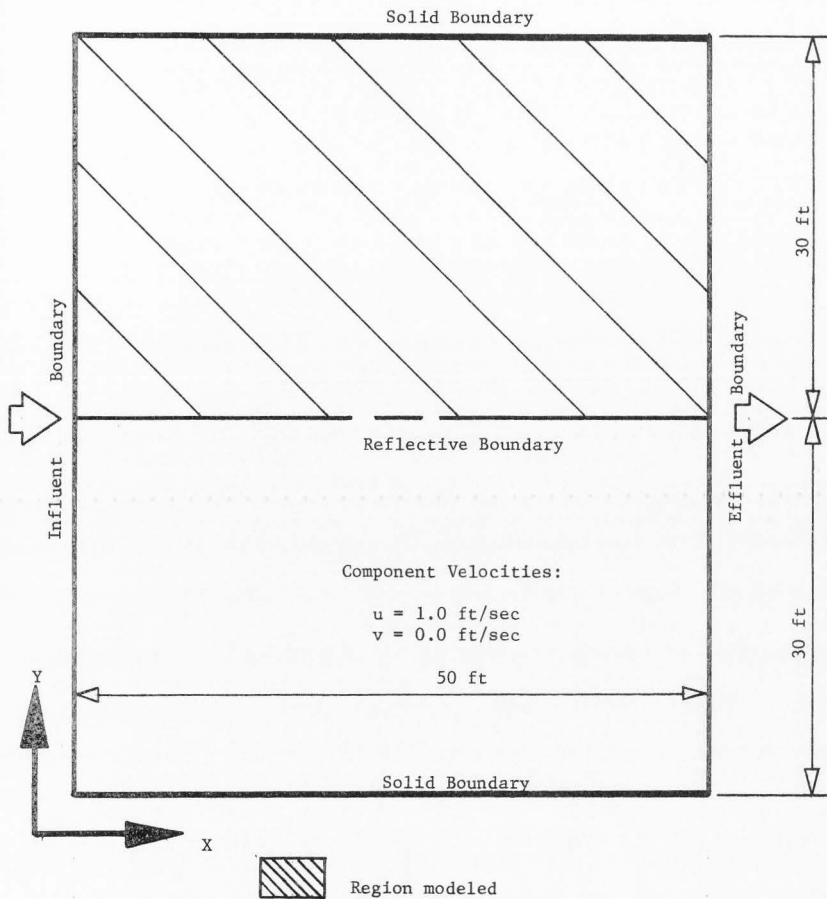


Figure 4. Domain for examples.

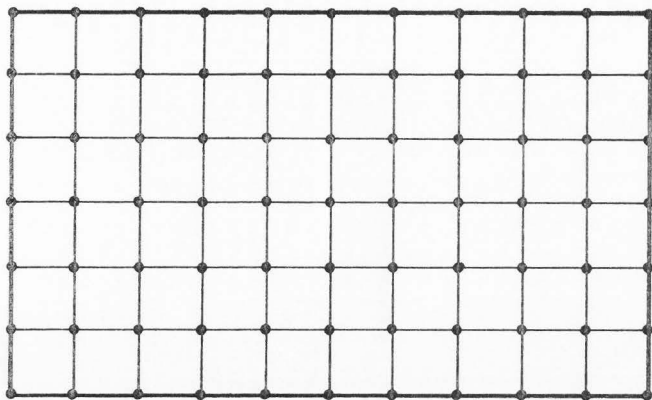


Figure 5a. Linear discretized domain for steady-state examples.

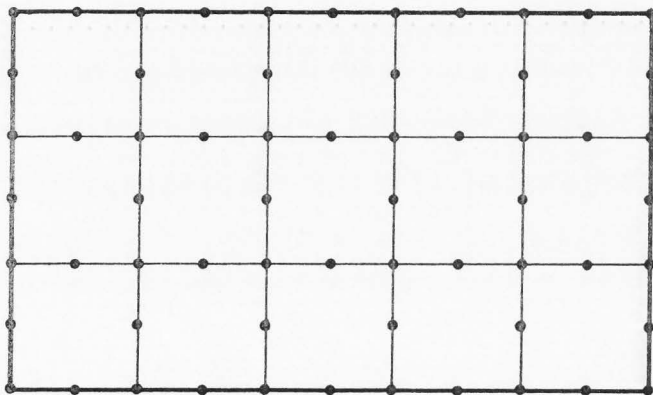
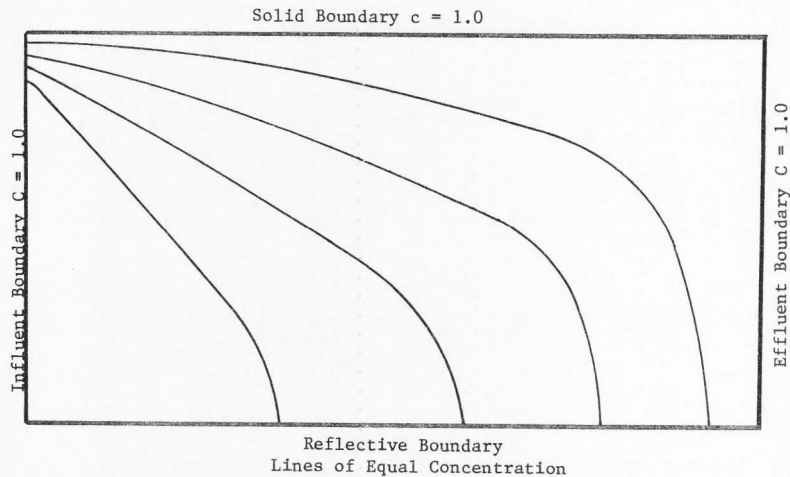


Figure 5b. Quadratic discretized domain for steady-state examples.



$$D_x = D_y = 10 \text{ ft}^2/\text{sec}$$

$$C \text{ in lbm/ft}^3$$

Figure 6. Solution of example 1 - dispersion dominated, linear Galerkin.

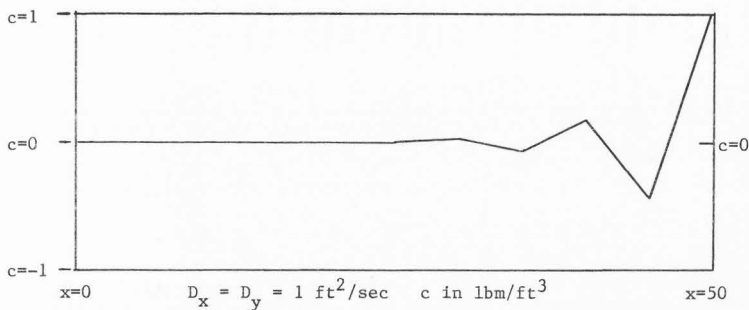


Figure 7a. Linear Galerkin solution of example 2 along reflective boundary.

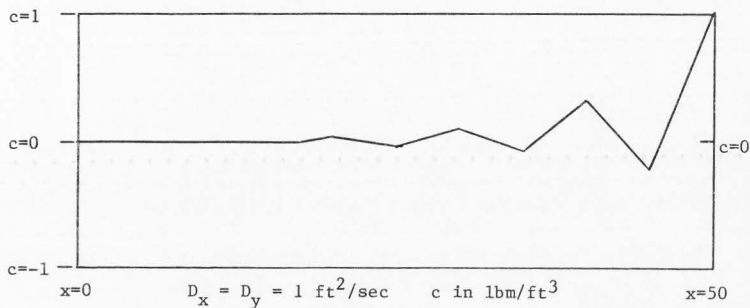


Figure 7b. Quadratic Galerkin solution of example 2 along reflective boundary.

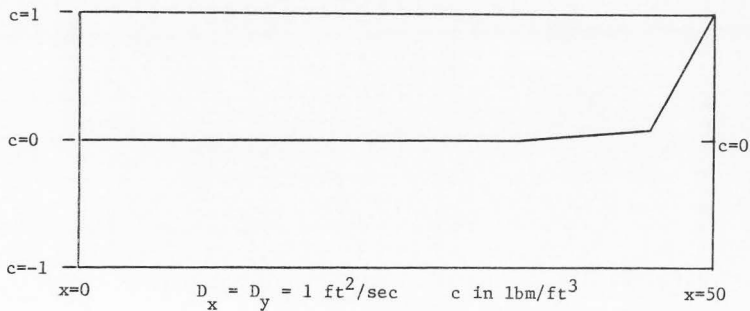


Figure 7c. Linear "upwinding" solution of example 2 along reflective boundary.

approximation to the solution because of the oscillations and the meaningless negative concentration values. The "upwinding" method as applied to the linear elements gives a non-oscillatory solution, as shown in Figures 7c and 8. This approximate solution is more accurate than either Galerkin solution, thus, illustrating the importance of using the "upwinding" method in convection-dominated problems.

Example 3. In this example the isotropic dispersion coefficient is  $10.0 \text{ ft}^2/\text{sec}$ . The boundary conditions are the same as in the previous examples, except that the effluent boundary is now a Neumann condition describing this boundary as a transport boundary ( $\frac{\partial c}{\partial n} = 0$ ). This of course assumes that the dispersive transport at the effluent boundary is small compared to the convective transport across the boundary. This simulates the same situation as in example 1 except that a uniform exit concentration is no longer assumed. Figure 9 shows the approximate solution for the linear Galerkin method. It can readily be seen how this boundary condition at the effluent boundary yields a different solution when compared to example 1 (Figure 6).

Example 4. This example is different from the three previous examples because of its boundary conditions. An isotropic dispersion coefficient of  $10.0 \text{ ft}^2/\text{sec}$  is used with the Neumann boundary condition describing all of the boundaries except for the influent boundary. The influent boundary is a gradient boundary condition with a zero value for concentration from  $0.0$  to  $1.0 \text{ lbm}/\text{ft}^3$  along the remaining 10 feet of the boundary. The concentration's maximum value is at the reflective boundary. This example simulates the inflow of a concentrated substance in the middle of a flow field.

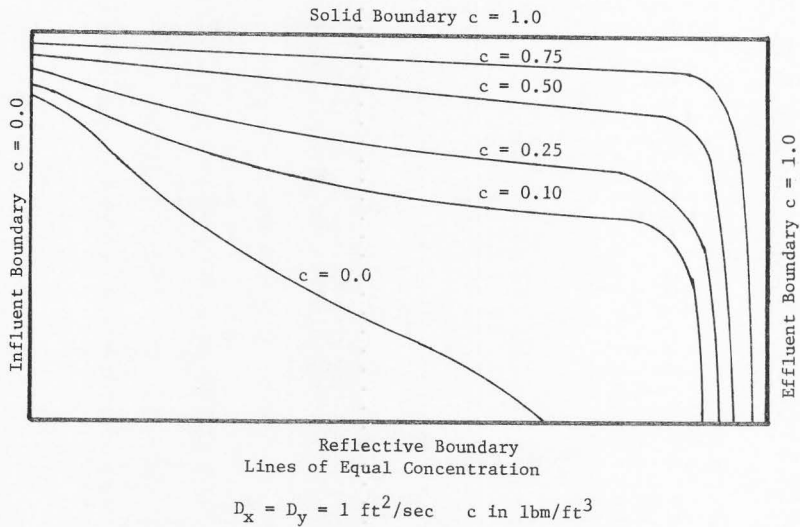


Figure 8. Solution of example 2 - convection dominated, "upwinding" method.

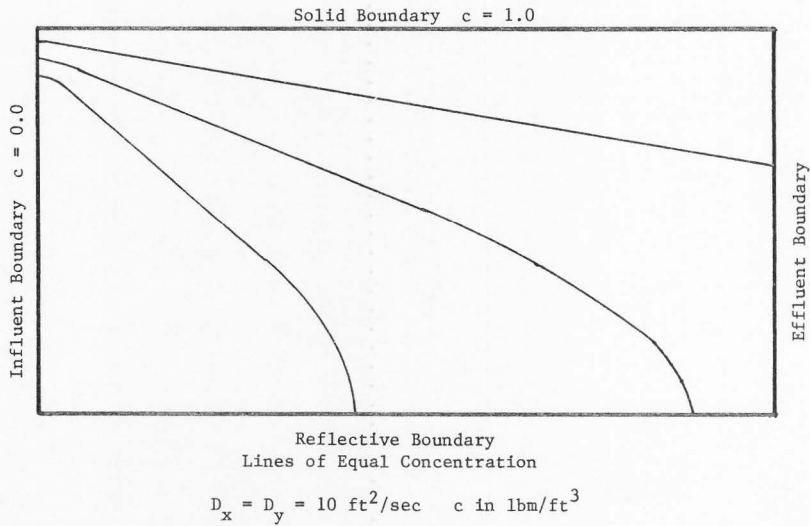


Figure 9. Solution of example 3 - dispersion dominated, linear Galerkin, Neumann condition at effluent boundary.

Figure 10 shows the results of this example using the linear Galerkin method. This technique of using a gradient Dirichlet boundary condition can be used to simulate various steady-state problems, such as substance introduced through an inlet pipe to a pond, estuary discharge into a lake, etc.

#### Unsteady-state examples

The unsteady-state examples will use the domain shown in Figure 4, but the discretization will not be the same as in the steady-state examples. The elements will be smaller near the reflective boundary (Figure 11) to allow a better resolution for approximating the exact solution in that area of the domain.

Example 5. This example is identical to example 4 except that the exact solution is approximated by an unsteady-state solution process instead of a steady-state solution process. Figure 12 shows the concentration of substance with respect to time at a position 30 ft downstream of the influent boundary along the reflective boundary. It can be seen that the value of the concentration at that point approaches the concentration calculated by the steady-state solution. Given enough time, the unsteady-state solution should match the steady-state solution. This result will always occur whenever a problem has no time varying (dynamic) parameters.

Example 6. This example is a dynamic problem that utilizes the source term. The source is located over an element next to the reflective boundary (element marked with a circle in Figure 11). The source supplies  $10.0 \text{ lbs/ft}^3\text{-sec}$  of substance applied over the element linearly from  $t = 0$  to  $t = 1.0$  seconds while using a time increment of 0.5 seconds. All of



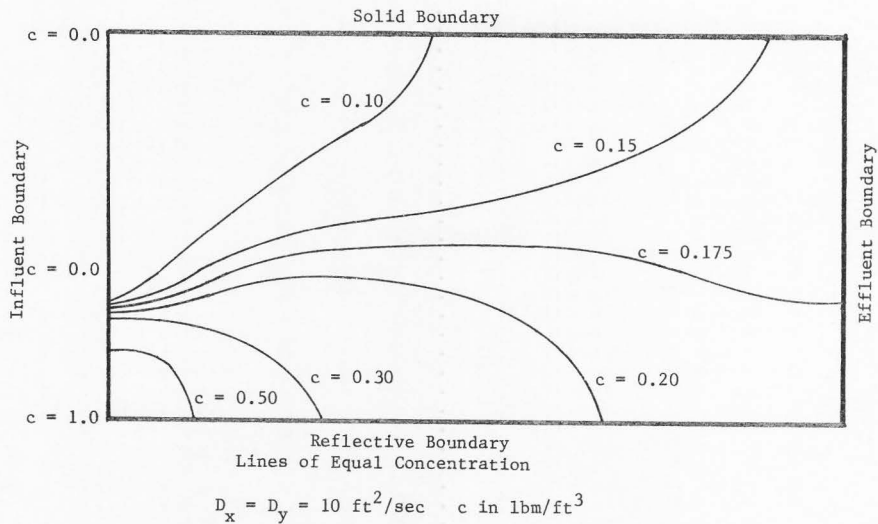


Figure 10. Solution of example 4 - dispersion dominated, linear Galerkin, Dirichlet condition only on influent boundary.

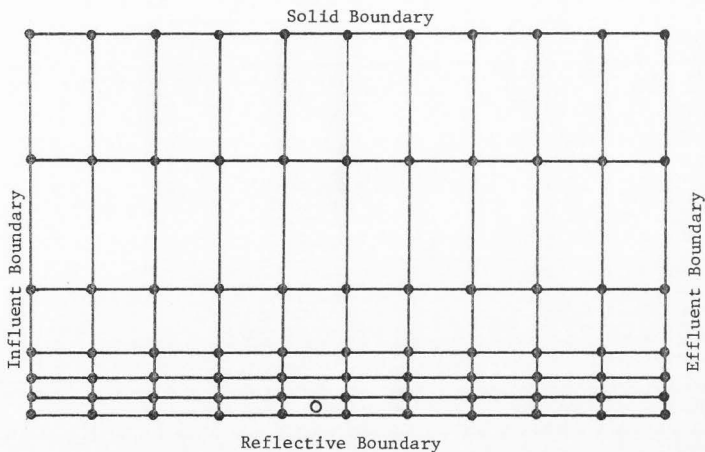


Figure 11. Linear discretized domain for unsteady-state examples.

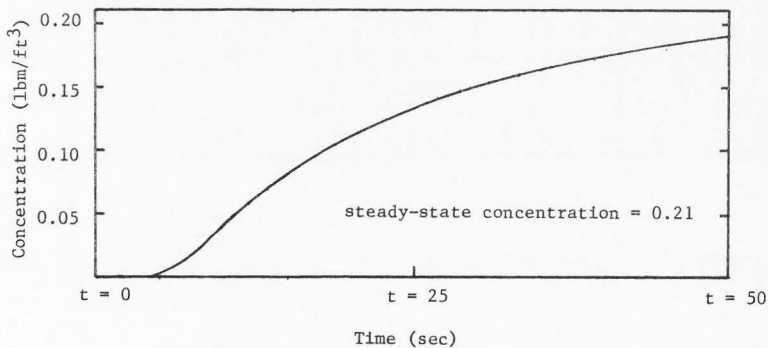


Figure 12. Concentration at a position 30 ft downstream of the influent boundary along the reflective boundary for example 5.

the boundaries utilize the Neumann boundary condition except for a 15 foot section at the upstream end of the solid boundary, which is a constant-value boundary condition with a value of 0.0. The value of the isotropic dispersion coefficient is  $1.0 \text{ ft}^2/\text{sec}$ .

This example shows that a substance will disperse upstream for a short period of time if the dispersive effect is strong enough to overcome the convective effect. Figure 13 shows the concentration profile along the reflective boundary at various times.

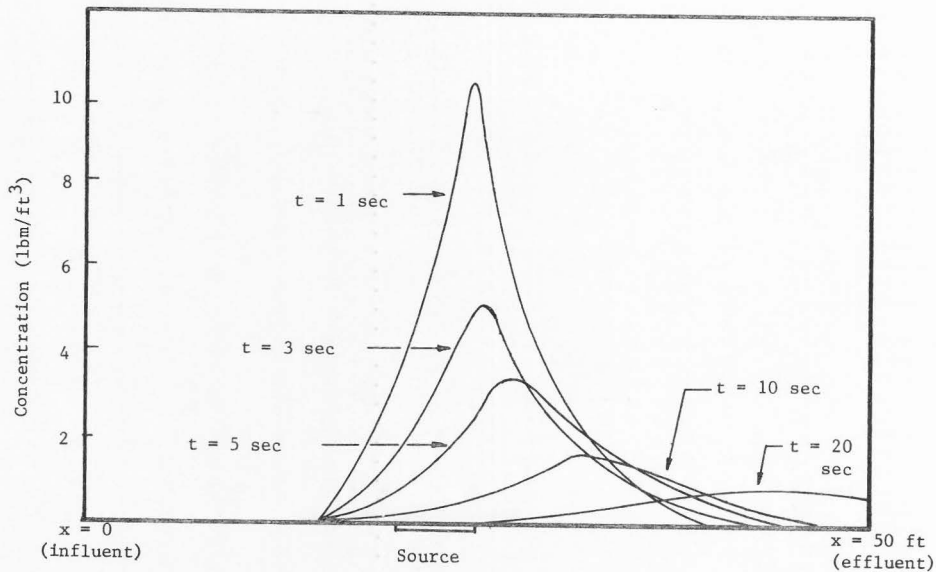


Figure 13. Concentration profile along the reflective boundary for example 6 - time varying source.

## CHAPTER V

## PROBLEMS ASSOCIATED WITH THE APPLICATION OF THE MODEL

The computer model does a reasonable job in calculating an approximate solution to a convection-dispersion situation, but there are some problems that must be addressed when using the model. These problems can be categorized into two related and basic areas, conservation of mass and discretization of the domain. This chapter explains the specific items of importance and the problems associated with them.

Velocity Field

The velocity field is perhaps the most critical parameter used in the model. It is extremely important that the nodal velocities preserve continuity for each and every element in the discretized domain. If the flow field does not preserve continuity then erroneous results are computed by the model. Generally speaking, severe oscillations in the approximate solution will occur when fluid continuity is disregarded.

Determining the velocity field by hand is a cumbersome task for almost any realistic situation. Continuity must be preserved for each element when developing a flow field representative of a particular situation. The use of a hydrodynamic computer model is a necessity to properly calculate the nodal velocities with the same degree of accuracy as employed in the convection-dispersion computer model.

Dispersive Transport at the Boundaries

When the model generates a solution it approximates the concentration values in the domain at the discrete nodes. Based on these discrete

concentration values and the interpolation functions, the gradients of concentration in any direction can be calculated. Given a certain number of nodes in an area near the boundary, the approximation of the normal gradient of the concentration ( $\frac{\partial c}{\partial n}$ ) can be determined. If the number of nodes in that same area is increased (finer mesh size), then a better approximation to the normal derivative can be calculated. The exact solution occurs when the domain is discretized with an infinite number of nodes, which of course is impossible.

The more nodes in the domain, the more costly the solution with respect to computer time. Determination of the optimum mesh size for the accuracy and cost desired is dependent on the problem and the capabilities of the computer being utilized. In this model care must be taken when discretizing the domain because any coarseness in the mesh size near the boundaries yields a poor approximation to the Neumann boundary condition. Experience has shown that an extremely fine mesh size is needed to adequately approximate the homogeneous Neumann boundary condition. Since an evaluation of the natural boundary condition is used to determine the dispersive mass transport across the boundaries, the accuracy of a mass balance check of the domain is seriously impaired when a coarse mesh size is used at the boundaries.

#### Introduction of Substance

Dynamically introducing a slug of substance over a finite period of time into the domain results in a numerical error when calculating the check for the conservation of mass of substance. It appears that the abruptness of starting and stopping the injection of substance causes this

lack of mass conservation. There are a few techniques available to reduce this problem.

The accuracy improves when a finer mesh size is used in the area of the injection. Since the immediate effect is only over the elements next to the injection area the finer mesh size helps minimize the abruptness of the changes in the domain. The mass balance check can also be improved by approximating a slug injection of substance with a gradual step-wise increase and decrease of substance. This also helps reduce the abruptness of a slug injection. For a maximum reduction of the mass balance discrepancy it is recommended that both the mesh refinement and gradual introduction of substance be used.

#### Discretization of the Domain

The nodal spacing ( $\Delta X$ ,  $\Delta Y$ ) and the time interval ( $\Delta t$ ) have quite an effect on the accuracy of an unsteady-state solution. As mentioned earlier in this chapter, the mesh size (nodal spacing) can play a critical role in various ways for properly modeling a region to obtain an accurate solution. The interrelation between the nodal spacing and the time interval can also be a source of trouble when attempting to model a region.

The time interval should be determined with respect to the mesh size, velocity field, and dispersion coefficients. The time interval should not be so large as to allow the substance to convect or disperse past too many nodes during each time step thus causing a poor representation of the concentration profile in the domain. The time interval should also not be so small as to prevent the substance from reaching any interior

nodes in the first time step thus causing oscillations with large negative concentration values in the domain. These negative values affect the solution for the rest of the time steps in the unsteady-state process causing the entire solution to be of questionable value. In any domain with a varying mesh size and a complex velocity profile the determination of a proper time interval can be difficult.

The two methods of transport, convection and dispersion, are the basis for determining the time interval with relation to mesh size. The calculation of the rate of transport for the convective process is straight-forward. It is simply the distance traveled divided by the velocity. The calculation of the rate of transport for the dispersive process is much more complicated. The dispersive transport is dependent on the concentration gradient. Since this gradient varies with time and position it becomes difficult to evaluate an appropriate rate of dispersive transport. In any case, the time interval selected will always yield a stable but not necessarily accurate solution because the model uses the inherently stable Crank-Nicolson method of time stepping.



CHAPTER VI  
CONCLUSIONS AND RECOMMENDATIONS

The construction of this convection-dispersion computer model has led to a good preliminary low-cost model which can be applied to Great Salt Lake. It has also been useful in obtaining knowledge to be passed on for future works on more complex models. The model develops reasonable solutions to simple hypothetical situations, but more effort is needed on the verification and improvement of this model as well as more research in related areas.

The model is designed to simulate most of the situations that can occur in two-dimensional convection-dispersion problems. The model has some faults as well as some advantages as described in the following:

1. The model has the capability of using a variety of approximating techniques. The model can use linear or quadratic interpolation functions for substance concentration, the Galerkin or "upwinding" methods of weighted residuals, and a linear or quadratic velocity variation.
2. The model simulates reasonably well the steady-state uniform flow fields and accepts various methods of introducing the substance, such as internal injections, established concentration gradients, and diffusers.
3. Continuity of the fluid in the domain is a requisite in order for the model to compute a solution to a problem. Severe oscillations in the approximate solution will occur in the domain if continuity is disregarded.

4. There are difficulties in conserving mass when making abrupt changes in the time varying introduction of substances. When the abruptness of the changes in the domain are minimized the model conserves mass with a fair degree of accuracy. For the time period after the injection of the substance, the model does conserve mass.
5. Dispersive transport at the boundaries also causes mass conservation problems. Numerically approximating the concentration's normal gradient at the boundaries (which should be zero) results in undesired dispersive transport. This error can be minimized with a fine mesh size. This discrepancy in the conservation of mass is negligible compared to the problems discussed earlier relating to the introduction of a substance.
6. In convection-dominated problems the model develops longitudinal oscillations in its approximate solution. The "upwinding" technique helps minimize these severe oscillations. Abrupt changes in the boundary conditions also cause oscillations, but in this case the "upwinding" method gives no aid in reducing them.

With a few improvements in the model some of the problems listed above can be alleviated.

A general Neumann boundary condition, a nodal representation of the dispersion coefficient, and a quadratic "upwinding" technique would be definite improvements in the model. The Neumann condition can be improved

by allowing the normal concentration gradient to have a non-zero value at transport boundaries thus allowing dispersive transport across Neumann condition boundaries. A refinement of the dispersion coefficient can occur by letting it vary from node to node instead of being constant over each element as found in this model. The creation of a quadratic "upwinding" technique is the most difficult of the above mentioned improvements. Determining the optimum "upwinding" factors would be the hardest part of creating a quadratic "upwinding" method, but this method would have the ability to reduce the longitudinal oscillations found in convection-dominated Galerkin solutions while giving a better approximation to the concentration's normal gradient than found in the linear "upwinding" method.

A quantitative verification of the model should be done with a physical prototype. By attempting to accurately model a physical situation insights into the usefulness of this model as well as ideas for improvements can be obtained. A practical way of quantitatively determining the dispersion coefficient and the size of the time interval for unsteady-state solutions would be additional benefits from physical verification.

At the present time, it can be said that a good deal of practical knowledge about the capability of the finite element method to represent the convection-dispersion equation can be learned from using this computer model. Many improvements are needed, but this research is a first step in attempting to develop a model which can quantitatively determine the dispersion of a substance in the Great Salt Lake.

## REFERENCES

1. Carnahan-Luther-Wiles. Applied numerical methods. John Wiley and Sons, Inc. 1969.
2. Christic, Griffiths, Mitchell, and Zienkiewicz. Finite element methods for second order differential equations with significant first derivatives. International Journal for Numerical Methods in Engineering, Vol. 10. 1976. pp. 1389-1396.
3. George, Robert L. Two-dimensional wind generated flow patterns, diffusion and mixing in a shallow stratified pond. Utah State University. 1974.
4. Harleman, D.R.F. Significance of longitudinal dispersion in the analysis of pollution in estuaries. 2nd International Conference on Water Pollution Research, Proceedings. Pergamon Press, Ltd. 1974.
5. Heinrich, Huyakorn, Zienkiewicz and Mitchell. An "upwind" finite element scheme for two-dimensional convective transport equation. International Journal for Numerical Methods in Engineering, Vol. 11. 1977. pp. 131-143.
6. Holley, Edward R. Unified view of diffusion and dispersion. Journal of the Hydraulics Division, Proceedings of the American Society of Civil Engineers, Vol. 95. 1969.
7. Huebner, Kenneth H. The finite element method for engineers. John Wiley and Sons, Inc. 1975.
8. Huyakorn, P.S. Solution of steady-state, convective transport equation using an upwind finite element scheme. Applied mathematical modeling, Vol. 1. 1977.
9. Ippen, Arthur T. Estuary and coastline hydrodynamics. McGraw-Hill Book Company, Inc. 1966.
10. Narayanan and Shankar. A numerical model for the simulation of two-dimensional convective-dispersion in shallow estuaries. International Journal for Numerical Methods in Engineering, Vol. 10. 1976. pp. 873-883.
11. Zienkiewicz, O.C. The finite element method in engineering science. McGraw-Hill Book Company, Inc. 1971.

## APPENDICES

## Appendix A

UNITS OF VARIABLES IN THE CONVECTION-DISPERSION EQUATION

---

Table 2. Basic units

| Dimensions | English | Metric   |
|------------|---------|----------|
| L = length | Foot    | Meter    |
| F = force  | Pound   | Kilogram |
| T = time   | Second  | Second   |

Table 3. Variables used in model

| Variables              | Units           |
|------------------------|-----------------|
| Concentration          | $FL^{-3}$       |
| Velocity               | $LT^{-1}$       |
| Distance               | L               |
| Dispersion Coefficient | $L^2T^{-1}$     |
| Source or Sink         | $FL^{-3}T^{-1}$ |
| Decay Coefficient      | $T^{-1}$        |

## Appendix B

## USER'S GUIDE TO CONVECTION-DISPERSION MODEL



User's Guide to Convection-Dispersion Model

This appendix describes how to set up the data needed to execute the computer model. The model has a variety of options. The following is a step by step description of the input variables.

Type of model

There are five different types of models available in the computer program. They are dependent on the type of element, degree of interpolation function, type of weighting technique, and velocity variation. Table B-1 shows the five types of models and the commands needed to use each type of model.

Input for program

Language: FORTRAN IV

NameList IØ. This statement reads in the various control parameters that govern subsequent read statements and the solution process. Logicals - set to true if desired (all values default to false).

- UPWIND - "upwinding" technique in use, linear elements only.
- ZERLRF - make linear elements Galerkin. UPWIND must be set to true.
- LINVEL - linear velocity variation, UPWIND must be set to true.
- CØNVEL - velocity is the same at all nodes (uniform flow-field)
- PGSM - print global stiffness matrix contributions and load vector values.
- CØNDIF - dispersion coefficient is the same for all elements.
- CØNCB - background concentration is the same at all nodes, STEADY must be set to false.

Table 4. Types of modeling techniques

| Model No. | Type of Element*    | Degree of Concentration Interpolation Function | Weighting Technique | Degree of Velocity Interpolation Function | Commands   |
|-----------|---------------------|--|---------------------|---|--|
| 1         | Quadratic (8 nodes) | Quadratic                                      | Galerkin            | Quadratic                                 | UPWIND = FALSE<br>ZERALF = FALSE<br>LINVEL = FALSE |
| 2         | Linear (8 nodes)    | Linear   | Galerkin            | Quadratic                                 | UPWIND = TRUE<br>ZERALF = TRUE<br>LINVEL = FALSE   |
| 3         | Linear (8 nodes)    | Linear   | Upwinding           | Quadratic                                 | UPWIND = TRUE<br>ZERALF = FALSE<br>LINVEL = FALSE  |
| 4         | Linear (4 nodes)    | Linear   | Galerkin            | Linear                                    | UPWIND = TRUE<br>ZERALF = TRUE<br>LINVEL = TRUE    |
| 5         | Linear (4 nodes)    | Linear   | Upwinding           | Linear                                    | UPWIND = TRUE<br>ZERALF = FALSE<br>LINVEL = TRUE   |

\* Refer to Table 1 for further explanation.

- STEADY - Steady-state problem. If false, then it is an unsteady-state problem.
- ZERO - re-set concentrations to background concentration if calculated values are less than the background concentrations. STEADY must be set to false, CØNCB must be set to true.
- ZERØM - print adjusted values as designated by ZERØ. STEADY must be set to false, CØNCB must be set to true, ZERØ must be set to true.
- ELINFØ - print input data by elements. Input data is always printed by nodes.
- REGIØN - plot the domain by concentration values.
- CØNDEC - decay coefficient is the same for all elements. STEADY must be set to false.
- PALP - print factors used in "upwinding" technique, UPWIND must be set to true.
- CHDT - change time interval during unsteady-state execution, STEADY must be set to false.
- PØWØFF - no scaling of domain plot with respect to concentration, REGIØN must be set to true.
- PHIØFF - no decrease scaling of domain plot with respect to concentration, REGIØN must be set to true.
- PLØØFF - no increase scaling of domain plot with respect to concentration, REGIØN must be set to true.
- MSCK - check mass balance.
- FLAG - print mass balance information by element, MSCK must be set to true.

Numericals - specify value desired (defaults in parenthesis).

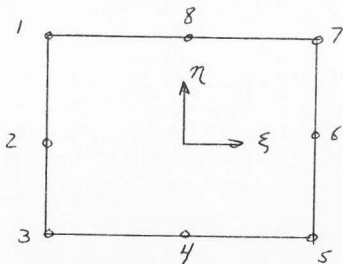
- NE - number of elements
- NN - total number of nodes
- NNC - number of corner nodes, if UPWIND is set to false then NNC is not necessary (0).
- NQP - number of Gaussian quadratic points for numerical integration. Only allowable values are 2 and 3. 3 is recommended.

- NSC - number of source-sink term (0).
- TC - unsteady-state solution's stop time, STEADY must be set to false (0.0)
- DT - initial time interval for unsteady-state solution, STEADY must be set to false (0.0)
- DT2 - second time interval for unsteady-state solution, STEADY must be set to false, CHDT must be set to true (0.0)
- TIMECH - time at which time interval becomes DT2, STEADY must be set to false, CHDT must be set to true (1.E6)
- DT3 - final time interval for unsteady-state solution, STEADY must be set to false, CHDT must be set to true, TIMECH must be less than TCH2 (0.0)
- TCH2 - time at which time interval becomes DT3, STEADY must be set to false, CHDT must be set to true, TIMECH must be less than TCH2 (1.E6)
- JPRN - print interval of time steps, STEADY must be set to false (1)
- NSNØDS- number of monitored nodes for unsteady-state summary table, STEADY must be set to false (0)
- NODES(i) - nodes monitored for unsteady-state summary table, STEADY must be set to false, NSNØDS must be greater than 0.
- NBL - number of boundary nodes for mass check, MSCK must be set to true (0)
- NCAT - number of dynamic dirichlet conditions. STEADY must be set to false.

Global coordinates. This read statement inputs the global coordinates of the nodes.

Format: x-coordinate, y-coordinate (10X,2F10.0)  
Number of cards: NN  
Remarks: One node per card

Global node numbers describing the elements. This read statement gives the element's local node numbers their global node numbering. The element's local node numbers are to be numbered counter-clockwise as shown below.



Format: Global nodes for element nodes 1,2,3,4,5,6,7,8. (5X,I5,7I10).

Number of Cards: NE

Remarks: One element per card. If LINVEL is set to true, then the mid-side (even numbered) element nodes are not input. Under this condition the format changes to (5X,I5,3I20) and the program will list the mid-side nodes as NNC+1.

When numbering the nodes in the domain and using models 2 or 3, number the corner nodes first, then number the mid-side nodes such that the numbering sequence is:

Corner nodes: 1 to NNC

Mid-side nodes: NNC+1 to NN

Velocity profile. This read statement inputs the X and Y-components of the velocity at each node.

Format: X-component of velocity, Y-component of velocity  
(10X,2F10.0)

Number of Cards: NN

Remarks: One node per card. If CØNVEL is set to true then only one card is needed to input the X and Y-components of the velocity.

Dispersion coefficient. This read statement inputs the dispersion coefficient for each element.

Format: Dispersion coefficient in X-direction for element i,  
Dispersion coefficient in Y-direction for element i,  
Dispersion coefficient in X-direction for element i+1,  
Dispersion coefficient in Y-direction for element i+1,  
etc. (8F10.0)

Number of Cards: Integer part of  $[(N-1)/4] + 1$

Remarks: Four elements per card. If CØNDIF is set to true then only one card is needed to input the X and Y-directions of the dispersion coefficient. The format becomes (2F10.0)

Sources and sinks. This read statement inputs the source or sink terms for the steady and unsteady-states.

Format: Element of source or sink, strength of source or sink, start time of source or sink, finish time of source or sink (5X,I5,3F10.0)

Number of Cards: NSC

Remarks: One source or sink per card. If NSC is set to 0 then there are no cards used. If STEADY is set to true then the start and finish times are not used. If STEADY is set to false then the start and finish times are adjusted by the program to the nearest time step. Use of source or sink should only occur during initial time interval.

Concentration background. This read statement gives all of the nodes their initial concentration values at time zero for an unsteady-state problem.

Format: Concentration at node i, concentration at node i+1, etc. (8F10.0)

Number of Cards: Integer part of  $((N-1)/8)+1$

where,  $N=NN$  for model 1,  
 $N=NNC$  for all other models.

Remarks: Eight nodes per card. If STEADY is set to true then read no cards. If CØNCB is set to true then only one card is needed and the format becomes (F10.0)

Decay coefficient. This read statement inputs the decay coefficient

for each element for an unsteady-state problem.

Format: Decay coefficient for element  $i$ ,  $i+1$ ,  $i+2$ , etc.  
(8F10.0)

Number of Cards: Integer part of  $[(NE-1)/8]+1$

Remarks: Eight elements per card. If STEADY is set to true, read no cards. If C0NDEC is set to true then only one card is needed and the format becomes (F10.0)

Boundary nodes. This read statement inputs the boundary nodes of a domain for a mass balance check.

Format: Node  $i$ , node  $i+1$ , etc (8I10)

Number of Cards: Integer part of  $[(NBL-1)/8]+1$

Remarks: Eight nodes per card. If MSCK is set to false then no cards are read. Nodes must be listed in an order consistent with the direction of the node numbering of the elements. The nodes listed describe the edges between them, so to have a proper mass check the boundary edges should form a closed loop. If there is more than one closed boundary loop then a dummy node with a value of zero must be listed between the multiple loops of boundary nodes.

Dynamic Dirichlet condition nodes. This input statement describes the nodes with specified-values for finite periods of time in an unsteady-state solution.

Format: Node number, value of concentration, start time, finish time (5X,I5,3F10.0)

Number of Cards: NCAT

Remarks: One node per card. If STEADY is set to true or NCAT equals zero then no cards are read. The dynamic Dirichlet condition supercedes the static Dirichlet condition. The dynamic Dirichlet condition can not affect nodes at time zero. This must be done with the static Dirichlet condition. If a node is specified as a dynamic Dirichlet node for any finite period of time, it must be described in that manner through the duration of the run.

Static Dirichlet condition nodes. This read statement reads in the boundary nodes with static specified-values,

Format: Node, value of concentration (I10,F10.0)

Number of Cards: Number of static Dirichlet nodes.

Remarks: One node per card. This read statement terminates  
by reading an "end of file" card.



## Appendix C

FORTRAN IV - LISTING OF COMPUTER MODEL





```

-----
C WRITE DECAY COEFFICIENTS
-----
5212 FORMAT('D', 'DECAY COEFFICIENT FOR ENTIRE DOMAIN IS', F8.4)
IF (CONC) WRITE(6, 5212) DFCAY
IF (CONC) GO TO 5213
WRITE(6, 5213)
5213 FORMAT('D', 'DECAY COEFFICIENTS', /, ' ELEMENTS')
      3X ME-1, /10D1
      4X 23A 1A3, /X
      35X ME-1, /10D1
      /F14A, /F, 1X3 /E9E
5214 WRITE(6, 5214) JZ, JZ15, JZ1
5215 CONTINUE
C
-----
C WRITE UNCLAYED TIME INFO
-----
192 FORMAT('D', 'TIME INTERVALS', F8.2, /, ' MAY TIME', F8.2, /)
IF (COT) WRITE(6, 192) DT, DTPECH
IF (COT) AND, DT1, DT2, DT3 WRITE(6, 193) DT1, DT2, DT3
193 FORMAT('D', 'TIME INTERVAL CHANGES TO', F8.2, /, ' WHEN TIME IS', F8.2)
C
-----
C WRITE INITIAL BACKGROUND CONCENTRATIONS
-----
IF (CONC) GO TO 8A4
WRITE(6, 8A4)
8A4 FORMAT('D', 'BACKGROUND CONCENTRATIONS', /, ' NODES')
      4X 14X 1A3, /X
      15X 1A-1, /10D1
      /F14A, /F, 1X3 /E9E
8A5 WRITE(6, 8A5) JZ, JZ15, JZ15, JZ15
      GO TO 8A8
8A8 FORMAT('D', 'BACKGROUND CONCENTRATION FOR ENTIRE DOMAIN', F8.2)
8A9 CONTINUE
C
-----
C WRITE VARIOUS UNSATURY STATE INFORMATION
-----
IF (UNSAT) GO TO 5129
WRITE(6, 5129)
5129 WRITE(6, 5129) 'THERE IS NOT A SUMMARY TABLE OF SELECTED NODES'
      GO TO 5127
5128 WRITE(6, 5128) 'NODES', JZ1, JZ15, UNSAT
5129 CONTINUE
5130 FORMAT('D', 'NODES FOR SUMMARY TABLE', /, ' IS')
5131 CONTINUE
      GO TO 5129
5132 WRITE(6, 5132) 'THERE IS NOT A MASS CHECK'
      GO TO 9277
5133 WRITE(6, 5133)
5134 WRITE(6, 5134)
5135 FORMAT('D', 'BOUNDARY NODES')
5136 FORMAT('D', 'THERE IS A MASS CHECK', /, ' BOUNDARY OF DOMAIN')
      4X 15X 1A3, /X
      15X 1A-1, /10D1
      /F14A, /F, 1X3 /E9E
5137 WRITE(6, 5137) JZ15, JZ15, JZ15, JZ15, JZ15, JZ15
5138 FORMAT('D', 'UNSAT')
C
-----
C WRITE VARIOUS CONTROL INFORMATION
-----
9277 CONTINUE
IF (UNSAT) GO TO 9381
9382 FORMAT('D', 'QUADRATIC VELOCITY PROFILE')
      GO TO 9383
9384 WRITE(6, 9384)
9385 FORMAT('D', 'LINEAR VELOCITY PROFILE')
9386 CONTINUE
IF (UNSAT) GO TO 927
9387 FORMAT('D', 'QUADRATIC CALCERN')
9388 FORMAT('D', 'LINEAR')
9389 FORMAT('D', '1', /F8.2)
IF (FZ) AND, CONC AND, ZERR) GO TO 9442
IF (FZ) AND, CONC WRITE(6, 9442)
9443 FORMAT('D', 'CONCENTRATIONS ADJUSTED AFTER EACH DT--CALCULATED VALUE
      4X 8X 8X SHOWN IN OUTPUT')
9444 WRITE(6, 9444)
9445 FORMAT('D', 'CONCENTRATIONS ADJUSTED AFTER EACH DT--ADJUSTED VALUES
      4X 8X 8X SHOWN IN OUTPUT')
9446 CONTINUE
WRITE(6, 9447) JZ, JZ15
9447 FORMAT('D', 'PRINT INTERVAL IS', /F8.2)
9448 CONTINUE
IF (UNSAT) CALL SETUP
IF (UNSAT) CALL UPSHP
C
-----
C BRING IN WORKING VARIABLES
-----
926 CALL STUP
CALL SUWD
CALL SHPNC
IF (UNSAT) GO TO 7722
WRITE(6, 7722)
7722 WRITE(6, 7722)
IF (FZ) GO TO 9391
WRITE(6, 7723)
      GO TO 779
9391 WRITE(6, 7723)
776 FORMAT('D', '1', /F8.2)
779 CONTINUE
IF (UNSAT) CALL SETUP
IF (UNSAT) CALL UPSHP
C
-----
C CALL FOR MATRIX, SCAL, AND BLOC
C THIS AREA FORMULATES THE SYSTEM OF EQUATIONS
-----
IF (C) GO TO 927
      GO TO 928
ALORD(J) = CONB(J)
IF (CONB) WRITE(6, 928) JZ, JZ15, JZ15
      GO TO 929
929 WRITE(6, 929)
930 WRITE(6, 930)
931 WRITE(6, 931)
932 WRITE(6, 932)
933 WRITE(6, 933)
934 WRITE(6, 934)
935 WRITE(6, 935)
936 WRITE(6, 936)
937 WRITE(6, 937)
938 WRITE(6, 938)
939 WRITE(6, 939)
940 WRITE(6, 940)
941 WRITE(6, 941)
942 WRITE(6, 942)
943 WRITE(6, 943)
944 WRITE(6, 944)
945 WRITE(6, 945)
946 WRITE(6, 946)
947 WRITE(6, 947)
948 WRITE(6, 948)
949 WRITE(6, 949)
950 WRITE(6, 950)
951 WRITE(6, 951)
952 WRITE(6, 952)
953 WRITE(6, 953)
954 WRITE(6, 954)
955 WRITE(6, 955)
956 WRITE(6, 956)
957 WRITE(6, 957)
958 WRITE(6, 958)
959 WRITE(6, 959)
960 WRITE(6, 960)
961 WRITE(6, 961)
962 WRITE(6, 962)
963 WRITE(6, 963)
964 WRITE(6, 964)
965 WRITE(6, 965)
966 WRITE(6, 966)
967 WRITE(6, 967)
968 WRITE(6, 968)
969 WRITE(6, 969)
970 WRITE(6, 970)
971 WRITE(6, 971)
972 WRITE(6, 972)
973 WRITE(6, 973)
974 WRITE(6, 974)
975 WRITE(6, 975)
976 WRITE(6, 976)
977 WRITE(6, 977)
978 WRITE(6, 978)
979 WRITE(6, 979)
980 WRITE(6, 980)
981 WRITE(6, 981)
982 WRITE(6, 982)
983 WRITE(6, 983)
984 WRITE(6, 984)
985 WRITE(6, 985)
986 WRITE(6, 986)
987 WRITE(6, 987)
988 WRITE(6, 988)
989 WRITE(6, 989)
990 WRITE(6, 990)
991 WRITE(6, 991)
992 WRITE(6, 992)
993 WRITE(6, 993)
994 WRITE(6, 994)
995 WRITE(6, 995)
996 WRITE(6, 996)
997 WRITE(6, 997)
998 WRITE(6, 998)
999 WRITE(6, 999)
1000 WRITE(6, 1000)

```











